



Politecnico
di Bari

Repository Istituzionale dei Prodotti della Ricerca del Politecnico di Bari

Dynamic input allocation in presence of periodic exogenous signals: spline-based and sensitivity functions approaches

This is a PhD Thesis

Original Citation:

Dynamic input allocation in presence of periodic exogenous signals: spline-based and sensitivity functions approaches / Akbari, Shima. - ELETTRONICO. - (2026).

Availability:

This version is available at <http://hdl.handle.net/11589/295740> since: 2026-01-16

Published version

DOI:

Publisher: Politecnico di Bari

Terms of use:

(Article begins on next page)



Italian National Ph.D. Program in Autonomous Systems

ACADEMIC DISCIPLINE: SYSTEMS AND CONTROL ENGINEERING (IINF-04/A)

Final Dissertation

Dynamic Input Allocation in Presence of Periodic Exogenous Signals: Spline-based and Sensitivity Functions Approaches

by

Shima Akbari

firma

Administrative Headquarters:

Politecnico di Bari – Department of Electrical and Information Engineering

Hosting University:

University of Rome Tor Vergata – Department of Civil and Computer Engineering

Referees:

Prof. Edouard Laroche

Prof. Andrea Cristofaro

Supervisors:

Prof. Sergio Galeani

firma

Prof. Laura Menini

firma

Coordinator of Ph.D Program

Prof. Mariagrazia Dotoli

firma

LIBERATORIA PER L'ARCHIVIAZIONE DELLA TESI DI DOTTORATO

Al Magnifico Rettore
del Politecnico di Bari

la sottoscritta Shima Akbari nata a Tehran (IRAN) il 23/08/1995
residente a Roma in via Luigi Chiozza, 23 e-mail SHIMA.AKBARIEE@GMAIL.COM
iscritto al 3° anno di Corso di Dottorato di Ricerca in Autonomous Systems (DAUSY) ciclo 38
ed essendo stato ammesso a sostenere l'esame finale con la prevista discussione della tesi dal titolo:
Dynamic Input Allocation in Presence of Periodic Exogenous Signals: Spline-based and Sensitivity Functions Approaches

DICHARA

- 1) di essere consapevole che, ai sensi del D.P.R. n. 445 del 28.12.2000, le dichiarazioni mendaci, la falsità negli atti e l'uso di atti falsi sono puniti ai sensi del codice penale e delle Leggi speciali in materia, e che nel caso ricorreranno dette ipotesi, decade fin dall'inizio e senza necessità di nessuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni;
- 2) di essere iscritto al Corso di Dottorato di ricerca Autonomous Systems (DAUSY) ciclo 38, corso attivato ai sensi del "Regolamento dei Corsi di Dottorato di ricerca del Politecnico di Bari", emanato con D.R. n.286 del 01.07.2013;
- 3) di essere pienamente a conoscenza delle disposizioni contenute nel predetto Regolamento in merito alla procedura di deposito, pubblicazione e autoarchiviazione della tesi di dottorato nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica;
- 4) di essere consapevole che attraverso l'autoarchiviazione delle tesi nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica del Politecnico di Bari (IRIS-POLIBA), l'Ateneo archiverà e renderà consultabile in rete (nel rispetto della Policy di Ateneo di cui al D.R. 642 del 13.11.2015) il testo completo della tesi di dottorato, fatta salva la possibilità di sottoscrizione di apposite licenze per le relative condizioni di utilizzo (di cui al sito <http://www.creativecommons.it/Licenze>), e fatte salve, altresì, le eventuali esigenze di "embargo", legate a strette considerazioni sulla tutelabilità e sfruttamento industriale/commerciale dei contenuti della tesi, da rappresentarsi mediante compilazione e sottoscrizione del modulo in calce (Richiesta di embargo);
- 5) che la tesi da depositare in IRIS-POLIBA, in formato digitale (PDF/A) sarà del tutto identica a quelle **consegnate**/inviata/da inviarsi ai componenti della commissione per l'esame finale e a qualsiasi altra copia depositata presso gli Uffici del Politecnico di Bari in forma cartacea o digitale, ovvero a quella da discutere in sede di esame finale, a quella da depositare, a cura dell'Ateneo, presso le Biblioteche Nazionali Centrali di Roma e Firenze e presso tutti gli Uffici competenti per legge al momento del deposito stesso, e che di conseguenza va esclusa qualsiasi responsabilità del Politecnico di Bari per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi;
- 6) che il contenuto e l'organizzazione della tesi è opera originale realizzata dal sottoscritto e non compromette in alcun modo i diritti di terzi, ivi compresi quelli relativi alla sicurezza dei dati personali; che pertanto il Politecnico di Bari ed i suoi funzionari sono in ogni caso esenti da responsabilità di qualsivoglia natura: civile, amministrativa e penale e saranno dal sottoscritto tenuti indenni da qualsiasi richiesta o rivendicazione da parte di terzi;
- 7) che il contenuto della tesi non infrange in alcun modo il diritto d'Autore né gli obblighi connessi alla salvaguardia di diritti morali od economici di altri autori o di altri aventi diritto, sia per testi, immagini, foto, tabelle, o altre parti di cui la tesi è composta.

Luogo e data Bari 15/01/2026

Firma


Shima Akbari

Il/La sottoscritto, con l'autoarchiviazione della propria tesi di dottorato nell'Archivio Istituzionale ad accesso aperto del Politecnico di Bari (POLIBA-IRIS), pur mantenendo su di essa tutti i diritti d'autore, morali ed economici, ai sensi della normativa vigente (Legge 633/1941 e ss.mm.ii.),

CONCEDE

- al Politecnico di Bari il permesso di trasferire l'opera su qualsiasi supporto e di convertirla in qualsiasi formato al fine di una corretta conservazione nel tempo. Il Politecnico di Bari garantisce che non verrà effettuata alcuna modifica al contenuto e alla struttura dell'opera.
- al Politecnico di Bari la possibilità di riprodurre l'opera in più di una copia per fini di sicurezza, back-up e conservazione.

Luogo e data Bari 15/01/2026

Firma


Shima Akbari



Shima Akbari

Dynamic Input Allocation in Presence of Periodic Exogenous Signals: Spline-based and Sensitivity Functions Approaches

Thesis submitted for the degree of Philosophiae Doctor

Italian National Ph.D. Program in Autonomous Systems
University of Rome Tor Vergata

Tutors

Prof. Engr. *Sergio Galeani*

Prof. Engr. *Laura Menini*



2026



Ministero
dell'Università
e della Ricerca



Italiadomani
PIANO NAZIONALE
DI RIPRESA E RESILIENZA



Politecnico
di Bari



The doctoral scholarship was funded by the European Union - Next Generation EU, Mission 4 Component 1 CUP *D93C22000850005* (*PNRR* – 351).

Dissertation submitted for the degree of *Philosophiae Doctor*
Italian National Ph.D. Program in Autonomous Systems

Cycle:
38th

Administrative Headquarters:
Politecnico di Bari

Hosting University:
University of Rome Tor Vergata

Title:
Dynamic Input Allocation in Presence of Periodic Exogenous Signals: Spline-based and Sensitivity Functions Approaches

Ph.D Candidate:
Shima Akbari, University of Rome Tor Vergata (Rome, Italy)

Tutors:
Prof. Engr. Sergio Galeani, University of Rome Tor Vergata (Rome, Italy)
Prof. Engr. Laura Menini, University of Rome Tor Vergata (Rome, Italy)

Coordinator:
Prof. Engr. Mariagrazia Dotoli, Politecnico di Bari (Bari, Italy)

External Reviewers:
Prof. Engr. Edouard Laroche, University of Strasbourg (Strasbourg, France)
Prof. Andrea Cristofaro, Sapienza University (Rome, Italy)

Last version:
January 9, 2026

All rights reserved. No part of this publication may be reproduced or transmitted, in any form or by any means, without permission.

Abstract

Modern engineering systems are increasingly characterized by overactuation, where multiple actuators influence the same output variable. This redundancy offers opportunities for performance enhancement, robustness, and fault tolerance, yet it also poses the fundamental challenge of allocating actuator effort in a manner that improves efficiency without degrading closed-loop behavior. This thesis develops and analyzes novel strategies for Dynamic Control Allocation (DCA) in systems with input redundancy, with particular emphasis on nonlinear dynamics and periodic operating conditions.

The overarching objective is to design allocation schemes that operate within the nullspace of the input-output map of the plant, thereby preserving the plant output while reshaping actuator commands to optimize steady-state performance. This property of output invisibility allows allocation to be introduced as an auxiliary mechanism on top of existing feedback controllers, enriching system behavior without altering regulation or tracking performance. Building upon this principle, the thesis advances DCA along three interconnected directions: the development of spline-based internal models for allocation, the application of DCA to complex overactuated platforms, and the extension of allocation to nonlinear systems subject to periodic exogenous signals.

The first contribution lies in the introduction of polynomial and spline-type internal models for allocation signals. By moving beyond constant or purely sinusoidal structures, the proposed spline-based allocator offers enhanced expressiveness and flexibility, enabling more effective redistribution of actuator effort in steady state. Theoretical guarantees ensure stability, invisibility, and optimality, while numerical studies confirm superior performance compared with earlier approaches, particularly in the presence of time-varying references.

The second contribution is an application-oriented study of allocation strategies on a highly nonlinear aerial vehicle equipped with tilting-twisting propellers. Two distinct DCA methods—an instantaneous allocator and a periodic spline-based allocator—are compared under both idealized and realistic actuator dynamics. The results highlight a fundamental trade-off: while the instantaneous allocator is simple and optimal in the absence of dynamics, the spline-based method achieves smoother and more robust performance when actuator dynamics are non-negligible. This investigation demonstrates the feasibility of deploying DCA in demanding aerospace scenarios and clarifies the relative advantages of different allocation paradigms.

The third contribution extends DCA to nonlinear systems subject to periodic exogenous signals. The proposed framework combines gradient-based optimization with parameter-dependent sensitivity dynamics, thereby extending control allocation beyond conventional point-wise (instantaneous) schemes. This formulation systematically incorporates periodicity in both the reference trajectories and the cost functionals, allowing the allocator to exploit the temporal structure of the problem rather than optimizing solely at each instant. A key feature of the proposed method is the preservation of the output invisibility property, ensuring that the introduction of allocation dynamics does not interfere with the nominal closed-loop system output guaranteed by the baseline controller. The approach is implemented and validated on a nonlinear two-mass spring-damper system, which serves as an academic benchmark demonstrating the effectiveness, stability, and practical relevance of the proposed sensitivity-based dynamic allocation framework.

Taken together, these developments establish a coherent and versatile framework for dynamic control allocation in both linear and nonlinear systems. By combining rigorous analysis with illustrative applications, the thesis demonstrates how allocation can be leveraged to reduce actuator usage, improve efficiency, and enhance robustness, all while ensuring that closed-loop output remains unaffected. The contributions pave the way toward experimental validation on robotic and aerial platforms, integration with high-level motion planning, and extensions to distributed and fault-tolerant control architectures. In this way, the work advances dynamic control allocation from a primarily theoretical construct to a practical methodology for managing redundancy in complex modern systems.

To my husband,

my heart, my anchor, and my greatest blessing. Thank you for your endless patience, for the quiet strength behind every one of my steps, and for loving me fiercely through every storm. This journey would have been unbearable without your hand in mine.

To my brother,

my lifelong protector and silent cheerleader. Your unwavering faith in me, your strength, and your quiet love have been a constant source of courage. I carry your support with me in every achievement.

To my parents,

for your sacrifices, your resilience, and the kind of love that asks for nothing but gives everything. I owe this path to you.

And to my supervisor,

thank you for your guidance, trust, and the freedom to find my own way.

With deepest gratitude to you all.

Contents

Acronyms	ix
Preface	x
List of Papers Written by the Author	xi
Introduction	
1 Introduction	2
1.1 Context and Motivation	2
1.2 Limitations of Classical Approaches	2
1.3 Contributions and Thesis Objectives	3
1.4 Thesis Structure	4
Chapter 2	
2 Background and State of the Art	8
2.1 Conceptual Distinction Between Allocation Paradigms	8
2.2 Control Allocation in Redundant Systems	9
2.3 Static vs. Dynamic Control Allocation	12
2.4 Dynamic Control Allocation for Linear Systems	15
2.5 Dynamic Control Allocation for Nonlinear Systems	17
Chapter 3	
3 Dynamic Control Allocation Using Periodic Spline Signals	21
3.1 Introduction	21
3.2 Model, Assumptions, and Objective	22
3.3 Allocator Architecture: Periodic Splines and Nullspace Injection . .	27
3.4 Theoretical Guarantees	34
3.5 Computational Experiments	37
3.6 Discussion and Concluding Remarks	45
Chapter 4	
4 Spline-based Input Allocation on an Overactuated Tilting and Twisting Unmanned Aerial Vehicle	47
4.1 Introduction	47
4.2 UAV Architecture and Control Objectives	48
4.3 Dynamics and Baseline Controller	51
4.4 Allocated Wrench Mapper	56
4.5 Flight Simulations	61
4.6 Discussion and Concluding Remarks	65
Chapter 5	
5 A Sensitivity Approach to Periodic Control Allocation in Nonlinear Systems	69
5.1 Introduction	69
5.2 Nonlinear Plant Class and Periodic Cost	70
5.3 Sensitivity-based Allocator Design	76

5.4	Academic Example	81
5.5	Discussion and Concluding Remarks	87
Chapter 6		
6	Dynamic Control Allocation for Nonlinear Systems via a Sensitivity Approach: Refinements and a Case Study	89
6.1	Introduction	89
6.2	Problem Formulation	89
6.3	Allocator Design	94
6.4	Application to a Mechanical System	99
6.5	Discussion and Concluding Remarks	100
Chapter 7		
7	Conclusions and Perspectives	103
7.1	Summary of Contributions	103
7.2	Applications and Validation	104
7.3	Limitations and Open Questions	104
7.4	Future Directions	104
7.5	Concluding Remarks	105
References		106

List of Figures

2.1	Direct Allocation for Actuation Mapping	8
2.2	Add-on Allocation for Performance Enhancement.	9
3.1	General Closed-Loop Control System with Allocation.	24
3.2	Architecture of a DCA unit (cfr. [10], [22]). Dashed arrows denote updates occurring at times $t = kT$, $t \in \mathbb{N}$	25
3.3	The period is divided into several segments and a spline (a fixed degree polynomial of time) is considered on each segment	30
3.4	Constant Reference: comparison of output trajectories with and without allocation. The regulated output y remains unaffected by the allocation mechanism, thereby confirming the Output Invisibility property R2) of Problem 1.	38
3.5	Constant Reference: input trajectories u_1 and u_2 before allocation. The absence of allocation results in redundant control inputs that are not optimized with respect to the cost functional.	38
3.6	Constant Reference: cost functional J over successive periods. The action of the allocator leads to a monotonic decrease of the cost, illustrating the convergence of the gradient-based update law and the fulfillment of requirement R3).	39
3.7	Constant Reference: input trajectories u_1 and u_2 after allocation. Compared with Fig. 3.5, the allocation signal redistributes the control effort, producing inputs that achieve the same output regulation while reducing the steady-state cost.	39
3.8	Saw-tooth reference: plant inputs with a single sinusoidal allocator. The allocator reduces the cost but exhibits limited flexibility in shaping the input trajectory due to the restricted signal class.	41
3.9	Saw-tooth reference: comparison of the cost functional J for three cases: unallocated inputs, allocation with a single sinusoidal internal model, and allocation with the proposed spline-based internal model. The spline allocator consistently achieves the lowest cost, thereby validating its enhanced expressivity and steady-state optimality.	41
3.10	Saw-tooth reference: plant inputs obtained with the spline-based allocator. Compared to the sinusoidal case (Fig. 3.8), the spline allocator produces richer input trajectories, allowing more efficient cost reduction while maintaining output invisibility.	42
3.11	Input trajectories (u_1, u_2) before and after allocation for the constant-reference case. The allocated inputs settle deeper inside the cost-free region, confirming the effectiveness of the steady-state optimization. . . .	43
3.12	Input trajectories (u_1, u_2) before and after allocation for the sawtooth reference. The allocated trajectory gradually moves toward the center of the cost-free region, illustrating the effect of the periodic spline-based optimization on actuator usage.	43
3.13	Saw-tooth reference: comparison of output trajectories without allocation, with spline allocation method and with sinusoidal allocation method. The regulated output y remains unaffected by the allocation mechanism, thereby confirming the Output Invisibility property R2) of Problem 1.	44

4.1	Comparison of closed-loop architectures: (a) a standard control system where the controller C directly drives the plant P ; (b) an augmented system with a dynamic control allocator ensuring output invisibility while exploiting input redundancy for performance optimization.	49
4.2	Schematic representation of the UAV with tilting and twisting propellers. The illustration highlights the world frame F_W , the body frame F_B , and the twelve independent control inputs: twist angles α_i , tilt angles β_i , and propeller angular velocities ω_i , for $i = 1, \dots, 4$	52
4.3	Control architecture of the UAV, adapted from [23]. The diagram highlights the cascade of reference generation, feedback control, wrench mapping, inverse kinematics, and actuator dynamics.	54
4.4	Modified control architecture of the UAV with DCA included after the Wrench Mapper block.	57
4.5	Reference trajectory used in the simulation experiments, consisting of a square motion in the x - y plane combined with a sinusoidal variation along the z axis.	62
4.6	Comparison of costs in the instantaneous actuation case. The cost is minimized by both allocation strategies, with the instantaneous greedy allocator performing better than the periodic one when no actuator dynamics are present.	64
4.7	Comparison of costs in the dynamic actuation case. When actuator dynamics are present, the periodic allocator outperforms the instantaneous one, achieving a lower cost.	64
4.8	Angular velocities of the four propellers under instantaneous actuation. The angular velocity of the first propeller is reduced by both allocation strategies, but the instantaneous (greedy) allocator achieves lower cost than the periodic one (see Fig. 4.6).	66
4.9	Angular velocities of the four propellers under dynamic actuation (first-order actuator dynamics). The angular velocity of the first propeller is reduced by both allocation strategies, but the periodic spline-based allocator outperforms the instantaneous one (see Fig. 4.7).	66
5.1	<i>A priori</i> given closed-loop control system Σ (dotted, light grey) augmented with DCA as Σ_{all} (dashed, dark grey).	75
5.2	Scenario I. Comparison of instantaneous cost function $g(u_{ss}(t), z_{ss}(t))$ before and after allocation in case of constant reference: both methods achieve the same performance.	78
5.3	Scenario II. Output invisibility is achieved by both methods.	81
5.4	Scenario II. First input channel before and after allocation.	83
5.5	Scenario II. Second input channel before and after allocation.	83
5.6	Scenario II. Comparison of instantaneous cost function $g(u_{ss}(t), z_{ss}(t))$ before and after allocation in case of periodic reference.	85
5.7	Scenario II. Steady-state input trajectory in the (u_1, u_2) plane, before and after allocation.	85
5.8	Scenario II. Comparison of periodic cost function $\int_0^T g(u_{ss}(\tau), z_{ss}(\tau)) d\tau$ before and after allocation.	86
6.1	<i>A priori</i> given closed-loop control system Σ (dotted, light grey) augmented with DCA as Σ_{all} (dashed, dark grey).	90
6.2	A mass spring damper system with variable stiffness.	99
6.3	First input channel before and after allocation.	100
6.4	Second input channel before and after allocation. u_2 increases as a price to be paid in order to minimize u_1	100
6.5	Input evolution, from unallocated to allocated. For clarity, unequally-spaced iteration numbers are shown.	101

Acronyms

ANN *Annihilator.* 12

CoM *Center of Mass.* 51

DCA *Dynamic Control Allocation.* 3

DoF *Degrees of Freedom.* 47

FB *Body Frame.* 51

FW *World Frame.* 51

ISC *Input-to-State Convergent.* 74

OSSR *Optimal Steady-State Regulation.* 17

PD *Proportional–Derivative.* 54

QP *Quadratic Program.* 12

ROSPo *ROtor graSPing Omnidirectional.* 13

SCA *Static Control Allocation.* 2

SISO *Single-Input Single-Output.* 37

SO(3) *Special Orthogonal Group in 3D.* 51

TF *Transfer Function.* 16

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of *Philosophiae Doctor* in the National Ph.D. Program in Autonomous Systems, headquartered at Politecnico di Bari, and conducted at the University of Rome Tor Vergata.

The research presented herein was carried out from November 2022 to October 2025 as part of the Ph.D. program in the Department of Electrical and Information Engineering of Polytechnic University of Bari and Department of Civil Engineering and Computer Engineering of University of Rome Tor Vergata. The work focuses on the development of advanced control allocation strategies for overactuated and nonlinear systems, with applications to aerial vehicles and robotic systems.

The thesis reflects the research articles authored by the candidate during the doctoral studies. Each chapter corresponds to a core contribution in the field, detailing theoretical advances, algorithmic implementations, and simulation results. The research was conducted under the supervision of Prof. Sergio Galeani and Prof. Laura Menini.

The full list of papers written by the author is reported hereafter.

List of Papers Written by the Author

International Journal Articles

- Akbari, S., Galeani, S., Manca, G., and Sassano, M., “Dynamic control allocation for nonlinear systems via a sensitivity approach,” *IEEE Control Systems Letters*, vol. 9, pp. 2549-2554, 2025, doi: 10.1109/LCSYS.2025.3633369.

International Conference Proceedings

- Akbari, S., Galeani, S., Manca, G., and Sassano, M., “A sensitivity approach to periodic control allocation in nonlinear systems,” in *IEEE Conference on Decision and Control (CDC)*, Accepted for publication, 2025.
- Akbari, S., Akbari, N., Oriolo, G., and Galeani, S., “Visual Urban Navigation for Mobile Robots: Implementation in the Duckietown Environment,” in *International Conference on Control, Automation and Diagnosis (ICCAD)*, IEEE, 2025, pp. 1–6. doi: 10.1109/ICCAD64771.2025.11099311.
- Akbari, S., Manca, G., Galeani, S., and Sassano, M., “Spline-based Input Allocation On An Overactuated Tilting and Twisting Unmanned Aerial Vehicle,” in *International Conference on Control, Decision and Information Technologies (CoDIT)*, ISSN: 2576-3555, IEEE, 2024, pp. 2329–2334. doi: 10.1109/CoDIT62066.2024.10708417.
- Akbari, S., Galeani, S., and Sassano, M., “Towards Spline-based Dynamic Input Allocation,” in *International Conference on Control, Automation and Diagnosis (ICCAD)*, ISSN: 2767-9896, IEEE, 2023, pp. 1–6. doi: 10.1109/ICCAD57653.2023.10152316

Acknowledgements

Reaching the end of this Ph.D. journey has been one of the most challenging and transformative experiences of my life, and it would not have been possible without the support, kindness, and guidance of many incredible people to whom I owe my deepest gratitude.

First and foremost, to my loving husband — thank you for walking this path by my side, not just in life but also in this very same Ph.D. program. Your presence was my strength during the darkest days and my joy during the brightest ones. Collaborating with you has been both intellectually stimulating and emotionally grounding. I am endlessly grateful for your patience, brilliance, and the quiet way you always reminded me to believe in myself.

To my dear brother — your unconditional love and quiet strength have carried me through many unseen battles. Even from afar, you've always managed to make me feel supported, understood, and never alone. Your belief in me, spoken or silent, has meant more than I can say.

To my wonderful parents — thank you for being the pillars of my life. Your sacrifices, unwavering love, and endless encouragement have given me the foundation to pursue my dreams. This journey was made possible by the values you instilled in me: resilience, integrity, and hope.

My heartfelt thanks go to my supervisor, Prof. Sergio Galeani, for his insightful guidance, constant availability, and exceptional support throughout these years. Your mentorship has been fundamental to my academic and personal growth. I am especially grateful for your patience, your way of explaining complex ideas with clarity, and for always encouraging me to think deeply and critically.

I would also like to warmly thank Prof. Laura Menini for her support and valuable feedback, and for always providing thoughtful perspective on both research and life. Your kindness and care are truly appreciated.

A special thanks to Prof. Mario Sassano, whose brilliance and depth of understanding have helped me through countless difficult moments. Your insight always arrived exactly when I needed it most, and your intellectual sharpness and generosity have left a deep impression on me.

I would also like to extend my sincere gratitude to Prof. Antonio Franchi and Prof. Chiara Gabellieri at the University of Twente, where I had the opportunity to spend a significant period abroad. Their guidance and generosity made that experience not only academically enriching but also personally rewarding. Working in their group exposed me to new perspectives, broadened the scope of my research, and provided a stimulating environment that greatly contributed to the development of this thesis. I am deeply appreciative of their openness, encouragement, and the invaluable insights I gained during my time in the Netherlands.

To Prof. Mariagrazia Dotoli, coordinator and founder of the DAUSY Ph.D. program, thank you for your tireless dedication, precision, and the structured environment you have built for us. Your work behind the scenes has shaped our academic journey in

profound ways.

I am also sincerely grateful to Prof. Raffaele Carli and Dr. Gaetano Pavone for their kind availability and support throughout the program. Thank you both, along with the Post-Lauream Office, for always being responsive, helpful, and present whenever I needed administrative assistance.

To my colleagues and friends from the DAUSY program and beyond — thank you for the camaraderie, shared frustrations, and encouragement. You’ve made the tough days more bearable and the good days more memorable.

And finally, to every friend, teacher, relative, or kind stranger who offered a word of advice, a moment of help, or a gesture of support — I carry your kindness with me.

This work is not just the result of my effort, but a reflection of all those who believed in me and stood by me. I am deeply grateful to each and every one of you.

Introduction

Chapter 1

Introduction

1.1 Context and Motivation

The field of control allocation has gained increasing relevance as modern engineering systems have become more complex, interconnected, and redundant. Many autonomous platforms ranging from aerial and marine vehicles to spacecraft and robotic formations are equipped with more actuators than the number of independent control objectives or degrees of freedom. This property, known as *overactuation*, provides structural redundancy that can be deliberately exploited to enhance flexibility, robustness, and performance. Rather than being a byproduct of mechanical design, redundancy is often introduced intentionally to improve fault tolerance, distribute loads, reduce energy consumption, and extend actuator life. In practice, however, this abundance of actuation also creates a central challenge: determining how to distribute the required control effort among multiple actuators while respecting physical limits and optimizing system performance.

Control allocation refers precisely to the process of translating high-level control commands into individual actuator inputs in a way that satisfies both feasibility and optimality criteria. The allocation must obey actuator constraints such as saturation, rate limits, and dynamics, while ensuring that the system's control objectives are met. When performed effectively, control allocation allows redundant systems to exploit their full potential, achieving improved efficiency, smoother motion, and enhanced fault tolerance. When performed poorly, it can lead to wasted energy, degraded performance, or even instability.

As autonomy increases and systems operate under tighter coupling between control, perception, and planning layers, the design of allocation mechanisms has become more critical. Future autonomous systems must not only achieve primary tracking objectives but also dynamically optimize internal resource usage in real time. This requirement drives the evolution from traditional static allocation to dynamic frameworks capable of adapting actuator usage over time, even in nonlinear and uncertain environments.

1.2 Limitations of Classical Approaches

The allocation approaches can mainly be categorized into two subcategories according to their design structure: Static Control Allocation and Dynamic Control Allocation.

Classical static control allocation (SCA) techniques originated from static optimization methods developed in the mid-twentieth century. These formulations, including pseudo-inverse and weighted least-squares allocation [28], [32], were designed for linear plant model systems in which actuator dynamics could be neglected and control objectives were constant or slowly varying. They provided computationally efficient ways to distribute control commands and became standard tools in aerospace and marine applications [46]. However, allocation is computed independently at every time instant without considering temporal evolution, actuator lags, or persistent disturbances. While these methods demonstrated the benefits of redundancy, they also revealed fundamental limitations when applied to dynamic or constrained systems. Although effective for simple platforms, such assumptions are inadequate for modern autonomous systems characterized by strong couplings, hard saturation limits, and time-varying missions. Neglecting actuator

dynamics can cause performance degradation, loss of robustness, or instability under realistic conditions.

Dynamic Control Allocation (DCA) emerged to overcome these limitations by explicitly accounting for actuator dynamics and the temporal nature of control signals. Unlike static methods, DCA generates trajectories for actuator commands, improving smoothness, constraint handling, and coordination. Early works, such as [48], demonstrated that shaping actuator profiles over time can better reconcile control performance with actuator limitations. These developments marked an important shift: allocation became a dynamic process integrated into the overall control architecture rather than a memoryless mapping.

Nevertheless, most existing DCA approaches remain confined to linear time-invariant frameworks and rely on restrictive signal representations. Harmonic or sinusoidal parameterizations, while mathematically convenient, are often too rigid for applications requiring local adaptivity, abrupt redistribution of effort, or piecewise trajectory shaping. Their global smooth structure limits flexibility in agile robotic or aerial systems where control loads must change rapidly across different mission phases. Furthermore, theoretical guarantees for stability, convergence, and robustness have lagged behind practical implementations, and many algorithms remain too computationally expensive for real-time use in embedded systems. These gaps motivate the development of new DCA methodologies that combine theoretical rigor with computational practicality and that can handle both linear and nonlinear behaviors under realistic constraints.

1.3 Contributions and Thesis Objectives

This thesis addresses the aforementioned challenges by developing a coherent set of DCA methodologies that extend the applicability of DCA to a wider range of systems while preserving theoretical guarantees and computational efficiency. The guiding objective is to design allocation schemes that can exploit redundancy dynamically, improve steady-state and periodic performance, and operate transparently within existing control loops without altering the regulated outputs.

The proposed work advances the field along two complementary directions. The first approach concerns the development of spline-based DCA frameworks for overactuated systems that can be treated with linear allocation tools [3], [4]. In these methods, actuator commands are expressed through periodic splines (piecewise polynomial functions) that combine smoothness with local adaptability. This representation allows for flexible shaping of input trajectories while keeping computational complexity low. A spline-type internal model is integrated within the allocator, ensuring that the added dynamics do not interfere with the main control objectives. Theoretical results demonstrate stability, preservation of output invisibility, and convergence to steady-state periodic optima. Simulation studies confirm that spline-based allocation provides tangible benefits compared with static and harmonic approaches, particularly in scenarios involving actuator constraints and non-sinusoidal references.

The second approach extends the dynamic allocation concept to nonlinear systems through a sensitivity-based framework [1], [2]. Here, the allocator exploits sensitivity dynamics of steady-state trajectories with respect to allocation parameters, enabling the optimization of periodic performance indices in nonlinear plants. The formulation ensures that the additional input components remain invisible at the regulated outputs and provides provable guarantees of stability and convergence. The resulting architecture achieves dynamic redistribution of control effort while maintaining the baseline closed-loop behavior, bridging the gap between linear and nonlinear DCA formulations.

Both directions share a unifying philosophy: separating the regulation and allocation tasks while coupling them through rigorous structural conditions. In the linear case, this is achieved via null-space projections of the input map, whereas in the nonlinear case, uniform relative degree and normal-form representations expose the redundant channels through which allocation can act without disturbing the controlled output. This separation enables systematic synthesis and analysis, making it possible to add dynamic allocation capabilities to existing control systems with minimal modification.

Beyond these core theoretical advances, the thesis emphasizes computational tractability and practical implementation. The spline-based signals require only a limited number of parameters to capture complex behaviors, making them well suited for real-time applications. The sensitivity-based gradient algorithms are formulated on reduced steady-state models and enhanced with adaptive step-size mechanisms such as Armijo backtracking to ensure convergence under nonconvex cost functions. Together, these design principles guarantee that the proposed allocation methods remain lightweight, scalable, and suitable for embedded controllers on autonomous platforms.

The methodologies are validated through simulation studies on two representative systems. The first is an overactuated unmanned aerial vehicle (UAV) equipped with tilting and twisting propellers, used to demonstrate the spline-based allocator's capacity to handle redundancy and constraints effectively. The second is a nonlinear mass-spring-damper system with adjustable stiffness, which illustrates the sensitivity-based allocator's ability to optimize actuator usage in nonlinear environments. In both cases, the proposed methods achieve measurable reductions in input cost and improved smoothness without affecting the regulated outputs, confirming their theoretical and practical value.

The overarching goal of this thesis is to close persistent gaps in DCA research. It unifies theoretical insights and algorithmic designs into a consistent framework that is simultaneously rigorous, efficient, and practically relevant. By introducing spline-based and sensitivity-based approaches that reconcile stability analysis with computational feasibility, the thesis contributes to the deployment of advanced DCA techniques in modern autonomous and safety-critical systems. In doing so, it strengthens the connection between control theory and engineering practice, ensuring that redundancy can be leveraged systematically for improved performance and resilience.

1.4 Thesis Structure

The thesis is organized into seven chapters that collectively develop the theoretical framework, methodological innovations, and practical validations of advanced DCA. The progression follows a logical path from background material to the presentation of new methods, their application to representative systems, and the synthesis of insights through comparative analysis. Each chapter contributes a specific piece to the overall narrative, ensuring that the reader can follow the development of the ideas from their conceptual roots to their concrete implementation.

Chapter 2: Background and State of the Art

Chapter 2 establishes the theoretical foundation of the thesis by reviewing the main concepts and methods of control allocation in over-actuated systems. It first introduces the general formulation of the allocation problem and the principles of input redundancy. The discussion then distinguishes between static and dynamic allocation strategies, analyzing their respective advantages and limitations. Subsequent sections review the development of DCA for both linear and nonlinear systems, emphasizing the evolution

of methodologies and the mathematical frameworks that support them. The chapter concludes by identifying open challenges and research gaps that motivate the spline-based and sensitivity-based approaches proposed in the following chapters.

Chapter 3: Dynamic Control Allocation Using Periodic Spline Signals

Chapter three presents the first methodological contribution of the thesis, introducing a DCA framework in which periodic splines are employed as the signal basis for actuator commands. The architecture is composed of an annihilator, which guarantees the output invisibility property, an internal model that generates periodic spline trajectories, and a gradient block that adjusts the spline parameters to minimize a periodic performance cost. The use of spline functions with finite support and continuous derivatives provides a compact yet flexible representation that reconciles smoothness with local adaptability, overcoming the rigidity of globally defined harmonic signal classes. Theoretical analysis establishes stability of the closed loop and shows that the allocator preserves the nominal output behavior enforced by the baseline controller. Numerical simulations confirm that the spline-based dynamic allocator improves steady-state performance and actuator coordination compared with both static and harmonic allocation schemes, while handling actuator constraints in a more natural and computationally efficient manner.

Chapter 4: Spline-Based Input Allocation on Overactuated UAVs

The fourth chapter translates the spline-based allocation framework into the context of a complex, practically relevant system: a tilting and twisting overactuated UAV. This platform exemplifies the challenges of high redundancy, strong coupling, and dynamic constraints. The chapter introduces the UAV model, describing the actuator configuration, flight dynamics, and mission objectives. It then adapts the spline allocator to the UAV setting, embedding it within the hierarchical control structure and designing cost functions that account for both trajectory tracking and energy efficiency. The simulations presented in this chapter demonstrate that the spline-based allocator enables the UAV to execute aggressive maneuvers with improved actuator usage efficiency and robustness under saturation and partial faults. Comparisons with static and sinusoidal allocation methods highlight the superior performance of the spline-based approach in terms of adaptability, smoothness, and cost reduction, thereby validating its practical relevance.

Chapter 5: Sensitivity-Based Dynamic Control Allocation With An Academic Example

Chapter 5 generalizes the dynamic allocation concept to nonlinear plants, thereby addressing one of the central challenges identified in the state-of-the-art review. Classical nullspace-based approaches are not directly transferable to nonlinear settings, where the input-output mapping depends on the state and where superposition does not hold. To overcome this, the chapter introduces a sensitivity-based framework that characterizes how variations in actuator commands influence steady-state trajectories and periodic costs. Using nonlinear control theory, including uniform relative degree analysis and input-output normal forms, the allocation problem is reformulated to ensure output invisibility even in nonlinear plants. A periodic allocation strategy is combined with sensitivity equations to compute gradients of long-term cost functionals with respect to allocation parameters. Theoretical analysis demonstrates stability and convergence of the closed-loop scheme, and nonlinear simulations confirm its effectiveness in reducing actuator usage costs while preserving output performance. This chapter thus extends dynamic allocation into a regime that had previously remained largely unexplored.

Chapter 6: Dynamic Control Allocation for Nonlinear Systems via a Sensitivity Approach: Refinements and a Case Study

Chapter 6 applies the sensitivity-based nonlinear DCA methodology to a nonlinear mass-spring-damper system, where the stiffness is treated as a dynamically adjustable actuation channel, introducing nonlinear behavior through parameter-dependent dynamics. This setting provides a realistic and physically interpretable benchmark in which input redundancy originates from heterogeneous actuation capabilities (force input and variable stiffness). The chapter focuses on numerical implementation aspects, including gradient computation via sensitivity functions.

The results illustrate how the proposed allocator redistributes effort between the two actuators over repeated periods, significantly reducing the weighted quadratic cost associated with actuator usage, while rigorously maintaining the nominal tracking performance prescribed by the baseline controller. Unlike the academic example in Chapter 5, this case study highlights the method's ability to operate in a mechanical system with physically meaningful constraints and tunable energy exchange between masses. Overall, this chapter demonstrates the practical viability of the sensitivity-based DCA architecture and confirms its ability to generate performance gains in nonlinear, periodically-forced systems subject to realistic actuation structures.

Chapter 7: Conclusions and Future Work

The final chapter reflects on the journey of the thesis, revisiting the theoretical and practical contributions and assessing their impact on the field of DCA. It acknowledges the limitations of the current work and points toward avenues for future research. Among these, experimental validation on physical platforms is identified as a natural next step, as is the extension of the proposed frameworks to distributed control allocation in multi-agent systems. The integration of learning-based adaptation into the spline framework and the exploration of hybrid spline-sinusoid models for multimodal systems are also highlighted as promising directions. The chapter concludes by envisioning the continued evolution of DCA as a cornerstone of resilient and efficient autonomous system design, emphasizing its potential to meet the challenges posed by increasingly complex and uncertain environments.

Chapter 2

Chapter 2

Background and State of the Art

Modern engineered systems increasingly exploit input redundancy to achieve improved performance, flexibility, and robustness in actuation. Whenever the number of available actuators exceeds the number of independent control objectives, the system is said to be over-actuated or redundantly actuated [21], [33], [42], [48]. This situation arises in a wide variety of applications: aircraft equipped with multiple aerodynamic and thrust-vectoring effectors, underwater or surface marine vessels with several azimuth thrusters, robotic manipulators or mobile platforms with cooperative actuators, and modern automotive or aerial robots with distributed propulsion systems [20], [38], [47]. Input redundancy enables the designer to exploit multiple actuation paths to achieve the same generalized control objective while satisfying secondary goals such as energy efficiency, actuator wear minimization, or smooth transitions between configurations.

2.1 Conceptual Distinction Between Allocation Paradigms

As outlined in Chap. 1, control allocation serves as a module aimed at improving performance of a control system in accordance with a specified optimization objective. According to the literature, two fundamentally different paradigms of control allocation can be identified.

2.1.1 Direct Allocation for Actuation Mapping

The first paradigm concerns the *direct allocation* problem, which is typical in aerospace and robotic applications such as multirotor drones or marine vehicles. In this context, the goal of allocation is to distribute a desired generalized force or moment often referred to as a *wrench* among a redundant set of actuators. Here, the allocator is an integral part of the control structure: the high-level controller generates the desired wrench, and the allocator instantaneously transforms this virtual command into actuator inputs that satisfy physical and geometric constraints. There is no distinct “control” versus “allocation” layer in terms of time-scale separation; rather, the allocator directly realizes the control objective through actuator coordination. The system can be represented schematically as a cascade of controller, allocator, and plant P , where the allocator converts the desired wrench into the actual actuator commands u applied to the plant. The control architecture of such a system is shown in 2.1.

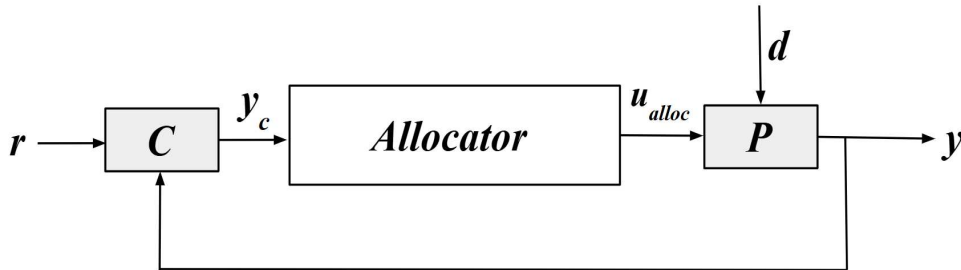


Figure 2.1: Direct Allocation for Actuation Mapping

2.1.2 Add-on Allocation for Performance Enhancement

The second paradigm, which underlies the work in [1], [3], [42], [43], [48] and many others, introduces the allocator as an *additional dynamic module* on top of an already functional control system. In this setting, the nominal controller alone is capable of stabilizing or regulating the plant; the allocator acts as a supplementary layer that modifies or redistributes the control signals to improve secondary performance objectives such as energy efficiency, smoothness, or actuator usage. The allocator typically receives the virtual control signal y_c (later denoted also v_c) produced by the controller, along with other plant measurements, and produces y_a which summed to y_c , is a feasible command u_{alloc} . Thus, unlike the first paradigm, where allocation is essential for actuation, here the allocator is a plug-in module that enhances the closed-loop behavior of an already designed control architecture.

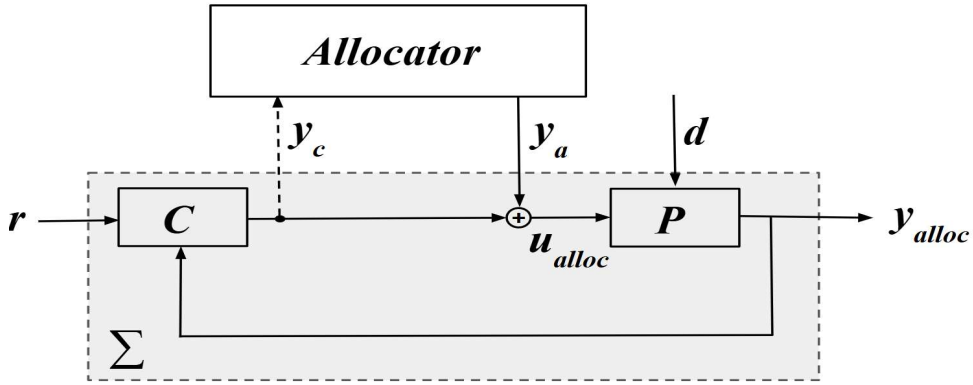


Figure 2.2: Add-on Allocation for Performance Enhancement.

These two paradigms correspond to fundamentally different design philosophies. The first direct wrench-to-actuator allocation is dominant in the robotics and aerospace literature, where the emphasis lies on converting desired forces and moments into actuator-level commands while respecting mechanical constraints. The second add-on dynamic allocation forms the basis of the theoretical developments by [1], [3], [42], [43], [48], as well as the methodologies developed in this thesis. In this framework, the controller and allocator operate in parallel: the former ensures closed-loop stability and reference tracking, while the latter optimizes the internal distribution of control effort without interfering with the primary control objective. This separation of functions enables modularity, allowing advanced allocation strategies to be designed and analyzed independently of the core control law.

2.2 Control Allocation in Redundant Systems

At a general level, consider a nonlinear dynamical system described by

$$\dot{x}(t) = f(x(t), u(t)), \quad y(t) = h(x(t)), \quad (2.1)$$

where $x \in \mathbb{R}^n$ denotes the state, $u \in \mathbb{R}^m$ the vector of actuator inputs, and $y \in \mathbb{R}^p$ the controlled output, with $m > p$. A higher-level control or guidance law generates a desired *virtual control* or *generalized input* $v_c(t) \in \mathbb{R}^p$ that represents the total control effort needed to achieve a regulation or tracking objective. The physical actuators produce generalized inputs through a *effectiveness map*

$$v(t) = G(x(t)) u(t), \quad (2.2)$$

where $G(x) \in \mathbb{R}^{p \times m}$ encodes how each actuator contributes to the generalized forces or moments. When $G(x)$ is constant and full-row rank, (2.2) reduces to the classical linear map $v = Bu$ used in many allocation studies [16], [26], but in general G may depend on configuration, aerodynamic conditions, or other states.

The control allocation problem consists of determining a feasible actuator command $u(t)$ such that

$$G(x(t)) u(t) \approx v_c(t), \quad (2.3)$$

subject to actuator limitations such as magnitude, rate, and sometimes energy constraints. Because $m > p$ and $\text{rank } G(x) = p$, the system (2.3) is underdetermined, admitting infinitely many solutions whenever the command v_c lies in the range of $G(x)$. Formally, the set of admissible actuator vectors is

$$\mathcal{U}(v_c) = \{ u \in \mathbb{R}^m \mid G(x)u = v_c \}, \quad (2.4)$$

which defines an affine subspace of dimension $m - p$. If actuator limits render v_c infeasible, one typically seeks the actuator vector that minimizes the deviation from the commanded generalized input according to some norm or cost functional.

A standard approach introduces a secondary performance index

$$J(u) = (u - u_{\text{ref}})^\top W (u - u_{\text{ref}}), \quad (2.5)$$

where $W > 0$ defines a weighting of actuator usage and u_{ref} may represent a nominal or preferred configuration. The *optimal allocation* is then given by the constrained optimization problem

$$\begin{aligned} u^*(t) &= \arg \min_u J(u) \\ \text{s.t. } &G(x)u = v_c, \\ &u_{\min} \leq u \leq u_{\max}. \end{aligned} \quad (2.6)$$

When no constraints are active and $G(x)$ is constant, the analytical solution is

$$u^* = G^+ v_c + (I - G^+ G) u_{\text{ref}}, \quad (2.7)$$

where G^+ denotes the Moore–Penrose pseudoinverse. Equation (2.7) yields the minimum-norm actuator vector that achieves the desired generalized input. This pseudo-inverse method became the cornerstone of early control allocation research in the aerospace community [16], [17], [26]. However, it neglects actuator saturation and rate limits, motivating the development of constrained approaches based on linear or quadratic programming [7], [33], [40]. Such formulations guarantee a unique optimal solution whenever the feasible set is convex and nonempty.

The redundancy structure of (2.3) can be characterized by decomposing the actuator space into two orthogonal components. Let $N(x) \in \mathbb{R}^{m \times (m-p)}$ be a full-column-rank matrix whose columns form a basis for the nullspace of $G(x)$, that is, $G(x)N(x) = 0$. Then every feasible actuator vector can be expressed as

$$u = G^+(x)v_c + N(x)z, \quad (2.8)$$

where $z \in \mathbb{R}^{m-p}$ parameterizes the redundant degrees of freedom. The first term produces the commanded generalized input, while the second redistributes control effort within the nullspace without affecting v_c . This decomposition underpins most modern allocation methods: static optimization algorithms choose z by minimizing a cost such as (2.5), whereas dynamic or adaptive allocators endow $z(t)$ with evolution laws that balance performance and smoothness over time [20], [21], [38], [48].

The benefits of input redundancy extend beyond mere feasibility. Redundant actuators can improve energy efficiency by exploiting favorable actuator operating regions, enhance agility by using multiple effectors simultaneously, and allow gradual transitions between different actuator combinations. In over-actuated aerial vehicles such as multirotors or tiltrotors, for instance, continuous redistribution of thrust enables stable operation even during partial saturation of some propellers [20].

Similarly, in the JET Tokamak plasma-shape controller studied by [47], multiple magnetic coils act redundantly to shape the plasma boundary; an appropriate allocator ensures both geometric precision and current-limit compliance. These examples highlight that the allocation mechanism is not merely a numerical solution to an algebraic equation but an integral component of the closed-loop control architecture.

While early formulations were primarily static computing u^* instantaneously from v_c , the dynamic nature of real actuators soon motivated *dynamic allocation* approaches, where internal allocator dynamics generate $u(t)$ in a continuous manner. Such methods can prevent discontinuities in actuator commands, accommodate actuator bandwidth limitations, and exploit the nullspace component $z(t)$ for smooth redistribution of effort. Dynamic allocation has been shown to reconcile optimality and stability even for complex multi-actuator systems [9], [21], [38], [48]. Nevertheless, at the conceptual level considered in this section, the focus remains on the structural essence of redundancy: an underdetermined mapping between actuator inputs and generalized control outputs. In Sec. 2.3 the static and dynamic control allocation has been discussed in details.

The concept of control allocation has also evolved alongside theoretical advances in optimal and geometric control. For instance, [21] formalized the notion of *input allocation-based regulation* for linear over-actuated systems, providing a rigorous link between the output regulation problem and control allocation. Extensions to nonlinear settings include inverse-model allocation strategies for over-actuated plants [42], [47], and optimality-based dynamic schemes for nonlinear redundant actuators [11], [38]. More recent efforts address robustness and safety through constraint handling and learning-based methods [14], [18], [44], demonstrating that allocation remains an active and expanding research frontier.

In summary, control allocation in redundant systems establishes a fundamental bridge between high-level control objectives and low-level actuation mechanisms. It formalizes the process of transforming a desired generalized input into feasible actuator commands when multiple solutions exist. The mathematical foundation provided by the mapping (2.2), the feasible set (2.4), and the nullspace decomposition (2.8) enables a unified treatment of both static and dynamic allocation techniques, as well as linear and nonlinear actuation structures. The subsequent sections of this chapter will detail these classes of methods, highlighting their theoretical formulations and relative merits.

2.3 Static vs. Dynamic Control Allocation

2.3.1 Static Control Allocation Methods

Static control allocation refers to the instantaneous computation of actuator commands $u(t)$ at each control update, based solely on the current value of the virtual control $v_c(t)$ and on actuator constraints, without introducing additional dynamic states or memory in the allocation mechanism. In other words, the mapping from the desired generalized input to actuator signals is algebraic rather than differential, implying that at every time instant the allocator computes the solution of an optimization or feasibility problem using only instantaneous information. This assumption considerably simplifies analysis and implementation, which explains the popularity of static allocation methods in real-time flight control, marine guidance, and robotic actuation systems [7], [20], [26], [33].

A generic static control allocation problem can be expressed as

$$\min_{u \in \mathbb{R}^m} J(u) \quad (2.9)$$

$$\begin{aligned} \text{subject to } & Gu = v_c, \\ & u_{\min} \leq u \leq u_{\max}, \end{aligned} \quad (2.10)$$

where $G \in \mathbb{R}^{p \times m}$ is the control effectiveness matrix, $v_c \in \mathbb{R}^p$ is the commanded generalized control input, and $u_{\min}, u_{\max}$ are actuator magnitude limits. The cost function $J(u)$ typically reflects actuator effort or energy expenditure.

Matrix G can also be interpreted as in the following plant definition where B_0 is a full-column-rank matrix and $\text{rank}(B) < m$:

$$\dot{x}(t) = Ax(t) + Bu(t) = Ax(t) + B_0Gu(t), \quad y(t) = Cx(t), \quad (2.11)$$

Moreover, $\ker(B)$ acts as an ANN (Annihilator) that guarantees the *output invisibility*, which means that the control system produces the same output without or with the presence of an allocator. In Chap. 3 and Chap. 4, the advantages and the design of ANN have been explained in detail.

A common and computationally convenient choice is a quadratic cost

$$J(u) = \frac{1}{2}(u - u_{\text{ref}})^\top W(u - u_{\text{ref}}), \quad (2.12)$$

where $W > 0$ is a diagonal weighting matrix that can encode actuator preference or efficiency, and u_{ref} is a reference configuration such as a trim point or previously used allocation. For simplicity, the minimum-norm case $u_{\text{ref}} = 0$ and $W = I$ leads to $J(u) = \|u\|_2^2$, but many practical systems use non-uniform weights to favor specific actuators [26], [33].

The equality constraint (2.10) enforces that the total control effect produced by the actuators equals the desired virtual control v_c . This ensures that, as long as v_c lies in the column space of G , the system will realize the requested generalized input exactly, up to numerical precision. The box constraints model actuator saturation limits, which are unavoidable in real systems due to physical, electrical, or mechanical restrictions. These constraints make the feasible set convex but bounded, resulting in a well-defined QP (quadratic programming) problem when J is quadratic.

2.3.2 Unconstrained SCA

If actuator limits are not active and G has full row rank ($\text{rank}(G) = p$), the optimization problem (2.9)–(2.10) admits the analytical solution

$$u^* = G^\dagger v_c = G^\top (GG^\top)^{-1} v_c, \quad (2.13)$$

where $G^\dagger$ denotes the Moore–Penrose pseudoinverse. This solution yields the minimum 2-norm actuator command satisfying $Gu = v_c$, which is equivalent to minimizing the total control effort under perfect tracking of v_c . The pseudoinverse formulation first appeared in the context of reconfigurable flight control and redundant effector systems [16], [17] and remains a benchmark reference point for control allocation studies. However, because it assumes linearity and neglects actuator limits, it can produce physically infeasible commands when any actuator saturates.

2.3.3 Constrained SCA

To handle saturation and rate constraints, researchers extended the basic pseudoinverse approach to constrained optimization problems. [7] proposed one of the first systematic formulations for *constrained control allocation*, casting the problem as a quadratic program that minimizes actuator effort while respecting both equality and bound constraints. Because the feasible set defined by these constraints is convex, QP algorithms can efficiently compute the global optimum even in real time. Later works by [33] and [26] provided comprehensive analyses of these QP-based methods, highlighting their geometric interpretation and computational efficiency.

An important variant of static allocation is the *fixed-point iteration* or *sequential saturation* algorithm, originally used in aircraft control systems. When an unconstrained solution violates actuator bounds, the algorithm fixes the saturated actuators at their limits, removes the corresponding columns from G , and recomputes the allocation for the remaining actuators until feasibility is achieved [6], [7], [33]. This approach is deterministic, easy to implement, and has predictable convergence properties, which made it attractive for embedded flight control systems before efficient QP solvers became common.

2.3.4 SCA Practical Properties and Limitations

SCA methods have several favorable properties. They are conceptually simple, rely on well-established numerical tools, and can be executed at high frequency with limited computational burden. When the effectiveness matrix G varies slowly compared with the control bandwidth, recomputing the static allocation at each sampling instant provides satisfactory performance and robustness. Such features have made static allocators the standard approach in many industrial systems, including underwater vehicles and over-actuated robotic platforms [20]. For example, in the ROSPO (ROtor graSPing Omnidirectional) aerial platform analyzed in [20], a static QP allocator distributes the thrust commands among six propellers while maintaining precise force and torque generation in the presence of actuator saturation.

Despite these advantages, static control allocation inherently ignores the temporal and dynamical aspects of actuator behavior. Real actuators possess bandwidth limitations, rate constraints, and nonlinear dynamics that cannot react instantaneously to changing commands. As a result, purely static allocation can produce discontinuous or rapidly varying actuator commands, particularly when the active set of constraints changes between time steps. Such chattering behavior not only degrades performance but may

also excite unmodeled dynamics or accelerate actuator wear. Furthermore, static methods cannot anticipate future constraint activations or exploit the temporal correlation of successive commands to improve smoothness and optimality. These drawbacks have motivated the development of DCA strategies, in which the allocator is endowed with its own dynamics to ensure continuous evolution of $u(t)$, smoother constraint handling, and explicit modeling of actuator limits [21], [48].

In summary, static control allocation provides the foundational framework upon which most modern allocation schemes are built. Its mathematical clarity and computational efficiency make it the preferred choice for many practical applications. However, the instantaneous and memoryless nature of static allocation constitutes its main limitation, justifying the introduction of dynamic methods that better capture the physics of real actuators and ensure improved transient performance.

2.3.5 Dynamic Control Allocation Methods

Dynamic Control Allocation (DCA) extends the static framework by introducing internal dynamics within the allocation process. Instead of computing actuator commands $u(t)$ instantaneously from the current virtual control $v_c(t)$, a dynamic allocator evolves over time according to its own differential equations. This dynamic behavior allows the allocator to handle actuator dynamics, rate constraints, and performance trade-offs in a continuous manner, thereby mitigating discontinuities and chattering effects that often arise in static approaches [9], [21], [38], [47], [48].

A simple yet effective structure is based on gradient or descent flows that asymptotically minimize J over time:

$$\dot{v}(t) = -K_v \nabla_v J(u, v, t), \quad (2.14)$$

where J is a time-varying cost function encoding secondary objectives such as energy, smoothness, or usage balance. $K_v > 0$ is a positive definite gain or step-size matrix. Under appropriate convexity and regularity assumptions on J , this continuous-time gradient flow ensures that $v(t)$ converges to the minimizer of J , and consequently $u(t)$ converges to the optimal allocation $u^*(t)$ associated with the current v_c [9], [47], [48]. The allocator thereby acts as a low-pass filter in the space of feasible allocations, smoothing rapid changes in v_c and naturally incorporating actuator limits and rate constraints.

A geometric interpretation of DCA can be obtained by decomposing the control input u as in (2.8), namely

$$u = G^+ v + N z, \quad (2.15)$$

where N spans the nullspace of G . The variable $z(t) \in \mathbb{R}^{m-p}$ represents the redundant degrees of freedom. m represents the number of physical actuator inputs and p represents the number of independent control outputs or objectives. While static allocators compute z^* instantaneously (e.g., by solving a QP), DCA assigns dynamic evolution laws to $z(t)$ of the form

$$\dot{z}(t) = -K_z \nabla_z J(G^+ v + N z(t)), \quad (2.16)$$

so that the actuator distribution gradually approaches the optimal steady-state solution. This dynamic law can be shown to guarantee asymptotic convergence to the minimum of J within the feasible set, while preserving the constraint $Gu = v$ for all time [9],

[38], [48]. An appealing feature of this design is that the allocator introduces a tunable time constant that governs how quickly the control effort redistributes among actuators, offering a direct trade-off between optimality and smoothness.

2.3.6 DCA Theoretical Properties and Advantages

The DCA framework exhibits several valuable theoretical properties and offers distinct advantages over its static counterpart:

- **Smooth transitions and rate-limited actuators:** Because $u(t)$ evolves continuously through the allocator dynamics, the resulting actuator commands are smooth by construction. [48] showed that the dynamic evolution can be tuned such that the generalized output $v = Gu$ remains constant during redistribution of actuator effort, achieving what he termed “output-invisible reallocation.” This property is crucial in applications such as precision flight control and magnetic plasma shaping, where abrupt changes in u can excite undesirable transients [47].
- **Closed-loop stability:** The allocator can be designed using Lyapunov arguments to ensure that the augmented plant-allocator system remains globally asymptotically stable. Under mild assumptions on convexity of J and full-rankness of G , the equilibrium (x^*, u^*, v^*) is globally stable and the allocation error $e_v = v_c - Gu$ converges to zero. This was rigorously proven for several classes of linear and mildly nonlinear systems by [21], [48].
- **Incorporation of secondary performance objectives:** The nullspace component z in (2.15) allows multiple criteria such as minimizing energy [4], [2], and [15], balancing actuator usage [5], increasing reliability and fault tolerance [12] or penalizing fast variations to be optimized concurrently. Works such as [38] and [3] introduced optimality-based dynamic laws that minimize a time-integrated quadratic cost while guaranteeing constraint satisfaction.
- **Compatibility with constrained optimization:** The gradient-based structure can be combined with projection or saturation operators to enforce actuator limits, leading to continuous constrained dynamics rather than abrupt switching between saturated and unsaturated regimes [9].

Overall, DCA represents a natural bridge between continuous-time optimization and control theory. By embedding an optimization process directly into the feedback loop, it achieves smooth, adaptive allocation behavior that aligns with both control and actuation objectives.

2.4 Dynamic Control Allocation for Linear Systems

When the plant can be adequately modeled as linear and time-invariant, dynamic control allocation admits a more explicit analytical treatment using standard linear systems theory. A linear over-actuated plant is typically described by (2.11) ensures asymptotic tracking of a reference $r(t)$ or regulation of $y(t)$ to zero while simultaneously optimizing a secondary performance index on the actuator effort. For instance, while optimizing actuator usage through internal allocator dynamics.

The set of steady-state input vectors capable of generating a desired virtual control v_{ss} (or steady-state output y_{ss}) can be expressed as

$$u_{ss} = G^\dagger v_{ss} + Nz, \quad (2.17)$$

where G is the control effectiveness matrix (constant in the linear case) and N spans the nullspace of G . The free variable z parametrizes all possible actuator redistributions that produce the same generalized input v_{ss} . Dynamic allocation laws therefore focus on prescribing evolution dynamics for $z(t)$ that drive the system toward the optimal steady-state actuator configuration, while maintaining $Gu(t) = v_{ss}$ at all times.

Moreover, here in DCA, the TF (Transfer Function) of the component that acts as an annihilator is $W_{ann}(s) = N^\perp(s)(d_{ann}(s))^{-1}$ where the TF of the plant is $P(s) = (d(s))^{-1}N(s)$ and $N^\perp(s) = \text{null}(N(s))$.

A simple and widely studied dynamic allocation law is the gradient flow

$$\dot{z}(t) = -K_z \nabla_z J(u_{ss}(z)), \quad (2.18)$$

where J is a convex performance index. Typical choices include

$$J(u) = \begin{cases} \|u\|_1, & (\text{sparsity or actuator minimization}) \\ \|u\|_2^2, & (\text{energy or effort minimization}) \\ \|u\|_\infty, & (\text{amplitude minimization}). \end{cases} \quad (2.19)$$

Each norm emphasizes different allocation priorities. The quadratic cost yields smooth distributions and analytic convergence results, while L_1 and L_∞ norms lead to piecewise-linear or saturating behaviors useful for actuator selection and amplitude control, respectively [9], [38].

Within the output regulation framework, the combined plant–allocator system can be viewed as an augmented system whose steady-state regulation equations include the allocator states. [21] demonstrated that dynamic allocation can be seamlessly incorporated into the regulator design by embedding the allocator dynamics in the internal model structure. Specifically, they proved that if a standard regulator achieves asymptotic tracking for a fat plant (more inputs than outputs), then an augmented regulator with dynamic allocation can achieve the same regulation for the over-actuated case while additionally optimizing actuator performance. This theoretical result bridges classical output regulation and control allocation theory.

[9] further extended this approach by formulating the DCA problem as a dynamic optimization scheme, where the allocator implements a real-time gradient flow that minimizes an instantaneous quadratic cost subject to the constraint $Gu = v_c$. They derived explicit convergence conditions and proved structural stability, robustness, and uniqueness of the steady-state solution. Similarly, [47] applied a variant this framework to plasma shape control in the JET Tokamak, demonstrating the practical viability of DCA in large-scale experimental systems.

Related developments have also emerged in the area of optimal steady-state regulation for linear systems formulated a steady-state regulation problem that combines state feedback with an internal-model structure to guarantee convergence to an optimal operating point while minimizing a quadratic cost on the steady-state input [25]. Although their formulation addresses regulation rather than allocation, it is conceptually aligned with the objectives of DCA: both seek, among the many admissible input configurations, the one that optimizes a given performance criterion without degrading nominal regulation.

Consequently, optimal steady-state regulation can be regarded as a complementary viewpoint to linear DCA in over-actuated systems.

Beyond the instantaneous formulations of DCA, recent works have emphasized the relevance of *periodic allocation* strategies for systems subject to cyclic or steady-state periodic operation. These approaches recognize that, when exogenous signals or references are periodic, optimal steady-state performance cannot be captured by point-wise optimization alone, but must instead be evaluated over an entire fundamental period.

In this direction, [3] introduced a spline-based dynamic input allocation framework in which the allocation signals are parameterized as periodic extensions of polynomial functions, allowing the use of a hybrid internal model that explicitly encodes periodicity. This formulation unifies and extends previous sinusoidal approaches by providing a more expressive yet computationally tractable structure for periodic DCA. The concept was subsequently implemented and validated on a nonlinear overactuated UAV platform in [4], confirming the method's ability to optimize long-term Lagrangian costs while maintaining output invisibility. Furthermore, the sensitivity-based extension proposed in [1] generalized periodic DCA to nonlinear systems, enabling gradient-based optimization of steady-state trajectories through sensitivity dynamics and ensuring convergence to optimal periodic allocations. Collectively, these studies demonstrate that periodic DCA frameworks can systematically exploit temporal redundancy to enhance performance, reduce actuator effort, and preserve closed-loop output behavior, thereby bridging steady-state optimization and control allocation theory. The spline-based formulation is presented in detail in Chap. 3, its UAV implementation in Chap. 4, and the sensitivity-based periodic extension together with its case study are discussed in Chapters 5 and 6, respectively.

In [24], the OSSR (Optimal Steady-State Regulation) problem for linear systems was formulated and solved using an internal model and state feedback operating in tandem. The problem extends the classical output regulation framework by introducing a quadratic steady-state cost on the effort of individual control modules specifically, an adaptive internal model and a state feedback rather than on the plant inputs or outputs. The authors derived both instantaneous and time-averaged (over a period) steady-state optimization formulations, expressed as convex least-squares problems in the feedback gain, and characterized their solvability through the regulator equations. An adaptive control architecture was then developed to minimize the long-term internal-model workload when full offloading by the state feedback is not possible. The proposed adaptive regulator employs gradient-based adaptation laws with two-timescale averaging analysis to ensure convergence and stability, providing an implementable solution for optimal steady-state operation under unknown plant and exosystem parameters.

Dynamic control allocation for linear systems offers a rigorous, flexible, and physically meaningful framework that extends static methods by incorporating dynamics and performance criteria into the allocation layer. It provides analytical tractability, proven stability, and adaptability to actuator constraints, forming a solid foundation for subsequent extensions to nonlinear systems discussed in Sec. 2.5.

2.5 Dynamic Control Allocation for Nonlinear Systems

While DCA has been extensively investigated in the context of linear time-invariant systems, extending these methods to nonlinear dynamics poses fundamental conceptual and analytical challenges. Nonlinear plants are pervasive in robotics, aerospace, and mechanical systems, where input redundancies coexist with nonlinear couplings and parameter-dependent behaviors. The goal of nonlinear DCA is to preserve the appealing modular structure of the linear case, where allocation acts as an auxiliary

layer that optimizes actuator usage without altering closed-loop stability, while generalizing it to a wider class of systems that exhibit nonlinear dynamics and periodic operation.

2.5.1 Formulation for Nonlinear Systems

Consider a plant P with m inputs and p outputs, which is *input redundant* (namely, $m > p$), has *uniform vector relative degree* [31, Sec. 11.5] denoted as $r = [r_1 \ r_2 \ \dots \ r_p]'$ and is in *normal form* [30, (5.7)-(5.8)]; then, P is described by the following equations:

$$\begin{aligned}\dot{x} &= A_0 x + B_0 (f_0(x, z) + g_0(x, z)u), \\ \dot{z} &= f_1(x, z) + g_1(x, z)u, \\ y &= C_0 x,\end{aligned}\tag{2.20}$$

where $x \in \mathbb{R}^n$ denotes the measurable state, $z \in \mathbb{R}^k$ the internal or unmeasured dynamics, $u \in \mathbb{R}^m$ the input vector with $m > p$, and $y \in \mathbb{R}^p$ the output. The mappings f_0, g_0, f_1, g_1 are assumed to be smooth and satisfy $\text{rank}(g_0(x, z)) = p$ globally, ensuring that the system possesses a uniform vector relative degree $r = [r_1, \dots, r_p]^\top$. This assumption enables a simplified normal form representation and well-defined pseudo-inverse and kernel basis matrices $g_0^\#(x, z)$ and $g_0^\perp(x, z)$, respectively.

A stabilizing controller C is assumed to exist, guaranteeing asymptotic tracking of a smooth periodic reference $y_d(t)$ with period $T > 0$. The baseline control law provides a nominal input $y_c(x, z, Y_d^{[r]})$ which, when applied to the plant, ensures that $y(t) \rightarrow y_d(t)$ exponentially. The h -th scalar component $y_{d,h}$ of y_d is r_h times continuously differentiable, for $h = 1, \dots, p$. As usual in tracking problems, the derivatives of $y_{d,h}^{(k)}$ for $k = 1, \dots, r_h$ are assumed to be available for control purposes; for brevity, the following shorthand notations will be used,

$$\begin{aligned}y_d^{[r]} &:= \begin{bmatrix} y_{d,1}^{(r_1)} & y_{d,2}^{(r_2)} & \dots & y_{d,p}^{(r_p)} \end{bmatrix}', \\ Y_{d,h}^{(v_h)} &:= \begin{bmatrix} y_{d,h} & y_{d,h}^{(1)} & \dots & y_{d,h}^{(v_h)} \end{bmatrix}', h = 1, \dots, p, \\ Y_d^{[v]} &:= \begin{bmatrix} (Y_{d,1}^{(v_1)})' & \dots & (Y_{d,p}^{(v_p)})' \end{bmatrix}',\end{aligned}$$

where either $(v, v_h) = (r, r_h)$ or $(v, v_h) = (r^-, r_h^-)$, with $r^- := [r_1 - 1 \ r_2 - 1 \ \dots \ r_p - 1]'$ and $r_h^- := r_h - 1$, for $h = 1, \dots, p$.

The DCA mechanism then augments this nominal controller by introducing an additional allocation signal u_{al} :

$$u = y_c + u_{al},\tag{2.22}$$

where $u_{al} \in \text{Im}(g_0^\perp(x, z))$ so as to preserve output invisibility under suitable hypothesis. The augmented closed-loop system, denoted Σ_{all} , thus inherits the stability properties of the nominal closed loop Σ while gaining additional degrees of freedom for optimization within the redundant actuation space.

In Chap. 5 and Chap. 6 the nonlinear allocation has been studied in detail while an innovative method has been introduced and has been verified on both academic and physical case studies.

2.5.2 Advantages and Applications of DCA

The Dynamic Control Allocation (DCA) framework offers several distinct advantages over static allocation methods, making it particularly suitable for complex over-actuated systems:

- **Output Invisibility:** A fundamental property of DCA is that the allocation action is designed to operate strictly within the kernel of the input–output map. This ensures that the redistribution of control effort among redundant actuators does not perturb the primary regulated output, maintaining the performance of the high-level control loop.
- **Exploitation of System Dynamics:** Unlike static methods that solve for an instantaneous distribution, DCA accounts for the temporal evolution of the system. This allows for the optimization of performance metrics over time, such as minimizing energy consumption or managing actuator fatigue, while accounting for the bandwidth and constraints of the physical hardware.
- **Modularity and Retrofitting:** The DCA architecture is inherently decoupled from the baseline control design. This modularity allows it to be integrated into existing closed-loop systems without requiring a redesign of the primary tracking controller, provided the closed loop maintains basic stability properties.

The versatility and efficacy of these properties are demonstrated throughout this thesis across various over-actuated benchmarks. For instance, the framework is applied to handle the complex redundancy of an *Unmanned Aerial Vehicle (UAV)* in Chap. 4, and its ability to manage nonlinearities is explored through a *Mass-Spring-Damper (MSD)* system with variable stiffness in Chap. 6. In these applications, DCA facilitates an efficient redistribution of control effort while ensuring that the allocation signals remain "invisible" to the tracking objectives, thereby preserving overall system stability and performance.

2.5.3 Perspectives and Research Directions

The sensitivity-based DCA framework opens promising directions for further research. Extensions to fault-tolerant allocation can be achieved by embedding actuator health parameters within the sensitivity model, enabling real-time reallocation in the presence of partial failures. Another perspective involves coupling the nonlinear allocator with higher-level adaptive or learning modules that tune cost weights or update sensitivity dynamics online. Finally, combining spline-based periodic parameterizations with sensitivity-based gradient laws may yield hybrid allocators that merge the local flexibility of spline signals with the analytical rigor of gradient-based optimization.

In summary, Dynamic Control Allocation for nonlinear systems establishes a systematic methodology for exploiting redundancy in complex, nonlinear, and periodically driven systems. By coupling gradient optimization with sensitivity dynamics, it bridges the gap between linear allocation theory and real-world nonlinear control problems, paving the way toward scalable, adaptive, and energy-efficient control architectures.

Chapter 3

Chapter 3

Dynamic Control Allocation Using Periodic Spline Signals

Abstract

In this chapter, we introduce and analyze a novel framework for dynamic input allocation for systems characterized by input redundancy. More precisely, we focus on closed-loop configurations in which the plant possesses a greater number of control inputs than controlled outputs, a situation frequently encountered in modern engineering applications such as aerospace systems, robotics, and advanced manufacturing. As we explained in the previous chapter, within this setting, the role of the allocator is to suitably modify the plant's control inputs through the injection of a carefully designed allocation signal. The defining feature of this approach is that the allocator guarantees invisibility of the plant's output trajectory with respect to the original closed-loop design, while at the same time shaping the steady-state behavior of the control inputs in order to achieve optimality with respect to a predefined Lagrange cost functional. This cost encodes performance and efficiency considerations, such as minimization of actuator effort, balancing of energy consumption, or distribution of wear across actuators, thereby offering a principled means of exploiting input redundancy for performance enhancement.

The methodology presented here builds upon and significantly extends prior research in the field. In particular, earlier approaches have typically relied on harmonic allocation signals, which, although analytically tractable, exhibit inherent limitations in terms of flexibility and adaptability to complex system requirements. The innovation of the present contribution lies in the replacement of these purely harmonic components with periodic piecewise polynomial signals. Such signals, when parameterized in the form of splines, offer a richer representational capability and allow for more refined tuning of allocation dynamics. The adoption of spline-based allocation therefore equips the designer with a versatile tool that can capture a wider range of signals and can be systematically adapted to address specific structural properties of the plant.

3.1 Introduction

This chapter introduces the first major original contribution of the thesis: a spline-based framework for DCA. While Chap. 2 provided the necessary background, highlighted existing results, and formalized the general problem of allocation in redundant control systems, the focus now shifts to the design of a concrete allocator that can overcome the limitations of existing methods. In the literature, allocation signals have typically been restricted to constant vectors or harmonic functions, which limits their ability to exploit redundancy in the presence of more complex reference signals. By contrast, this chapter adopts polynomial and spline-type trajectories as the internal model for allocation, significantly broadening the admissible signal set and enhancing flexibility.

The structure of the chapter is as follows: Sec. 3.2 formulates the allocation problem in precise mathematical terms. Sec. 3.3 presents the spline-based allocator architecture, highlighting how the internal model generator, the annihilator, and the gradient computation block interact to preserve output invisibility while optimizing actuator usage. Sec. 3.4 provides the main theoretical results, showing that all key requirements are satisfied. Sec. 3.5 offers numerical evidence on benchmark references, and Sec. 3.6 concludes the

chapter with a discussion of implications and possible extensions.

Overall, Chap. 3 bridges the general background of Chap. 2 with the first concrete contribution of the thesis. It sets the stage for moving from theoretical formulation to practical application: while the spline-based allocator is presented here in an abstract setting, the next chapter demonstrates how this methodology can be integrated into the control of a complex aerial vehicle with tilting-twisting propellers.

3.2 Model, Assumptions, and Objective

This chapter is devoted to the systematic study of DCA, which can be formally interpreted as a specific instance of *controller augmentation* (see Fig. 3.1). In essence, DCA enhances a pre-existing closed-loop control architecture by embedding an additional allocation mechanism designed to operate on the redundant input channels of the plant. From an intuitive perspective, the fundamental purpose of control allocation in this context is to deliberately exploit such redundancy in order to achieve supplementary performance objectives. These additional objectives may include, for example, the reduction of overall actuator effort, the preferential utilization of actuators with lower energy costs, or the extension of actuator lifespan by balancing usage. Crucially, all these benefits are obtained while preserving the nominal closed-loop input–output behavior of the plant as enforced by the original stabilizing controller. In this way, allocation provides a principled method to improve efficiency, robustness, and flexibility without compromising the primary regulation or tracking performance.

In alignment with the seminal contributions [10], [22], [48], the problem setting considered in this chapter is formally defined by three core components: the plant P , representing the dynamical system to be controlled; the stabilizing controller C , which is assumed to ensure closed-loop stability and desired regulation properties in the absence of allocation; and the exogenous signal generator E , responsible for producing both reference signals to be tracked and disturbance inputs to be rejected. This triplet P, C, E captures the essential structure of the problem and serves as the starting point for analyzing the effects of introducing a dynamic allocation mechanism. While the framework can be generalized to broader classes of systems, our analysis will primarily focus on the Linear Time Invariant (LTI) case, which already encompasses a wide range of practically relevant scenarios and allows for a clear and rigorous characterization of the theoretical properties involved.

Within this LTI setting, the exosystem E generates an exogenous signal $w \in \mathbb{R}^{n_w}$, which simultaneously carries reference trajectories for the controller and disturbance signals acting on the plant. The mathematical structure of E is expressed as

$$\dot{w} = Sw \quad (3.1)$$

where $w \in \mathbb{R}^{n_w}$ denotes the internal state of the exosystem and $S \in \mathbb{R}^{n_w \times n_w}$ is the exosystem matrix. The following structural assumption is imposed.

Assumption 3.2.1

The exosystem E generates a T -periodic reference. □

Remark 3.2.1

If the exosystem is given by (3.1), Assumption 3.2.1 is guaranteed by asking that matrix $S \in \mathbb{R}^{n_w \times n_w}$ in (3.1) is diagonalizable, with all eigenvalues lying exactly on the imaginary axis, and such that they are integer multiples of a fundamental frequency ω_0 . Clearly, 0 can be seen as a (trivial) multiple of any value ω_0 . Hence, such an exosystem can generate any signal which is periodic of period $T = 2\pi/\omega_0$, possibly with a constant bias, since the natural modes of (3.1) are the constant function (if $0 \in \sigma(S)$) and any pair of functions $(\sin(h\omega_0 t), \cos(h\omega_0 t))$ with $h \in \mathbb{N} > 0$ and $\pm h\omega_0 \in \sigma(S)$.

Furthermore, note that in output regulation it is crucial to have the exosystem in form (3.1) to be able to replicate its eigenvalues in the internal model, in the present context the internal model only needs to produce a suitable class of T -periodic signals, and then simply Assumption 3.2.1 is needed¹. By adopting a spline-based internal model, the allocator gains the flexibility to approximate optimal periodic trajectories even for references that might be structurally incompatible with a classical linear exosystem. $\square$

The plant P has the following form²:

$$\dot{x} = Ax + Bu + Pw \quad (3.2a)$$

$$y = Cx + Qw \quad (3.2b)$$

where $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $y \in \mathbb{R}^{n_y}$. Considering that the goal of DCA is to exploit input redundancy to achieve some performance improvement, the following assumption is considered.

Assumption 3.2.2

Plant (3.2) is input redundant, namely, $n_u > n_y$. $\square$

Finally, let us turn our attention to the controller C , which is assumed to be pre-designed and available for direct use in connection with the plant in (3.2). This controller, whose structure has been fixed prior to the introduction of any allocation mechanism, is described by the following state-space realization:

$$\dot{x}_c = A_c x_c + B_c y + P_c w \quad (3.3a)$$

$$y_c = C_c x_c + D_c y + Q_c w \quad (3.3b)$$

where $x_c \in \mathbb{R}^{n_{x_c}}$ denotes the vector of internal controller states, and $y_c \in \mathbb{R}^{n_{y_c}}$ represents the output of the controller, which is ultimately connected to the plant as its input signal. This formal representation captures the dynamic processing carried out by the controller on the available measurements y and the exogenous signals w , and determines the control action delivered to the plant.

¹The author would like to thank the thesis reviewer, Prof. Andrea Cristofaro, for suggesting the clarification of Remark 3.2.1 and Assumption 3.2.1, in particular for stressing that no knowledge of the exosystem eigenvalues is used in the design of the allocator.

²With an abuse of notation, throughout the thesis, the same symbols P , C are used to denote the plant P and controller C as well as the matrices in (3.2); the context clarifies which sense is used.

In the present framework, controller C is assumed to be designed in such a way that, when directly interconnected with the plant P (without any additional allocation signal), it already guarantees the fundamental closed-loop properties of stability and regulation. Thus, the essential control objectives of disturbance rejection and reference tracking are satisfied independently of the allocation unit. The introduction of the allocation mechanism therefore serves a complementary role: its task is not to establish or restore closed-loop stability, which is already ensured by C , but rather to exploit the inherent input redundancy of P to enhance steady-state input trajectories according to secondary performance criteria. These criteria may include, for example, reducing the amplitude of control signals, distributing effort across actuators to avoid saturation, or minimizing cost functions associated with energy consumption. To formalize this setup, the following baseline assumption is imposed on the controller–plant interconnection.

Assumption 3.2.3

The closed-loop interconnection Σ (without allocation) illustrated in Fig. 3.2, obtained by combining (3.2) and (3.3) under the interconnection $u = y_c$, is well-posed and asymptotically stable. $\square$

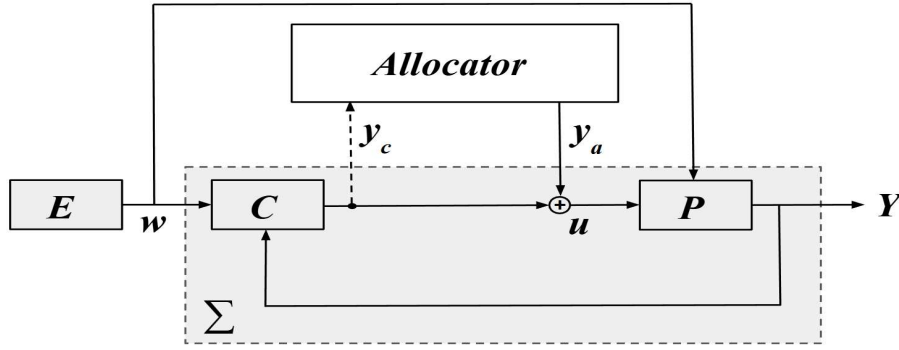


Figure 3.1: General Closed-Loop Control System with Allocation.

Remark 3.2.2

Assumption 3.2.3 ensures that the interconnection of P and C gives rise to a closed-loop system that is both mathematically well-defined and dynamically well-behaved. The condition of asymptotic stability guarantees that trajectories of the interconnected system converge to a steady state, thereby providing a solid foundation upon which allocation can be introduced. Importantly, this property also implies that the plant P in (3.2) is detectable and stabilizable with respect to the signals of interest, which eliminates the need to separately require these conditions in Assumption 3.2.2.

It is also worth emphasizing that the definition of the exogenous signal w has been formulated with no loss of generality to include both reference inputs and disturbances. From a modeling standpoint, references typically do not act directly on the plant but instead shape the controller's objectives, while disturbances may affect the plant but not necessarily be available for measurement by the controller. Both of these cases are naturally captured by allowing zero columns in the matrices P , Q , P_c , and Q_c , which describe the interconnections of w with the plant and controller. This modeling choice therefore preserves generality while simplifying the exposition. $\square$

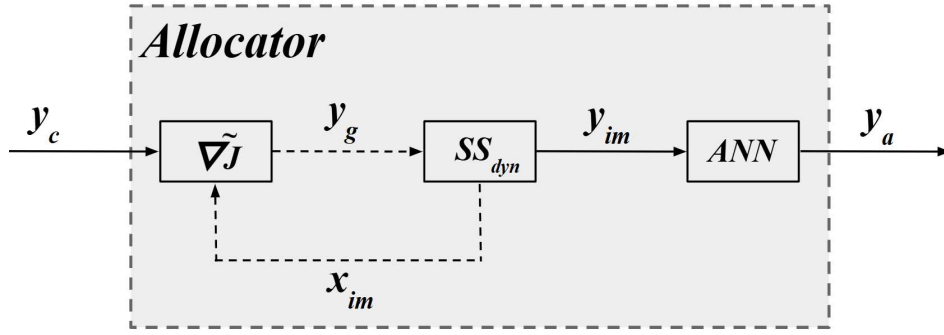


Figure 3.2: Architecture of a DCA unit (cfr. [10], [22]). Dashed arrows denote updates occurring at times $t = kT$, $t \in \mathbb{N}$.

Since all dynamical components involved in the present framework are modeled as Linear Time Invariant (LTI) systems, the asymptotic stability property postulated in Assumption 3.2.3 is, in fact, equivalent to the stronger requirement of global exponential stability. This equivalence is particularly relevant because it rules out slow convergence phenomena or marginal stability and instead guarantees uniform exponential decay of all trajectories toward the steady state. In this sense, the controller (3.3) does not merely ensure closed-loop stability in the weak Lyapunov sense, but provides robust exponential convergence rates that are independent of initial conditions. Such a property constitutes the minimal and essential prerequisite for regarding C as a meaningful and effective stabilizer of system (3.2). Without this guarantee, any further attempt to exploit redundancy through allocation would risk being undermined by an unstable or poorly conditioned closed-loop foundation.

In addition, when Assumption 3.2.3 is combined with the structural properties of the exosystem described in Assumption 3.2.1, a stronger conclusion can be drawn regarding the steady-state behavior of the interconnected closed-loop system Σ . Specifically, the interconnection formed by plant (3.2), controller (3.3), and exosystem w generated by (3.1) admits a unique and well-defined steady-state solution that is not only stable but also periodic. The period of this steady state is given by $T = 2\pi/\omega_0$, which directly corresponds to the fundamental frequency of the exosystem. Furthermore, the spectral properties of this periodic solution are precisely characterized by the same modes as those produced by (3.1), thereby ensuring a direct correspondence between the structure of exogenous signals and the asymptotic closed-loop response. It is worth noting that the existence of such a unique steady state does not rely in any critical way on the linearity or finite dimensionality of (3.1); similar conclusions hold in more general contexts as long as the exosystem satisfies appropriate periodicity conditions. Indeed, Sec. 3.5 illustrates this fact by presenting a numerical example where the exosystem extends beyond the finite-dimensional linear case, yet the closed-loop still converges to a periodic regime consistent with the theory.

At this point, it is essential to recall that the plant under consideration is input redundant, which implies that an infinite family of admissible input trajectories can produce exactly the same closed-loop output behavior. While the stabilizing controller C guarantees that, for any initial condition of the exosystem E , the output y of the plant (and equivalently the output y_c of the controller) converges to a unique T -periodic steady state, the plant input u corresponding to this output is by no means uniquely determined. This fundamental non-uniqueness motivates the introduction of an additional allocation signal y_a , which modifies the plant input as $u = y_c + y_a$ (see Fig. 3.2). The central idea is to design y_a so that it leaves the output steady state completely unaffected while reshaping the input trajectory according to secondary performance objectives. Such objectives may involve minimizing a Lagrangian cost over one period, redistributing actuator usage to avoid saturation, or ensuring smoother control effort to account for physical limitations

of actuators. To ensure computational feasibility and practical implementability, it is natural to constrain the design of y_a to a finite-dimensional subspace $\mathbb{P}$ of T -periodic signals.

In earlier studies, such as [22], the subspace $\mathbb{P}$ was specifically chosen as the span of trajectories generated by the exosystem (3.1). This approach had the advantage of maintaining perfect structural alignment between the allocation mechanism and the exogenous dynamics, thereby simplifying both the analysis and implementation. However, this choice also introduced a fundamental limitation: the expressiveness of the allocation mechanism was tied to the modes of the exosystem, leaving little room to exploit redundancy beyond the spectral content of w . In this chapter, we depart from this restrictive viewpoint and generalize the construction of $\mathbb{P}$ by considering a significantly richer class of signals namely, piecewise smooth periodic polynomials of a prescribed maximum degree r . More precisely, each scalar component of a signal in $\mathbb{P}$ is defined as the T -periodic extension of a polynomial of degree at most r . This generalization substantially enlarges the design space for allocation signals and enables the mechanism to capture a wide variety of temporal patterns, far beyond those achievable with purely harmonic signals. Even relatively low-degree polynomials, when extended periodically, can approximate complex trajectories and offer superior flexibility in shaping input behavior.

With these preliminaries now firmly established, we are in a position to formalize the central problem addressed in this chapter. Following the formulations presented in [10], [22], the problem is defined in terms of the initial states $w(0)$, $x(0)$, and $x_c(0)$ of the exosystem (3.1), the plant (3.2), and the controller (3.3), respectively. These initial conditions uniquely determine the subsequent closed-loop evolution and thereby frame the allocation problem to be solved. The following sections will build upon this foundation, progressively developing the theoretical tools and algorithmic strategies required to design spline-based dynamic allocation mechanisms, and demonstrating their efficacy through both analysis and simulation.

Problem 1. *Consider the plant (3.2), the exosystem (3.1), and the controller (3.3), all operating under Assumptions 3.2.1, 3.2.2, and 3.2.3. The objective is to design, whenever possible, an input allocation device with input u_a and output y_a such that, when interconnected with (3.2) and (3.3) according to $u = y_c + y_a$ and $u_a = y_c$, the following requirements are simultaneously satisfied:*

- R1)** (Stability) *For every initial condition $w(0)$ of the exosystem, the allocated closed-loop interconnection depicted in Fig. 3.2 admits a unique steady-state solution of period T . Furthermore, this steady-state response is globally asymptotically stable, meaning that all trajectories of the allocated closed-loop system converge exponentially to it, regardless of initial conditions.*
- R2)** (Output Invisibility) *For each admissible initial condition $w(0)$, $x(0)$, and $x_c(0)$, the output response of the plant, y , in the allocated closed-loop coincides exactly with the output response of the unallocated closed-loop. In other words, the allocation mechanism is designed so that it is "invisible" from the perspective of the regulated output, which preserves the regulation and tracking properties already guaranteed by the baseline controller C .*
- R3)** (Steady-State Optimality) *For every $w(0)$, let u_{ss} denote the periodic³ steady-state*

trajectory of the plant input u . Then the following Lagrange cost functional

$$J(u_{ss}) = \int_0^T g(u_{ss}(\tau)) d\tau \quad (3.4)$$

where $g : \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ is a convex function representing the instantaneous cost, is minimized over the class of admissible steady-state allocation signals $y_a \in \mathbb{P}$. Hence, the allocator not only preserves output behavior but also selects an input trajectory that is optimal in the sense of minimizing the cumulative cost over one period.

Remark 3.2.3

The convexity assumption on the instantaneous cost g in (3.4) is introduced to streamline the problem statement. Under convexity, the optimization problem admits a unique global minimum (though the minimizing allocation signal may not be unique). If g is non-convex, the same allocation strategy still ensures convergence to a steady-state solution corresponding to a local minimum, thereby preserving the optimality interpretation in a weaker sense. $\square$

It is worth emphasizing that no currently available result in the literature on DCA fully addresses Problem 1 for the considered class of allocation signals $\mathbb{P}$. Consequently, the following sections introduce a novel allocator design, specifically developed to satisfy all three requirements of Problem 1, thus filling a notable gap in the existing body of research.

3.3 Allocator Architecture: Periodic Splines and Nullspace Injection

An effective and systematic approach to the design of a DCA unit was proposed in [10], [22], where several of the fundamental ideas first introduced in the seminal work [48] were generalized to the broader context of non-constant signals and to optimization frameworks based on Lagrange cost functionals. In particular, while [48] primarily focuses on the greedy minimization of instantaneous cost functions at each time instant—a strategy that is well known to lead to suboptimal performance when the cost is instead evaluated over a complete period—the methodology in [10], [22] extends the analysis to full-period performance indices. This generalization allows for the design of allocation strategies that explicitly account for the periodic steady-state regime, thereby achieving superior global performance.

The proposed framework leverages the three-component architecture depicted in Fig. 3.2, which provides a clear structural decomposition of the allocation problem. This architecture is expressly constructed so as to ensure that the requirements R1)-R3) of Problem 1 are simultaneously satisfied, while at the same time exposing to the designer a minimal yet meaningful set of tuning parameters. These parameters represent the degrees of freedom available for fine-tuning the performance of the allocator, thereby striking a balance between theoretical rigor and practical implementability.

The allocation architecture is organized around three fundamental building blocks, each of which plays a distinct role in the overall design. These blocks are: the *Annihilator* (*ANN*), which guarantees that the allocation signal does not interfere with the regulated

³Since u_{ss} is periodic, the cost functional (3.4) has the same value on any interval of length T . Without loss of generality, the integral is therefore taken over the interval $[0, T]$.

output dynamics; the SS_{dyn} (*Steady-State Dynamic generator*), which ensures that the allocation signal belongs to the desired class of T -periodic trajectories; and the *Gradient Computation Block* ($\nabla \tilde{J}$), which is responsible for driving the allocation toward minimization of the steady-state cost functional defined in (3.4). The design principles, mathematical formulation, and practical interpretation of each of these components will be discussed in detail in the following subsections.

3.3.1 Design of ANN

The Annihilator constitutes the first and most fundamental component of the proposed dynamic allocation architecture. Its primary role is to guarantee the Output Invisibility property R2) in Problem 1, thereby ensuring that the introduction of the allocation mechanism does not interfere with or alter the output response already guaranteed by the pre-existing closed-loop system composed of the plant and the controller.

In this sense, the Annihilator acts as the structural safeguard of the entire allocation framework, since without its correct operation the allocator would fail to preserve the essential closed-loop behavior. Beyond this direct function, however, the Annihilator also indirectly facilitates the achievement of the other requirements (R1 and R3), as will be clarified in Remark 3.4.1. Thus, while its role appears specialized, the Annihilator in fact provides the foundational layer upon which the stability and optimality of the allocation strategy ultimately depend.

Formally, let $W_{ann}(s)$ denote the transfer function matrix associated with the Annihilator, and let $P(s)$ denote the transfer function matrix of the plant (3.2) mapping the control input u to the regulated output y . It is immediate to verify that the augmented architecture of Fig. 3.2 satisfies requirement R2) of Problem 1 if and only if the following condition holds:

$$P(s)W_{ann}(s) = 0.$$

This relation ensures that any signal generated by the Annihilator lies within the kernel of the plant input-output map, and therefore does not affect the regulated output y . The challenge then reduces to designing a transfer function matrix $W_{ann}(s)$ that satisfies the above identity while preserving realizability and stability.

To this end, consider a polynomial matrix fraction decomposition of $P(s)$. Specifically, $P(s)$ can be written in the form of a (possibly non-minimal) left polynomial matrix fraction [34] as

$$P(s) = (d(s))^{-1}N(s),$$

where $d(s)$ is chosen as the least common multiple of all denominators of the entries of $P(s)$, and where $N(s)$ is then a polynomial matrix of size $n_y \times n_u$. The advantage of this factorization is that it isolates the polynomial structure of the plant's numerator matrix, thereby exposing its algebraic kernel in a form suitable for annihilator construction.

It is well known from polynomial matrix theory [34] that it is possible to construct a polynomial matrix $N^\perp(s)$ of size $n_u \times (n_u - n_y)$ which forms a minimal polynomial basis for the kernel of $N(s)$. Intuitively, the columns of $N^\perp(s)$ span the subspace of input trajectories that are invisible at the plant output, i.e., they generate no contribution to y . From a computational perspective, such a basis can be obtained using standard algebraic techniques; for example, in MATLAB this can be achieved with the `null` command in the Symbolic Toolbox, which automatically returns a polynomial basis for the kernel of $N(s)$.

With this construction, the required Annihilator transfer function matrix can be defined as

$$W_{ann}(s) = N^\perp(s)(d_{ann}(s))^{-1} \quad (3.5)$$

where $d_{ann}(s)$ is any Hurwitz polynomial. The role of $d_{ann}(s)$ is twofold: first, it ensures that $W_{ann}(s)$ is strictly proper, and second, it guarantees stability by enforcing all poles of the annihilator dynamics to lie in the open left-half complex plane.⁴ Once $d_{ann}(s)$ is chosen, a minimal state-space realization $(A_{ann}, B_{ann}, C_{ann})$ of $W_{ann}(s)$ can be constructed, leading to the following internal dynamics of the annihilator:

$$\dot{x}_{ann} = A_{ann}x_{ann} + B_{ann}C_{im}x_{im}, \quad (3.6a)$$

$$y_{ann} = C_{ann}x_{ann}, \quad (3.6b)$$

where x_{ann} represents the internal state of the annihilator and $C_{im}x_{im}$ denotes the input provided by the intermediate block of the allocator architecture.

This realization provides the explicit dynamical form of the Annihilator, which, by construction, guarantees that its output y_{ann} is always confined to the kernel of the plant input-output map. Consequently, the Annihilator achieves invisibility at the output level while shaping the internal structure of the allocator in a manner that facilitates the fulfillment of stability and steady-state optimality requirements.

3.3.2 Design of SS_{dyn}

The block SS_{dyn} constitutes the second fundamental component of the allocation architecture. Its purpose is to provide a parametrization of the admissible allocation signals, thereby defining the set $\mathbb{P}$ over which the steady-state optimization requirement R3) of Problem 1 is performed. In other words, the role of SS_{dyn} is to act as a generator of the admissible input trajectories, whose structure is entirely determined by its internal dynamics and by the specification of its initial conditions. The use of such a subsystem makes the allocation problem finite-dimensional and implementable in practice, while still retaining the expressive power needed to approximate or represent the desired class of allocation signals.

Formally, the block SS_{dyn} is characterized by the following hybrid dynamics:

$$\dot{x}_{im}(t) = A_{im}x_{im}(t), \quad t \in [kT, (k+1)T), \quad (3.7a)$$

$$x_{im}(t) = \Omega_{im}x_{im}(t^-) + y_g(t^-), \quad t = kT, \quad (3.7b)$$

$$y_{im}(t) = C_{im}x_{im}(t), \quad t \in \mathbb{R}, \quad (3.7c)$$

where $x_{im}(t^-) := \lim_{\varepsilon \rightarrow 0} x_{im}(t - \varepsilon)$ denotes the pre-jump value of the state. Thus, the system evolves according to continuous-time linear dynamics within each interval $[kT, (k+1)T)$, while at the discrete instants $t = kT$ its state undergoes a reset governed by (3.7b). The output $y_{im}(t)$ is then obtained as a linear projection of the internal state. This hybrid structure enables the subsystem to encode periodicity explicitly, since the jumps occur exactly at integer multiples of the fundamental period T . Moreover, y_g is the gradient-descent correction signal computed once per period, used to update the

⁴A polynomial is called Hurwitz if all of its roots have strictly negative real part, meaning that it is the denominator of a stable transfer function.

internal model state toward the steady-state minimizer of the cost (See Fig. 3.2).

The internal model reproduces T -periodic extensions of polynomials of degree at most $r - 1$. A natural generalization of this idea is to represent the periodic steady-state signals through *splines* (see Fig. 3.3), i.e., piecewise-polynomial functions defined over a subdivision of the fundamental period T . Formally, the period is divided into N_s uniform segments of length $\Delta = T/N_s$, and on each segment $[kT + i\Delta, kT + (i + 1)\Delta)$ the internal model generates a polynomial trajectory of degree at most $r - 1$. At the boundaries between segments, the reset map in (3.7b) enforces continuity and ensures the periodic repetition of the trajectory across successive periods. This spline-based construction allows a richer description of intra-period dynamics without altering the hybrid structure of (3.7).

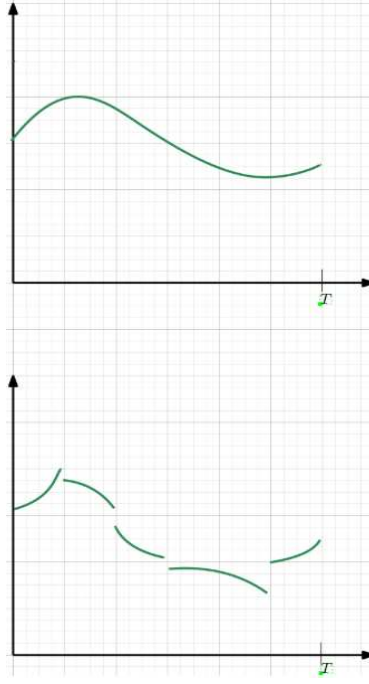


Figure 3.3: The period is divided into several segments and a spline (a fixed degree polynomial of time) is considered on each segment

In this work, a single segment per period is adopted ($N_s = 1$), defined as the periodic extension of a time polynomial over $[0, T)$. This configuration minimizes the computational load while preserving the structural flexibility to introduce multiple segments if finer intra-period resolution is required. Thus, the term *spline* is retained to reflect the framework's inherent generalized capability, despite the current reduction to a single interval. Under this setting, the matrices in (3.8) remain unchanged:

$$A_{im} = I_{n_u - n_y} \otimes J_{0,r}, \quad (3.8a)$$

$$\Omega_{im} = e^{-A_{im}T}, \quad (3.8b)$$

$$C_{im} = I_{n_u - n_y} \otimes [1 \quad 0_{1 \times (r-1)}]. \quad (3.8c)$$

Hence, no modification is required in the allocator or *ANN* design method; only the internal-model block is interpreted as a spline generator, and the gradient computation block is updated accordingly to account for the hybrid (piecewise-continuous) evolution of the internal model. Where $\otimes$ denotes the Kronecker product, $I_{n_u - n_y}$ is the identity matrix of dimension $(n_u - n_y)$, $J_{0,r}$ is a Jordan block of size r associated with the zero eigenvalue, and $0_{1 \times (r-1)}$ is a row vector of zeros of length $(r - 1)$. This parametrization reflects the fact that the admissible signals in $\mathbb{P}$ are generated as T -periodic extensions of

polynomials of degree strictly less than r .

It is instructive to analyze the behavior of (3.7) in two cases. If $y_g = 0$, the reset map reduces to $x_{im}(kT) = \Omega_{im}x_{im}((k-1)T)$. Since Ω_{im} is constructed as $e^{-A_{im}T}$, this ensures that each scalar component of x_{im} evolves as the T -periodic extension of a polynomial trajectory of degree at most $r-1$. Thus, in the absence of the steering signal y_g , SS_{dyn} naturally reproduces the entire class of periodic splines of interest. Conversely, the input y_g plays a crucial role in the closed-loop allocation mechanism, steering the state of SS_{dyn} toward the trajectory in $\mathbb{P}$ that minimizes the steady-state cost functional (3.4). In this sense, y_g acts as an adaptation signal that guides the internal model dynamics toward the optimal periodic orbit. Once y_g has fulfilled its role and becomes practically zero⁵, the evolution of x_{im} becomes T -periodic. Consequently, the response of the annihilator (which is linear and stable since $d_{ann}(s)$ is Hurwitz) and the output of the allocator both converge to a T -periodic steady state. Due to the resets in x_{im} , the annihilator's response generally contains both the natural modes of SS_{dyn} and those of the annihilator itself, producing a composite periodic motion. The state of the annihilator then satisfies:

$$x_{ann}(kT) = x_{ann}(0), \quad (3.9)$$

and more precisely, the following property holds.

Proposition 3.3.1

If x_{im} is reset periodically to the same initial condition, the annihilator evolves on a periodic orbit whose continuous state at the beginning of each period is given by

$$x_{ann}(0) = (I - e^{A_{ann}T})^{-1} M x_{im}(0), \quad (3.10)$$

where

$$M = \int_0^T e^{A_{ann}(T-\tau)} B_{ann} C_{im} e^{A_{im}\tau} d\tau. \quad (3.11)$$

Finally, the complete steady-state output of the cascade of the spline generator and annihilator is given by

$$y_a(t) = C_{ann} e^{A_{ann}t} (I - e^{A_{ann}T})^{-1} M x_{im}(0) + C_{ann} \int_0^t e^{A_{ann}(t-\tau)} B_{ann} C_{im} e^{A_{im}\tau} x_{im}(0) d\tau.$$

□

Proof. Let the combined state of the annihilator and the spline generator be

$$\bar{X}(t) = \begin{bmatrix} x_{ann}(t) \\ x_{im}(t) \end{bmatrix}, \quad (3.12)$$

whose evolution satisfies

$$\dot{\bar{X}}(t) = \begin{bmatrix} A_{ann} & B_{ann} C_{im} \\ 0 & A_{im} \end{bmatrix} \bar{X}(t) = \bar{A} \bar{X}(t). \quad (3.13)$$

Hence,

$$e^{\bar{A}T} = \begin{bmatrix} e^{A_{ann}T} & M \\ 0 & e^{A_{im}T} \end{bmatrix}, \quad (3.14)$$

which directly yields

$$x_{ann}(T) = e^{A_{ann}T} x_{ann}(0) + Mx_{im}(0). \quad (3.15)$$

If $x_{im}(T) = x_{im}(0)$, periodicity implies $x_{ann}(T) = x_{ann}(0)$, leading to the claimed expression. $\square$

Remark 3.3.1

As pointed out in [10], [22], the conceptual role of the subsystem SS_{dyn} can be directly related to the notion of an internal model in the theory of output regulation [8], [13], [19], [29], [39]. Indeed, in the classical internal model principle, the regulator must incorporate within its dynamics a generator capable of reproducing the exogenous signals to be tracked or rejected. Analogously, in the present context, the block SS_{dyn} is capable, in open loop, of generating the class of admissible allocation signals $\mathbb{P}$, i.e., the T -periodic extensions of polynomials of degree at most r . It can therefore be interpreted as an internal model specifically tailored for spline-type signals.

This interpretation also clarifies the connection between the present spline-based allocator and previously proposed schemes. For example, the allocator in [22] employs a sinusoidal internal model generator, suitable for harmonic allocations, while the approach in [48] relies on a constant internal model generator, suitable only for steady constant allocations. By contrast, the proposed framework incorporates a generator able to produce T -periodic extensions of polynomial signals, thereby offering a significantly richer parametrization that generalizes both previous approaches. This distinction is also reflected in the notation of (3.7), which emphasizes the hybrid nature and the spline-generating capability of the subsystem.

Moreover, in DCA the internal model does not (necessarily) replicate the class of exogenous signals, but rather is used to produce signals from a suitable, user-selected class of signals $\mathbb{P}$; and the class of spline signals can be more flexible than other classes considered in [8], [13], [19], [29], [39]. $\square$

3.3.3 Design of $\nabla \tilde{J}$

The third and final block of the allocation architecture is the *Gradient Computation* unit, denoted by $\nabla \tilde{J}$. Its purpose is to generate the steering signal y_g which drives the internal state of SS_{dyn} toward the trajectory that yields a periodic steady-state evolution minimizing the cost functional (3.4).

A natural and effective way to achieve this goal is to employ a gradient-based adaptation law, which updates y_g in the direction opposite to the gradient of the cost functional. This strategy is conceptually simple, yet it provides a powerful mechanism to ensure convergence toward the optimal solution within the admissible signal class $\mathbb{P}$.

The relevant gradient must be evaluated with respect to the steady-state behavior of the closed-loop system, that is, under the condition that the update y_g in (3.7b) is frozen (set to zero).

⁵In general, y_g converges to zero only asymptotically, but for simplicity one can regard it as essentially zero after a finite interval. In implementation, it can be switched off once its norm falls below a small threshold, as is common in numerical optimization.

In this case, the cost gradient takes the following form ⁶:

$$\nabla J = \int_0^T N' e^{A_a' \tau} C_a' \nabla g(y_c(\tau) + C_a e^{A_a \tau} N x_{im}(0)) d\tau, \quad (3.16)$$

where the following notation is used:

$$\begin{aligned} A_a &= \begin{bmatrix} A_{ann} & B_{ann} C_{im} \\ 0 & A_{im} \end{bmatrix}, & C_a &= [C_{ann} \quad 0], \\ M &= e^{A_a T} = \begin{bmatrix} M_1 & M_2 \\ 0 & M_3 \end{bmatrix}, & N &= \begin{bmatrix} (I - M_1)^{-1} M_2 \\ I \end{bmatrix}. \end{aligned}$$

Here, the augmented system matrices A_a and C_a describe the joint dynamics of the Annihilator and the signal generator SS_{dyn} , while M and N encode the effect of one-period evolution and its relation to the internal state x_{im} . The expression (3.16) thus provides a compact representation of the gradient of the steady-state cost functional with respect to the initial condition of SS_{dyn} , which in turn determines the admissible allocation trajectory.

In order to compute (3.16) in real time, one must take into account the hybrid nature of the generator dynamics, in particular the fact that the state x_{im} is reset at the beginning of each period according to (3.7b). A convenient way to implement the gradient update online is to employ the following hybrid dynamics:

$$\dot{x}_g = -A_a' x_g + C_a' \nabla g(y_c + C_a x_a), \quad t \in [kT, (k+1)T), \quad (3.17a)$$

$$\dot{x}_a = A_a x_a, \quad t \in [kT, (k+1)T), \quad (3.17b)$$

$$x_g(t) = 0, \quad t = kT, \quad (3.17c)$$

$$x_a(t) = N(\Omega_{im} x_{im}(t^-) + y_g(t^-)), \quad t = kT, \quad (3.17d)$$

$$y_g = -\gamma N' e^{A_a' T} x_g, \quad t \in \mathbb{R}, \quad (3.17e)$$

$$y_a = C_a x_a, \quad t \in \mathbb{R}, \quad (3.17f)$$

where γ is a sufficiently small positive constant, serving as the step size of the gradient update.

The interpretation of (3.17) is the following: the state x_g acts as an integrator accumulating information about the instantaneous gradient ∇g , filtered through the adjoint dynamics $-A_a'$ and the output map C_a' . The reset condition $x_g(t) = 0$ at each period ensures that the computation is restricted to a single period and does not accumulate beyond it. The auxiliary state x_a reproduces the nominal dynamics of the combined system (ANN, SS_{dyn}) , and its reset incorporates the effect of both the hybrid jump Ω_{im} and the steering signal y_g . Finally, the update $y_g = -\gamma N' e^{A_a' T} x_g$ implements the negative gradient step, driving the generator state toward the minimum of the cost functional.

Remark 3.3.2

It is worth emphasizing that, although the high-level architecture of the allocator remains conceptually similar to that proposed in [10], [22], the gradient computation presented here is considerably more intricate. This increased complexity arises from the richer structure of the admissible signal space $\mathbb{P}$ adopted in this chapter, namely

⁶Since all signals are evaluated at their T -periodic steady state, the integral can be taken over any interval of length T . Without loss of generality, the interval $[0, T]$ is chosen here.

the class of T -periodic extensions of polynomials of fixed degree r . In contrast, the approaches in [22] and [48] deal with sinusoidal and constant signal classes, respectively, which yield simpler steady-state behaviors. By extending the admissible class to splines, the present design significantly broadens the range of trajectories available for optimization, at the cost of requiring more sophisticated gradient evaluation and update laws. Nevertheless, this added complexity is justified by the enhanced flexibility and performance that the spline-based allocation strategy is capable of delivering. $\square$

3.4 Theoretical Guarantees

This section presents the central theoretical contribution of the chapter. The result establishes that, under the structural assumptions stated earlier and for a sufficiently small step size in the gradient update law, the allocator architecture introduced in Sec. 3.3 is able to meet all three requirements of Problem 1. In other words, the theorem guarantees that the proposed design not only preserves the stability and output properties of the nominal closed-loop system but also ensures steady-state optimality with respect to the cost functional (3.4).

Theorem 3.4.1

Under Assumptions 3.2.1, 3.2.2, and 3.2.3, there exists a constant $\gamma^* > 0$ such that for every step size $\gamma \in (0, \gamma^*)$, the allocator designed according to Sec. 3.3 solves Problem 1. $\square$

Proof. To make the proof precise and ensure global asymptotic stability of the optimal steady-state, we proceed under the following technical conditions: the instantaneous cost $g(\cdot)$ is strictly convex and radially unbounded, and the cascade of the steady-state generator SS_{dyn} and the Annihilator is observable.

Verification of Requirement R2: Output Invisibility

Requirement R2 demands that the plant output y is unaffected by the allocation signal y_a . The plant input is $u = y_c + y_a$. The signal y_a is generated by the Annihilator (ANN) such that its transfer function $W_{ann}(s)$ satisfies:

$$P(s)W_{ann}(s) = 0$$

where $P(s)$ is the plant transfer function. By linearity, the contribution of y_a to the output is $P(s)W_{ann}(s)Y_{im}(s) = 0$. Consequently, the output response $y(t)$ depends only on the primary control y_c and the exogenous signal w . The nominal closed-loop performance is preserved, and R2 is satisfied structurally.

Verification of Requirements R1 and R3: Stability and Optimality

Under the stated assumptions, we prove R1 and R3 by treating the cost functional J as a Lyapunov function for the parameter dynamics.

1. **Existence of a Unique Minimum:** Since $g(\cdot)$ is strictly convex and radially unbounded, the steady-state cost functional $\bar{J}(\alpha)$ (where α parameterizes the initial state $x_{im}(0)$) is also strictly convex and radially unbounded in $\mathbb{R}^{\bar{N}}$. This ensures the existence of

a *unique* global minimizer α^* , corresponding to a unique optimal periodic orbit for the allocation signal.

2. **Lyapunov-Based Convergence:** Let $V(\alpha) = \bar{J}(\alpha) - \bar{J}(\alpha^*)$ be a candidate Lyapunov function for the discrete-time parameter update law. By construction, $V(\alpha^*) = 0$ and $V(\alpha) > 0$ for all $\alpha \neq \alpha^*$. Radial unboundedness of g implies $V(\alpha) \rightarrow \infty$ as $\|\alpha\| \rightarrow \infty$. The gradient update law (Equation 3.17e) ensures:

$$V(\alpha_{k+1}) - V(\alpha_k) \leq -\gamma \|\nabla \bar{J}(\alpha_k)\|^2 + O(\gamma^2)$$

For a sufficiently small $\gamma \in (0, \gamma^*)$, the gradient law acts as a strict descent method. Since the mapping from α to the cost is observable (due to the observability of the SS_{dyn} -ANN cascade), $\nabla \bar{J}(\alpha) = 0$ if and only if $\alpha = \alpha^*$. Thus, α^* is a globally asymptotically stable equilibrium for the parameter dynamics.

3. **Cascade Stability:** The overall system Σ_{all} is a cascade interconnection:

- The unallocated closed-loop Σ is exponentially stable (Assumption 3.2.3).
- The allocator parameters α converge to α^* (as shown above).
- The SS_{dyn} and ANN blocks are linear systems driven by convergent parameters.

By standard results on the stability of cascaded systems, the convergence of the driving system (the parameter update) and the stability of the driven subsystems (Σ , ANN, SS_{dyn}) ensure that the entire state of Σ_{all} converges to the unique T -periodic steady-state response associated with the optimal parameter α^* .

Relaxation of Hypotheses

If the additional hypotheses (strict convexity and radial unboundedness) are relaxed to simple convexity:

- The set of minimizers $\mathcal{A}^*$ may be a non-singleton or unbounded set.
- While the gradient law still ensures that the cost J is non-increasing, x_{im} could potentially diverge if the level sets of J are not compact.
- Convergence would only be guaranteed to the set of stationary points where $\nabla \bar{J} = 0$, which solves Problem 1 in a weaker, local sense.

Under the full hypotheses, however, the allocator is guaranteed to achieve global asymptotic stability of the unique optimal steady-state. $\square$

The significance of this result is twofold. First, it provides a rigorous performance guarantee for the spline-based allocation scheme, demonstrating that the architecture proposed in this chapter is not merely heuristic but is theoretically sound and systematically convergent. Second, it highlights the key role of the gradient-based adaptation mechanism: by choosing γ sufficiently small, one can guarantee that the allocator operates in a stable regime while still providing effective convergence toward steady-state optimality.

The parameter γ therefore represents a classical trade-off between speed of adaptation and robustness: if chosen too large, instability or oscillatory behavior may occur, while if chosen sufficiently small, convergence to the optimal solution is assured at the price of slower convergence.

The proof relies on constructing a Lyapunov-like functional combining the natural quadratic stability measure associated with the closed-loop interconnection and the convexity properties of the cost functional (3.4).

By exploiting the hybrid structure of the generator SS_{dyn} , together with the resetting properties of the gradient computation dynamics (3.17), one can show that the allocation

state evolves within a bounded and contracting set across successive periods. As a consequence, the gradient update acts as a descent method on the steady-state cost functional, thereby ensuring convergence toward an optimal (or locally optimal, in the non-convex case) solution within the admissible signal set $\mathbb{P}$.

Remark 3.4.1

The design of the Annihilator (3.6) guarantees that, regardless of the specific signal y_{im} produced by the block SS_{dyn} in Fig. 3.2, its net effect on the plant output y is identically zero. This fundamental property is achieved by construction, since the Annihilator acts as a dynamic filter projecting every admissible allocation signal onto the kernel of the plant input-output map. In this way, the Annihilator enforces the Output Invisibility requirement R2) of Problem 1 in an exact and structural manner. As already pointed out in the pioneering contribution [48], although in a simpler context, this property has consequences that extend beyond invisibility itself: it facilitates both the stability analysis of the overall allocated closed-loop system and the convergence to steady-state optimality.

Concerning stability, the key observation is that, because the Annihilator completely decouples the allocation signal from the regulated plant output y , the block C which is assumed to stabilize P in the nominal setting remains entirely unaffected by the allocation dynamics.

This effectively creates a virtual cascade interconnection, in which the plant-controller pair operates as in the nominal closed loop, while the allocator evolves in parallel without disturbing the regulated variables. The immediate implication of this cascade structure is that the overall stability of the allocated closed-loop system reduces to verifying the stability of the allocator itself.

This verification is straightforward: the allocator is composed of three blocks in cascade, namely the Annihilator, the signal generator SS_{dyn} , and the gradient adaptation block $\nabla \tilde{J}$.

The Annihilator is stable by construction due to the Hurwitz choice of $d_{ann}(s)$, the subsystem SS_{dyn} is stabilized by the corrective action of the steering signal y_g , and the adaptation block $\nabla \tilde{J}$ enforces convergence by implementing a gradient descent algorithm on the cost functional.

Hence, the combined allocation mechanism inherits stability from the cascade structure together with the stabilizing design choices embedded in its components.

With regard to optimality, the same virtual cascade reasoning plays a pivotal role. Since the plant output y remains completely unaffected by the allocation signal, the controller output y_c converges exponentially to the same unique T -periodic steady state for any initial condition $w(0)$ of the exosystem E , both in the allocated and in the non-allocated closed-loop systems. This invisibility is crucial because it ensures that the optimization of the cost functional J in (3.4) can be performed using the actual plant input $u = y_c + y_a$, while being guaranteed that the periodic trajectory of y_c is unchanged.

As a consequence, any online optimization method, such as the gradient descent algorithm adopted in this chapter, will converge to the same minimizer that would have been obtained if the steady-state signal y_c were directly provided to the optimization routine. This structural property eliminates the risk of spurious minimizers or perturbations in the regulation loop, thereby allowing the allocator to operate as a transparent yet effective mechanism for steady-state performance improvement. $\square$

3.5 Computational Experiments

To complement the theoretical developments of the previous sections, this part of the chapter presents a simple yet representative numerical example. The objective is to illustrate in a concrete setting how the proposed allocation architecture achieves the requirements R1)-R3) of Problem 1, and in particular how the annihilator, the signal generator SS_{dyn} , and the gradient-based update law cooperate in practice. While the example is intentionally kept simple for clarity of exposition, it nevertheless captures the essential features of input redundancy, allocation, and steady-state optimization.

3.5.1 Data

Consider the plant defined by the following parameter values:

$$A = \begin{bmatrix} 2 & 3 & -2 \\ 0 & -1 & 0 \\ 1 & 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & -1 \end{bmatrix}, \quad (3.18a)$$

$$C = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}, \quad D = 0, \quad (3.18b)$$

which describes a system with three states, two inputs, and a single regulated output. According to the definition in [48], this plant is not *strongly redundant*, since its two inputs do not provide complete decoupling freedom at steady state. Nevertheless, as demonstrated in [10], it is still possible to enforce complete invisibility of the allocator action through the design of a suitable *dynamic annihilator*, thereby fulfilling the Output Invisibility requirement R2). This property makes the plant an interesting case study for validating the proposed allocator design.

To regulate the plant, an LQG stabilizer with integral action is adopted. The inclusion of an integrator guarantees exact asymptotic tracking of step reference signals, thereby ensuring zero steady-state error. The stabilizer is computed in two stages by sequentially closing loops around the two plant inputs. Specifically, the second input u_2 is assigned as

$$u_2(s) = \frac{1}{s}(r(s) - y(s)),$$

where $r(s)$ denotes the Laplace transform of the reference signal, thus enforcing integral action with respect to the regulation error. Once this loop is closed, the resulting SISO system with input u_1 and output y is further stabilized by assigning

$$u_1(s) = W_{cc}(s)y(s),$$

where $W_{cc}(s)$ is designed as a standard LQG controller with unit weights, and has the following minimal realization:

$$A_{cc} = \begin{bmatrix} 10.34 & 5.19 & -21.82 & -0.06 \\ 1.44 & -1.00 & -1.44 & 1.00 \\ 21.06 & 3.19 & -28.54 & -1.06 \\ 1.41 & 0 & -1.41 & 0 \end{bmatrix}, \quad B_{cc} = \begin{bmatrix} -3.59 \\ -1.44 \\ -15.32 \\ -2.41 \end{bmatrix},$$

$$C_{cc} = \begin{bmatrix} 4.75 & 2.19 & -16.23 & -0.06 \end{bmatrix}, \quad D_{cc} = 0.$$

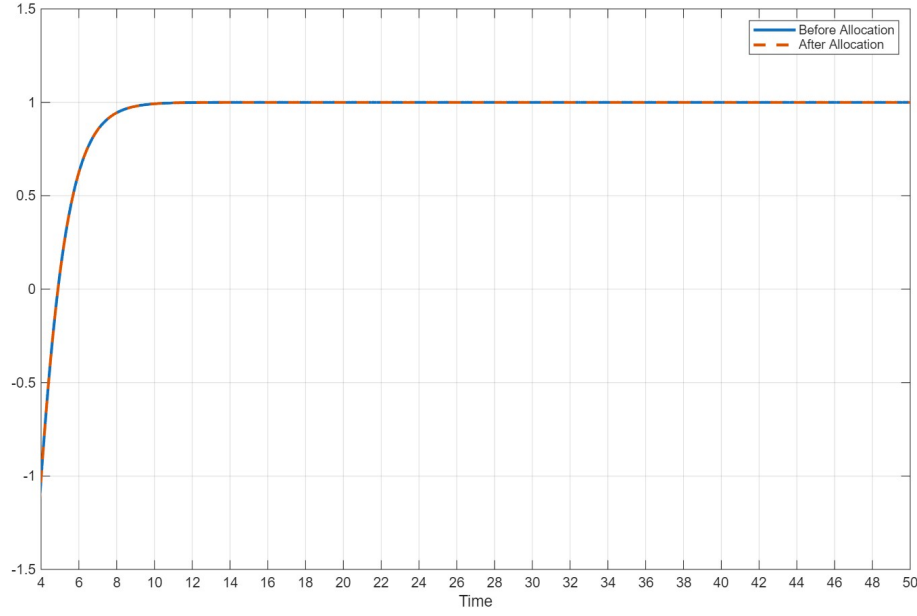


Figure 3.4: Constant Reference: comparison of output trajectories with and without allocation. The regulated output y remains unaffected by the allocation mechanism, thereby confirming the Output Invisibility property R2) of Problem 1.

The numerical simulations reported below illustrate how the proposed allocator, applied to the plant-controller pair above, achieves output invisibility while simultaneously improving the steady-state input behavior through cost minimization. The steady-state performance is analyzed, and the role of the allocator is highlighted by comparing input trajectories and cost evolution before and after allocation.

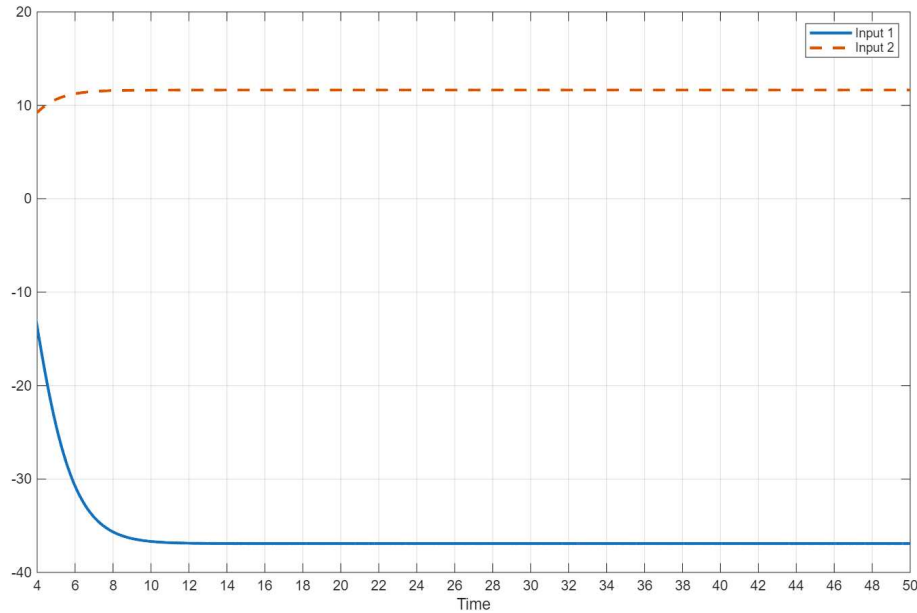


Figure 3.5: Constant Reference: input trajectories u_1 and u_2 before allocation. The absence of allocation results in redundant control inputs that are not optimized with respect to the cost functional.

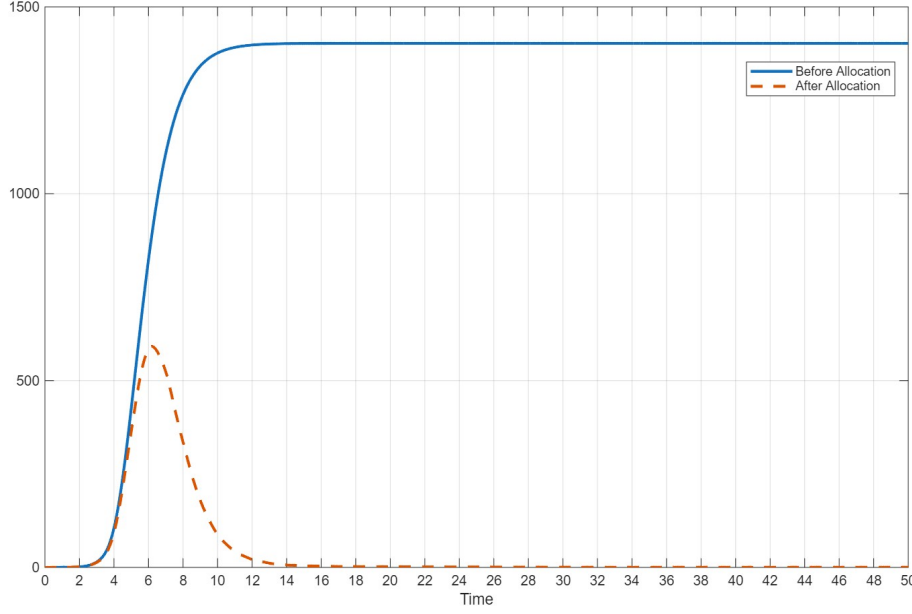


Figure 3.6: Constant Reference: cost functional J over successive periods. The action of the allocator leads to a monotonic decrease of the cost, illustrating the convergence of the gradient-based update law and the fulfillment of requirement R3).

3.5.2 Allocator Design

In order to test the allocation framework on the plant (3.18), a specific performance criterion must be defined. For this purpose, the cost functional to be minimized by the allocator is chosen as a quadratic integral functional, evaluated over one period T , of the so-called unit deadzone of the input:

$$J(u) = \int_0^T dz(u)^\top dz(u) d\tau, \quad (3.19)$$

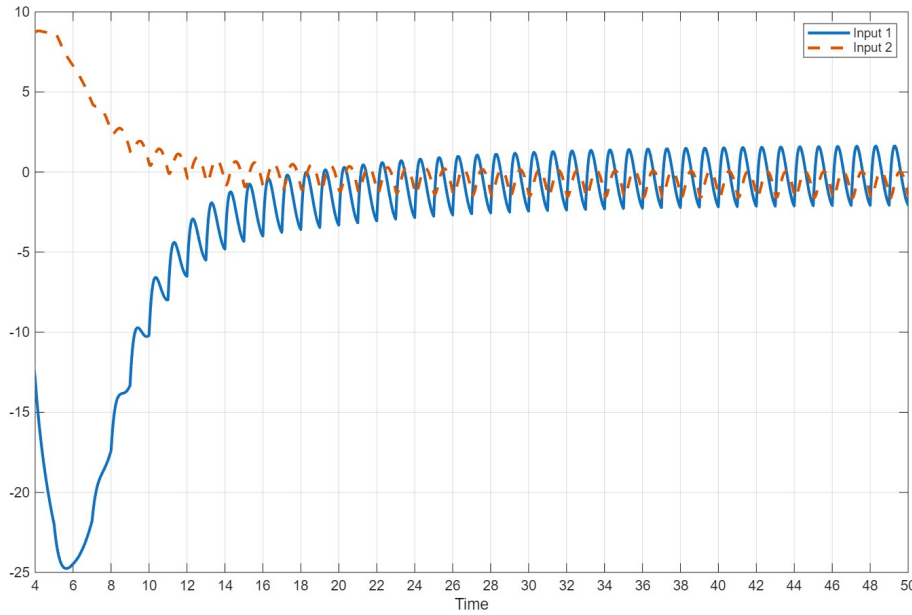


Figure 3.7: Constant Reference: input trajectories u_1 and u_2 after allocation. Compared with Fig. 3.5, the allocation signal redistributes the control effort, producing inputs that achieve the same output regulation while reducing the steady-state cost.

where the unit deadzone operator $\text{dz}(\cdot)$ is defined componentwise according to the relation

$$\text{dz}(u) := \text{sign}(u) \max(|u| - 1, 0).$$

The functional (3.19) penalizes only those input components that exceed a prescribed threshold (equal to one in this case), while leaving smaller inputs unpenalized. Such a cost is particularly relevant in engineering applications where small input magnitudes can be generated at negligible energetic expense, whereas large magnitudes require costly resources. A representative example arises in aerospace systems: small torques can often be provided by solar-powered actuators such as magnetotorquers, essentially free of fuel cost, while larger torques necessitate the use of thrusters, which consume scarce and expensive propellant. By weighting the squared magnitude of the deadzone, the cost function further penalizes large excursions, thus enforcing a strong preference for input trajectories that minimize the use of high-energy actuators.

A suitable Annihilator for the plant (3.18) is chosen with the following realization:

$$A_{ann} = \begin{bmatrix} -2 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -4 \end{bmatrix}, \quad B_{ann} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad (3.20a)$$

$$C_{ann} = \begin{bmatrix} -7 & 12 & -4 \\ -2 & 8 & -6 \end{bmatrix}, \quad D_{ann} = 0. \quad (3.20b)$$

This choice ensures that the resulting allocator satisfies the invisibility property (requirement R2) of Problem 1) by structurally eliminating the effect of the allocation signal on the plant output y . It is worth emphasizing that, despite the fact that the plant (3.18) is only *weakly redundant* according to the taxonomy in [48], the above dynamic annihilator is nevertheless sufficient to achieve complete output invisibility, in line with the theoretical results of [10].

To illustrate this point, consider the case of a constant reference. Figures 3.5 and 3.7 depict the plant inputs with and without allocation, respectively. It can be observed that the allocated inputs differ substantially from their unallocated counterparts, as the allocator redistributes control effort to minimize the cost functional. However, as shown in Fig. 3.4, the corresponding output trajectories are indistinguishable, both during the transient phase and at steady state, thereby confirming that output invisibility is exactly enforced.

The parameters of the subsystems SS_{dyn} and $\nabla \tilde{J}$ are chosen consistently with the general structure in (3.8) and (3.17), with the specific values

$$r = 4, \quad \gamma = 3, \quad T = 1.$$

Here, $r = 4$ specifies that cubic polynomials (degree $r - 1 = 3$) are employed in the spline representation of admissible signals, providing sufficient richness to approximate the optimal trajectories. The step size $\gamma = 3$ determines the gradient descent update rate, and is chosen small enough to ensure stability while still enabling fast convergence. Finally, the period T is set to one. For the case of a constant reference, this choice of T is arbitrary since a constant signal is periodic with any period. However, in the second scenario involving a time-varying reference, the period $T = 1$ corresponds exactly to the fundamental period of the reference signal, making it the natural choice.

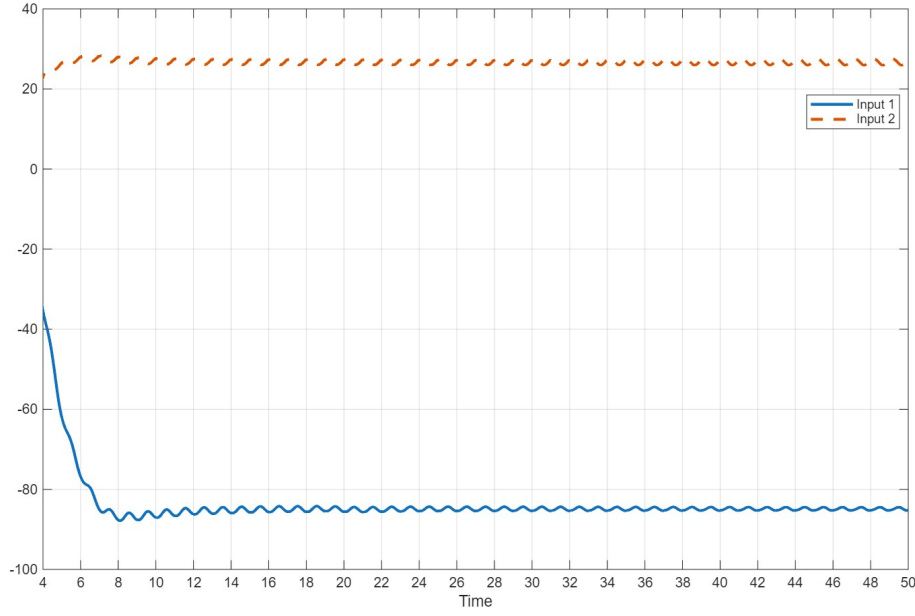


Figure 3.8: Saw-tooth reference: plant inputs with a single sinusoidal allocator. The allocator reduces the cost but exhibits limited flexibility in shaping the input trajectory due to the restricted signal class.

The following simulations demonstrate how the proposed allocator performs under more demanding conditions, such as saw-tooth references, and how the spline-based internal model enables superior performance compared to traditional sinusoidal allocation strategies.

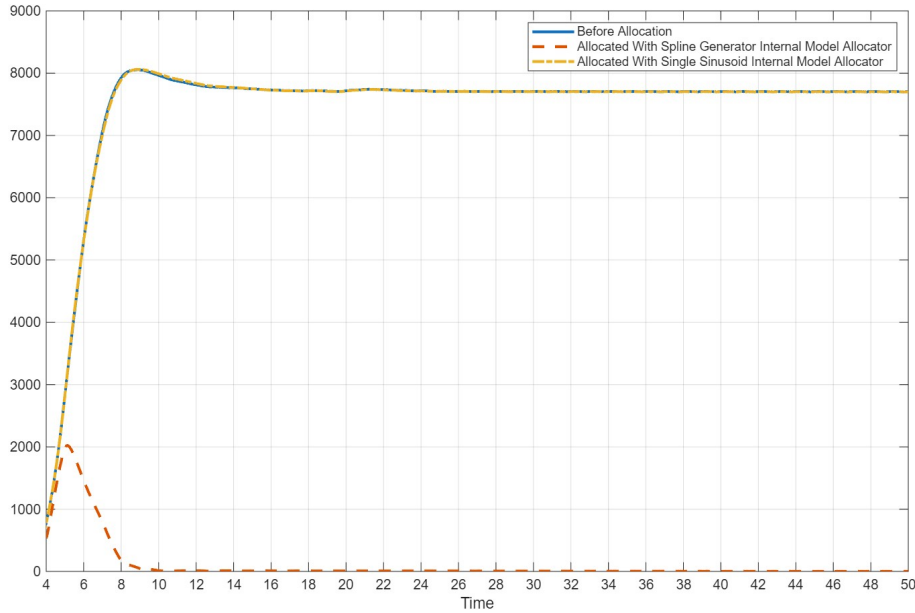


Figure 3.9: Saw-tooth reference: comparison of the cost functional J for three cases: unallocated inputs, allocation with a single sinusoidal internal model, and allocation with the proposed spline-based internal model. The spline allocator consistently achieves the lowest cost, thereby validating its enhanced expressivity and steady-state optimality.

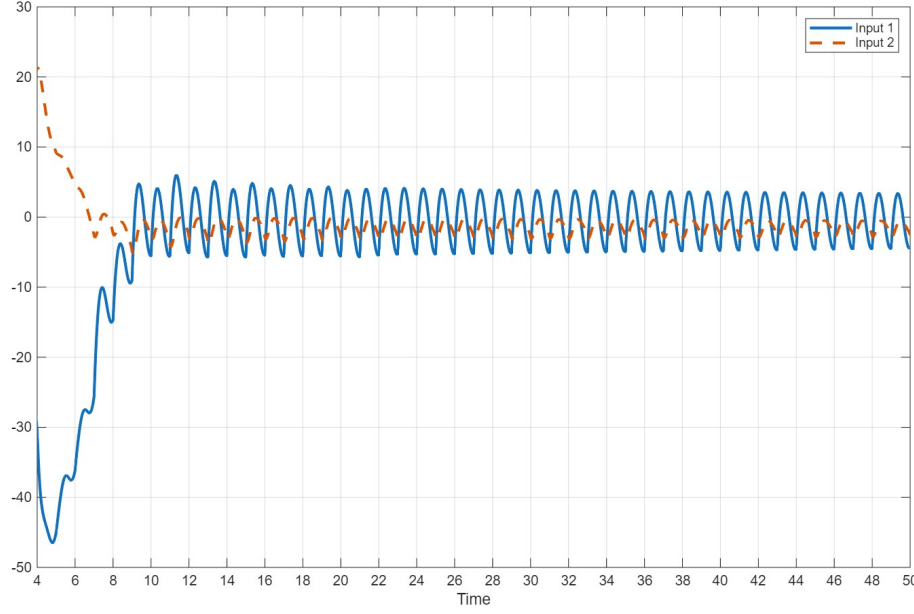


Figure 3.10: Saw-tooth reference: plant inputs obtained with the spline-based allocator. Compared to the sinusoidal case (Fig. 3.8), the spline allocator produces richer input trajectories, allowing more efficient cost reduction while maintaining output invisibility.

3.5.3 Scenario 1: Constant Reference

In the first experiment, the reference signal is chosen as a unit step. The *desirable performance* provided by the baseline controller C , designed as described earlier, already includes the crucial property of ensuring offset-free tracking in response to constant references. This property can be clearly observed in Fig. 3.4, where the plant output converges exactly to the reference value.

The introduction of the allocator does not compromise this behavior: thanks to the annihilator, the output invisibility requirement is structurally enforced. As a result, the entire closed-loop output response induced by C , including both transient and steady-state dynamics, is preserved.

Indeed, the allocated and unallocated output trajectories in Fig. 3.4 are exactly coincident, thereby confirming requirement R2) of Problem 1.

While the invisibility of the output trajectory is expected, it is nevertheless interesting to analyze how the allocator modifies the input signals. Figures 3.5 and 3.7 compare the plant input trajectories before and after allocation.

It can be appreciated that the allocator introduces structured oscillations in the input channels, which at first may appear counterintuitive since the reference is constant. However, these oscillations are not arbitrary: they are the mechanism through which the allocator reduces the steady-state cost (3.19).

This cost penalizes input excursions beyond the unit threshold, and by appropriately oscillating the inputs, the allocator redistributes the effort so as to minimize the overall penalty. The resulting improvement is quantified in Fig. 3.6, which shows the monotonic reduction of the cost functional achieved after allocation. Thus, the experiment demonstrates that even in the case of a constant reference, the spline-based allocator discovers nontrivial input trajectories that reduce the quadratic deadzone cost without affecting the regulated output.

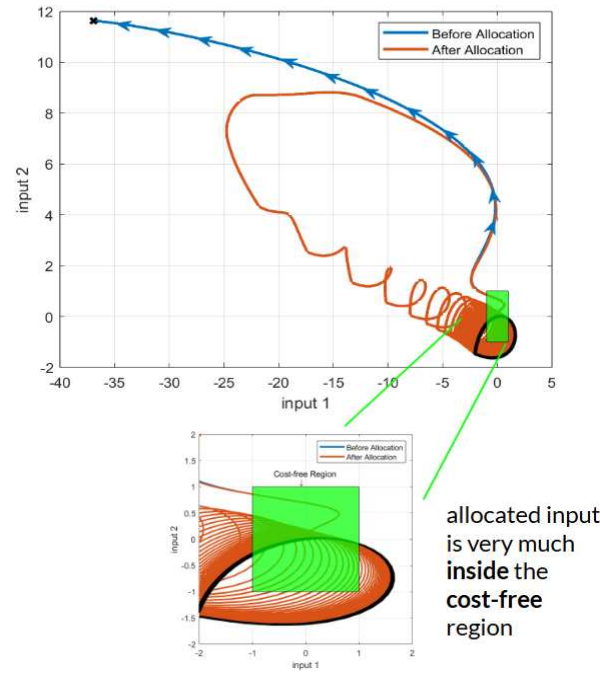


Figure 3.11: Input trajectories (u_1, u_2) before and after allocation for the constant-reference case. The allocated inputs settle deeper inside the cost-free region, confirming the effectiveness of the steady-state optimization.

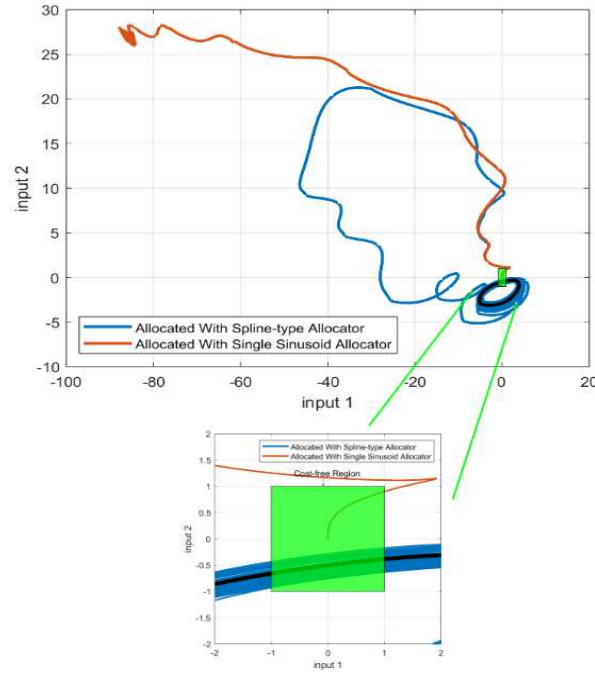


Figure 3.12: Input trajectories (u_1, u_2) before and after allocation for the sawtooth reference. The allocated trajectory gradually moves toward the center of the cost-free region, illustrating the effect of the periodic spline-based optimization on actuator usage.

3.5.4 Scenario 2: Sawtooth Reference

The second experiment considers a more challenging scenario in which the reference is a periodic triangular wave (sawtooth) of period $T = 1$. The signal increases linearly from 0 to 5 during the first half-period and decreases linearly from 5 back to 0 dur-

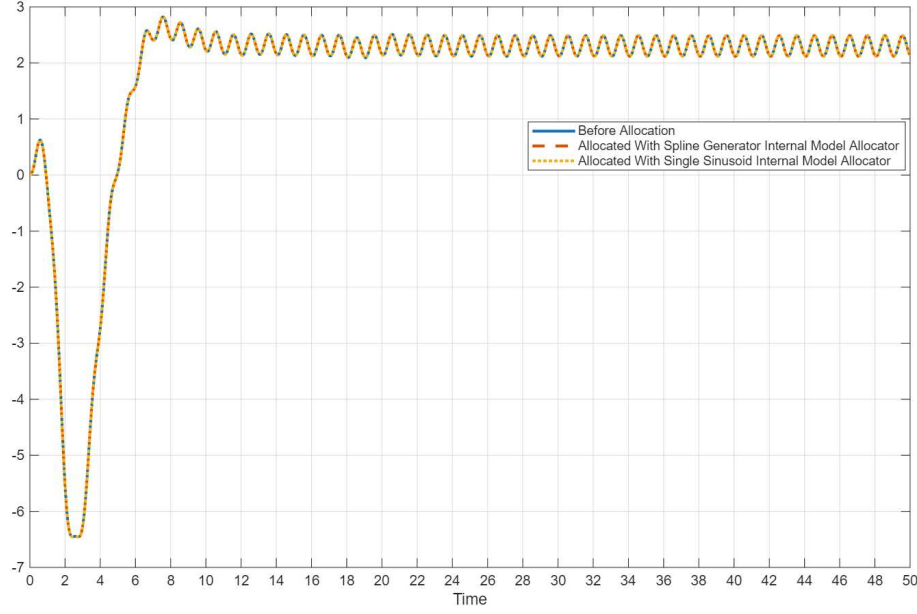


Figure 3.13: Saw-tooth reference: comparison of output trajectories without allocation, with spline allocation method and with sinusoidal allocation method. The regulated output y remains unaffected by the allocation mechanism, thereby confirming the Output Invisibility property R2) of Problem 1.

ing the second half-period. This test is particularly relevant since it probes the ability of the allocator to exploit the richer signal class $\mathbb{P}$ in the presence of time-varying references.

The plant output responses are reported in Fig. 3.13. As shown, the invisibility property holds: the plant output coincides with the response induced by the baseline controller C , regardless of the allocation mechanism. This confirms that the allocator preserves the nominal closed-loop output behavior while optimizing the inputs with respect to the cost functional.

For a more comprehensive comparison, three cases are considered:

1. the unallocated closed loop (baseline case),
2. the closed loop with an allocator designed according to [22], employing a single sinusoidal internal model of period $T = 1$, and
3. the closed loop with the spline-based allocator proposed in this chapter.

The allocated input obtained with the sinusoidal allocator is shown in Fig. 3.8, while the input obtained with the spline-based allocator is shown in Fig. 3.10. The qualitative difference between the two is evident: while the sinusoidal allocator introduces smooth harmonic corrections, the spline-based allocator generates richer trajectories capable of better adapting to the piecewise-linear nature of the reference.

This difference is quantitatively reflected in the associated costs, plotted in Fig. 3.9. The sinusoidal allocator produces only a marginal improvement over the unallocated case, reflecting the limited expressiveness of its internal model. By contrast, the spline-based allocator reduces the cost by several orders of magnitude, thereby validating the theoretical claim that spline-based internal models provide a strictly more expressive parametrization of admissible signals.

This experiment highlights the fundamental advantage of the proposed framework: by allowing polynomial extensions rather than restricting to sinusoidal harmonics, the allocator achieves substantially better steady-state input optimization while still guaranteeing

output invisibility and stability.

It is also interesting to note that the evolution of the actuator inputs in the (u_1, u_2) plane clearly highlights the effect of the proposed allocation scheme. In both reference scenarios, the action of the spline-based allocator drives the steady-state inputs deeper inside the cost-free region compared with the unallocated case, indicating a more efficient use of the available actuation space. As shown in Fig. 3.11, for the constant reference the allocated inputs remain well within the admissible low-cost area, confirming that the steady-state optimization effectively minimizes unnecessary input effort. Under the sawtooth reference (Fig. 3.12), the same behavior is observed dynamically: the trajectory of the allocated inputs gradually moves toward the center of the cost-free zone, showing how the periodic spline-based update redistributes actuator usage more evenly over time. This comparison highlights the ability of the proposed allocator to achieve smoother and more economical input distributions across different reference conditions.

3.6 Discussion and Concluding Remarks

In this chapter, a novel design methodology for DCA has been introduced. The key innovation lies in the adoption of an internal model generator capable of producing, and optimizing over, the class of signals given by T -periodic extensions of fixed-degree polynomials of time.

This choice significantly extends the classes of allocation signals considered in previous works, which were restricted to constant allocations [48] or sinusoidal signals [10], [22]. By enlarging the admissible signal set to include spline-type trajectories, the allocator achieves enhanced flexibility and expressiveness, thereby enabling superior steady-state performance.

Numerical experiments confirm the theoretical claims: in the case of constant references, the allocator reduces the deadzone cost without altering the output trajectory, while in the case of time-varying sawtooth references, the spline-based allocator outperforms sinusoidal allocation schemes by a wide margin.

The flexibility provided by polynomial and spline-type internal models paves the way for further extensions. In particular, adopting splines with multiple nodes per period would allow an even finer approximation of complex input trajectories, which is expected to play a crucial role in more demanding applications. Examples include redundant robotic manipulators, cooperative drone swarms, and other large-scale networked systems where dynamic allocation can be leveraged not only for performance improvement but also for fault tolerance and resource optimization.

The developments in this chapter therefore represent an important step toward bridging the gap between theoretical dynamic allocation frameworks and their application to complex, real-world engineering scenarios.

Chapter 4

Chapter 4

Spline-based Input Allocation on an Overactuated Tilting and Twisting Unmanned Aerial Vehicle

Abstract

This chapter investigates the performance of the developed approach in Chap. 3 applied to an overactuated quadrotor unmanned aerial vehicle (UAV) equipped with tilting and twisting propellers. In the DCA framework, actuator redundancy is systematically exploited so that the closed-loop output of the plant, as shaped by a nominal stabilizing controller, is preserved while the steady-state evolution of the control input is optimized with respect to a Lagrange cost functional. This enables the achievement of secondary objectives such as fuel economy, actuator load balancing, or energy-efficient operation without compromising the primary regulation or tracking requirements.

The case study considered here is particularly challenging because the UAV platform exhibits strongly nonlinear dynamics due to its novel thrust-vectoring mechanism. The system, originally proposed in [23], employs omnidirectional thrust-vectorized rotors that introduce additional DoF (degrees of freedom) by allowing each propeller to both tilt and twist, thereby overcoming the intrinsic maneuverability limitations of conventional multirotors. Such a configuration provides nearly full directional authority over each thrust vector, decoupling position and attitude dynamics and enabling complex maneuvers (e.g., nonlevel takeoff, inverted flight, or interaction with inclined surfaces) that are unattainable for traditional quadrotors.

The contribution of this chapter is to show that, despite the nonlinear nature of the UAV dynamics, a suitable DCA strategy enables the allocation module to be designed by leveraging tools from the Linear Time-Invariant (LTI) setting. Specifically, by employing a spline-based internal model generator, the allocator optimizes over classes of periodic piecewise polynomial signals, thus generalizing earlier allocation schemes restricted to constant or sinusoidal trajectories. This design ensures output invisibility, stability, and convergence to steady-state optimality while reducing input costs.

4.1 Introduction

While Chap. 3 established the theoretical basis for spline-based allocation in an abstract setting, this chapter turns to a demanding application domain: unmanned aerial vehicles (UAVs) with tilting-twisting propellers. These platforms are paradigmatic examples of overactuated systems, where the number of actuators exceeds the number of controlled degrees of freedom. The redundancy they offer enables agile maneuvering, fault tolerance, and novel control strategies, but it also raises the fundamental challenge of distributing effort in a way that respects actuator limits and enhances efficiency. The objective of this chapter is to demonstrate how the spline-based allocation framework can be embedded into a UAV control loop and to analyze its behavior under realistic operating conditions.

The UAV application highlights the importance of modeling and integration. After describing the system dynamics and the baseline control architecture, we incorporate allocation into the actuation layer, ensuring that the nominal trajectory tracking is unaffected while the internal actuator distribution is optimized. Two strategies are studied

in parallel: an instantaneous allocator, which selects the optimal actuator distribution at each time instant, and a spline-based allocator, which optimizes input trajectories over an extended horizon. Through simulations we investigate both idealized conditions, where actuator dynamics are neglected, and more realistic scenarios, where each actuator is modeled with first-order dynamics.

The chapter is organized as follows. Sec. 4.2 formulates the UAV allocation problem and outlines the control objectives. Sec. 4.3 details the dynamic model of the UAV and the baseline control loop. Sec. 4.4 describes the integration of allocation strategies into the UAV architecture, while Sec. 4.5 presents simulation results comparing the two methods under different conditions. Sec. 4.6 concludes with a discussion of insights gained and lessons learned.

By applying the spline-based allocator to a nonlinear aerial vehicle, this chapter shows how the abstract methodology of Chap. 3 can be made operational in practice. The lessons learned here also motivate the methodological extension of the next chapter, where allocation is no longer limited to linearized or application-specific settings but is systematically developed for nonlinear systems subject to periodic references.

4.2 UAV Architecture and Control Objectives

The principle of this research is illustrated in Fig. 4.1b, where an allocation module is introduced as an add-on on top of a pre-existing closed-loop system of the type shown in Fig. 4.1a, in which the plant is assumed to be *fat* (i.e., possessing more control inputs than regulated outputs). In this specific application, the control output consists of the total net forces and torques acting on the vehicle, comprising six components in total. These are represented as the vectors u_1 and u_2 in Fig. 4.3, where each vector contains three independent elements corresponding to the three translational and three rotational degrees of freedom.

The purpose of such augmentation is not to redesign the baseline controller, but rather to exploit the inherent input redundancy of the plant in order to achieve secondary performance objectives. Typical examples include reducing the magnitude of the control effort, distributing actuation more evenly across available actuators, or favoring the use of certain inputs that are cheaper or more reliable, while preserving the same regulated output response guaranteed by the original controller.

A natural and practically relevant setting for formulating such allocation problems is that of periodic maneuvers of duration T . In this context, the performance of a given closed-loop system can be captured by means of a Lagrange cost functional of the form

$$J = \int_t^{t+T} g(x, u, \tau) d\tau, \quad (4.1)$$

where x denotes the state of the controlled plant, u its control input, and $g(\cdot)$ is a suitably defined instantaneous cost function. If the maneuver is periodic and the closed loop is at steady state, then the value of J becomes independent of the initial time t , and the functional can be interpreted as a measure of long-term average performance.

The flexibility of the formulation (4.1) allows for a broad class of performance objectives. For instance, if the designer's aim is simply to minimize the amplitude of a suitable p -norm of the control input along the entire period, the cost can be chosen as

$$g(x, u, \tau) = \|u(\tau)\|_p,$$

with the norm $\|\cdot\|_p$ selected according to the application requirements. However, more sophisticated design goals can be accommodated by allowing g to depend explicitly on both the plant states and on time. This enables fine-grained performance tuning: for example, penalizing large attitude rates in order to reduce mechanical stress, or adopting a time-varying cost that discourages aggressive maneuvers during intervals in which the UAV is required to capture stable visual footage, while allowing faster motion in other phases of the trajectory.

The central requirement in DCA is that the cost minimization process must be achieved without altering the regulated output response imposed by the original controller. Comparing the architectures in Fig. 4.1a and Fig. 4.1b, this invisibility can be expressed as the condition

$$y = y_{alloc},$$

which states that the plant output remains unaffected by the presence of the allocator. In other words, the allocation module is invisible from the perspective of the output dynamics: while it reshapes the internal distribution of input effort, it leaves the external closed-loop behavior intact. This property is crucial to ensure that all guarantees provided by the nominal controller (such as stability, regulation, or disturbance rejection) are preserved, while redundancy is exploited exclusively to optimize secondary performance measures.

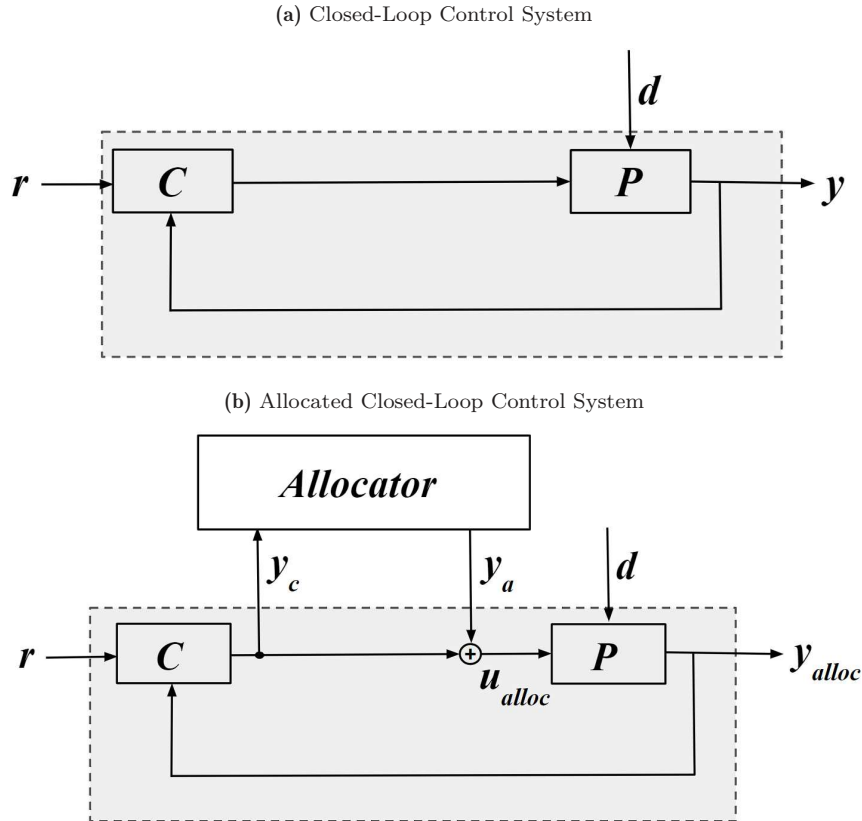


Figure 4.1: Comparison of closed-loop architectures: (a) a standard control system where the controller C directly drives the plant P ; (b) an augmented system with a dynamic control allocator ensuring output invisibility while exploiting input redundancy for performance optimization.

As in [10], [22], [48], the problem formulation involves a plant P interconnected with a stabilizing controller C . In earlier works, both P and C were assumed to be Linear Time-Invariant (LTI) systems, which facilitated the development of rigorous theoretical tools. In contrast, the present chapter addresses a more challenging scenario in which both the plant and the controller (derived from [23]) are inherently nonlinear. This nonlinear setting is particularly relevant for UAVs with thrust-vectoring mechanisms, where actuator redundancy is extreme and the system dynamics cannot be adequately described by linear models.

In this context, the spline-based DCA methodology proposed in [3] provides a promising extension. By parameterizing allocation signals as T -periodic polynomial splines, the method achieves a richer representation of admissible input trajectories compared to harmonic-based approaches. This enhanced flexibility is expected to yield improved optimization performance while preserving the structural advantages of DCA, namely output invisibility and stability.

The remainder of the chapter is devoted to analyzing this approach in detail and demonstrating its effectiveness on the considered UAV case study.

Two central insights emerge from the analysis developed in this chapter, both of which shed light on the structural features of the problem and on the advantages of spline-based DCA:

- First, by adopting a suitable choice of allocated variables namely, considering the thrust vectors q as allocation signals instead of the raw actuator commands $(\alpha_i, \beta_i, \omega_i)$ it becomes possible to formulate the allocation problem within a framework that is effectively Linear Time-Invariant (LTI). This reformulation is crucial: although both the plant and the baseline controller are intrinsically nonlinear, the allocation problem can nevertheless be addressed using the well-established tools of LTI theory. This observation significantly simplifies the allocator design and provides a rigorous justification for employing LTI-based DCA strategies in a setting that would otherwise appear intractably nonlinear.
- Second, it is important to recall that, in general, the minimization of the Lagrange cost functional (4.1) cannot be achieved by a purely *greedy* optimization strategy. In other words, selecting the instantaneous input trajectory at each time $\tau \in [t, t+T]$ as

$$u(\tau) \in \arg \min_v g(x(\tau), v, \tau)$$

does not usually guarantee optimality over the full horizon, since such a strategy ignores the temporal coupling introduced by dynamics and periodicity.

Nevertheless, for the specific UAV model analyzed in this chapter, when actuator dynamics are neglected, the greedy solution coincides with the global minimizer of the cost functional. This somewhat surprising fact highlights a structural property of the system and provides an intuitive benchmark for evaluating allocation performance. On the other hand, when actuator dynamics are explicitly taken into account, the greedy approach is no longer sufficient, and the additional flexibility afforded by the spline-based allocation method proposed in [3] becomes essential.

As will be shown in Sec. 4.5, this advanced approach achieves superior results by exploiting richer classes of admissible signals while preserving stability and output invisibility.

4.3 Dynamics and Baseline Controller

The UAV platform introduced in [23] and schematically represented in Fig. 4.2 is briefly described in this section, with emphasis on the notational conventions and modeling assumptions adopted throughout the chapter.

Additional details on the derivation of the full nonlinear model can be found in the original reference [23].

4.3.1 Preliminaries

The plant P considered in this chapter is a quadrotor UAV equipped with four propellers. Each propeller assembly consists of one brushless motor that generates thrust, complemented by two servomotors that provide additional degrees of freedom: one for *tilting* the propeller around an axis by an angle β_i , and another for *twisting* the propeller by an angle α_i . Consequently, for each propeller indexed by $i \in \{1, \dots, 4\}$, three independent inputs are available: the twist angle α_i , the tilt angle β_i , and the rotational speed ω_i of the propeller. Altogether, the UAV therefore possesses a total of 12 control inputs, which are employed to regulate six primary outputs namely the three translational coordinates of the CoM (center of mass) and the three components of the vehicle's orientation. This high degree of redundancy makes the UAV an archetypal benchmark for the study of DCA.

The geometric configuration of the UAV is described with respect to two coordinate frames. The inertial frame, or world frame F_W , is denoted by $F_W = (O_W, [x_W, y_W, z_W])$ and is assumed to be fixed to the ground. The body frame F_B , denoted by $F_B = (O_B, [x_B, y_B, z_B])$, is rigidly attached to the UAV, with its origin O_B displaced by an orthogonal distance l_h upwards from the center of mass O_c . The relation between the orientation of the four propellers and the body frame can be described in terms of rotations around the vertical axis. In particular, the rotation matrix associated with the i -th propeller is defined as

$$R_i = R_z \left[(i-1) \frac{\pi}{2} \right], \quad (4.2)$$

where $R_z(\cdot)$ denotes an elemental rotation around the z -axis. This representation reflects the symmetry of the four-propeller arrangement, with each propeller placed 90° apart around the UAV body.

The translational kinematics of the UAV in F_W are described by the position vector $p = [x, y, z]^\top$, the velocity $\dot{p} = [\dot{x}, \dot{y}, \dot{z}]^\top$, and the acceleration $\ddot{p} = [\ddot{x}, \ddot{y}, \ddot{z}]^\top$. The attitude of the UAV is described by the rotation matrix $R \in \text{SO}(3)$, which represents the orientation of the body frame F_B relative to the world frame F_W [45]. The angular velocity of the body, expressed in the body frame, is denoted by $\Omega \in \mathbb{R}^3$. The relation between the time derivative of R and the angular velocity Ω is given by

$$\dot{R} = R\Omega^\wedge, \quad (4.3)$$

where the operator $(\cdot)^\wedge : \mathbb{R}^3 \rightarrow \text{SO}(3)$ maps a three-dimensional vector into a 3×3 skew-symmetric matrix according to

$$x^\wedge = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}. \quad (4.4)$$

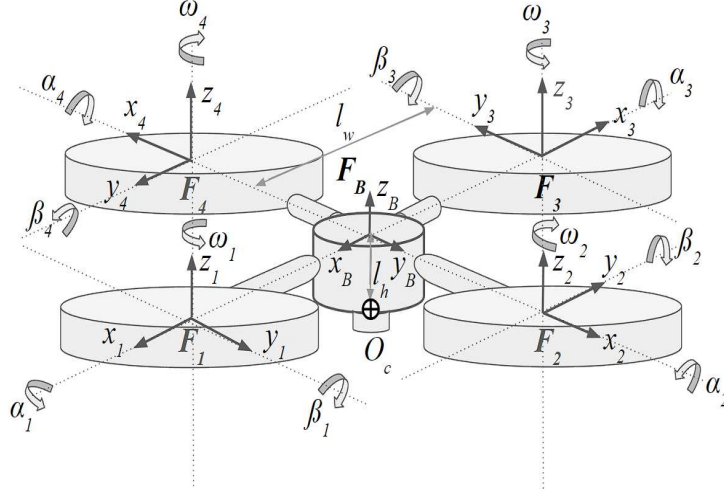


Figure 4.2: Schematic representation of the UAV with tilting and twisting propellers. The illustration highlights the world frame F_W , the body frame F_B , and the twelve independent control inputs: twist angles α_i , tilt angles β_i , and propeller angular velocities ω_i , for $i = 1, \dots, 4$.

This operator satisfies the property $x^\wedge y = x \times y$ for all $x, y \in \mathbb{R}^3$, where $\times$ denotes the cross product. The inverse operator $(\cdot)^\vee : SO(3) \rightarrow \mathbb{R}^3$ retrieves the vector from its skew-symmetric representation.

4.3.2 UAV Dynamics

The UAV dynamics are determined by the forces and moments generated by the four propellers, each of which can be reoriented through tilt and twist angles. The i -th propeller produces a thrust force of magnitude proportional to the square of its angular velocity, expressed as

$$t_i = c_p \omega_i^2 \hat{t}_i, \quad (4.5)$$

where $c_p > 0$ is a thrust coefficient depending on the aerodynamic characteristics of the propeller, ω_i is the angular velocity of the i -th rotor, and $\hat{t}_i \in \mathbb{R}^3$ is the unit vector aligned with the thrust direction determined by the tilt and twist angles (α_i, β_i) . Thus, by varying both ω_i and the orientation $\hat{t}_i$, the UAV can generate thrust in directions not accessible to standard quadrotors, providing significant redundancy in achievable force distributions.

In addition to thrust, each propeller also generates an aerodynamic drag moment aligned with the thrust direction due to the resistance of air. This torque is modeled as

$$\tau_i = (-1)^i c_\tau \omega_i^2 \hat{t}_i, \quad (4.6)$$

where $c_\tau > 0$ is a drag coefficient, identical across all propellers, and the alternating factor $(-1)^i$ accounts for the fact that adjacent propellers rotate in opposite directions (two clockwise, two counterclockwise). This torque represents the reaction that must be balanced by the structure or other propellers to prevent undesired yawing motions.

4.3.2.1 Translational dynamics

The translational motion of the UAV's center of mass in the inertial frame F_W follows from Newton's second law, yielding the linear equation of motion

$$m\ddot{p} = mg_0 + R \sum_{i=1}^4 R_i t_i + F_d, \quad (4.7)$$

where m is the total mass of the UAV, $g_0 \in \mathbb{R}^3$ is the gravitational acceleration vector, and $F_d \in \mathbb{R}^3$ represents external disturbances such as aerodynamic drag, gusts, or wind effects. The term $R \sum_{i=1}^4 R_i t_i$ is the total thrust expressed in the inertial frame, obtained by transforming the individual thrusts t_i (defined in the propeller-fixed frames) into the world coordinates via the body orientation R and the placement matrices R_i .

4.3.2.2 Rotational dynamics

The angular motion of the UAV in the body-fixed frame F_B can be described using Euler's rigid-body equations, which account for the coupling between angular velocity and angular momentum. The resulting equation is

$$\mathbb{I}\dot{\Omega} + h = \tau_g + \sum_{i=1}^4 [d_i \times (R_i t_i) + R_i \tau_i] + \tau_d, \quad (4.8)$$

where $\mathbb{I} \in \mathbb{R}^{3 \times 3}$ is the symmetric, positive-definite inertia matrix of the UAV, and

$$h \equiv \Omega \times (\mathbb{I}\Omega)$$

is the gyroscopic term representing the rate of change of angular momentum due to the nonlinear coupling of Ω and $\mathbb{I}$.

The first contribution on the right-hand side, $\tau_g \in \mathbb{R}^3$, is the gravitational torque that depends on the location of the body frame relative to the center of mass. In this study, we adopt the assumption $O_B = O_C$ as in [23], which implies that $\tau_g = 0$. The second contribution, $\sum_{i=1}^4 d_i \times (R_i t_i)$, represents the moments generated by the thrust forces acting at the propeller positions d_i relative to the center of mass. The term $\sum_{i=1}^4 R_i \tau_i$ corresponds to the aerodynamic drag torques of the individual propellers, expressed in the body frame. Finally, $\tau_d \in \mathbb{R}^3$ accounts for external disturbance torques, which may arise from aerodynamic effects such as wind shear, turbulence, or blade flapping phenomena.

Equations (4.7) and (4.8) together describe the full nonlinear dynamics of the UAV. In contrast to conventional quadrotors, which are underactuated (four control inputs for six degrees of freedom), the tilting-twisting configuration introduces 12 control inputs to regulate 6 outputs. This redundancy enables greater flexibility in shaping the input trajectories but also introduces the need for systematic allocation strategies. It is precisely in this context that DCA methods become valuable, as they can exploit the additional degrees of freedom to optimize steady-state performance while preserving the desired closed-loop output behavior.

4.3.3 Control Architecture

The control architecture adopted from [23] is illustrated in Fig. 4.3. It provides a systematic framework that links high-level trajectory generation with low-level actuator commands, ensuring stability and tracking while exposing the system's inherent input

redundancy. The architecture is structured into four main layers: reference generation, feedback control law, wrench mapping, and actuator-level conversion (including inverse kinematics and actuator dynamics).

This layered design is particularly convenient because it separates trajectory tracking objectives from allocation strategies, thereby allowing new allocation schemes such as the spline-based DCA considered in this chapter to be integrated seamlessly. In the following the role of each block has been explained.

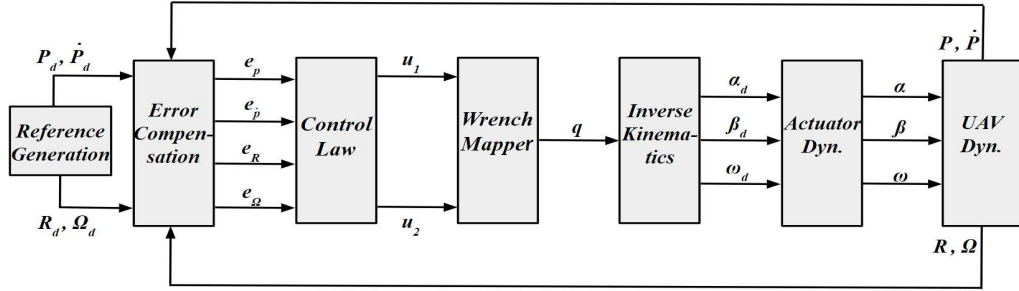


Figure 4.3: Control architecture of the UAV, adapted from [23]. The diagram highlights the cascade of reference generation, feedback control, wrench mapping, inverse kinematics, and actuator dynamics.

4.3.3.1 Reference Generation And Error Computation

The reference generator specifies the desired UAV trajectory in terms of position $p_d = [x_d, y_d, z_d]^\top$ and velocity $\dot{p}_d = [\dot{x}_d, \dot{y}_d, \dot{z}_d]^\top$ in the inertial frame F_W , as well as desired attitude $R_d \in SO(3)$ and angular velocity $\Omega_d \in \mathbb{R}^3$ in the body frame F_B .

4.3.3.2 Error Compensation

To evaluate tracking performance, four error signals are defined:

$$e_p = p_d - p, \quad (4.9a)$$

$$e_v = \dot{p}_d - \dot{p}, \quad (4.9b)$$

$$e_R = (R^\top R_d - R_d^\top R)^\vee, \quad (4.9c)$$

$$e_\Omega = \Omega_d - \Omega, \quad (4.9d)$$

corresponding to position, velocity, attitude, and angular velocity deviations, respectively. The use of the vee-map $(\cdot)^\vee$ ensures that the matrix-valued attitude error is represented as a vector in $\mathbb{R}^3$, which is more suitable for control design.

4.3.3.3 Baseline Control Law

The feedback law is a proportional-derivative PD controller that computes a *virtual input* in the form of a wrench $[u_1^\top \ u_2^\top]^\top$, consisting of a net force $u_1 \in \mathbb{R}^3$ and a net torque $u_2 \in \mathbb{R}^3$. The expressions are:

$$u_1 = mR^\top (g_0 - K_p e_p - K_v e_v), \quad (4.10a)$$

$$u_2 = h - \tau_g - K_R e_R - K_\Omega e_\Omega, \quad (4.10b)$$

where K_p , K_v , K_R , and K_Ω are symmetric positive-definite gain matrices. The position and attitude subsystems are fully decoupled, which simplifies both design and stability analysis. Increasing the gain matrices accelerates error convergence, but excessive gain magnitudes amplify measurement noise and may push the system closer to instability [41].

4.3.3.4 Wrench Mapping

The wrench $[u_1^\top \ u_2^\top]^\top \in \mathbb{R}^6$ must be mapped into the propeller thrust vectors $q = [q_1^\top, \dots, q_4^\top]^\top \in \mathbb{R}^{12}$ through a static control allocation (SCA) step. This is performed by the *Wrench Mapper*, which is derived directly from the translational and rotational dynamics (4.7)–(4.8). Defining $q_i = \omega_i^2 R_i \hat{t}_i$ as the effective thrust vector of the i -th propeller, one obtains

$$m\ddot{p} = mg_0 + Ru_1, \quad (4.11a)$$

$$\mathbb{I}\dot{\Omega} + h = \tau_g + u_2, \quad (4.11b)$$

with

$$u_1 = c_p \sum_{i=1}^4 q_i, \quad (4.12a)$$

$$u_2 = c_p \sum_{i=1}^4 d_i^\wedge q_i + c_\tau \sum_{i=1}^4 (-1)^i q_i. \quad (4.12b)$$

Thus the relation between wrench and thrusts can be compactly expressed as

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = Aq, \quad A = \begin{bmatrix} c_p I_3 & \cdots & c_p I_3 \\ J_1 & \cdots & J_4 \end{bmatrix}, \quad (4.13)$$

where $J_i = c_p d_i^\wedge + (-1)^i c_\tau I_3 \in \mathbb{R}^{3 \times 3}$. Since A has full row rank ($\text{rank}(A) = 6$), it admits a right pseudoinverse:

$$q = A^\dagger \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad A^\dagger = A^\top (AA^\top)^{-1}. \quad (4.14)$$

This static allocation step ensures feasibility but does not exploit dynamic redundancy, motivating the introduction of DCA in later sections.

4.3.3.5 Inverse Kinematics

Given the thrust vectors q_i , the next step is to determine the actuator-level commands $(\alpha_i, \beta_i, \omega_i)$. This requires solving an inverse kinematics problem for the tilt and twist angles. Denoting $\hat{z} = [0, 0, 1]^\top$ as the unit thrust vector in the local propeller frame F_i , one has

$$\hat{t}_i = R_x(\alpha_i) R_y(\beta_i) \hat{z} = R_p(\alpha_i, \beta_i) \hat{z},$$

which expands to

$$\hat{t}_i = \begin{bmatrix} \sin(\beta_i) \\ -\sin(\alpha_i) \cos(\beta_i) \\ \cos(\alpha_i) \cos(\beta_i) \end{bmatrix}.$$

The inverse problem of finding (α_i, β_i) admits two generic solutions and one singular solution. The generic solutions are:

$$\alpha = \text{atan2}\left(\mp \frac{\hat{t}_{i,y}}{\sqrt{1 - \hat{t}_{i,x}^2}}, \pm \frac{\hat{t}_{i,z}}{\sqrt{1 - \hat{t}_{i,x}^2}}\right), \quad (4.15a)$$

$$\beta = \text{atan2}\left(\hat{t}_{i,x}, \pm \sqrt{1 - \hat{t}_{i,x}^2}\right), \quad (4.15b)$$

while the singular case occurs when $\hat{t}_{i,x} = \pm 1$, in which case $\beta = \pm \pi/2$ and α is undetermined in $[-\pi, \pi]$.

4.3.3.6 Actuator Dynamics

Finally, the actuator dynamics determine how the commanded tilt and twist angles are realized. Two scenarios are typically considered:

- *Slow dynamics*: both α_i and β_i evolve as first-order systems with non-negligible time constants, introducing delays.
- *Fast (ideal) dynamics*: tilt and twist are assumed instantaneous, eliminating actuator lag.

In the ideal case, the two generic inverse kinematics solutions coincide. In the more realistic case with dynamics, the Inverse Dynamics block selects the configuration closest to the current actuator state to ensure smooth evolution.

The architecture in Fig. 4.3 guarantees reference tracking by combining PD feedback with static allocation and inverse kinematics. However, the use of a purely static pseudoinverse mapping in (4.14) does not exploit the full potential of input redundancy. This motivates the adoption of DCA, which augments the architecture with a dynamic optimization layer to minimize steady-state costs while preserving output invisibility.

4.4 Allocated Wrench Mapper

In this section, two classes of DCA strategies are presented and discussed for the UAV system under study.

It is important to emphasize that, within the adopted control architecture introduced in this chapter, the action of DCA is introduced *after* the static allocation stage performed by the Wrench Mapper block. This modified structure is depicted in Fig. 4.4, where the additional allocation signal is injected following the pseudoinverse-based SCA stage. In this way, DCA augments but does not replace the baseline allocation mechanism, ensuring that the nominal stability and trajectory tracking performance of the closed-loop remain unchanged, while redundancy is exploited to achieve secondary performance improvements.

Following the general framework of [10], similar to Chap. 3, the DCA allocator is decomposed into three conceptual blocks:

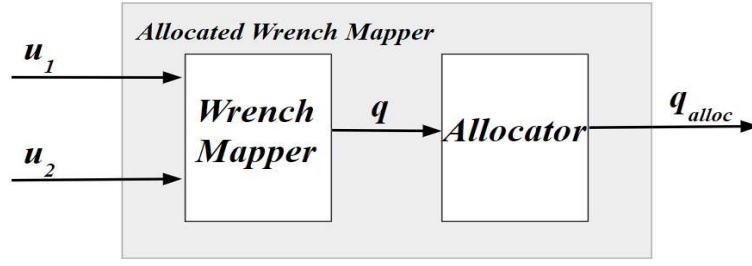


Figure 4.4: Modified control architecture of the UAV with DCA included after the Wrench Mapper block.

- the *Annihilator*, which ensures *output invisibility* by suitably filtering the signals injected into the plant so that the regulated output is left unchanged;
- the *Steady-State Generator*, which parameterizes the class of allocation signals used in the optimization process by means of an internal model (as in the theory of output regulation). Different internal models correspond to different families of signals: constant [48], sinusoidal [22], spline-based [3], and so on;
- the *Optimizer*, which applies an online optimization algorithm (most often a gradient descent law) to drive the parameters of the Steady-State Generator toward the minimizer of the cost functional (4.1).

This modular structure allows tailoring the allocation strategy to both the plant's dynamics and the performance criteria of interest.

In principle, DCA could be designed to inject additional signals directly into the actuator-level inputs $(\alpha_i, \beta_i, \omega_i)$. However, such an approach would act on the nonlinear dynamics of both the translational and rotational equations of motion in a highly coupled fashion, leading to a complex and computationally intensive design problem. For this reason, a more effective strategy is to apply DCA at the level of the *thrust vector* q , i.e., directly after the Wrench Mapper stage.

Remark 4.4.1

For all fully redundant UAVs with fixed propeller geometry, a relation of the form (4.13) holds, where the wrench matrix A is full row rank. This property has a particularly desirable consequence: if the DCA acts directly on the thrust distribution q by adding an allocation signal u_{alloc} , then the annihilator can be chosen as a purely static map.

Specifically, any full-column-rank matrix whose columns form a basis of the null space of A is sufficient. With such a choice, the condition $y = y_{alloc}$ (output invisibility) is automatically guaranteed, since $Au_{alloc} = 0$ ensures that the added allocation input does not affect the wrench applied to the plant. As a result, the allocator can redistribute thrust commands among actuators without altering the plant's regulated output trajectory. This property enables performance-oriented reallocation strategies, such as reducing energy consumption, promoting actuator reliability, or penalizing the use of certain actuators relative to others [33], [48]. $\square$

Remark 4.4.2

Considering fully redundant UAVs with fixed propeller geometry, and under the

condition that the wrench matrix A has full row rank (see Remark 4.4.1), it follows that when the input q is taken as the input of the augmented plant (the cascade of the inverse kinematics and the UAV), the system becomes strongly redundant [48].

Strong redundancy means that allocation under output invisibility implies not only invisibility of the regulated output but also invisibility of the entire state trajectory. In other words, reallocating inputs by adding u_{alloc} affects neither the steady-state output nor the internal dynamics of the plant. An immediate and useful consequence of this property is that the minimization of the Lagrange cost (4.1) can, in such cases, be achieved by simple greedy optimization. That is, the allocator may select at each instant

$$u(t) \in \arg \min_v g(x(t), v, t),$$

without resorting to an explicit Steady-State Generator or a finite-dimensional parameterization of candidate allocation signals. This observation is particularly powerful, as it allows designing allocation strategies that are both computationally efficient and provably optimal in steady state, at least when actuator dynamics are neglected. $\square$

Together, Remark 4.4.1 and 4.4.2 highlight that the peculiar structure of the UAV with tilting-twisting propellers provides an unusually favorable setting for DCA. By operating on the thrust distribution q rather than directly on actuator-level inputs, the allocator can exploit linear algebraic properties of the wrench mapping to guarantee output invisibility in a straightforward manner.

Moreover, strong redundancy ensures that steady-state optimization problems can sometimes be solved by simple greedy methods, though in practice actuator dynamics often reintroduce the need for richer allocation strategies, such as the spline-based approach discussed in Sec. 4.5.

Based on the above discussion, in this chapter two distinct DCA methods have been implemented and tested on the UAV system. These two methods embody different levels of architectural complexity, yet they are both consistent with the structural properties of the plant and the performance objectives described earlier.

First method: instantaneous greedy DCA. The first approach leverages the structural observations discussed in Remark 4.4.1 and 4.4.2, namely the existence of a full-row-rank wrench matrix A and the ensuing strong redundancy of the UAV. In this case, it is possible to design a DCA strategy based entirely on greedy optimization of the instantaneous running cost associated with the periodic Lagrange functional (4.1).

No dynamics are introduced in either the annihilator or the steady-state generator, since output invisibility is enforced algebraically by restricting allocation inputs to the null space of A . Similarly, the optimizer reduces the instantaneous cost at each time instant. This solution is computationally efficient and is particularly attractive when actuator dynamics are negligible or can be approximated as instantaneous.

Second method: dynamic spline-based DCA. The second approach follows the general architecture for DCA proposed in Chap. 3 (see also [3]).

Here, the annihilator is again chosen as a static operator acting in the kernel of the wrench matrix A (thus guaranteeing output invisibility), but the optimizer and steady-state generator are dynamic. Specifically, the desired periodic trajectory is divided into multiple time segments, on each of which a polynomial representation is adopted. The concatenation of these polynomials yields a spline parameterization of the allocation

signals. A gradient-based optimization algorithm is then employed to minimize the Lagrange cost (4.1) over this finite-dimensional class of candidate signals. This approach is richer than the greedy allocation method, as it allows optimization over an entire period rather than instantaneously, and is particularly effective in the presence of actuator dynamics or when periodic trajectories are introduced.

In both allocation strategies, the running cost has been chosen as a quadratic penalty on the use of the propellers, with the aim of biasing the allocator against excessive reliance on the first propeller. The cost is therefore defined as

$$\sum_{i=1}^4 c_i \omega_i^2 = \omega^\top \bar{Q} \omega = q^\top Q q, \quad (4.16)$$

where $\bar{Q} = \text{diag}(c_1, c_2, c_3, c_4)$ with strictly positive weights $c_i > 0$, $i = 1, \dots, 4$, and $Q = \bar{Q} \otimes I_3$ with $\otimes$ denoting the Kronecker product. By suitably tuning the coefficients c_i , one can explicitly penalize or promote the use of specific propellers, thereby incorporating preferences or reliability considerations into the allocation strategy.

In summary, the first method provides a lightweight, greedy DCA solution that is optimal under strong redundancy and negligible actuator dynamics, while the second method implements a more general spline-based architecture that achieves improved performance in more realistic conditions. The comparative evaluation of these two approaches, reported in Sec. 4.5, illustrates the trade-offs between simplicity and flexibility in the design of DCA strategies.

4.4.1 Instantaneous Control Input Allocation

In this subsection, we introduce the first allocation strategy, referred to as the Instantaneous Control Input Allocator. The key feature of this method is that it optimizes the distribution of control inputs at every time instant independently, without introducing any additional dynamics.

The main advantage of this approach is its computational efficiency and conceptual simplicity: the allocation is carried out by solving a quadratic optimization problem at each instant along the trajectory.

The quadratic cost to be minimized is defined as

$$J_{INST} = \omega^\top \bar{Q} \omega \quad (4.17a)$$

$$= q_{alloc}^\top Q q_{alloc} \quad (4.17b)$$

$$= (q + A^\perp v)^\top Q (q + A^\perp v), \quad (4.17c)$$

where $q_{alloc} = q + A^\perp v$ is the thrust distribution after allocation, and q is the baseline thrust vector produced by the pseudoinverse mapping of the Wrench Mapper.

Here, $A^\perp \in \mathbb{R}^{12 \times 6}$ denotes a basis for the null space of the wrench matrix A (e.g., computed in MATLAB with `null(A)`), which serves as the static annihilator ensuring output invisibility. The matrix $Q \in \mathbb{R}^{12 \times 12}$ is symmetric positive definite and defines the specific performance criterion under consideration. Finally, v is a free parameter vector, chosen at each time step to minimize the instantaneous cost.

By differentiating J_{INST} with respect to v and setting the gradient equal to zero, one obtains the optimal allocation input:

$$\nabla_v J_{INST} = 2A^{\perp\top}Q(q + A^{\perp}v) \stackrel{!}{=} 0, \quad (4.18a)$$

$$\implies v = -2(A^{\perp\top}QA^{\perp})^{-1}A^{\perp\top}Qq. \quad (4.18b)$$

This closed-form expression highlights that the optimal v is a linear function of the baseline thrust vector q . The computation involves only matrix multiplications and the inversion of the reduced Hessian ($A^{\perp\top}QA^{\perp}$), which is guaranteed to be invertible since Q is positive definite and $A^{\perp}$ has full column rank.

Once the optimal v has been computed, the modified thrust vector $q_{alloc} = q + A^{\perp}v$ is fed into the Inverse Kinematics block of the control architecture, replacing the baseline thrust q . This ensures that the reallocation affects only the actuator commands, while the plant's output response remains unchanged due to the annihilation property $AA^{\perp}v = 0$. As a result, the UAV continues to follow the same reference trajectory, but the actuator usage is optimized according to the chosen quadratic cost.

This method effectively minimizes the weighted squared usage of the propellers, where the weights c_i embedded in Q can be tuned to penalize or favor particular actuators. For example, one may penalize the use of a specific propeller known to be less reliable or more energy-consuming. Thus, the instantaneous allocator provides a simple yet powerful way to exploit redundancy for energy saving, load balancing, or fault-tolerance, while remaining computationally lightweight and directly implementable in real time.

4.4.2 Periodic Control Input Allocation

In contrast to the instantaneous allocator described in Sec. 4.4.1, the second approach adopts a *periodic* perspective, in which the allocation problem is formulated over an entire maneuver period of length T . The central idea is to minimize a Lagrangian cost functional defined over the complete period, rather than optimizing instantaneously at each time step.

This approach, which follows the framework introduced in [3], allows the allocator to exploit temporal correlations along the trajectory and to generate allocation signals that are optimal with respect to the *entire periodic behavior* of the closed-loop system. For brevity, we only recall here the main ingredients needed to bridge the general theory with the specific UAV application studied in this chapter; further details can be found in Chap. 3.

The design of the Steady-State Generator plays a crucial role, since it parameterizes the family of allocation signals considered by the optimizer. Different choices of internal models allow capturing different classes of periodic behaviors. For example, one may select a single sinusoidal internal model, a spline-based representation, or even divide the overall period T into several smaller subintervals, with different models applied on each subinterval to better match the plant's steady-state response (see Sec. 4.5). In general, the dynamics of the internal model on a single segment can be represented in state-space form as

$$\dot{x}_{im} = A_{im}x_{im}, \quad (4.19a)$$

$$y_{im} = C_{im}x_{im}, \quad (4.19b)$$

where x_{im} denotes the state of the internal model and y_{im} its output. At the beginning

of each period, the initial state x_{im0} is updated by the optimizer so as to minimize the cost functional.

The Lagrangian cost to be minimized is defined as

$$J = \int_0^T \omega^\top(t) \bar{Q} \omega(t) dt = \int_0^T (q + A^\perp \Psi_{im} x_{im0})^\top Q (q + A^\perp \Psi_{im} x_{im0}) dt,$$

where $\Psi_{im} := C_{im} e^{A_{im} t}$ is the transition operator mapping the initial state x_{im0} to the steady-state trajectory of the internal model, and $A^\perp$ is the annihilator ensuring output invisibility. As before, $Q \in \mathbb{R}^{12 \times 12}$ is a positive-definite weighting matrix that encodes the chosen actuator penalties.

Differentiating J with respect to x_{im0} yields

$$\nabla_{x_{im0}} J = 2 \int_0^T (q + A^\perp \Psi_{im} x_{im0})^\top Q A^\perp \Psi_{im} dt, \quad (4.20)$$

which, after transposition, gives

$$\nabla_{x_{im0}}^\top J = 2 \int_0^T \Psi_{im}^\top A^{\perp\top} Q (q + A^\perp \Psi_{im} x_{im0}) dt. \quad (4.21)$$

In the general DCA framework, this gradient is used to iteratively update x_{im0} via a gradient descent law, ensuring convergence to the optimal allocation. In the present context, however, the problem admits a closed-form solution: one can directly impose that (4.21) vanishes and solve for x_{im0} . This yields

$$\begin{aligned} x_{im0} &= M^{-1} \int_0^T \Psi_{im}^\top A^{\perp\top} Q q dt, \\ M &= \int_0^T \Psi_{im}^\top A^{\perp\top} Q A^\perp \Psi_{im} dt, \end{aligned}$$

where M is a symmetric positive-definite matrix that depends only on the choice of internal model and the weighting matrix Q . Importantly, M can be computed once offline, thereby reducing the online computational burden.

The periodic allocator thus optimizes actuator usage over an entire period. Compared with the instantaneous allocator, which selects v greedily at each time step, the periodic method is capable of producing smoother and more coordinated allocation signals, particularly in the presence of actuator dynamics or when the system is subject to periodic references and disturbances.

With all ingredients defined, the next section presents simulation studies illustrating the performance of both allocation strategies under different trajectory scenarios, highlighting the trade-offs between computational simplicity and improved performance.

4.5 Flight Simulations

In this section, the proposed control allocation techniques have been implemented and validated in *Simulink* through a series of simulation experiments. The goal of these

simulations is twofold: first, to demonstrate the ability of both instantaneous and periodic allocation strategies to preserve the nominal output tracking performance of the UAV, and second, to highlight the improvements in performance achieved by exploiting each DCA method.

The UAV is commanded to follow a three-dimensional reference trajectory that combines planar and vertical components. In the horizontal plane (x - y), the reference describes a square-shaped path, while in the vertical direction (z), the UAV performs a sinusoidal oscillation. This composite trajectory is illustrated in Fig. 4.5 and has been selected to excite both translational and rotational dynamics, thereby providing a meaningful benchmark for allocation strategies.

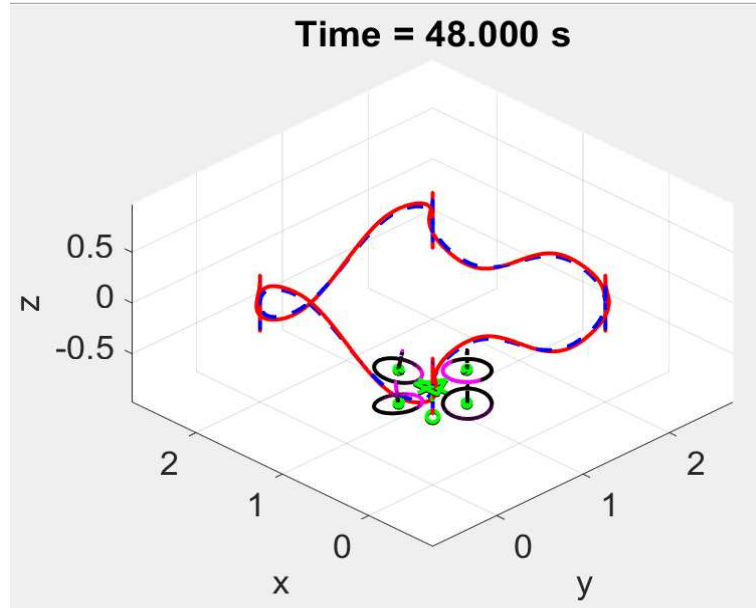


Figure 4.5: Reference trajectory used in the simulation experiments, consisting of a square motion in the x - y plane combined with a sinusoidal variation along the z axis.

Simulation parameters. Unless otherwise stated, all experiments use the UAV specifications reported in [23], which are summarized below:

- Total system mass: $m = 5$ kg.
- Physical dimensions: $l_h = 0.3$ m (vertical offset of propellers from center of mass), $l_w = 0.4$ m (arm length).
- Propeller thrust constant: $c_p = 9.14 \times 10^{-5}$ kg · m.
- Propeller torque constant: $c_\tau = 1.0 \times 10^{-4}$ kg · m².
- Body inertia: $\mathbb{I} = \text{diag}(0.08, 0.08, 0.15)$ kg/m².
- Positional control gains: $K_p = 10I_3$ s⁻², $K_v = 5I_3$ s⁻¹.
- Angular control gains: $K_R = 10I_3$ N · m, $K_\Omega = 5I_3$ kg · m² · s⁻¹.
- Joint rate limits: $\dot{\alpha}_{\max} = \dot{\beta}_{\max} = 10$ rad/s.
- Propeller angular velocity limits: $\omega_{\min} = 80$ rad/s, $\omega_{\max} = 525$ rad/s.

Both the instantaneous and periodic allocation strategies have been tested under two different actuator models:

1. *Ideal actuation*: changes in tilt angle α_i , twist angle β_i , and propeller speed ω_i are assumed instantaneous, i.e., no actuator dynamics are present.
2. *Dynamic actuation*: more realistic first-order actuator dynamics are introduced, with time constants

$$\tau_\alpha = 0.1 \text{ s}, \quad \tau_\beta = 0.1 \text{ s}, \quad \tau_\omega = 0.08 \text{ s},$$

representing the twist, tilt, and rotational velocity dynamics of each propeller, respectively.

This two-case analysis provides insight into the performance of the allocation schemes under both optimistic and realistic conditions, and allows us to quantify the robustness of the proposed methods with respect to actuator dynamics. The results of these simulation studies are presented and compared in the following subsections.

4.5.1 Scenario 1: Instantaneous Allocation With and Without Dynamics

In the first simulation scenario, the goal is to reduce the usage of the first propeller by appropriately redistributing control effort among the remaining actuators. This objective is captured by selecting a quadratic cost function of the form (4.17a), with weighting matrix

$$\bar{Q} = \text{diag}(10, 1, 1, 1).$$

The first diagonal entry is intentionally assigned a value ten times larger than the others, thereby penalizing the angular velocity of the first propeller more heavily than that of the others. In this way, the allocator is biased to minimize the workload of the first actuator, which may reflect, for instance, a reliability concern or an energy-efficiency requirement.

Two cases are examined:

1. *Ideal actuation*: the tilt, twist, and angular velocity dynamics of the propellers are assumed to be instantaneous.
2. *Dynamic actuation*: the actuators are modeled as first-order systems with the time constants defined in Sec. 4.5.

Figures 4.6 and 4.7 illustrate the evolution of the cost function with and without allocation, for the instantaneous and with first order dynamics cases, respectively. In both situations, the introduction of allocation results in a clear reduction of the cost relative to the baseline (non-allocated) case. Interestingly, the relative performance of the two allocation strategies (instantaneous versus periodic) depends strongly on whether actuator dynamics are present.

In the *absence of dynamics*, the instantaneous greedy allocation achieves a lower cost than the periodic allocator. This is consistent with the theoretical result discussed in Sec. 4.4, where instantaneous greedy optimization is provably optimal under strong redundancy when actuator dynamics are neglected.

In the *presence of dynamics*, however, the periodic allocator significantly outperforms the instantaneous method. This is expected, since the periodic approach accounts for temporal correlations across the maneuver and can therefore compensate for the delays introduced by actuator dynamics, whereas the instantaneous method cannot.

Figures 4.8 and 4.9 show the angular velocities of the individual propellers before and after allocation. It can be observed that the thrust command of the first propeller

is marginally reduced after allocation, while the workload is redistributed among the remaining three propellers. This confirms that the allocator successfully enforces the desired prioritization embedded in the cost function, without altering the output trajectory of the UAV.

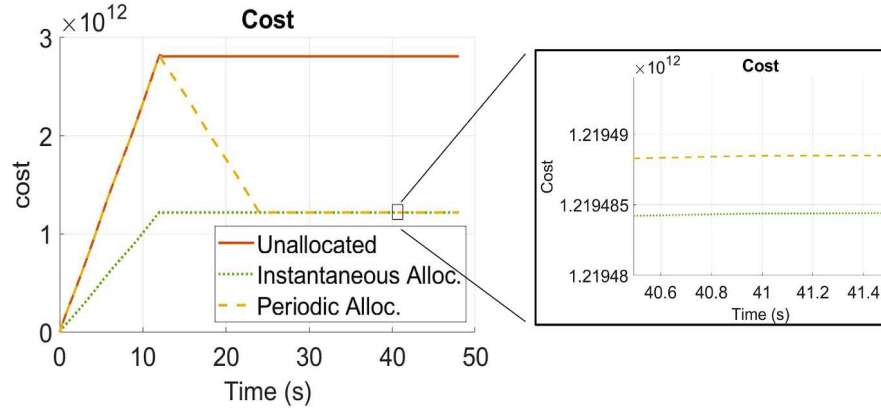


Figure 4.6: Comparison of costs in the instantaneous actuation case. The cost is minimized by both allocation strategies, with the instantaneous greedy allocator performing better than the periodic one when no actuator dynamics are present.

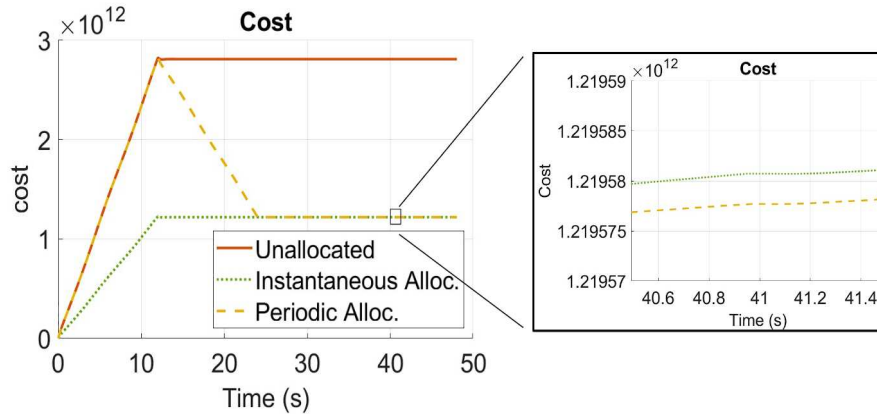


Figure 4.7: Comparison of costs in the dynamic actuation case. When actuator dynamics are present, the periodic allocator outperforms the instantaneous one, achieving a lower cost.

4.5.2 Scenario 2: Periodic Allocation With and Without Dynamics

The second scenario investigates the performance of the spline-based periodic allocator described in Sec. 4.4.2. In contrast to Scenario 1, where allocation was performed greedily at each time instant, here the allocator optimizes actuator usage over an entire period of the maneuver. This enables the allocator to account for temporal correlations and to generate smoother and more coordinated actuator commands.

The UAV trajectory corresponds to the same square-shaped maneuver used previously, now considered as consisting of 4 identical periods of duration $T = 12$ s. During each period, the UAV performs a sinusoidal motion in the z axis while simultaneously completing one edge of the square in the x - y plane and executing a 90° counter-clockwise yaw rotation at the corner. To facilitate spline-based allocation, each period is divided into six subperiods: three associated with traversing the straight edges (accelerating, constant velocity, decelerating) and three associated with performing the corner turns (again, accelerating,

constant, decelerating). This segmentation ensures that the steady-state generator captures the dominant phases of the maneuver and can adapt allocation strategies accordingly.

On each subperiod, allocation is performed by a Spline-Type Steady-State Generator [3]. Specifically, the allocator represents the allocation input as a polynomial trajectory of fixed degree r within each segment. In our implementation, a 6th-degree polynomial ($r = 6$) was chosen, balancing accuracy and computational complexity: higher-degree splines can provide better approximation of the optimal allocation trajectory but at the cost of increased offline and online computations. The internal model dynamics for each segment are given by (4.19a), with parameters:

$$A_{im} = I_{n_u - n_y} \otimes J_{0,r}, \quad (4.22a)$$

$$C_{im} = I_{n_u - n_y} \otimes \begin{bmatrix} 1 & 0_{1 \times (r-1)} \end{bmatrix}, \quad (4.22b)$$

where $\otimes$ denotes the Kronecker product, $I_{n_u - n_y}$ is the identity matrix of size equal to the redundancy ($n_u - n_y$), and $J_{0,r}$ is a Jordan block of size r with zero eigenvalue. This choice yields a spline basis for the allocation signal, with continuity properties that guarantee smooth transitions between subperiods.

As in Scenario 1, the allocation objective is to reduce the reliance on the first propeller. The cost functional is therefore the periodic quadratic cost (4.20), with $\bar{Q} = \text{diag}(10, 1, 1, 1)$. This again penalizes the angular velocity of the first propeller more heavily than the others, biasing the allocator to redistribute effort among the remaining actuators.

Figures 4.8 and 4.9 illustrate the angular velocities of the four propellers with and without actuator dynamics. In the case of ideal actuation (Fig. 4.8), both the instantaneous and periodic allocators reduce the angular velocity of the first propeller, although the instantaneous method achieves a lower cost (consistent with the theoretical optimality of greedy allocation when no dynamics are present). When actuator dynamics are included (Fig. 4.9), the situation is reversed: the periodic allocator slightly outperforms the instantaneous one, achieving a marginal reduction in the first propeller's angular velocity. This improvement stems from the ability of the spline-based allocator to anticipate actuator delays and distribute effort smoothly across time, which the instantaneous greedy allocator cannot achieve.

This scenario confirms the theoretical expectations from Sec. 4.4: the greedy instantaneous allocator is optimal only in the absence of actuator dynamics, whereas the spline-based periodic allocator provides superior performance when realistic actuator dynamics are taken into account. Moreover, the spline-based approach produces smoother actuator trajectories, which is advantageous from the standpoint of mechanical stress, energy consumption, and robustness.

4.6 Discussion and Concluding Remarks

In this chapter, two DCA strategies have been implemented and compared on a UAV with tilting-twisting propellers. Both approaches successfully achieved the intended goal of reducing actuator usage according to the chosen quadratic cost functional, thereby confirming the validity of the allocation framework presented in Sec. 4.4. The key differences between the methods were highlighted through simulation studies under both ideal (instantaneous actuation) and realistic (first-order actuator dynamics) conditions.

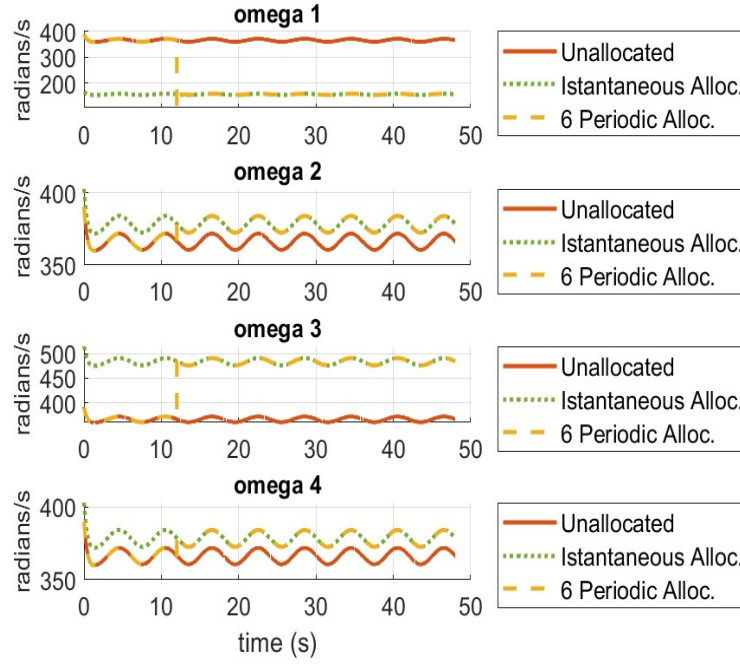


Figure 4.8: Angular velocities of the four propellers under instantaneous actuation. The angular velocity of the first propeller is reduced by both allocation strategies, but the instantaneous (greedy) allocator achieves lower cost than the periodic one (see Fig. 4.6).

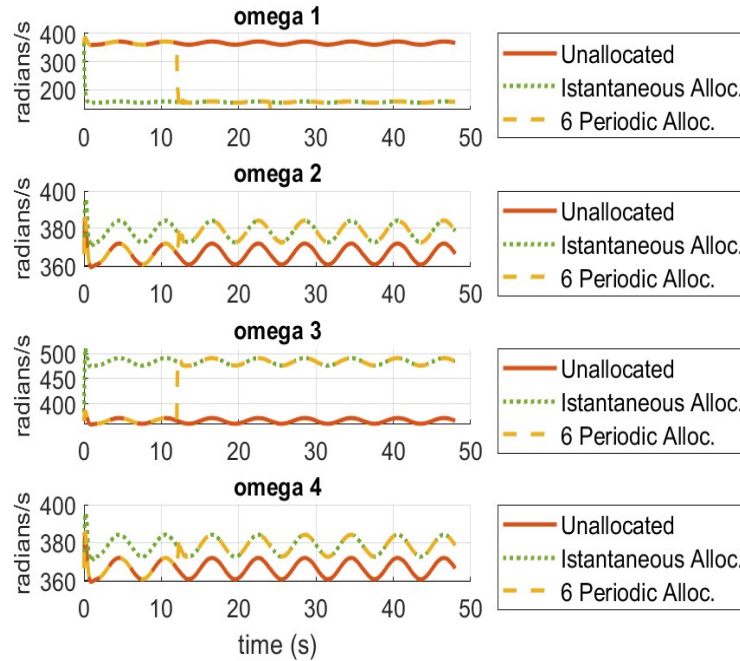


Figure 4.9: Angular velocities of the four propellers under dynamic actuation (first-order actuator dynamics). The angular velocity of the first propeller is reduced by both allocation strategies, but the periodic spline-based allocator outperforms the instantaneous one (see Fig. 4.7).

Since allocation methods are designed to optimize steady-state behavior, the transient phases corresponding to the first two periods of the trajectory can be disregarded. From the third period onward, meaningful performance comparisons can be drawn.

In order to highlight the contrasting behaviors of the two allocation strategies, the simulation results were systematically compared under both ideal and realistic actuation conditions. When actuator dynamics were neglected, the instantaneous (or greedy) allocator consistently achieved lower allocation cost, in line with the theoretical property that greedy redistribution is optimal in the absence of dynamic constraints. In this setting, the spline-based periodic allocator still reduced the load on the penalized actuator but did so with a higher residual cost, confirming that its additional structural complexity does not provide an advantage when actuators are assumed to respond instantaneously.

The situation was reversed once actuator dynamics were explicitly taken into account. Under first-order actuator dynamics, the instantaneous allocator exhibited performance degradation due to its inability to anticipate or compensate for delays in actuator response. By contrast, the spline-based periodic allocator significantly outperformed the greedy method, achieving a greater reduction in the angular velocity of the most penalized propeller. The improved performance stems from its ability to generate smooth, temporally distributed control signals that respect actuator dynamics.

Taken together, these findings confirm the theoretical expectations discussed earlier in the chapter. The instantaneous allocator represents the optimal choice only in the absence of actuator dynamics, whereas the spline-based periodic allocator becomes clearly superior when dynamic effects are present. This distinction underscores the importance of embedding actuator dynamics into the design of allocation frameworks for realistic applications, as it allows for both improved steady-state performance and smoother input trajectories that are more favorable from the standpoint of mechanical stress and energy efficiency.

Each method presents its own advantages. The instantaneous (greedy) allocator is extremely simple to implement, requires no additional states, and introduces no transient phase. In the absence of actuator dynamics, this method is provably optimal, as confirmed by the simulations (see Fig. 4.6). On the other hand, when unmodeled actuator dynamics are present, the periodic spline-based allocator exhibits greater flexibility. By optimizing over entire trajectory segments rather than at individual time instants, it produces smoother allocation signals and better overall performance (see Fig. 4.7).

It should be noted that in this work, the actuator dynamics of α_i , β_i , and ω_i were modeled as simple first-order systems. Accordingly, the difference in performance between the two methods was not drastic. However, in the presence of more complex or higher-order dynamics, the advantage of the periodic allocator is expected to become more pronounced.

The results presented here suggest a clear trade-off between simplicity and robustness: while the instantaneous allocator is optimal in idealized conditions, the periodic spline-based approach becomes preferable when actuator dynamics cannot be neglected. Future work will focus on extending the theoretical analysis of spline-based DCA in the presence of nonlinear and uncertain actuator models, and on validating the approach through more extensive simulations and real-world experiments.

Chapter 5

Chapter 5

A Sensitivity Approach to Periodic Control Allocation in Nonlinear Systems

Abstract

This chapter presents a framework for DCA in nonlinear over-actuated systems subject to periodic or time-varying exogenous signals. While classical allocation schemes treat the problem in a static or point-wise manner, solving an instantaneous optimization at each time step, such approaches neglect the temporal structure of the system and often yield suboptimal results when nonlinear dynamics or periodic references are present.

The proposed methodology extends the DCA paradigm beyond the instantaneous setting by combining gradient-based optimization with parameter-dependent sensitivity dynamics. This construction allows the allocator to account for the steady-state periodic response of nonlinear closed-loop systems and to minimize cost functionals defined over complete periods of exogenous signals. In doing so, it captures temporal correlations that static methods ignore, enabling smoother actuator coordination and improved periodic performance. A key property of the framework is the preservation of output invisibility that is, the nominal regulation or tracking performance of the baseline controller remains unaltered, while redundancy is exploited to optimize auxiliary criteria such as energy usage or actuator wear.

Two allocation strategies are analyzed within this general framework. The first corresponds to the classical instantaneous allocation, suitable for constant references where optimization can be performed pointwise in time. The second, more general strategy addresses periodic references by parameterizing the allocation signals through periodic basis functions and minimizing a Lagrangian cost over one fundamental period. This periodic approach captures the steady-state behavior of the system and fully exploits temporal correlations, thereby achieving improved actuator coordination and periodic performance in cyclic operating conditions.

Overall, this chapter provides a systematic and computationally tractable extension of DCA to nonlinear, periodically forced systems, thereby enriching the theoretical foundations of DCA and offering practical tools for a wide range of over-actuated engineering applications.

5.1 Introduction

In this chapter, we consider nonlinear plants in normal form with uniform relative degree. This assumption, which results in a slightly more structured class of systems than those treated in [11], thereby allowing additional inputs to act exclusively without perturbing the regulated output. Within this setting, the proposed allocation mechanism is explicitly designed to address periodic reference signals and cost functions expressed as a Lagrangian functional over one period, rather than relying on instantaneous or pointwise cost evaluations. As a result, the methodology extends DCA concepts to a broader family of nonlinear problems where performance must be understood and optimized over entire periodic cycles.

The main technical contribution lies in the development of a general algorithmic strategy grounded in gradient optimization. The allocation signal is generated through a finite-dimensional parameterization, and its parameters are iteratively updated so as

to decrease the periodic steady-state cost. A central component of the approach is the use of steady-state sensitivity functions, which characterize how periodic trajectories change with respect to allocation parameters. These sensitivities enable efficient and principled computation of the gradient of the Lagrangian cost, thus allowing systematic improvement of steady-state performance while rigorously preserving output invisibility.

This chapter is organized as follows. In Sec. 5.2, we formalize the problem statement and introduce the nonlinear system model, the class of admissible controllers, and the performance criterion to be optimized. Sec. 5.3 presents the proposed allocation architecture in detail, including the sensitivity-based gradient descent algorithm. Numerical experiments validating the effectiveness and generality of the proposed method are reported in Sec. 5.4, where we illustrate how the allocator redistributes control effort and reduces periodic steady-state cost without affecting output tracking. Finally, Sec. 5.5 provides concluding remarks and discusses extensions of the method to broader classes of nonlinear systems and performance objectives.

Notations. For a real matrix M , its transpose is denoted as M' . If M is full row rank, its pseudoinverse (such that $MM^\sharp = I$) is defined as $M^\sharp := M'(MM')^{-1}$, whereas $M^\perp$ denotes a basis matrix for its kernel $\ker(M)$, whose columns can be chosen as a set of orthonormal vectors. All functions are assumed to be smooth (or at least continuously differentiable as many times as required). The k -th time derivative of a scalar- or vector-valued function $f(t)$ is denoted by $f^{(k)} := \frac{d^k}{dt^k}f$, with $f^{(0)} := f$. The symbol $f \equiv F$ indicates that, for all possible values of its arguments, the function f assumes the constant value F .

5.2 Nonlinear Plant Class and Periodic Cost

This chapter addresses the problem of DCA, understood as a form of controller augmentation (see Sec. 2.2) applied to systems with input redundancy. As illustrated in Fig. 5.1, the allocator operates on top of a pre-designed stabilizing controller and its role is to redistribute the plant inputs in a way that optimizes a secondary performance criterion, while ensuring that the plant output response remains identical to that of the baseline closed loop.

The essential idea (as explained in Chap. 1 and 2) is that the redundancy of inputs provides freedom of choice in achieving the same closed-loop output, and this freedom can be exploited to improve performance in several ways, such as reducing the energy required by actuators, limiting the use of expensive or fragile components, or keeping some internal states in more desirable regions. Allocation therefore extends the scope of the controller from ensuring basic stabilization and tracking, to achieving higher-level optimization goals.

The formal setting considered here consists of a nonlinear plant with more control inputs than regulated outputs, a stabilizing controller that guarantees asymptotic tracking of a reference signal, and a class of exogenous references that are not restricted to constants but may also be periodic. Performance is evaluated through a Lagrangian cost functional defined on the steady-state trajectory of the closed loop, which allows the effect of actuator usage and state evolution to be captured over an entire period of motion rather than at individual instants. The presence of a well-defined cost functional naturally transforms the allocation task into an optimization problem, but one that must be solved under the stringent requirement that the closed-loop output response is not altered.

The class of nonlinear systems under study is slightly more restrictive than the one investigated in [11]. Specifically, we assume that the plant admits a globally defined

normal form with uniform vector relative degree [30, Sec. 11.5]. This assumption, while somewhat limiting, is sufficient to encompass a wide range of relevant examples and has the important advantage of enabling explicit and constructive derivations of the proposed allocation architecture. By focusing on this class, it becomes possible to move beyond the earlier frameworks developed for asymptotically constant references and instantaneous costs [11], [48], and to directly address the more general scenario of nonlinear plants driven by periodic references and optimized with respect to Lagrangian performance measures.

The central problem can therefore be expressed as follows. Given a nonlinear plant with input redundancy and a stabilizing controller that ensures reference tracking in closed loop, the task is to design a dynamic control allocator that, while leaving the plant output unaltered, minimizes a Lagrangian cost functional defined on the steady-state response of the system. The following sections present the architectural ingredients of such an allocator, showing how gradient-based optimization can be systematically combined with sensitivity dynamics to obtain a practical and effective solution.

5.2.1 The plant

We consider a nonlinear plant P with m inputs and p outputs, assumed to be input redundant, that is, $m > p$. The plant is supposed to admit a *uniform vector relative degree* $r = [r_1 \ r_2 \ \cdots \ r_p]'$ in the sense of [30, Sec. 11.5], and to be represented in *normal form* as introduced in [30, (5.7)–(5.8)]. Under these assumptions, the dynamics of P can be written as

$$\dot{x} = A_0 x + B_0 (f_0(x, z) + g_0(x, z)u), \quad (5.1a)$$

$$\dot{z} = \bar{f}_1(x, z) + \bar{g}_1(x, z)u, \quad (5.1b)$$

$$y = C_0 x, \quad (5.1c)$$

where the state vector is partitioned into x , collecting the chains of integrators associated with the relative degree, and z , representing the internal dynamics, while the input is $u \in \mathbb{R}^m$ and the output is $y \in \mathbb{R}^p$.

The matrices appearing in (5.1) are defined in block diagonal form as

$$A_0 = \text{blkdiag}(J_0(r_1), J_0(r_2), \dots, J_0(r_p)), \quad (5.2a)$$

$$B_0 = \text{blkdiag}(e^{(r_1)}, e^{(r_2)}, \dots, e^{(r_p)}), \quad (5.2b)$$

$$C_0 = \text{blkdiag}(e_{(r_1)}, e_{(r_2)}, \dots, e_{(r_p)}), \quad (5.2c)$$

with the notation

$$J_0(q) := \begin{bmatrix} 0_{(q-1) \times 1} & I_{q-1} \\ 0 & 0_{1 \times (q-1)} \end{bmatrix}, \quad e^{(q)} := \begin{bmatrix} 0_{(q-1) \times 1} \\ 1 \end{bmatrix},$$

$$e_{(q)} := \begin{bmatrix} 1 & 0_{1 \times (q-1)} \end{bmatrix}.$$

The matrices A_0 , B_0 , and C_0 therefore capture the canonical integrator chains associated with the relative degree of each output, while the nonlinearities f_0 , g_0 , $\bar{f}_1$, and $\bar{g}_1$ capture the coupling terms that depend on both the input and the internal dynamics z .

Remark 5.2.1

As explained in [31, Sec. 11.5 and Remark. 6.1.8], the presence of a uniform vector relative degree simplifies the normal form in (5.1) significantly with respect to the more general construction in [30, Sec. 6.1], and also with respect to the intermediate form considered in [11]. In particular, the uniformity assumption implies that the input matrix $g_0(x, z)$ always has full rank equal to p , for all (x, z) , which is a key property that will be exploited in the allocation design. $\square$

Summarizing, the class of plants considered in this chapter is characterized by being input redundant and admitting a nonlinear normal form with uniform vector relative degree. The plant equations involve smooth functions f_0 , g_0 , $\bar{f}_1$, and $\bar{g}_1$, with the additional structural property that $\text{rank}(g_0(x, z)) = p$ globally.

Assumption 5.2.1

Plant P is input redundant and admits the nonlinear normal form (5.1) with uniform vector relative degree r , smooth functions f_0 , g_0 , $\bar{f}_1$, and $\bar{g}_1$, and $\text{rank}(g_0(x, z)) = p$ for all (x, z) . $\square$

5.2.2 The reference

The purpose of the baseline control system is to ensure accurate tracking of a reference trajectory. In the context of this chapter, the reference is assumed to be periodic, reflecting scenarios of practical interest such as repetitive maneuvers, cyclic tasks, or exogenous signals with a fundamental period.

To guarantee compatibility with the relative degree of the plant, the reference is required to possess a sufficient level of smoothness.

Assumption 5.2.2

The reference signal $y_d : \mathbb{R} \rightarrow \mathbb{R}^p$ is T -periodic, for some $T > 0$. The h -th scalar component $y_{d,h}$ of y_d is r_h times continuously differentiable, where r_h is the relative degree associated with the h -th output, for $h = 1, \dots, p$. $\square$

As is standard in nonlinear tracking problems, it will be assumed that the derivatives $y_{d,h}^{(k)}$, for $k = 1, \dots, r_h$, are available for feedback purposes. This can be achieved either directly by differentiation, or indirectly when the reference is generated by an exosystem of the form $\dot{w} = Sw$, $y_d = Fw$, since in that case the derivatives satisfy $y_d^{(k)} = FS^k w$.

For notational convenience, we now introduce compact representations for the derivatives of the reference. Given the vector of relative degrees $r = [r_1 \ r_2 \ \dots \ r_p]'$, define

$$y_d^{[r]} := \begin{bmatrix} y_{d,1}^{(r_1)} & y_{d,2}^{(r_2)} & \dots & y_{d,p}^{(r_p)} \end{bmatrix}',$$

$$Y_{d,h}^{(v_h)} := \begin{bmatrix} y_{d,h}^{(1)} & y_{d,h}^{(2)} & \dots & y_{d,h}^{(v_h)} \end{bmatrix}', \quad h = 1, \dots, p,$$

$$Y_d^{[v]} := \left[(Y_{d,1}^{(v_1)})' \quad \cdots \quad (Y_{d,p}^{(v_p)})' \right]',$$

where either $(v, v_h) = (r, r_h)$ or $(v, v_h) = (r^-, r_h^-)$, with $r^- := [r_1 - 1 \ r_2 - 1 \ \cdots \ r_p - 1]'$ and $r_h^- := r_h - 1$.

For brevity, two shorthands will be used extensively in the sequel:

$$\xi := Y_d^{[r]}, \quad \bar{\xi} := Y_d^{[r^-]}. \quad (5.4)$$

By Assumption 5.2.2, the reference and all its required derivatives are periodic, continuous, and bounded. Hence, the vectors $Y_d^{[v]}$, ξ , and $\bar{\xi}$ inherit these properties, providing a well-defined basis for the subsequent development of the allocation scheme.

5.2.3 The controller

Given the plant structure in normal form and the requirement of tracking the reference signal y_d , a natural choice for the baseline controller C is to adopt the class of input–output feedback linearizing tracking controllers, as described in [30, Sec. 5.1]. The idea is to cancel the nonlinearities affecting the x subsystem and then enforce stable tracking dynamics by means of a linear error system.

Accordingly, the controller output y_c can be written as

$$y_c = g_0^\#(x, z)(u_{fl} + u_{tr}) + g_0^\perp(x, z)u_z, \quad (5.5a)$$

$$u_{fl} = -f_0(x, z), \quad (5.5b)$$

$$u_{tr} = y_d^{(r)} + K(x - Y_d^{[r^-]}), \quad (5.5c)$$

$$u_z = \sigma(x, z, Y_d^{[r]}), \quad (5.5d)$$

where the matrix $K = \text{blkdiag}(K_1, \dots, K_p)$ with $K_h \in \mathbb{R}^{1 \times r_h}$ is chosen so that the closed-loop matrix $A_K := A_0 + B_0 K$ is Hurwitz. In this structure, considering (5.1), it is easy to see that the term u_{fl} compensates for the nonlinear drift in the x subsystem, while u_{tr} imposes stable tracking dynamics driving the output toward the reference y_d . The additional input u_z is only available in the redundant case ($m > p$), because when $m = p$ the matrix $g_0(x, z)$ is invertible and $g_0^\perp(x, z)$ reduces to zero. The role of u_z is to stabilize the internal dynamics associated with z , as clarified later in Assumption 5.2.3.

When this controller is interconnected with the plant, the input applied to P becomes

$$u = y_c + y_{al}, \quad (5.6)$$

where the additional term y_{al} represents the allocation signal, to be designed in the next sections. The closed-loop dynamics then take the form

$$\dot{x} = A_K(x - Y_d^{[r^-]}) + \dot{Y}_d^{[r^-]} + B_0 g_0(x, z) y_{al}, \quad (5.7a)$$

$$\dot{z} = f_1(x, z, Y_d^{[r]}) + \bar{g}_1(x, z) y_{al}, \quad (5.7b)$$

$$y = C_0 x, \quad (5.7c)$$

where

$$f_1(x, z, Y_d^{[r]}) := \bar{f}_1(x, z) + \bar{g}_1(x, z)y_c. \quad (5.8)$$

By introducing the error coordinates $e := x - Y_d^{[r-]}$ and the shorthand notation of (5.4), the closed-loop system becomes

$$\dot{e} = A_K e + B_0 g_0(\bar{\xi} + e, z)y_{al}, \quad (5.9a)$$

$$\dot{z} = f_1(\bar{\xi} + e, z, \xi) + \bar{g}_1(\bar{\xi} + e, z)y_{al}, \quad (5.9b)$$

$$y = y_d + C_0 e. \quad (5.9c)$$

Equations (5.9a) and (5.9c) make it evident that the feedback-linearizing controller in (5.5) ensures exponential convergence of the tracking error to zero, thereby fulfilling the first requirement of Assumption 5.2.3.

Assumption 5.2.3

Controller C when connected to plant P in (5.1) as in (5.6) ensures that:

- i. $x - Y_d^{[r-]}$ (hence in particular $y - y_d$) converges exponentially to 0 if $g_0(x, z)y_{al} \equiv 0$;
- ii. the z subsystem in (5.9b) is ISC (input-to-state convergent) with respect to the input y_{al} .

□

Remark 5.2.2

Convergent systems (see e.g. [36], [39]) enjoy several properties similar to linear time invariant systems. For the developments in this chapter, item ii. of Assumption 5.2.3 is used to ensure that for bounded, continuous and T -periodic (or asymptotically T -periodic) $Y_d^{[r]}$ and y_{al} , also z remains bounded and asymptotically converges to a unique T -periodic steady-state trajectory, as explained in [39, Sec. 2.2.2]. Control law design approaches to find a u_z which is stabilizing in the input-to-state convergence sense can be found in the literature, e.g. based on backstepping-like procedures; since linear time invariant asymptotically stable systems are input to state convergent, feedback linearization can also be used in the special case when the z dynamics after application of u_{fl} and u_{tr} in (5.5), namely

$$\dot{z} = \left(\bar{f}_1 + \bar{g}_1 g_0^\#(u_{fl} + u_{tr}) \right) + \bar{g}_1 g_0^\perp u_z$$

is feedback linearizable from u_z .

□

⁶Although this property is guaranteed by design when using the controller in (5.5), it is explicitly included in Assumption 5.2.3 to highlight that the subsequent results remain valid for other controllers as long as the same properties hold.

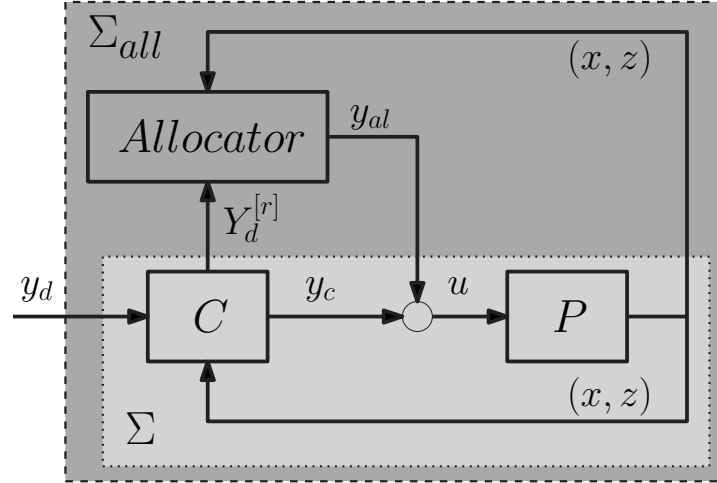


Figure 5.1: *A priori* given closed-loop control system Σ (dotted, light grey) augmented with DCA as Σ_{all} (dashed, dark grey).

5.2.4 The problem

Since the plant is input redundant, multiple input trajectories can achieve the same output evolution; the normal form makes it apparent that this can be achieved by modifying the evolution of the z -subsystem in (5.1), once it is decoupled from the rest of the plant by the controller (5.5) (for a deeper discussion from the geometric control point of view in the linear case, see the insightful analysis in [42]). The proposed DCA strategy will exploit the redundancy, to refine the plant input u by augmenting it with an allocation signal u_{al} , to achieve an optimized steady-state response, as specified in the next problem definition.CDC

Problem 2. *Consider the control system Σ described by (5.1), (5.5), (5.6) under Assumptions 5.2.1, 5.2.2, 5.2.3. Design, if possible, an input allocation device generating y_{al} such that, when interconnected to Σ to form the allocated closed loop system Σ_{all} as in Fig. 5.1, the following requirements are achieved for each initial state and y_d :*

- R1)** (Stability) Σ_{all} admits a unique, globally asymptotically stable T -periodic steady-state response;
- R2)** (Output Invisibility) The response in the output y of Σ_{all} coincides with the response of Σ ;
- R3)** (Steady-State Optimality) Denoting by u_{ss}, z_{ss} the T -periodic⁷ steady-state evolution of u and z , the Lagrange cost functional

$$J(u_{ss}, z_{ss}) = \int_0^T g(u_{ss}(\tau), z_{ss}(\tau)) d\tau \quad (5.10)$$

is locally minimized when y_{al} is chosen in a given class of signals $\mathbb{P}$.

No currently available result in the literature on DCA allows to solve Problem 2, especially considering the Lagrange cost for a nonlinear control system as Σ ; hence, in the following sec. we will introduce a novel design for the allocator which is able satisfy all requirements in Problem 2.

⁷Since u_{ss}, z_{ss} are T -periodic, the value of the cost (5.10) is the same on any interval of length T , hence the integration is considered simply between 0 and T . Moreover, dependency of J on x_{ss} is not considered, since by Assumption 5.2.3 $x_{ss} = Y_d^{[r-]}$ and then is fixed for any given y_d .

A special case of Problem 2 is obtained by considering constant references y_d , as in [11]. If y_d is constant, Problem 2 can be simplified as follows.

Problem 3. *All as in Problem 2, but replace R3) with:*

R3') (Steady-State Optimality) *For each constant y_d , denoting by u_{ss}, z_{ss} the constant steady-state values of u and z , the instantaneous cost function*

$$J(u_{ss}, z_{ss}) = g(u_{ss}, z_{ss}) \quad (5.11)$$

is locally minimized when y_{al} is chosen in a given class of signals $\mathbb{P}$.

5.3 Sensitivity-based Allocator Design

After some preliminary steps in Sec. 5.3.1, we introduce two different approaches to address Problem 3 in Sec. 5.3.2 and Problem 2 in Sec. 5.3.3.

5.3.1 Allocator structure

The allocation signal y_{al} is defined according to

$$y_{al} = g_0(x, z)^\perp v, \quad (5.12)$$

which ensures that $g_0(x, z)y_{al} \equiv 0$ as in item i. in Assumption 5.2.3, so that the output of Σ_{all} is unaffected by y_{al} and matches the output of Σ . The signal v is selected within a finitely parameterized set of T -periodic functions, such as:

$$v = \sum_{h=1}^{n_v} \sum_{i=1}^{N_h} e_h \alpha_{h,i} \psi_i(t) = \Psi(t)\alpha, \quad (5.13a)$$

$$\alpha = [\alpha_{1,1} \cdots \alpha_{1,N_1} | \cdots | \alpha_{n_v,1} \cdots \alpha_{n_v,N_{n_v}}]' \in \mathbb{R}^{\bar{N}}, \quad (5.13b)$$

$$\bar{N} = \sum_{h=1}^{n_v} N_h, \quad n_v := m - p, \quad (5.13c)$$

where e_h is the h -th column of the identity matrix (of dimension n_v), and clearly larger N_h 's result in greater flexibility of the DCA; parameter α must satisfy $\alpha \in \mathcal{A} \subset \mathbb{R}^{n_v}$, where $\mathcal{A}$ is a compact set containing an open neighborhood of the origin. Since the allocator design will achieve requirement R1) in Problem 2, it is easy to see that for any fixed y_d , the steady-state evolutions of y_{al} in (5.12) will also be a set of functions finitely and compactly parameterized by $\alpha \in \mathcal{A}$, constituting the class of signals $\mathbb{P}$ in requirement R3) in Problem 2. The optimal selection of α is based on steepest descent. More precisely, the optimization problem underlying the minimization of (5.10) boils down to solve the *reduced minimization problem*:

$$\min_{\alpha \in \mathcal{A}} J(u_{ss}, z), \text{ subject to} \quad (5.14a)$$

$$J(u_{ss}, z) = \int_0^T g(u_{ss}(\tau), z(\tau)) d\tau \quad (5.14b)$$

$$\dot{z} = f(t, z, \alpha), \quad (5.14c)$$

$$z(0, \alpha) = z(T, \alpha), \quad (5.14d)$$

where u_{ss} and the dynamics of z in (5.14c) already consider the steady-state evolution of

$x_{ss} = Y_d^{[r^-]}$ and the parameterization (5.12), (5.13), so that dependency from all variables but z and α is condensed in the dependency on time, namely:

$$\begin{aligned} f(t, z, \alpha) &:= f_1(Y_d^{[r^-]}, z, Y_d^{[r]}) \\ &\quad + \bar{g}_1(Y_d^{[r^-]}, z) g_0(Y_d^{[r^-]}, z)^\perp \Psi \alpha, \end{aligned} \quad (5.15a)$$

$$\begin{aligned} u_{ss}(t, z, \alpha) &= g_0^\#(Y_d^{[r^-]}, z) \left(-f_0(Y_d^{[r^-]}, z) + y_d^{(r)} \right) \\ &\quad + g_0^\perp(Y_d^{[r^-]}, z) \left(u_z(Y_d^{[r]}, z) + \Psi \alpha \right), \end{aligned} \quad (5.15b)$$

The periodic boundary conditions in (5.14d) ensure that the solution in z computed when addressing the reduced problem (5.14) is actually the restriction on the interval $[0, T]$ of the unique periodic steady-state solution of z , which exists by item ii. of Assumption 5.2.3.

Remark 5.3.1

In DCA results for linear systems, suitable convexity hypotheses are made on the cost which result in steepest descent to converge to a bounded solution which is the unique global minimum. In the present context, due to the interplay between the nonlinear dynamics and the cost in the reduced problem (5.14), in general convexity properties do not hold, and boundedness of α as determined by steepest descent is not guaranteed a priori.

For these reasons, α is restricted a priori in the compact set $\mathcal{A}$, which implies boundedness of the value of α selected by steepest descent, and the fact that the minimum in the reduced problem (5.14) exists (since it is a continuous map from on a compact domain $\mathcal{A}$). $\square$

5.3.2 Instantaneous Allocation

As pointed out in Sec. 5.2.4, for constant y_d , a constant steady-state for x and y in (5.1) follows, and all elements in $Y_d^{[r]}$ (and then of x_{ss}) are equal to zero except the p elements with no derivatives, which are equal to y_d . Hence, v in (5.12) is chosen as a constant vector and $\Psi(t)$ in (5.13a) becomes an identity. In this case (5.14) simplifies to

$$\min_{\alpha \in \mathcal{A}} J(\tilde{u}_{ss}, z), \text{ subject to} \quad (5.16a)$$

$$J(\tilde{u}_{ss}, z) = g(\tilde{u}_{ss}, z) \quad (5.16b)$$

$$0 = \tilde{f}(y_d, z, \alpha), \quad (5.16c)$$

where $\tilde{f}(y_d, z, \alpha)$ and $\tilde{u}_{ss}(y_d, z, \alpha)$ are given by (5.15) specialized to constant y_d , and showing the dependency on y_d explicitly (rather than the trivial dependency on time). The constant steady-state condition on z given by (5.16c) replaces the periodic steady-state condition on z given by (5.14c), (5.14d). By item ii. of Assumption 5.2.3, for each pair (y_d, v) there is a unique constant steady-state value z_{ss} , which implies that there is a well defined function $z_{ss} = \pi_z(y_d, \alpha)$ defined implicitly by (5.16c). To minimize (5.16a), namely $g(\pi_z(y_d, \alpha), \pi_u(y_d, \alpha))$, with respect to $\alpha \in \mathcal{A}$, a projected gradient law

$$\dot{\alpha} = \text{proj}_{\mathcal{A}}(-\gamma \nabla_{\alpha} g(\pi_z(y_d, \alpha), \pi_u(y_d, \alpha)))', \quad \gamma > 0. \quad (5.17a)$$

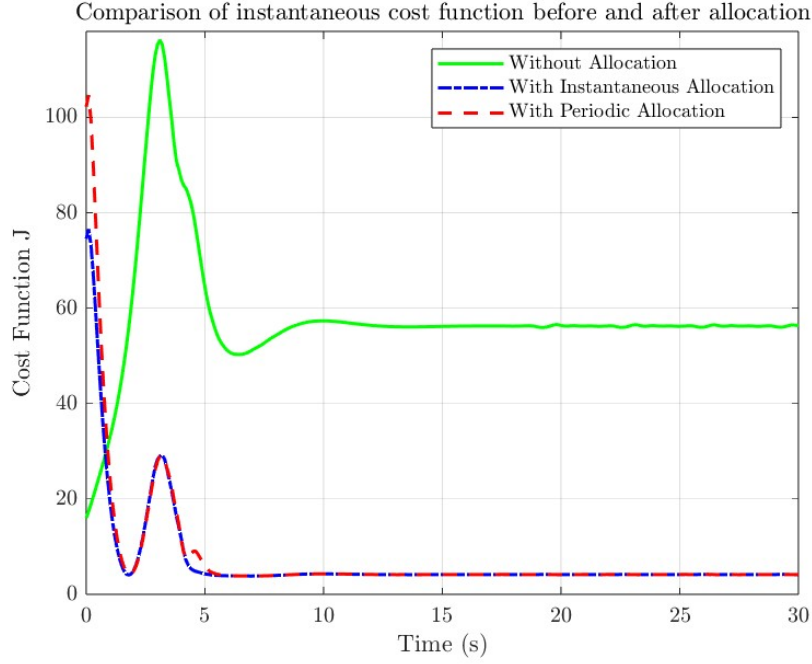


Figure 5.2: Scenario I. Comparison of instantaneous cost function $g(u_{ss}(t), z_{ss}(t))$ before and after allocation in case of constant reference: both methods achieve the same performance.

can be used. The gradient in (5.17a) is given by

$$\nabla_{\alpha} g(\pi_z, \pi_u) = \frac{\partial g}{\partial z} \frac{\partial \pi_z}{\partial \alpha} + \frac{\partial g}{\partial u} \frac{\partial \pi_u}{\partial \alpha} \quad (5.17b)$$

and cannot be computed directly if π_z, π_u are not available in closed form; however, considering that $\frac{\partial \pi_z}{\partial \alpha}$ is the *sensitivity* of the solution of (5.16c) to variations in α , differentiation of (5.16c) and the implicit function theorem can be used to determine it as:

$$\frac{\partial \pi_z}{\partial \alpha} = - \left(\frac{\partial \tilde{f}}{\partial z} \right)^{-1} \frac{\partial \tilde{f}}{\partial \alpha}, \quad (5.18a)$$

where invertibility of $\frac{\partial \tilde{f}}{\partial z}$ follows from item ii. of Assumption 5.2.3. With $\frac{\partial \pi_z}{\partial \alpha}$ available, the sensitivity of $\tilde{u}_{ss}(y_d, z, \alpha)$ is also readily available by the formula:

$$\frac{\partial \pi_u}{\partial \alpha} = \frac{\partial \tilde{u}_{ss}}{\partial z} \frac{\partial \pi_z}{\partial \alpha} + \frac{\partial \tilde{u}_{ss}}{\partial \alpha}. \quad (5.18b)$$

The (already discussed) properties induced by the structure of the proposed allocator (Sec. 5.3.1) and standard properties of projected steepest descent laws allow to prove the following formal result (see *e.g.* [11]).

Theorem 5.3.1

The DCA (5.12), (5.13) (with $\Psi(t) \equiv I$), (5.17), (5.18) solves Problem 3 in the weak sense, namely convergence to a constrained stationary point whose cost is not worse

than the non-allocated cost. □

Proof. The proof is divided into three parts: verifying output invisibility, ensuring stability of the augmented system, and proving convergence of the optimization law.

Output Invisibility (Requirement R2): The allocation signal is defined as $y_{al} = g_0(x, z)^\perp v$. By construction, $g_0(x, z)g_0(x, z)^\perp \equiv 0$, which implies $g_0(x, z)y_{al} \equiv 0$. Substituting this into the x -subsystem dynamics in Eq. 5.9a:

$$\dot{e} = A_K e + B_0 g_0(\bar{\xi} + e, z)y_{al} = A_K e. \quad (5.19)$$

The error dynamics $\dot{e} = A_K e$ are identical to the unallocated case. Since A_K is Hurwitz by Assumption 5.2.3, $e(t) \rightarrow 0$ exponentially, ensuring that the regulated output $y(t)$ is unaffected by y_{al} .

Stability (Requirement R1): The output invisibility property creates a virtual cascade. The (x, e) subsystem is independent of y_{al} and is exponentially stable. By Assumption 5.2.3, the z -subsystem is Input-to-State Convergent (ISC) with respect to y_{al} . Since y_{al} is bounded (because α is restricted to a compact set $\mathcal{A}$ and $g_0^\perp$ is smooth), the combined (x, z) trajectories are bounded and converge to a unique steady state for any fixed $\alpha \in \mathcal{A}$.

Weak Steady-State Optimality (Requirement R3'): The adaptation law $\dot{\alpha} = \text{proj}_{\mathcal{A}}(-\gamma \nabla_{\alpha} J)$ is a projected gradient descent on the cost $J(\alpha) = g(\pi_z(y_d, \alpha), \pi_u(y_d, \alpha))$.

- **Existence of Gradient:** The gradient $\nabla_{\alpha} J$ is well-defined because π_z and π_u are smooth functions of α , as guaranteed by the Implicit Function Theorem applied to the steady-state equation $\bar{f}(y_d, z, \alpha) = 0$.
- **Convergence:** On the compact set $\mathcal{A}$, the continuous function $J(\alpha)$ is bounded below. The projected gradient flow ensures that $J(\alpha(t))$ is a non-increasing Lyapunov function for the parameter dynamics. Specifically, $\dot{J} = \nabla_{\alpha} J \cdot \dot{\alpha} = -\gamma \|\text{proj}_{\mathcal{A}}(\nabla_{\alpha} J)\|^2 \leq 0$.

By LaSalle's Invariance Principle for projected systems, $\alpha(t)$ converges to the set of stationary points within $\mathcal{A}$ where the cost is locally minimized (or at least non-increasing relative to the unallocated case $\alpha = 0$), thus solving Problem 3 in the weak sense. □

Remark 5.3.2

In the literature, DCA designed as in Theorem 5.3.1 but for linear plants have been proposed and used also for time-varying references y_d . As will be shown in Sec. 5.4, in general such a construction does not guarantee optimality; on the other hand, suitable extension of the same ideas proposed in this sec. to the T -periodic case are possible, as shown in the next section, and they actually provide improved performance. □

5.3.3 Periodic Allocation

Since Problem 2 involves a cost functional and T -periodic reference and steady-state evolutions rather than a cost function and constant reference and steady-state evolutions as in Problem 3, it is to be expected that closed form solution are even less likely to be obtained. Mimicking the developments of Sec. 5.3.2, the idea is to define an iterative

optimization rule to solve the reduced problem defined by (5.14), (5.15) by updating an initial guess of α . The key tool will be the *Sensitivity Functions* (see [35, Sec. 3.3]) of the steady-state solutions $z_{ss}(t, \alpha)$ and the ensuing $u_{ss}(t, \alpha)$, which are easily computed even when the map associating y_d to z_{ss} through the nonlinear dynamics (5.14c) is usually complex and non-causal.

Denoting the steady-state evolutions of z and u as

$$z_{ss}(t, \alpha) := \pi_z(t, \alpha), \quad u_{ss}(t, \alpha) := \pi_u(t, \alpha), \quad (5.20)$$

the projected steepest descent law takes the form:

$$\dot{\alpha} = \text{proj}_{\mathcal{A}} \left(-\gamma \int_0^T \nabla_{\alpha} g(\pi_z(\tau, \alpha), \pi_u(\tau, \alpha)) d\tau \right)', \quad \gamma > 0, \quad (5.21a)$$

where the gradient in (5.21a) is given by

$$\nabla_{\alpha} g(\pi_z, \pi_u) = \frac{\partial g}{\partial z} \frac{\partial \pi_z}{\partial \alpha} + \frac{\partial g}{\partial u} \frac{\partial \pi_u}{\partial \alpha}, \quad (5.21b)$$

and, as in the previous case, the term $\frac{\partial \pi_z}{\partial \alpha}$ represents the sensitivity of the solution of (5.14c), (5.14d) to variations of α . Following the derivation of Sensitivity Functions in [35, Sec. 3.3], differentiation with respect to time of (5.14c), (5.14d) and the definition of the sensitivity function S of z as

$$S(\tau) = \frac{\partial \pi_z}{\partial \alpha}(\tau, \alpha), \quad \tau \in [0, T], \quad (5.22)$$

implies that S can be computed by solving the equations

$$\frac{dz}{d\tau} = f(\tau, z(\tau), \alpha(t)), \quad (5.23a)$$

$$z(0, \alpha(t)) = z(T, \alpha(t)), \quad (5.23b)$$

$$\frac{dS}{d\tau} = A(\tau, \alpha(t))S(\tau) + B(\tau, \alpha(t)), \quad (5.23c)$$

$$S(0) = S(T), \quad (5.23d)$$

where

$$A(\tau, \alpha(t)) = \frac{\partial f}{\partial z}, \quad B(\tau, \alpha(t)) = \frac{\partial f}{\partial \alpha}. \quad (5.23e)$$

With the solution of (5.23) available, both sensitivities required by (5.21b) are easily computed by the formulas:

$$\frac{\partial \pi_z}{\partial \alpha} = S(\tau), \quad (5.24a)$$

$$\frac{\partial \pi_u}{\partial \alpha} = \frac{\partial u_{ss}}{\partial z} S(\tau) + \frac{\partial u_{ss}}{\partial \alpha}. \quad (5.24b)$$

Note that, in principle, (5.21a) requires that (5.23) are solved at each time t (for fixed $\alpha(t)$); in practice, a discrete time approximation can be applied (with a suitably small gradient update gain in place of γ appearing in (5.21a)) so that (5.23) can be solved and

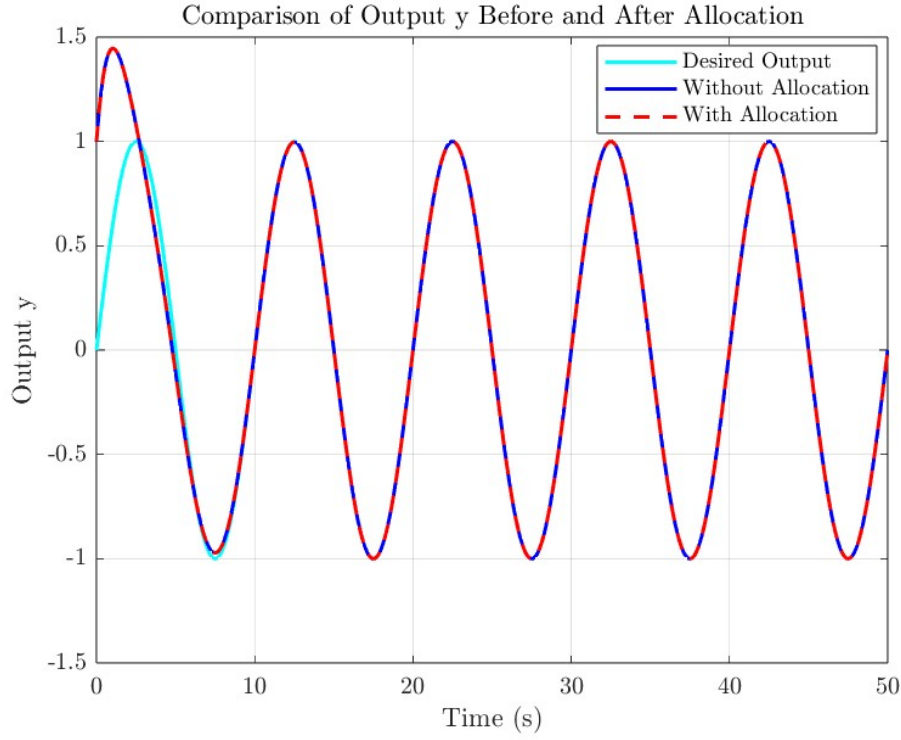


Figure 5.3: Scenario II. Output invisibility is achieved by both methods.

its solution used in (5.24) for update α in a discrete fashion.

The (already discussed) properties induced by the structure of the proposed allocator (Sec. 5.3.1) and standard properties of projected steepest descent laws again allow to prove the following formal result, extending previously available results to the considered class of nonlinear control systems satisfying Assumptions 5.2.1, 5.2.2 and 5.2.3.

Theorem 5.3.2

The DCA (5.12), (5.13), (5.21), (5.23), (5.24) solves Problem 2 in the weak sense. $\square$

5.4 Academic Example

5.4.1 Data

To illustrate the effectiveness of the proposed allocation framework, both of the DCA methods presented in Sec. 5.3 have been tested on a simple nonlinear plant. The purpose of this example is not to address a specific engineering application, but rather to provide an academic benchmark where the features of instantaneous and periodic allocation can be clearly observed and compared.

The system dynamics are given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1 + z_2 \\ x_1^2 - z_1 \\ -z_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 + x_2 x_1 & x_2 \\ x_2 z_1 & x_2 \\ x_1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad y = x_1 \quad (5.25)$$

with two control inputs and a scalar regulated output. The structure of (5.25) makes it evident that the system is nonlinear and input redundant, thus fitting within the class of plants described in Sec. 5.2.

The objective of the simulation is for the output to track two different reference trajectories, chosen so as to highlight the two allocation strategies. In the first scenario, the reference is constant, namely $y_d = 3$. This corresponds to the setting where the instantaneous allocation method can be applied, since the periodic cost functional reduces to an instantaneous cost. In the second scenario, the reference is periodic, given by

$$y_d(t) = \sin\left(\frac{2\pi}{T}t\right), \quad T = 10,$$

which represents the more general case where the full periodic allocation design is required. In this way, the example covers both extremes: constant references that recover earlier formulations, and genuinely periodic references where the new design of this chapter becomes necessary.

The baseline controller, which satisfies Assumption 5.2.3, is taken from the class of input–output feedback linearizing controllers introduced in (5.5). In the present example, the gain matrix is chosen as $K = [-1 \ -1]$, and the stabilizing term u_z is set identically to zero. This simple configuration is sufficient to ensure exponential tracking of the desired output, while leaving redundancy available for allocation.

Figures 5.4 and 5.5 illustrate the results for the second scenario, namely the case of a sinusoidal reference trajectory with period $T = 10$. The plots show the time evolution of the two input channels before and after allocation, making it evident that the proposed DCA modifies the distribution of effort between the two actuators without altering the regulated output trajectory. The allocator exploits the redundancy to shape the input profiles in a more efficient way, while the output tracking performance remains identical to that of the baseline controller.

5.4.2 Allocator Design

In order to validate the proposed methodology on the nonlinear plant (5.25), we now design the allocation signal y_{al} according to the two different DCA strategies introduced in Sec. 5.3. The allocation law is parameterized in terms of the auxiliary signal v , which is shaped differently depending on whether the instantaneous or the periodic approach is considered. For the instantaneous case, we assume the simplest form $v = \alpha_1$, corresponding to a constant adjustment in the redundant direction. For the periodic case, in contrast, we allow v to contain harmonics of the fundamental frequency, namely

$$v(t) = \alpha_1 + \alpha_2 \sin\left(\frac{2\pi}{T}t\right) + \alpha_3 \cos\left(\frac{2\pi}{T}t\right),$$

which gives the allocator additional freedom to exploit the periodicity of the reference signal. In both cases, the optimizer employs a gradient update with gain $\gamma = 1$.

The cost functional to be minimized is chosen consistently with (5.11) and (5.10), and is defined as

$$g(u_{ss}, z_{ss}) = u_{ss}^T W u_{ss},$$

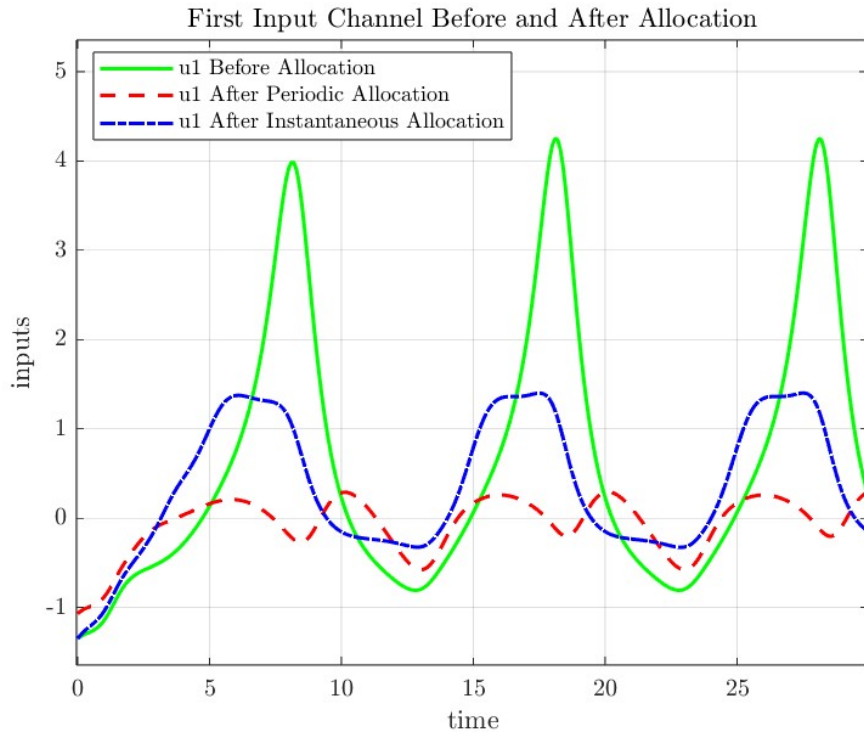


Figure 5.4: Scenario II. First input channel before and after allocation.

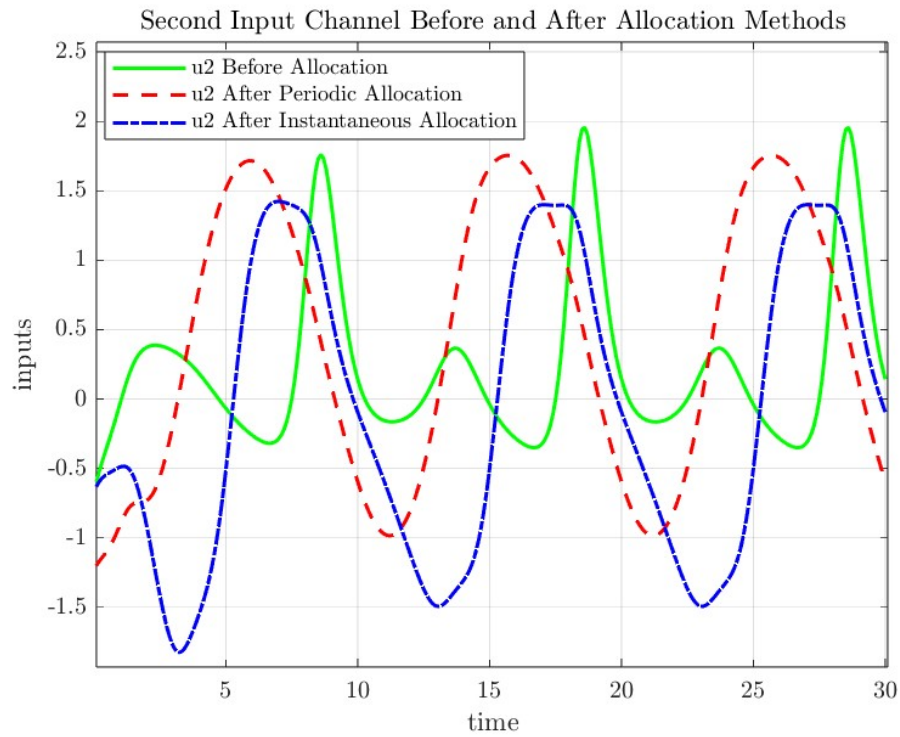


Figure 5.5: Scenario II. Second input channel before and after allocation.

with the weighting matrix $W = \text{diag}(200, 1)$. This choice penalizes the use of the first input much more heavily than the second, reflecting a situation where one actuator is considered “expensive” or less desirable.

Hence, the allocator is expected to reduce the contribution of u_1 , possibly at the expense of greater reliance on u_2 .

5.4.3 Scenario I: constant $y_d = 3$

The first simulation considers a constant reference trajectory $y_d = 3$. In this situation, the periodic allocation scheme collapses to the instantaneous one, since the harmonics in v are not excited and only the constant term α_1 plays a role. As a result, both allocation methods achieve essentially identical performance. This is confirmed in Fig. 5.2,

which compares the cost before and after allocation and shows no distinction between the two methods. This outcome is consistent with the theoretical discussion in Sec. 5.3.2: when the reference is constant, the periodic Lagrangian cost reduces to an instantaneous one, and therefore a static greedy optimization suffices to reach the optimum. The result also provides a useful baseline, illustrating that the proposed periodic approach does not introduce unnecessary complexity in the trivial case of constant references.

5.4.4 Scenario II: periodic $y_d = \sin((2\pi t)/T)$, $T = 10$

The second scenario is more demanding, with a sinusoidal reference of period $T = 10$. Here the difference between instantaneous and periodic allocation becomes evident.

First of all, Fig. 5.3 demonstrates that the requirement of output invisibility is satisfied in both cases. The steady-state output trajectory remains exactly the same before and after allocation, showing that the allocator operates only in the redundant directions without compromising the tracking performance of the baseline controller.

Turning to the input channels, Fig. 5.4 shows the evolution of the first input. Both allocation methods succeed in reducing its amplitude, but the periodic method achieves a much stronger suppression, thereby fulfilling more effectively the design goal of reducing the use of the “expensive” actuator. Fig. 5.5 provides the complementary picture for the second input. Here it is evident that the improvement in u_1 achieved by the periodic allocator comes at the price of allowing larger excursions of u_2 . This trade-off is expected, given the weighting structure of W , and highlights how periodic allocation can redistribute the effort more aggressively over time while still maintaining the same output behavior.

The quantitative effect is further clarified in Figs. 5.8 and 5.6. Figure 5.8 reports the value of the periodic cost functional $\int_0^T g(u_{ss}(\tau), z_{ss}(\tau)) d\tau$ before and after allocation. The reduction achieved by the periodic allocator is significantly larger than that obtained by the instantaneous one, confirming that the additional harmonics in v can be effectively exploited to lower the periodic average cost. On the other hand, Fig. 5.6 shows the instantaneous value of the cost function at different time instants. Here, while both methods improve performance relative to the baseline, the periodic allocator provides a trajectory of u that keeps the running cost consistently lower.

In summary, this scenario demonstrates the advantage of periodic allocation over instantaneous allocation when dealing with time-varying references. While both methods preserve output tracking and reduce the burden on the penalized input channel, the periodic approach achieves a more substantial minimization of the periodic cost by allowing temporal redistribution of the control effort.

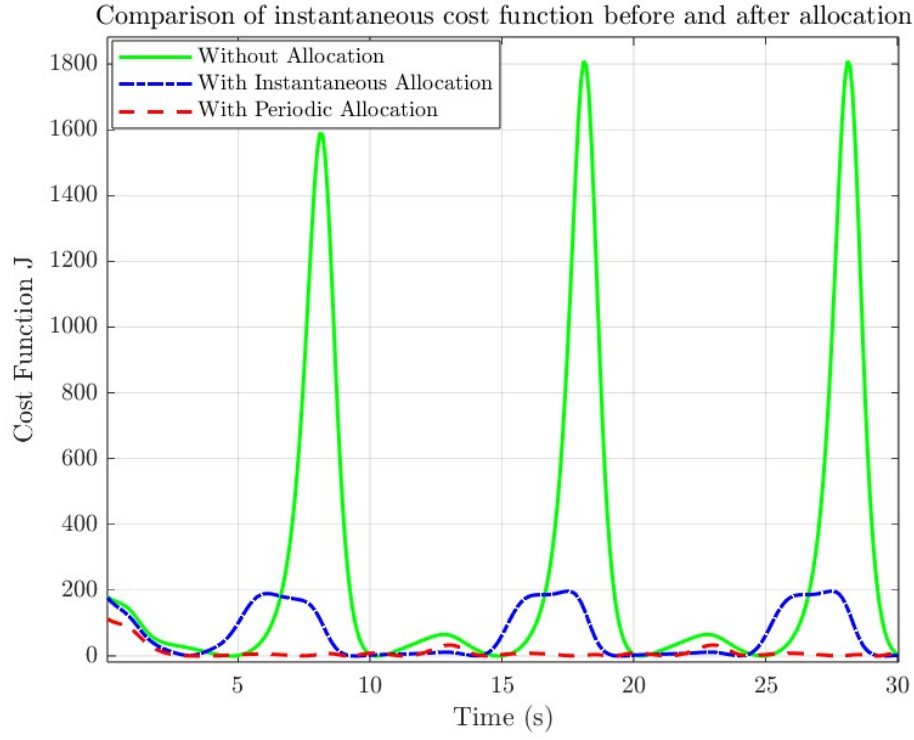


Figure 5.6: Scenario II. Comparison of instantaneous cost function $g(u_{ss}(t), z_{ss}(t))$ before and after allocation in case of periodic reference.

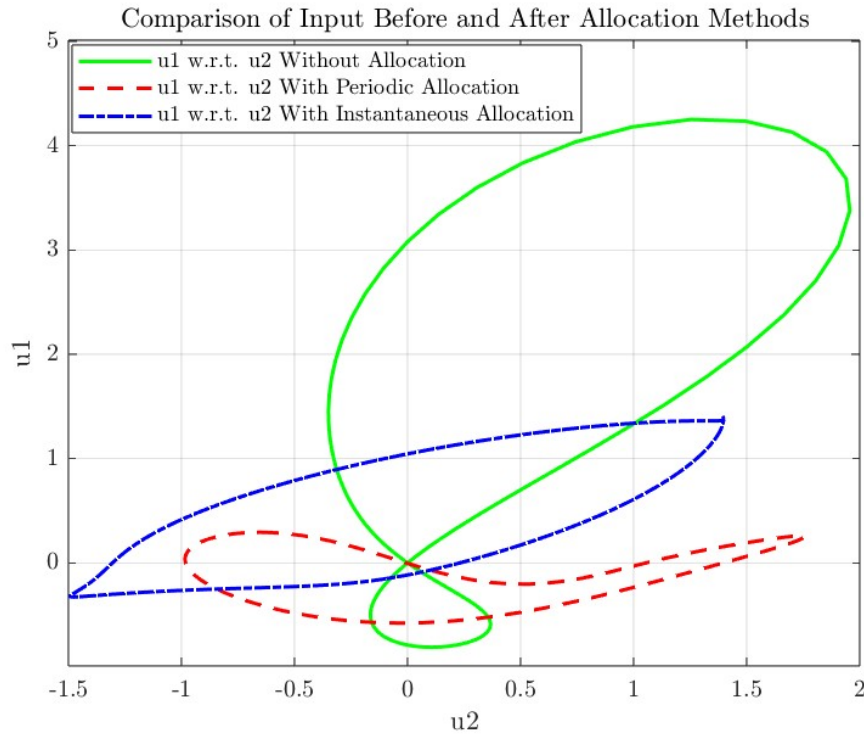


Figure 5.7: Scenario II. Steady-state input trajectory in the (u_1, u_2) plane, before and after allocation.

Figure 5.8 highlights the key message of this numerical study: when the reference trajectory is periodic, the proposed periodic allocation method achieves a drastic reduction of the periodic cost functional, thereby clearly outperforming the instantaneous alternative. This result confirms the intuition that a design tailored to constant references and instantaneous costs cannot capture the temporal structure of periodic signals, and thus cannot systematically minimize their periodic effect.

As anticipated in Sec. 5.3, the instantaneous method is inherently limited when applied outside its intended scope. Its greedy nature allows it to reduce the running cost at individual time instants, sometimes even driving the instantaneous value of $g(u_{ss}, z_{ss})$ below that obtained with periodic allocation. This behavior is visible in Figure 5.6, where the instantaneous allocator occasionally outperforms the periodic one on a pointwise basis. However, this advantage is short-lived and does not translate into improved overall performance. By contrast, the periodic allocator explicitly accounts for the entire trajectory over one fundamental period, optimizing the integral cost rather than isolated values. As a consequence, it consistently achieves a lower average running cost and therefore a superior periodic performance index.

The difference between the two methods is further illustrated in Figure 5.7, which plots the trajectory of the first input u_1 against the second input u_2 under three conditions: without allocation, with instantaneous allocation, and with periodic allocation. Without allocation, the trajectory covers a wide range in both directions, reflecting the lack of any attempt to penalize actuator usage. Instantaneous allocation already compresses the excursion of u_1 , in line with the higher weight assigned to this channel. Yet, the effect remains limited, since the method can only act locally and cannot exploit the redundancy in a temporally consistent manner. The periodic allocation, on the other hand, is most effective in squeezing the trajectory along the u_1 direction, strongly reducing the reliance on the heavily penalized input and re-distributing the effort onto u_2 in a way that is globally optimal over the period.

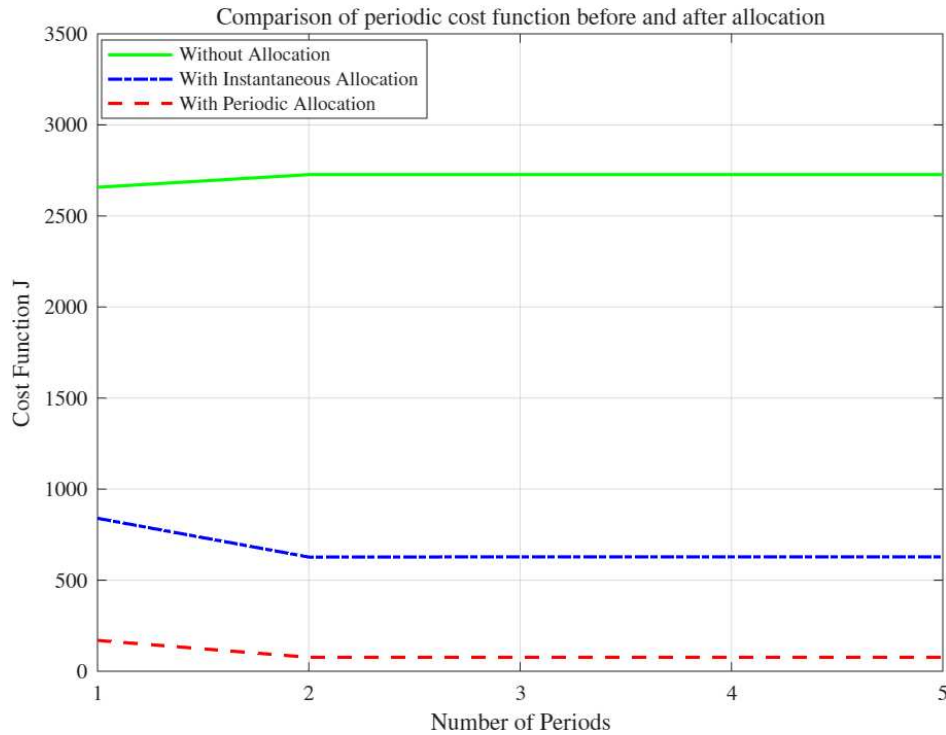


Figure 5.8: Scenario II. Comparison of periodic cost function $\int_0^T g(u_{ss}(\tau), z_{ss}(\tau))d\tau$ before and after allocation.

Taken together, these results demonstrate that periodic allocation brings a decisive advantage in scenarios with time-varying references. The proposed method not only preserves output invisibility, as required by the allocation framework, but also leverages the temporal structure of the problem to achieve a steady-state performance level unattainable with existing DCA approaches for nonlinear systems. By explicitly embedding periodicity into the allocator design, the method closes a gap in the current literature and extends the reach of DCA to a broader and more practical class of nonlinear control problems, where periodic signals and periodic performance criteria are the norm rather than the exception.

5.5 Discussion and Concluding Remarks

This chapter presented a sensitivity-based DCA framework for nonlinear systems subject to periodic reference trajectories and steady-state performance objectives. Building on the theory of nonlinear normal forms and uniform relative degree, an allocation architecture was developed that preserves output regulation while exploiting input redundancy to improve steady-state actuation performance. The key feature of the proposed method is the use of sensitivity functions to compute the gradient of a periodic Lagrangian cost, enabling a principled and systematic optimization procedure beyond pointwise or instantaneous allocation laws.

The resulting scheme guarantees output invisibility, preserves closed-loop stability of the nominal controller, and converges to a performance-improving allocation law in a rigorous “weak optimality” sense. Numerical results confirmed that, unlike classical or instantaneous allocation approaches, the proposed method can effectively exploit the temporal structure of periodic regimes to reduce actuator usage over a full period while maintaining nominal tracking behavior.

Overall, this chapter demonstrates that DCA can be systematically extended to nonlinear systems operating under periodic conditions by leveraging sensitivity analysis and gradient-based optimization. This framework forms a foundation for more advanced allocation strategies in nonlinear robotic and mechanical systems, and motivates further research on computationally efficient implementations, robustness to model uncertainty, and real-time extensions for more complex reference and actuation scenarios.

Chapter 6

Chapter 6

Dynamic Control Allocation for Nonlinear Systems via a Sensitivity Approach: Refinements and a Case Study

Abstract

The objective of this Chapter is to revisit the problem of Chap. 5 in a some things are repeated to ease the reading, making it possible to read this chapter independently from the previous one, and more formal discussion is provided with respect to the more introductory treatment in Chap. 5. Then the proposed technique of Chap. 5 is implemented and validated on a nonlinear mechanical system.

6.1 Introduction

This chapter as in Chap. 5 envisions a novel, systematic nonlinear DCA method based on gradient optimization and sensitivity functions, explicitly designed for periodic references and nonlinear system dynamics. The effectiveness of the approach is illustrated through a Mass-Spring-Damper (MSD) model in which the spring stiffness can be adjusted online. While classical MSD models are inherently linear, this parametric dependence induces a nonlinear behavior, enabling the study of advanced allocation strategies under more realistic actuation scenarios. Numerical simulations corroborate the effectiveness of the proposed strategy.

The chapter is structured as follows: Sec. 6.2 introduces the class of nonlinear models and the problem formulation. Sec. 6.3 outlines the proposed nonlinear DCA method. Numerical validations on an MSD system are provided in Sec. 6.4, followed by conclusions in Sec. 6.5.

Notation: For a real matrix M , transpose is denoted by M' , pseudo-inverse by $M^\#$, and kernel basis by $M^\perp$. The k -th derivative of $f(t)$ is $f^{(k)}(t)$.

6.2 Problem Formulation

DCA is a form of controller augmentation (see Fig. 6.1 and Fig. 2.2) for control systems with a plant having more inputs than outputs; hence, as in [48], the problem data consist of the plant P , a given stabilizing controller C , and a class of exogenous reference signals y_d to be tracked.

6.2.1 The plant

Consider a plant P with m inputs and p outputs, which is *input redundant* (namely, $m > p$) and has *uniform vector relative degree* [31, Sec. 11.5] denoted as $r = [r_1 \ r_2 \ \dots \ r_p]'$ and is in *normal form* [30, (5.7)-(5.8)]; then, P is described by the following equations:

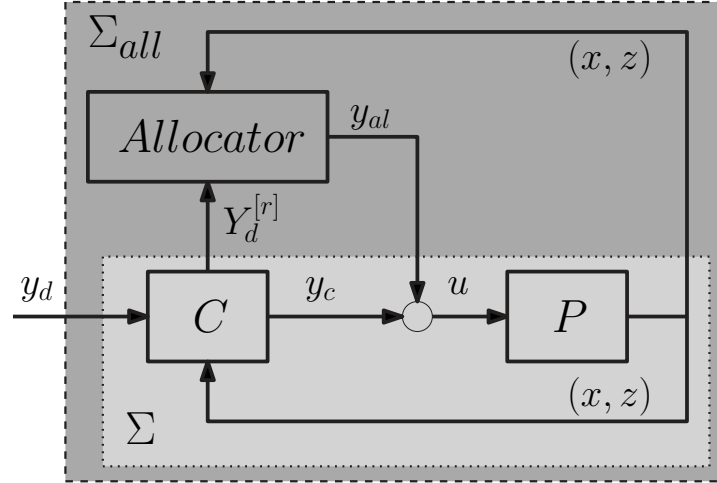


Figure 6.1: *A priori* given closed-loop control system Σ (dotted, light grey) augmented with DCA as Σ_{all} (dashed, dark grey).

$$\dot{x} = A_0 x + B_0(f_0(x, z) + g_0(x, z)u), \quad (6.1a)$$

$$\dot{z} = f_1(x, z) + g_1(x, z)u, \quad (6.1b)$$

$$y = C_0 x, \quad (6.1c)$$

In this formulation, the state variables are defined as $x \in \mathbb{R}^n$, where $n = \sum_{i=1}^p r_i$, and $z \in \mathbb{R}^k$, with control input $u \in \mathbb{R}^m$ and output $y \in \mathbb{R}^p$. The mappings involved in the system dynamics are given by $f_0 : \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^p$, $g_0 : \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^{p \times m}$, $f_1 : \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^k$, and $g_1 : \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^{k \times m}$.

The matrices used in the formulation are constructed as:

$$A_0 = \text{blkdiag}(J_0(r_1), J_0(r_2), \dots, J_0(r_p)), \quad (6.2a)$$

$$B_0 = \text{blkdiag}(\bar{e}_{(r_1)}, \bar{e}_{(r_2)}, \dots, \bar{e}_{(r_p)}), \quad (6.2b)$$

$$C_0 = \text{blkdiag}(\underline{e}_{(r_1)}, \underline{e}_{(r_2)}, \dots, \underline{e}_{(r_p)}), \quad (6.2c)$$

with the definitions:

$$J_0(q) := \begin{bmatrix} 0_{(q-1) \times 1} & I_{q-1} \\ 0 & 0_{1 \times (q-1)} \end{bmatrix}, \quad \bar{e}_{(q)} := \begin{bmatrix} 0_{(q-1) \times 1} \\ 1 \end{bmatrix},$$

$$\underline{e}_{(q)} := [1 \quad 0_{1 \times (q-1)}]$$

The resulting matrices have dimensions $A_0 \in \mathbb{R}^{n \times n}$, $B_0 \in \mathbb{R}^{n \times p}$, and $C_0 \in \mathbb{R}^{p \times n}$.

Remark 6.2.1

As explained in [31, Sec. 11.5], [30, Remark. 6.1.8], the existence of a uniform vector relative degree implies that the normal form in (6.1) is simpler than the general one in [30, Sec. 6.1], and also simpler than the intermediate normal form assumed in [11]. Moreover, uniform vector relative degree implies that $\text{rank}(g_0(x, z)) = p$ globally, and

then the pseudo-inverse $g_0^\sharp$ and the kernel basis $g_0^\perp$ are well-defined and have constant dimensions globally. $\square$

Summarizing the above discussion, the following assumption characterizes the class of plants considered in this chapter.

Assumption 6.2.1

Plant P is input redundant and in nonlinear normal form (6.1) with uniform vector relative degree r , smooth functions f_0, g_0, f_1, g_1 and $\text{rank}(g_0(x, z)) = p$ for all (x, z) . $\square$

6.2.2 The reference

The goal of the *a priori* given control system is to track a sufficiently smooth periodic reference signal y_d .

Assumption 6.2.2

Reference signal $y_d : \mathbb{R} \rightarrow \mathbb{R}^p$ is T -periodic, for $T > 0$. The h -th scalar component $y_{d,h}$ of y_d is r_h times continuously differentiable, for $h = 1, \dots, p$. $\square$

As usual in tracking problems, the derivatives of $y_{d,h}^{(k)}$ for $k = 1, \dots, r_k$ are assumed to be available⁸ for control purposes; for brevity, the following shorthand notations will be used,

$$\begin{aligned} y_d^{[r]} &:= \begin{bmatrix} y_{d,1}^{(r_1)} & y_{d,2}^{(r_2)} & \cdots & y_{d,p}^{(r_p)} \end{bmatrix}', \\ Y_{d,h}^{(v_h)} &:= \begin{bmatrix} y_{d,h} & y_{d,h}^{(1)} & \cdots & y_{d,h}^{(v_h)} \end{bmatrix}', h = 1, \dots, p, \\ Y_d^{[v]} &:= \begin{bmatrix} (Y_{d,1}^{(v_1)})' & \cdots & (Y_{d,p}^{(v_p)})' \end{bmatrix}', \end{aligned}$$

where either $(v, v_h) = (r, r_h)$ or $(v, v_h) = (r^-, r_h^-)$, with $r^- := [r_1 - 1 \ r_2 - 1 \ \cdots \ r_p - 1]'$ and $r_h^- := r_h - 1$, for $h = 1, \dots, p$. Note that by Assumption 6.2.2 (in particular, by continuity and periodicity of y_d and its derivatives), $Y_d^{[v]}$ is periodic, continuous, and bounded.

6.2.3 The controller

A static controller C of the form ⁹

⁸Either via differentiators or if y_d is generated by an exosystem of the form $\dot{w} = Sw, y_d = Fw$ via knowledge of w and the relation $y_d^{(k)} = FS^k w$.

⁹For simplicity, only a static controller is considered, so that the two items in Assumption 6.2.3 guarantee that Σ in Fig. 6.1 with $y_{d1} = g_0(x, z)^\perp v$ is input to state convergent with respect to y_d and v . The proposed results still hold when considering a dynamic controller C , with the only required modification of explicitly adding to Assumption 6.2.3 an input to state convergence hypothesis for Σ (hence, including the state of C).

$$y_c = h_c(x, z, Y_d^{[r]}) \quad (6.4)$$

is *a priori* given and ensures tracking and stabilization of Σ , see Fig. 6.1.

In view of the augmentation of C with the allocator to be designed, an additional input y_{al} that will be used for input allocation is introduced, and the interconnection of C to P is realized as

$$u = y_c + y_{al}. \quad (6.5)$$

Assumption 6.2.3

Controller C in (6.4) when connected to plant P in (6.1) as in (6.5) with $y_{al} = g_0(x, z)^\perp v$ ensures that:

- i. $x - Y_d^{[r-]}$ converges exponentially to zero;
- ii. the z subsystem is globally input-to-state convergent with respect to the input v .

□

Control law designs to achieve input-to-state convergence can be found in the literature, *e.g.* based on feedback linearization or backstepping-like procedures. Convergent systems (see *e.g.* [36], [39]) enjoy several properties similar to linear time invariant systems. The results in [39, Sec. 2.2.2], and in particular Properties 2.23, 2.24 and 2.25, immediately imply the following fact.

Fact 1. *Under Assumption 6.2.3, if y_d and v are asymptotically T -periodic, then all variables in Σ converge to a unique, bounded T -periodic steady-state evolution.*

The steady-state evolutions of (x, z, u) will be denoted (x_{ss}, z_{ss}, u_{ss}) ; item i. of Assumption 6.2.3 implies that $x_{ss} = Y_d^{[r-]}$ and that $y - y_d$ converges exponentially to zero.

It is stressed that DCA is a form of *controller augmentation*, and then controller C is *a priori* given in the problem formulation (cfr. [48]). Nevertheless, for concreteness, the following remark suggests a possible strategy to design C to satisfy Assumption 6.2.3; as a byproduct, the remark shows that the apparently peculiar requirement in item ii. of Assumption 6.2.3 could already be satisfied by the classic results in [30].

Remark 6.2.2

Considering the goal of tracking the reference signal y_d and the fact that the plant is in normal form, a natural choice is to design controller C as a static I/O feedback linearizing tracking controller [30, Sec. 5.1, 5.6]. In such a case, letting $y_{al} = g_0(x, z)^\perp v$, the output y_c of the controller C is chosen as:

$$y_c = g_0^\#(x, z) (u_{fl} + u_{tr}) + g_0^\perp(x, z)(u_z + v), \quad (6.6a)$$

$$u_{fl} = -f_0(x, z), \quad (6.6b)$$

$$u_{tr} = y_d^{[r]} + K(x - Y_d^{[r-]}), \quad (6.6c)$$

$$u_z = \sigma(x, z, Y_d^{[r]}), \quad (6.6d)$$

where $K = \text{blkdiag}(K_1, \dots, K_p)$ with $K_h \in \mathbb{R}^{1 \times r_h}$, for $h = 1, \dots, p$, is such that $A_K := A_0 + B_0 K$ is Hurwitz, and $\sigma(x, z, Y_d^{[r]})$ is such that

$$\dot{z} = \bar{f}_1(x, z, Y_d^{[r]}) + \bar{g}_1(x, z)v, \quad (6.7a)$$

is input-to-state convergent from v , where

$$\begin{aligned} \tilde{f}_1(x, z, Y_d^{[r]}) &:= f_1(x, z) + g_1(x, z)g_0^\#(x, z)(u_{fl} + u_{tr}), \\ \bar{f}_1(x, z, Y_d^{[r]}) &:= \tilde{f}_1(x, z, Y_d^{[r]}) + \bar{g}_1(x, z)\sigma(x, z, Y_d^{[r]}), \\ \bar{g}_1(x, z) &:= g_1(x, z)g_0^\perp(x, z). \end{aligned}$$

To see that the proposed controller C satisfies Assumption 6.2.3, note that under the action of (6.6a), (6.6b) and (6.6c), in the x subsystem in (6.1a) the nonlinearity (and the coupling with the z subsystem) is canceled by u_{fl} , and exponential stability and tracking of y_d are ensured by u_{tr} : in fact, the x dynamics becomes

$$\dot{x} = A_K x - B_0 K Y_d^{[r-]} + B_0 y_d^{[r]}, \quad (6.7b)$$

and taking into account that $\dot{Y}_d^{[r-]} = A_0 Y_d^{[r-]} + B_0 y_d^{[r]}$, the tracking error $e := x - Y_d^{[r-]}$ satisfies

$$\dot{e} = A_K e,$$

as required in item i. of Assumption 6.2.3. As for item ii. of Assumption 6.2.3, it holds by the very definition of $\sigma(x, z, Y_d^{[r]})$. State feedback linearization [30, Sec. 5.2] can also be used to design $\sigma(x, z, Y_d^{[r]})$ if the dynamics

$$\dot{z} = \tilde{f}_1(x, z, Y_d^{[r]}) + \bar{g}_1(x, z)u_z$$

is state feedback linearizable. □

6.2.4 The problem

Since the plant is input redundant, multiple input trajectories can achieve the same output evolution; the normal form makes it apparent that this can be achieved by modifying the evolution of the z -subsystem in (6.1), that under item i. of Assumption 6.2.3 is decoupled from the rest of the plant by controller (6.4) (for a deeper discussion from the geometric control point of view in the linear case, see the insightful analysis in [42]). The proposed DCA strategy exploits the redundancy to refine the plant input u by augmenting the controller output y_c with an allocation signal y_{al} and to achieve an optimized steady-state response, as specified in the next problem definition.

Problem 4. Consider the control system Σ described by (6.1), (6.4), (6.5) under Assumptions 6.2.1, 6.2.2, 6.2.3. Design, if possible, an input allocation device generating

y_{al} such that, when interconnected to Σ to form the allocated closed-loop system Σ_{all} as in Fig. 6.1, the following requirements are achieved for each initial state of Σ and y_d :

- R1)** (Stability) Σ_{all} admits a unique, globally asymptotically stable T -periodic steady-state response;
- R2)** (Output Invisibility) The response in the output y of Σ_{all} coincides with the response of Σ ;
- R3)** (Steady-State Optimality) The Lagrange cost functional¹⁰

$$J(u_{ss}, z_{ss}) = \int_0^T g(u_{ss}(\tau), z_{ss}(\tau)) d\tau, \quad (6.8)$$

where g is a smooth function, is minimized when y_{al} is chosen in a given class of signals $\mathbb{P}$.

An input allocation device will be said to solve Problem 4 in the *weak sense* if requirement R3) is relaxed by asking that $y_{al} \in \mathbb{P}$ is a *critical point* for the constrained minimization of (6.8), namely it satisfies the corresponding first order necessary conditions of optimality.

Remark 6.2.3

In DCA, optimization is focused on the steady-state response since most control systems are designed to drive the plant to a desirable (usually, constant or periodic) steady-state, and to remain close to it most of the time; so, optimizing the steady-state is usually the meaningful choice. The periodic case is most challenging, since while a (finite dimensional) constant steady-state $(u_{ss}, z_{ss}) \in \mathbb{R}^{m+k}$ is suitably evaluated by a cost function, the evaluation of a periodic (infinite dimensional) steady-state $(u_{ss}, z_{ss}) : \mathbb{R} \rightarrow \mathbb{R}^{m+k}$ requires a more complex cost functional, as in (6.8). $\square$

Remark 6.2.4

The extension of [48] to the nonlinear case was addressed in [11], but the study was limited to the constant steady-state case, and is not applicable to the periodic case. On the other hand, the adoption of Assumption 6.2.1 leads to much streamlined formulas (e.g. for the controller (6.2.2) in Remark 6.2.2), at the price of excluding some non-generic plants compatible with the corresponding assumption in [11]. $\square$

6.3 Allocator Design

To simplify the discussion, the structure of the allocator is introduced first and then its properties are presented.

¹⁰Since u_{ss}, z_{ss} are T -periodic, the value of the cost (6.8) is the same on any interval of length T , hence the integration is considered simply between 0 and T . Moreover, dependency of J on x_{ss} is not considered, since $x_{ss} = Y_d^{[r^-]}$ is fixed for any given y_d .

6.3.1 Allocator structure

Following the methodology of [10], [22] for linear systems, the allocator can conceptually be decomposed into three components:

- an *annihilator*, which ensures output invisibility of the allocation signal;
- an *internal model*, which generates a structured family of signals used to improve performance;
- an *optimizer*, which dynamically selects the optimal signal from the chosen family.

The *annihilator* is realized by defining the allocation signal so that it does not affect the x dynamics, namely as

$$y_{al} = g_0(x, z)^\perp v. \quad (6.9)$$

The *internal model* component is realized by selecting $v : [0, T] \rightarrow \mathbb{R}^{n_v}$ from a family of finitely parameterized T -periodic signals, expressed as:

$$v(t) = \sum_{h=1}^{n_v} \sum_{i=1}^{N_h} e_h \alpha_{h,i} \psi_i(t) = \Psi(t) \alpha, \quad (6.10a)$$

$$\alpha = [\alpha_{1,1} \cdots \alpha_{1,N_1} \mid \cdots \mid \alpha_{n_v,1} \cdots \alpha_{n_v,N_{n_v}}]', \quad (6.10b)$$

$$\alpha \in \mathcal{A} \subset \mathbb{R}^{\bar{N}}, \quad \bar{N} = \sum_{h=1}^{n_v} N_h, \quad n_v := m - p, \quad (6.10c)$$

where e_h denotes the h -th column of the identity matrix of size n_v , and $\mathcal{A}$ is a compact and convex set containing an open neighborhood of the origin (by (6.10a), $\alpha = 0$ implies $y_{al} \equiv 0$, *i.e.* no allocation). A viable choice is to set $N_1 = \cdots = N_{n_v} = 2N + 1$ and define the basis functions $\psi_i(t)$ as $\psi_i(t) := \cos(\omega_0 h t)$ for $i = 2h$, $h \in \{0, 1, \dots, N\}$, and $\psi_i(t) := \sin(\omega_0 h t)$ for $i = 2h - 1$, $h \in \{1, \dots, N\}$, with $\omega_0 := \frac{2\pi}{T}$.

For fixed y_d , the cost J in (6.8) depends on the choice of v , hence on the value of the parameter α ; thus, to optimize J , the constant α is replaced by an adaptive signal $\hat{\alpha}$, tuned by the remaining component of the allocator, *i.e.* the optimizer.

The *optimizer* is defined by the projected gradient descent:

$$\dot{\hat{\alpha}} = \text{proj}_{\mathcal{T}_{\mathcal{A}}(\hat{\alpha})} (-\gamma \nabla_{\alpha} J), \quad \alpha(0) = 0, \quad \gamma > 0, \quad (6.11)$$

where $\text{proj}_{\mathcal{T}_{\mathcal{A}}(\hat{\alpha})}(f(\hat{\alpha}))$ is the operator of (Euclidean) projection of vector $f(\hat{\alpha})$ on the tangent cone $\mathcal{T}_{\mathcal{A}}(\hat{\alpha})$ to $\mathcal{A}$ at point $\hat{\alpha} \in \mathcal{A}$ [27, Def. 3.1]. Roughly speaking, $\text{proj}_{\mathcal{T}_{\mathcal{A}}(\hat{\alpha})}(w)$ equals w as long as $\hat{\alpha}$ is in the interior of $\mathcal{A}$, but if $\hat{\alpha}$ is on the border of $\mathcal{A}$, then $\text{proj}_{\mathcal{T}_{\mathcal{A}}(\hat{\alpha})}(w)$ is the vector closest to w which points inside or is tangential to $\mathcal{A}$; this ensures that solutions to (6.11) starting inside $\mathcal{A}$ never leave $\mathcal{A}$ [27].

6.3.2 Allocator properties

The next lemma shows that the cost functional J admits a minimum (hence, a critical point) in the considered class of allocation signals; this will imply convergence of gradient descent, since the set of critical points is nonempty.

Lemma 6.3.1

Consider the cascade of (6.10), (6.9) and (6.7) under Assumptions 6.2.1, 6.2.2 and 6.2.3. There exists a global minimizer $\alpha^* \in \mathcal{A}$ for the cost functional $\bar{J}(\alpha) = J(u_{ss}(\alpha), z_{ss}(\alpha))$ defined in Problem 4. $\square$

Proof. Under the stated assumptions and for each fixed y_d , by Fact 1 the cascade of (6.10), (6.9) and (6.7) admits a unique bounded T -periodic steady-state solution (x_{ss}, z_{ss}, u_{ss}) which depends on signal v , and then is a continuous function of $\alpha \in \mathcal{A}$, with $\mathcal{A}$ compact. Hence, the cost functional J in (6.8) reduces to a continuous cost function $\bar{J}(\alpha)$. Since, by the extreme value theorem, a continuous function defined on a compact set attains its global minimum, there exists a minimizing $\alpha^* \in \mathcal{A}$. $\square$

The next lemma simplifies the minimization of cost (6.8) by replacing the need for computing the steady-state behaviour with the solution of a corresponding ODE on a finite interval $[0, T]$, under a periodic two-point boundary condition.

Lemma 6.3.2

Under the assumptions of Lemma 6.3.1, the minimization of the cost J in (6.8) can be reduced to a parameter optimization over α as follows:

$$\min_{\alpha \in \mathcal{A}} J(u_{ss}, z) \quad (6.12a)$$

$$\text{s.t. } \dot{z} = f(t, z, \alpha) \quad (6.12b)$$

$$z(0, \alpha) = z(T, \alpha) \quad (6.12c)$$

where the functions $f(t, z, \alpha)$ and $u_{ss}(t, z, \alpha)$ are given by:

$$f(t, z, \alpha) = f_1(Y_d^{[r-]}(t), z) + g_1(Y_d^{[r-]}(t), z)u_{ss}(t, z, \alpha), \quad (6.13a)$$

$$u_{ss}(t, z, \alpha) = h_c(Y_d^{[r-]}(t), z, Y_d^{[r]}(t)) + g_0^\perp(Y_d^{[r-]}(t), z)\Psi(t)\alpha, \quad (6.13b)$$

with $t \in [0, T]$. $\square$

Proof. It has to be shown that z specified by (6.12b), (6.12c) coincides with z_{ss} , since this implies that the values of (6.12a) and of the minimum of the cost in (6.8) coincide. By Fact 1 and the first part of the proof of Lemma 6.3.1, for each y_d and α the cascade of (6.10), (6.9) and (6.7) converges to the unique steady-state (x_{ss}, z_{ss}, u_{ss}) where $x_{ss} = Y_d^{[r-]}$ by item i. of Assumption 6.2.3. Equations (6.13) follow by replacing such value of x_{ss} and (6.10a) in (6.4), (6.9), (6.5) and in the right hand side of (6.1b). Since the steady-state response is a solution of the relative dynamics [30, Sec. 8.1], z_{ss} must satisfy (6.1b) with (x, u) replaced by (x_{ss}, u_{ss}) , namely (6.12b); periodicity and uniqueness of z_{ss} imply that it is fully characterized by the periodic boundary conditions (6.12c). $\square$

The implementation of the projected gradient descent (6.11) requires the computation of the gradient $\nabla_\alpha J$ for the cost J in (6.12a), which is enabled by the use of *sensitivity functions* (namely, derivatives of solutions to ODE's with respect to parameters; see [35, Sec. 3.3]) of the steady-state trajectories $z_{ss}(t, \alpha)$ and $u_{ss}(t, \alpha)$. Sensitivity functions allow computation of the gradient of J in (6.12) with respect to α even in the presence of complex dynamics and cost. For compactness, define for $\tau \in [0, T]$ the functions

$$\phi(\tau, \alpha) := z_{ss}(\tau, \alpha), \quad \pi(\tau, \alpha) := u_{ss}(\tau, \alpha), \quad (6.14)$$

with ϕ and π satisfying the equations for z and u_{ss} in (6.12) and (6.13b). To obtain the sensitivities of ϕ and π with respect to α , consider the two-point boundary value problem:

$$\frac{dS}{d\tau} = A(\tau, \alpha)S(\tau) + B(\tau, \alpha), \quad (6.15a)$$

$$S(0) = S(T), \quad (6.15b)$$

$$\frac{dz}{d\tau} = f(\tau, z(\tau), \alpha), \quad (6.15c)$$

$$z(0, \alpha) = z(T, \alpha), \quad (6.15d)$$

where $\tau \in [0, T]$ and

$$A(\tau, \alpha) := \frac{\partial f(\tau, z(\tau), \alpha)}{\partial z}, \quad (6.15e)$$

$$B(\tau, \alpha) := \frac{\partial f(\tau, z(\tau), \alpha)}{\partial \alpha}. \quad (6.15f)$$

Lemma 6.3.3

The gradient of the reduced cost functional J in (6.12) with respect to α is given by:

$$\nabla_{\alpha} J = \int_0^T \left(\frac{\partial g}{\partial z} \frac{\partial \phi}{\partial \alpha} + \frac{\partial g}{\partial u} \frac{\partial \pi}{\partial \alpha} \right) d\tau, \quad (6.16)$$

where the sensitivities of ϕ and π are given by

$$\frac{\partial \phi}{\partial \alpha}(\tau) = S(\tau), \quad (6.17a)$$

$$\frac{\partial \pi}{\partial \alpha}(\tau) = \frac{\partial u_{ss}}{\partial z} S(\tau) + \frac{\partial u_{ss}}{\partial \alpha}. \quad (6.17b)$$

where $S(\tau)$ satisfies (6.15) with $\tau \in [0, T]$. □

Proof. As for (6.17a), the statement follows by noting that (6.15) are the conditions provided in [35, Sec. 3.3] for sensitivity computation, adapted to the current setting; (6.17b) follows by the chain rule of differentiation.

As for (6.16), note that the cost functional J in (6.12a) is defined as an integral over $[0, T]$ of the instantaneous smooth cost $g(u_{ss}(t, \alpha), z(t, \alpha))$. Applying the chain rule under the integral, the gradient with respect to α is given by

$$\nabla_{\alpha} g' = \frac{\partial g}{\partial z} \frac{\partial \phi}{\partial \alpha} + \frac{\partial g}{\partial u} \frac{\partial \pi}{\partial \alpha}. \quad (6.18)$$

Integrating this expression over $[0, T]$ gives (6.16). □

The next lemma shows that a signal $\hat{\alpha} : \mathbb{R} \rightarrow \mathcal{A}$ described by the dynamics (6.11) (where $\nabla_{\alpha} J$ is evaluated at each time with α replaced by $\hat{\alpha}(t)$), converges to a constant value $\bar{\alpha}$ which provides a solution to Problem 4 in the weak sense.

Lemma 6.3.4

The projected gradient descent law (6.11) converges to a critical point $\bar{\alpha} \in \mathcal{A}$ of the constrained optimization problem (6.12). $\square$

Proof. The solution to (6.11) exists and is unique by [27, Thm. 6.13], whose hypotheses hold for the projected dynamics (6.11), since $\bar{J}(\alpha)$ is Lipschitz (being differentiable on the compact set $\mathcal{A}$), the Euclidean metric (denoted g in [27]) used for projections is also Lipschitz, and the set $\mathcal{A}$ is prox-regular (since it is convex); moreover, the projection on $\mathcal{T}_{\mathcal{A}(\hat{\alpha})}$ (which is well defined since $\mathcal{A}$ is closed and convex) ensures that $\hat{\alpha}(t) \in \mathcal{A}$ for all t . Since $\mathcal{A}$ is Clarke regular (since it is convex) and the Euclidean metric is continuous, [27, Proposition 8.6] applies, implying that for any initial condition $\hat{\alpha}(0) \in \mathcal{A}$, the projected gradient dynamics in (6.11) converge to a constant value in the set of critical points of the optimization problem (6.12); such a set is nonempty by Lemma 6.3.1. $\square$

Putting together all previous constructions and properties, the following main result is obtained.

Theorem 6.3.1

Under Assumptions 6.2.1, 6.2.2 and 6.2.3, the DCA scheme (6.9), (6.10), (6.16), (6.11), (6.15), and (6.17) with α in (6.10a) replaced with $\hat{\alpha}(t)$ from (6.11) solves Problem 4 in the weak sense. $\square$

Proof. It needs to be shown that each requirement in the problem definition is satisfied. As for R1), note that Σ_{all} is actually a cascade of two subsystems: the first subsystem made up by the projected gradient dynamics (6.11) and the internal model (6.10), feeding the second subsystem made up by the annihilator (6.9) connected with Σ (the *a priori* given closed-loop system of plant P and controller C). For any given y_d satisfying Assumption 6.2.2, by Lemma 6.3.4 the dynamics (6.11) has a unique solution which converges to a constant value $\bar{\alpha}$, and then the output v of the first subsystem converges to a T -periodic function expressed by (6.10a) with α replaced with $\bar{\alpha}$. By Assumption 6.2.3 and Fact 1, the state of the second subsystem will also converge to a unique T -periodic function, thus proving the claim.

As for R2), (6.1a) and (6.9) imply that y_{al} has no effect on the x subsystem, independently of the choice of v .

As for R3), by Lemma 6.3.4 the projected gradient dynamics in (6.11) converge to a critical point of problem (6.12); hence, requirement R3 is satisfied in the weak sense. $\square$

Remark 6.3.1

For easier implementation, instead of the continuous time update rule (6.11), a discrete-time update mechanism for the allocation parameter $\hat{\alpha}$ can be used. Specifically, a sequence of updates $\hat{\alpha}_k$ can be defined as

$$\hat{\alpha}_{k+1} = \text{proj}_{\mathcal{A}}(\hat{\alpha}_k - \gamma \nabla_{\alpha} J), \quad (6.19)$$

where γ is a small positive scalar (step size), and $\nabla_{\alpha} J$ is computed as in Lemma 6.3.3 at $\hat{\alpha}_k$. $\square$

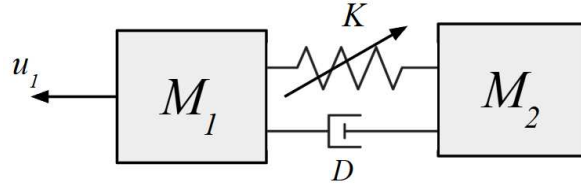


Figure 6.2: A mass spring damper system with variable stiffness.

6.4 Application to a Mechanical System

6.4.1 Problem Data

To demonstrate the applicability of the proposed method, we implement it on a two-mass spring-damper system, illustrated in Fig. 6.2. The system consists of two masses M_1 and M_2 , with corresponding positions x_1 and z_1 , and velocities $\dot{x}_2$ and $\dot{z}_2$, respectively. The control inputs are defined as u_1 , representing the force applied to M_1 (see Fig. 6.2), and $u_2 = K$, the spring stiffness coefficient, which is assumed to be dynamically adjustable [37].

The system dynamics can be expressed in the following form, consistent with (6.1):

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{D(z_2 - x_2)}{M_1} \\ z_2 \\ \frac{D(x_2 - z_2)}{M_2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{M_1} & \frac{(z_1 - x_1)}{M_1} \\ 0 & 0 \\ 0 & \frac{(x_1 - z_1)}{M_2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \quad (6.20)$$

The objective is for the system output $y = x_1$ to track the periodic reference trajectory $y_d(t) = 0.2 \sin((2\pi t)/T) + 1$ with $T = 10$. The *a priori* given controller¹¹ is

$$u = \begin{bmatrix} \frac{1}{M_1} & \frac{(z_1 - x_1)}{M_1} \\ 0 & \frac{(x_1 - z_1)}{M_2} \end{bmatrix}^{-1} \left(- \begin{bmatrix} \frac{D(z_2 - x_2)}{M_1} \\ \frac{D(x_2 - z_2)}{M_2} \end{bmatrix} + \begin{bmatrix} \kappa_0 \bar{e} + \ddot{y}_d \\ -\kappa_1 z \end{bmatrix} \right),$$

where $\kappa_0, \kappa_1 \in \mathbb{R}^{1 \times 2}$ are proportional gain vectors, $\bar{e} = (Y_d^{[r]} - x)$ with $Y_d^{[r]} = [y_d, \dot{y}_d]'$ and $y_d^{[r]} = \ddot{y}_d$.

6.4.2 Allocator Design

The allocator is designed as explained in Sec. 6.3, assuming $v = \alpha_1 + \alpha_2 \sin((2\pi t)/T) + \alpha_3 \cos((2\pi t)/T)$ and using a discretized form of projected gradient descent as specified in Remark 6.3.1. The purpose is to reduce the Lagrangian cost functional (6.8) with $g(u, z) = u^\top W u$ using a weighted quadratic penalty with $W = \text{diag}(100, 1)$, to penalize the use of the first input more heavily.

Figure 6.3 and Figure 6.4 illustrate the behavior of the two input channels before and after the periodic control allocation. The proposed allocator reduces the excursion of the first input u_1 , which is the main objective of the weighted cost design. As shown in Figure 6.4, this reduction is achieved by permitting more aggressive modulation of the second input u_2 , which is less costly in the optimization. Figure 6.5 offers a 3D visualization of the evolution of the input trajectory throughout the optimization process: the reduction of u_1 is particularly evident when observing the transformation from a mainly horizontal oscillation at the beginning to a mainly vertical one at the end. For the sake of space, plant output responses are not shown, but all input trajectories in Figure 6.5 induce the same output response, confirming that allocation does not interfere with closed-loop trajectory tracking.

¹¹This controller was actually designed as specified in Remark 6.2.2.

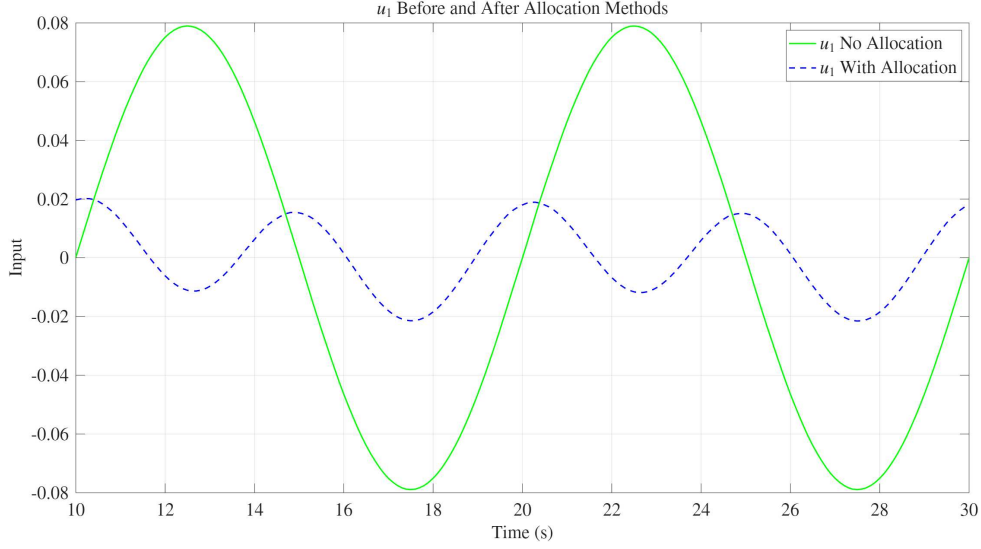


Figure 6.3: First input channel before and after allocation.

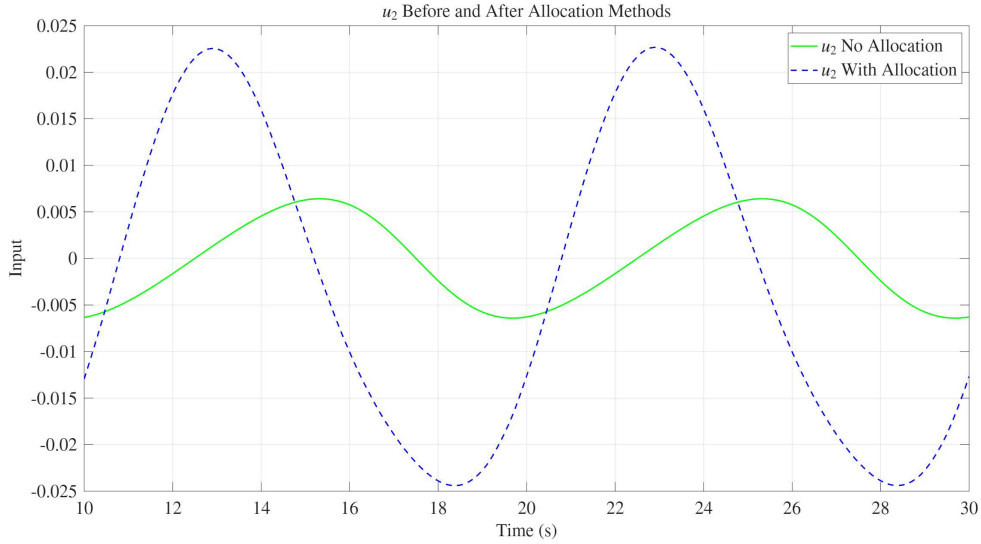


Figure 6.4: Second input channel before and after allocation. u_2 increases as a price to be paid in order to minimize u_1 .

6.5 Discussion and Concluding Remarks

In this chapter, we have revisited the novel DCA method tailored to nonlinear systems (from Chap. 5) subject to periodic reference signals and cost functions, by leveraging sensitivity functions and a projected gradient descent. The effectiveness of the proposed periodic DCA through simulations on a two-mass spring-damper system.

Results demonstrated a significant reduction in the usage of costly input channels while maintaining output invisibility and improving overall cost performance over time.

Future work will explore extending this framework to more complex and over-actuated robotic systems, where periodic input optimization and redundancy management are critical for agile and energy-efficient behavior.

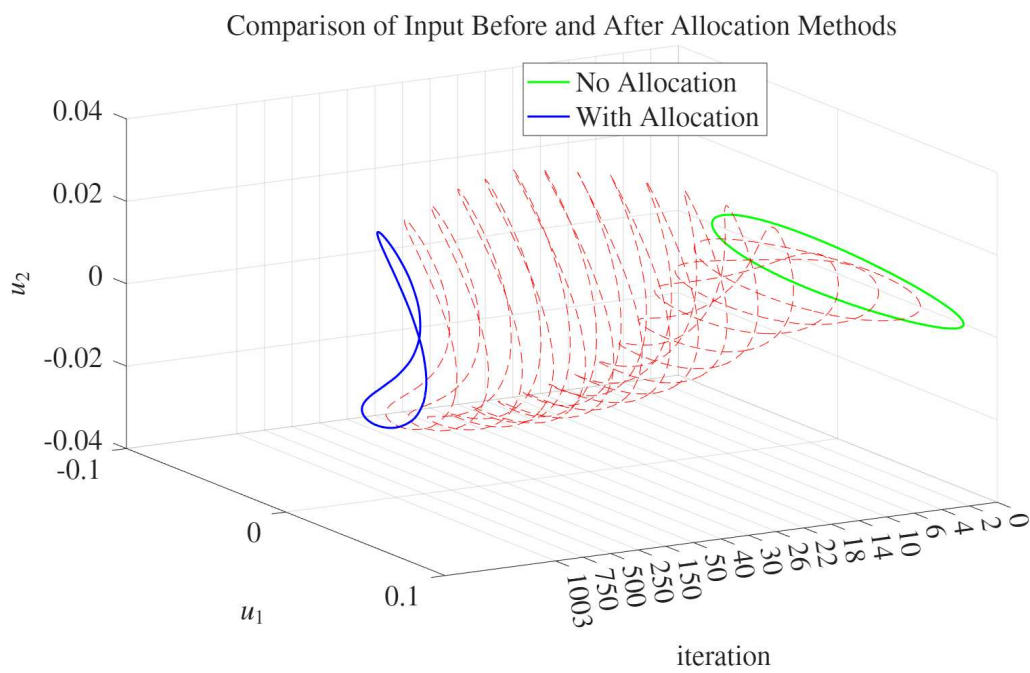


Figure 6.5: Input evolution, from unallocated to allocated. For clarity, unequally-spaced iteration numbers are shown.

Chapter 7

Chapter 7

Conclusions and Perspectives

7.1 Summary of Contributions

The research presented in this dissertation has advanced the theory and practice of DCA. While traditional allocation methods have mostly been confined to linear settings or instantaneous (pointwise) optimizations, the work developed here has progressively extended the scope of DCA in the development of richer internal models capable of representing nontrivial allocation signals as well as the exploitation of sensitivity-based optimization techniques for a large group of nonlinear systems to address cost functionals over periodic trajectories while maintaining the output invisibility.

Firstly, in Chapter 3, we introduced a novel spline-based DCA architecture. By enlarging the admissible class of allocation signals from constants or sinusoids (as in earlier literature) to polynomial and spline-type trajectories, the allocator gains flexibility and expressiveness. This advance allows for finer adaptation of redundant inputs to improve steady-state performance, without compromising closed-loop stability or output behavior. The theoretical analysis, supported by Theorem 1, proved that the proposed spline-based allocator satisfies the core requirements of stability, output invisibility, and steady-state optimality. Numerical experiments further confirmed the superiority of the spline-based approach, particularly in scenarios with non-constant references such as sawtooth trajectories.

Moreover, in Chapter 4, we focused on the implementation of DCA strategies on a concrete and challenging application: a UAV equipped with tilting-twisting propellers. Two methods were compared: an instantaneous allocator and a periodic spline-based allocator. Both were able to reduce actuator usage according to the chosen quadratic cost functional, thus validating the general allocation framework. The comparison revealed a clear trade-off: the instantaneous allocator is simpler, introduces no transient states, and is provably optimal when actuator dynamics are negligible; conversely, the periodic spline-based allocator exhibits greater robustness and smoother behavior when actuator dynamics are present. This application demonstrated that the proposed allocation strategies are not merely theoretical constructs but can be meaningfully deployed in advanced aerospace systems, highlighting their potential for practical adoption.

In Chapter 5, we presented the extension of DCA to nonlinear systems subject to periodic reference signals and Lagrangian cost functionals, consolidating the theoretical developments and their practical validation within a framework. The central innovation lies in the use of sensitivity functions to compute gradients of periodic cost functionals with respect to allocation parameters, thereby enabling gradient-based optimization in a nonlinear setting that had previously resisted tractable solutions. By embedding sensitivity dynamics into the allocator design and parameterizing the allocation signals through periodic basis functions, the framework optimizes actuator usage over an entire period while preserving the fundamental property of output invisibility. This guarantees that the allocator improves internal input distribution without compromising the nominal closed-loop output.

Lastly, in Chapter 6, the proposed method was analyzed and implemented through simulations on a nonlinear two-mass spring-damper system with adjustable stiffness, chosen as an academic benchmark for redundant actuation under periodic operation. Despite its simplicity, this system captures key features of nonlinear coupling and

input redundancy, demonstrating how the sensitivity-based allocator can effectively redistribute actuator effort, minimize the periodic cost, and maintain perfect output tracking. The results confirm the theoretical predictions, showing that the periodic allocation scheme achieves substantial steady-state performance improvements compared with instantaneous strategies when non-constant references are present. Overall, this chapter establishes a comprehensive methodology and validation for dynamic control allocation in nonlinear, periodically forced systems, bridging the gap between theory and application and highlighting the feasibility of implementing sensitivity-based optimization.

7.2 Applications and Validation

The application studies presented in Chapters 4 and 6 serve two complementary purposes. On one hand, they validate the theoretical guarantees: in both the UAV and the MSD academic example, the allocator was able to reduce input costs while preserving output trajectories. On the other hand, they provide insights into practical deployment. In the UAV case, the comparison between instantaneous and periodic methods revealed that robustness against actuator dynamics is an essential consideration, thereby motivating the adoption of periodic allocators in realistic platforms. In the MSD case, the results highlighted how input redundancy can be exploited to reduce costly actuation channels, a consideration relevant for robotics and other physical applications.

7.3 Limitations and Open Questions

Despite the progress achieved, several limitations of the present work must be acknowledged. First, the theoretical results rely on assumptions such as uniform relative degree and global normal forms, which, while standard in nonlinear control theory, may restrict the applicability of the framework to certain classes of systems. Relaxing these assumptions remains an important open problem.

Second, actuator dynamics were modeled in a simplified way (e.g., as first-order systems) in the UAV case study. While sufficient to illustrate the contrast between instantaneous and periodic allocation, more complex or uncertain actuator dynamics may require refined models and robust analysis.

Third, the optimization framework guarantees convergence in a weak sense, namely to stationary points within a compact set. Global optimality cannot be guaranteed in nonlinear, nonconvex settings, and the exploration of stronger convergence guarantees remains an open theoretical challenge.

Finally, the numerical validations were conducted in simulation. Although simulation results are encouraging, experimental validation on real hardware platforms is necessary to fully assess the robustness, performance, and computational feasibility of the proposed methods.

7.4 Future Directions

Building on the foundations laid in this dissertation, several promising directions for future research can be identified. On the theoretical side, extending the allocation framework to systems with non-uniform relative degree, state-dependent input constraints, or uncertain dynamics would broaden its applicability. The integration of second-order optimization methods, or hybrid strategies combining gradient and Newton-like approaches, could accelerate convergence and improve robustness.

On the computational side, the development of more efficient algorithms for solving sensitivity boundary value problems is an important step toward real-time implementa-

tion. Parallel computing and model reduction techniques could be leveraged to reduce computational burden, especially in high-dimensional applications.

On the application side, extending the validation to experimental platforms is essential. Potential testbeds include aerial vehicles with redundant propulsion, robotic manipulators with variable stiffness joints, and legged robots exhibiting periodic gaits. In these systems, the ability to reallocate input effort optimally while preserving output tracking can have a decisive impact on energy efficiency, fault tolerance, and agility.

Finally, a broader integration of DCA with higher-level control architectures represents an exciting avenue. By embedding allocation into trajectory planning and model predictive control frameworks, it may be possible to achieve a seamless connection between decision-making and low-level actuation, thereby enabling more sophisticated autonomous behaviors.

7.5 Concluding Remarks

In conclusion, this dissertation has significantly advanced the state of the art in DCA by extending its applicability from linear and instantaneous settings to nonlinear systems with periodic references and cost functionals. The combination of spline-based internal models, sensitivity-based gradient computation, and output invisibility guarantees has produced a coherent and versatile allocation framework. Through both theoretical analysis and application studies, the work has demonstrated that DCA can achieve substantial performance improvements in redundant nonlinear systems while preserving closed-loop stability and output behavior.

The contributions of this research thus provide not only novel theoretical insights but also practical tools for addressing redundancy in complex control systems. By bridging the gap between theory and applications, and by opening multiple directions for future development, this work lays the groundwork for a new generation of allocation strategies in advanced robotics, aerospace, and beyond.

Bibliography

- [1] Akbari, S., Galeani, S., Manca, G., and Sassano, M., “A sensitivity approach to periodic control allocation in nonlinear systems,” in *IEEE Conference on Decision and Control (CDC)*, Accepted for publication, 2025.
- [2] Akbari, S., Galeani, S., Manca, G., and Sassano, M., “Dynamic control allocation for nonlinear systems via a sensitivity approach,” *IEEE Control Systems Letters (LCCS)*, 2025, Accepted for publication.
- [3] Akbari, S., Galeani, S., and Sassano, M., “Towards Spline-based Dynamic Input Allocation,” in *2023 International Conference on Control, Automation and Diagnosis (ICCAD)*, 2023, pp. 1–6. DOI: [10.1109/ICCAD57653.2023.10152316](https://doi.org/10.1109/ICCAD57653.2023.10152316).
- [4] Akbari, S., Manca, G., Galeani, S., and Sassano, M., “Spline-based Input Allocation On An Overactuated Tilting and Twisting Unmanned Aerial Vehicle,” in *10th International Conference on Control, Decision and Information Technologies (CoDIT)*, 2024, pp. 2329–2334. DOI: [10.1109/CoDIT62066.2024.10708417](https://doi.org/10.1109/CoDIT62066.2024.10708417).
- [5] Al Khudir, K., Halvorsen, G., Lanari, L., and De Luca, A., “Stable Torque Optimization for Redundant Robots Using a Short Preview,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2046–2053, 2019. DOI: [10.1109/LRA.2019.2899667](https://doi.org/10.1109/LRA.2019.2899667).
- [6] Bodson, M., “Evaluation of Optimization Methods for Control Allocation,” *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 703–711, 2002. DOI: [10.2514/2.4937](https://doi.org/10.2514/2.4937).
- [7] Bordignon, K., “Constrained control allocation for systems with redundant control effectors,” in *Proceedings of the 1996 IEEE Conference on Decision and Control*, 1996, pp. 4032–4037.
- [8] Byrnes, C. I., Priscoli, F. D., and Isidori, A., *Output Regulation of Uncertain Nonlinear Systems* (Systems & Control: Foundations & Applications). Birkhäuser Boston, 1997. DOI: [10.1007/978-1-4612-2020-6](https://doi.org/10.1007/978-1-4612-2020-6).
- [9] Cocetti, M., Serrani, A., and Zaccarian, L., “Linear output regulation with dynamic optimization for uncertain linear over-actuated systems,” *Automatica*, vol. 97, pp. 214–225, 2018. DOI: [10.1016/j.automatica.2018.08.002](https://doi.org/10.1016/j.automatica.2018.08.002).
- [10] Cristofaro, A. and Galeani, S., “Output invisible control allocation with steady-state input optimization for weakly redundant plants,” in *53rd IEEE Conference on Decision and Control*, IEEE, 2014, pp. 4246–4253. DOI: [10.1109/CDC.2014.7040051](https://doi.org/10.1109/CDC.2014.7040051).
- [11] Cristofaro, A., Galeani, S., and Serrani, A., “Output invisible control allocation with asymptotic optimization for nonlinear systems in normal form,” in *IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 2563–2568. DOI: [10.1109/CDC.2017.8264030](https://doi.org/10.1109/CDC.2017.8264030).
- [12] Cristofaro, A. and Johansen, T. A., *Fault-Tolerant Control Allocation: An Unknown Input Observer based approach*, 2013. DOI: [10.1016/j.automatica.2014.05.007](https://doi.org/10.1016/j.automatica.2014.05.007).
- [13] Davison, E., “The robust control of a servomechanism problem for linear time-invariant multivariable systems,” in *IEEE Transactions on Automatic Control*, vol. 21, no. 1, pp. 25–34, 1976. DOI: [10.1109/TAC.1976.1101137](https://doi.org/10.1109/TAC.1976.1101137).
- [14] Decastro, R., “Safe and high-performance control allocation,” *IEEE Transactions on Automatic Control*, vol. 66, no. 7, pp. 1–12, 2021. DOI: [10.1109/TAC.2021.3096882](https://doi.org/10.1109/TAC.2021.3096882).
- [15] Ducard, G. and Hua, M.-D., “WCA: A New Efficient Nonlinear Adaptive Control Allocation for Planar Hexacopters,” *IEEE Access*, vol. 11, pp. 37 714–37 748, 2023. DOI: [10.1109/ACCESS.2023.3261240](https://doi.org/10.1109/ACCESS.2023.3261240).

- [16] Durham, W., “Constrained Control Allocation,” *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 4, pp. 717–725, 1993. DOI: [10.2514/3.21072](https://doi.org/10.2514/3.21072).
- [17] Enns, D., “Control allocation approaches,” in *Guidance, Navigation, and Control Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 1998. DOI: [10.2514/6.1998-4109](https://doi.org/10.2514/6.1998-4109).
- [18] Fotiadis, F., Vamvoudakis, K., and Jiang, Z.-P., “Data-driven actuator allocation for actuator redundant systems,” *IEEE Transactions on Automatic Control*, vol. 69, no. 4, pp. 1–12, 2024. DOI: [10.1109/TAC.2023.3328993](https://doi.org/10.1109/TAC.2023.3328993).
- [19] Francis, B. and Wonham, W., “The internal model principle of control theory,” *Automatica*, vol. 12, no. 5, pp. 457–465, 1976. DOI: [10.1016/0005-1098\(76\)90006-6](https://doi.org/10.1016/0005-1098(76)90006-6).
- [20] Furci, M., Nainer, C., Zaccarian, L., and Franchi, A., “Input allocation for the propeller-based overactuated platform ROSPO,” *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2462–2476, 2020. DOI: [10.1109/TCST.2019.2944341](https://doi.org/10.1109/TCST.2019.2944341).
- [21] Galeani, S., Serrani, A., Varano, G., and Zaccarian, L., “On input allocation-based regulation for linear over-actuated systems,” in *Automatica*, vol. 51, 2015, pp. 106–113. DOI: [/10.1016/j.automatica.2014.10.112](https://doi.org/10.1016/j.automatica.2014.10.112).
- [22] Galeani, S. and Pettinari, S., “On dynamic input allocation for fat plants subject to multi-sinusoidal exogenous inputs,” in *53rd IEEE Conference on Decision and Control*, 2014, pp. 2396–2403. DOI: [10.1109/CDC.2014.7039754](https://doi.org/10.1109/CDC.2014.7039754).
- [23] Gerber, M. J. and Tsao, T.-C., “Twisting and Tilting Rotors for High-Efficiency, Thrust-Vectored Quadrotors,” *Journal of Mechanisms and Robotics*, no. 6, p. 061 013, 2018. DOI: [10.1115/1.4041261](https://doi.org/10.1115/1.4041261).
- [24] Hafez, M. A. and Broucke, M. E., “Optimal Steady-State Regulation Using an Internal Model and State Feedback,” in *2024 American Control Conference (ACC)*, 2024, pp. 3237–3244. DOI: [10.23919/ACC60939.2024.10644515](https://doi.org/10.23919/ACC60939.2024.10644515).
- [25] Hafez, M. A., Uzeda, E. M., and Broucke, M. E., “Optimal Steady-State Regulation by State Feedback,” *IEEE Transactions on Automatic Control*, no. 9, pp. 6042–6057, 2024. DOI: [10.1109/TAC.2024.3393791](https://doi.org/10.1109/TAC.2024.3393791).
- [26] Härkegård, O. and Glad, T., “Resolving actuator redundancy - optimal control vs. control allocation,” *Automatica*, vol. 40, no. 10, pp. 1751–1757, 2004. DOI: [10.1016/j.automatica.2004.09.007](https://doi.org/10.1016/j.automatica.2004.09.007).
- [27] Hauswirth, A., Bolognani, S., and Dörfler, F., “Projected dynamical systems on irregular, non-euclidean domains for nonlinear optimization,” *SIAM J. Control Optim.*, vol. 59, no. 1, pp. 635–668, 2021.
- [28] Hitchin, N. J., Newstead, P. E., and Oxbury, W. M., Eds., *Vector bundles in algebraic geometry: Durham 1993* (London Mathematical Society lecture note series 208), 1. publ., transf. to digital print. 2009. Cambridge University Press, 2009.
- [29] Huang, J., *Nonlinear output regulation: theory and applications* (Advances in design and control). Society for Industrial and Applied Mathematics, 2004. DOI: [10.1137/1.9780898718683](https://doi.org/10.1137/1.9780898718683).
- [30] Isidori, A., *Nonlinear Control Systems* (Communications and Control Engineering), Sontag, E. D., Thoma, M., Isidori, A., and Van Schuppen, J. H., Eds. Springer London, 1995. DOI: [10.1007/978-1-84628-615-5](https://doi.org/10.1007/978-1-84628-615-5).
- [31] Isidori, A., *Nonlinear Control Systems II* (Communications and Control Engineering). Springer London, 1999. DOI: [10.1007/978-1-4471-0549-7](https://doi.org/10.1007/978-1-4471-0549-7).
- [32] Johansen, T., “Optimizing nonlinear control allocation,” in *43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, vol. 4, 2004, 3435–3440 Vol.4. DOI: [10.1109/CDC.2004.1429240](https://doi.org/10.1109/CDC.2004.1429240).
- [33] Johansen, T. A. and Fossen, T. I., “Control allocation—A survey,” *Automatica*, vol. 49, no. 5, pp. 1087–1103, 2013. DOI: [10.1016/j.automatica.2013.01.035](https://doi.org/10.1016/j.automatica.2013.01.035).

- [34] Kailath, T., *Linear systems* (Prentice-Hall information and system science series). Prentice-Hall, 1980.
- [35] Khalil, H. K., *Nonlinear systems*, eng, 2. ed. Prentice Hall, 1996.
- [36] Lohmiller, W. and Slotine, J.-J. E., “On Contraction Analysis for Non-linear Systems,” *Automatica*, vol. 34, no. 6, pp. 683–696, 1998. DOI: [10.1016/S0005-1098\(98\)00019-3](https://doi.org/10.1016/S0005-1098(98)00019-3).
- [37] Mathews, C. W. and Braun, D. J., *Design of a Variable Stiffness Spring with Human-Selectable Stiffness*, 2023. DOI: [10.48550/ARXIV.2302.13471](https://doi.org/10.48550/ARXIV.2302.13471).
- [38] Passenbrunner, T., Sassano, M., and Zaccarian, L., “Optimality-based dynamic allocation with nonlinear first-order redundant actuators,” *European Journal of Control*, vol. 30, pp. 1–12, 2016. DOI: [10.1016/j.ejcon.2016.04.002](https://doi.org/10.1016/j.ejcon.2016.04.002).
- [39] Pavlov, A., Wouw, N. v. d., and Nijmeijer, H., *Uniform output regulation of nonlinear systems: a convergent dynamics approach* (Systems and control : foundations and applications). Birkhäuser, 2006.
- [40] Petersen, J. and Bodson, M., “Constrained quadratic programming techniques for control allocation,” in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, 2003, pp. 3378–3383. DOI: [10.1109/CDC.2003.1271666](https://doi.org/10.1109/CDC.2003.1271666).
- [41] Ryll, M., Muscio, G., Pierri, F., *et al.*, “6D interaction control with aerial robots: The flying end-effector paradigm,” *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1045–1062, 2019. DOI: [10.1177/0278364919856694](https://doi.org/10.1177/0278364919856694).
- [42] Serrani, A., “Output regulation for over-actuated linear systems via inverse model allocation,” in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 1234–1239. DOI: [10.1109/CDC.2012.6426209](https://doi.org/10.1109/CDC.2012.6426209).
- [43] Serrani, A., Isidori, A., and Marconi, L., “Semiglobal nonlinear output regulation with adaptive internal model,” *IEEE Transactions on Automatic Control*, vol. 46, no. 8, pp. 1178–1194, 2001. DOI: [10.1109/9.940923](https://doi.org/10.1109/9.940923).
- [44] Seuret, A., Vazquez, R., and Zaccarian, L., “A hybrid dynamical system approach to the impulsive control of spacecraft rendezvous,” in *2024 European Control Conference (ECC)*, 2024, pp. 1–6. DOI: [10.23919/ECC64448.2024.10590979](https://doi.org/10.23919/ECC64448.2024.10590979).
- [45] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G., *Robotics* (Advanced Textbooks in Control and Signal Processing). Springer London, 2009. DOI: [10.1007/978-1-84628-642-1](https://doi.org/10.1007/978-1-84628-642-1).
- [46] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft control and simulation: dynamics, control design and autonomous systems*, Third edition. Wiley, 2016.
- [47] Tommasi, G., Galeani, S., Pironti, A., Varano, G., and Zaccarian, L., “Nonlinear dynamic allocator for optimal input/output performance trade-off: Application to the jet tokamak shape controller,” in *Automatica*, vol. 47, 2011, pp. 1002–1011. DOI: [10.1016/j.automatica.2011.01.080](https://doi.org/10.1016/j.automatica.2011.01.080).
- [48] Zaccarian, L., “Dynamic allocation for input redundant control systems,” *Automatica*, vol. 45, no. 6, pp. 1431–1438, 2009. DOI: [10.1016/j.automatica.2009.01.013](https://doi.org/10.1016/j.automatica.2009.01.013).