Andrea Spinelli

A Set-Based Design Space Exploration Framework for Hybrid-Electric Aicraft Design

School of Aerospace, Transport and Manufacturing
Centre for Propulsion and Thermal Power

PhD
Academic Year: 2022–2023

Supervisors: Dr Timoleon Kipouros
Prof Panagiotis Laskaridis

August, 2023

School of Aerospace, Transport and Manufacturing
Centre for Propulsion and Thermal Power

PhD

Academic Year: 2022–2023

Andrea Spinelli

A Set-Based Design Space Exploration Framework for Hybrid-Electric Aicraft Design

Supervisors: Dr Timoleon Kipouros
Prof Panagiotis Laskaridis

August, 2023

© Cranfield University 2023. All rights reserved. No part of this publication may be reproduced without the written permission of the copyright owner.

Academic Integrity Declaration

I declare that:

- the thesis submitted has been written by me alone.
- the thesis submitted has not been previously submitted to this university or any other.
- that all content, including primary and/or secondary data, is true to the best of my knowledge.
- that all quotations and references have been duly acknowledged according to the requirements of academic research.

I understand that to knowingly submit work in violation of the above statement will be considered by examiners as academic misconduct.

Successful design is not the achievement of perfection but the minimization and accommodation of imperfection.

Henry Petroski

We're not that much smarter than we used to be, even though we have much more information - and that means the real skill now is learning how to pick out the useful information from all this noise. [...] We must become more comfortable with probability and uncertainty.

Nate Silver
The Signal and The Noise

Abstract

Engineering design is characterised by uncertainty caused by a lack of experience and information. The traditional approach focuses on iterating and refining an initial conceptual design, which often is similar to the final one. Although this method serves well in the case of evolutionary design, it is unsuitable for innovation. In fact, without a suitable initial starting point, many rework iterations may be required to correct early inadequate design decisions. In addition, it may be challenging to map the requirements directly onto the design space.

This dissertation aims at developing a methodology to address this problem. The developed framework starts from the hypothesis, and the knowledge to carry out the mapping of requirements onto the input parameters is embedded in the simulation model, and hence no additional rules are required. Instead, a probabilistic surrogate model based on Gaussian processes is used in conjunction with Bayesian statistics to find and eliminate unfeasible areas of the design space. This selection criterion is used in a set-based design approach to explore pockets of the entire continuous design space. Finally, sets with a sufficient likelihood of satisfying the requirements are searched with a local multidisciplinary optimisation algorithm to recover the individual design points.

This process reduced the computational cost of the design space exploration by 80% without sacrificing the number of alternative solutions. Thanks to the large amount of data obtained, it was possible to produce new knowledge on hybrid-electric aircraft design. Specifically, it was found that linear segments are sufficient for defining energy management strategies, and the reduction of NO_x emissions and fuel consumption are associated with climb and cruise, respectively. Furthermore, when studying regional aircraft operating missions, it was found that partial recharge is necessary to maintain the design performance. However, this could reduce the duration of the battery. The battery ageing rate correlates with the EMS's demand for electrical energy. Finally, it was found that the battery's energy density is a determinant of the pack's durability and the feasibility of HE aircraft. The rate of improvement in emissions and fuel consumption is non-linear, suggesting that investing in considerable technological improvements has better returns. Indeed, the required technological level will not be available until the 2040s without an exponential increment of the cell energy density.

Keywords:
Set-based design; Design space exploration; Bayesian probability; Probabilistic surrogate modelling; Hybrid-electric aircraft; Energy Management Strategies; Technological uncertainty;

Acknowledgements

I joined Cranfield University in September 2020, at the beginning of the second Covid-19 wave. Undertaking a PhD programme during a world pandemic has been challenging. The work had to be carried out in isolation for the first two years. Peer support was also minimal, as most of the campus was deserted. Despite everything, I was able to push through and bring the work to a conclusion. However, I was able to reach the end with the support of some people whom I would like to thank here.

First and foremost, I would like to thank my supervisors, Dr Timoleon Kipouros and Prof Panagiotis Laskaridis, for their supervision. Specifically, Dr Kipouros has been helping disseminate the research and advising on overcoming the project hurdles. I am grateful to them for choosing me for this funded PhD position and trusting me to deliver this work.

I am also very thankful to all the doctoral students with whom I shared my time and space during the past three years. I want to thank Dr Mohammed Alieh, who gave me excellent tips in the early days, and Luchien Anderson. I also thank Dr Diego Lopez and Dr Gustavo P. Krupa for their expertise in data science and Bayesian statistics and their genuine friendship. Last but not least, I am very thankful to my office mates Zhengfei He, Nikos Monokrousos, Spyros Tsentis, and Abhishek Kiran, who were kind friends and companions in this adventure and to whom I wish their best in concluding their PhDs.

I also have exceptional thanks to those who supported me outside of academia. My parents, Gianlorenzo and Mariangela, supported my decision to take a PhD, and my family members (Mattia, Anna, Titti, Giorgio, and Aurora) cheered on me to succeed in it. My friends in Italy still kept in touch despite the distance: Edoardo, Davide, Marilena, Stefano, Dario, Marcin. My first-year flatmates, Ambrose, Dario, Ernesta, and Daniel, made the home environment cosy and welcoming during the pandemic. The members of the boardgames society: Charles, Muktim, Georgios, Marcin, Marko, Arnold, Kent, Pol, Angel, and Carola; you kept the society alive after I restarted it. Finally, my last flatmates, Albert, Emily, Noel, Madison, Vidhi, and Marta, were kind to me during the doctorate's last tough months.

Finally, I would like to thank the Cranfield researchers involved in the project FUTPRINT50: Dr Hossein Balaghi-Enalou, Dr Bahareh Zaghari, and Dr Evangelia Pontika. I also thank the other partners and the European Union funding body for sponsoring this PhD research.

Andrea Spinelli
Cranfield, August 2023

Table of Contents

Academic Integrity Declaration i

Abstract v

Acknowledgements vii

Table of Contents ix

List of Figures xiii

List of Tables xvii

List of Abbreviations xix

I Introduction and Literature Review 1

1 Introduction 3
- 1.1 Problem Definition . . . 3
- 1.2 Purpose of Study . . . 4
- 1.3 Research Questions . . . 4
- 1.4 Aims and Objectives . . . 4
- 1.5 Structure of the Dissertation . . . 6

2 Literature Review 9
- 2.1 Introduction . . . 9
- 2.2 Engineering Design . . . 9
  - 2.2.1 Challenges in Systems Engineering Design . . . 9
  - 2.2.2 Principles of Set-Based Design . . . 11
  - 2.2.3 Quantitative Methods in Set-Based Design . . . 14
  - 2.2.4 Uncertainty and Change in Engineering Design . . . 18
- 2.3 Multi-Disciplinary Optimisation . . . 20
  - 2.3.1 MDO in Engineering Design . . . 20
  - 2.3.2 Surrogate Modelling Principles . . . 27
  - 2.3.3 Visualisation Tools for Complex Data . . . 30
- 2.4 Hybrid-Electric Aircraft . . . 34
- 2.5 Summary of Review . . . 36

II The PDOPT Framework **39**

3 PDOPT Theory **41**
3.1 Introduction . . . 41
3.2 Design Space Exploration . . . 41
3.3 Knowledge Extraction . . . 43
3.4 Bayesian Requirements Mapping . . . 44
3.5 Optimisation for Design Exploration . . . 46
3.6 Conclusions: the PDOPT framework . . . 46

4 PDOPT Implementation **51**
4.1 Introduction . . . 51
4.2 Design Space Representation . . . 53
4.3 Step 1: The Exploration Phase . . . 54
4.4 Step 2: The Search Phase . . . 56
4.5 Postprocessing . . . 58
4.6 Problem Definition . . . 58
4.6.1 Definition of Input Parameters . . . 60
4.6.2 Definition of Requirements, Constraints, and Objectives) . . . 62
4.6.3 The evaluation function . . . 63
4.6.4 Structure of the Output . . . 64
4.7 Conclusions . . . 65

III Case Studies **67**

5 Parametric Analysis of HE Propulsion **69**
5.1 Introduction . . . 69
5.2 Problem Definition . . . 69
5.3 Simulation Model . . . 70
5.4 Results and Discussion . . . 73
5.5 Summary of Chapter . . . 78
5.6 Key Learning Outcomes . . . 78

6 Analysis of Energy Management Strategies **81**
6.1 Introduction . . . 81
6.2 Problem Definition . . . 81
6.3 Simulation Model . . . 83
6.3.1 Energy Management Strategy Definition . . . 83
6.3.2 Mission Analysis Methods . . . 83
6.3.3 Aircraft and Propulsion Model . . . 84
6.4 Results and Discussion . . . 87
6.4.1 Effectiveness of Design Space Exploration Methodology . . . 87
6.4.2 Parametrisation Complexity . . . 93
6.4.3 Hybridisation of Mission Segments . . . 97
6.4.4 Effects of Maximum Take-Off Mass . . . 100
6.4.5 Suggested Strategies . . . 101

6.5 Summary of Chapter . . . 107
6.6 Key Learning Outcomes . . . 108

7 Regional Aircraft Operations 109
7.1 Introduction . . . 109
7.2 Problem Definition . . . 109
7.3 System Model . . . 111
7.4 Results and Discussion . . . 114
7.4.1 Design Point . . . 114
7.4.2 One-Day Operations . . . 119
7.4.3 Commuter Flight . . . 120
7.5 Summary of Chapter . . . 120
7.6 Key Learning Outcomes . . . 122

8 Battery Pack Technology 123
8.1 Introduction . . . 123
8.2 Problem Definition . . . 123
8.3 System Model . . . 124
8.3.1 Battery pack ageing model . . . 124
8.3.2 Statistical model for e_{pack} technology forecasting . . . 125
8.4 Results and Discussion . . . 127
8.4.1 Effects of $e_{battery}$ on Energy Management Strategies . . . 128
8.4.2 Effects of r_{degr} on Energy Management Strategies . . . 128
8.4.3 Effects of $e_{battery}$ on r_{degr} and Battery Life . . . 130
8.4.4 Effects of $e_{battery}$ on Aircraft Performance . . . 135
8.4.5 Suggested Energy Management Strategies . . . 136
8.4.6 Probability of achieving the expected technological requirements 139
8.5 Summary of Results . . . 142
8.6 Key Learning Outcomes . . . 143

IV Conclusions 145

9 Discussion and Conclusions 147
9.1 Introduction . . . 147
9.2 Discussion of Results . . . 147
9.2.1 Identification of promising areas of the design space . . . 148
9.2.2 Reduction of computational cost . . . 148
9.2.3 Scenario analysis: robustness to changing requirements and technological uncertainty . . . 149
9.2.4 Energy Management Strategies: Topology and Systems Interaction 150
9.2.5 Interaction between EMS and Battery Pack Ageing . . . 151
9.3 Proposed solutions to Research Questions . . . 151
9.4 Contributions to Knowledge . . . 152
9.5 Impact of Research Findings . . . 153
9.6 Limitations of Study . . . 154
9.7 Future Work . . . 155

9.8 Dissemination of Research 157

References **159**

Appendices **175**

A Example PDOPT Setup **177**
A.1 Definition of the problem into PDOPT components 177
A.2 Definition of the .csv Files 178
A.3 Definition of the Evaluation Function 179
A.4 Definition of the Runfile 179
A.5 Test Case Run 182

B Ethical Approval Letter **187**

List of Figures

2.1 Set-Based Design process, after Bernstein (1). 12
2.2 Comparison between Set-Based Design with traditional Point-Based Design. 13
2.3 21
2.4 Example of a XDSM Diagram of an aerostructural MDO problem (From Jasa et al. (2)). The architecture used is of MDF type, where the optimiser is not tasked to manage the coupling between the disciplines. 25
2.5 IDF version of the MDO problem presented in Fig. 2.4. 26
2.6 SAND version of the MDO problem presented in Fig. 2.4. 27
2.7 Layout of a Deep Neural Network. 30
2.8 A pairwise scatter plot of the iris dataset. 31
2.9 Parallel coordinates plot of the iris dataset. 32
2.10 Selections in a PC plot. 32
2.11 Comparison between Cartesian and PC representation of correlations. . . 33

3.1 Decision Boundary of the constraint. 45
3.2 Probability of the constraint being satisfied. 45
3.3 Framework Structure with ideas highlighted. 47
3.4 Fuzzyness and ambiguity of the decision boundary when the design space is divided into sets. 48

4.1 Computational flowchart of the PDOPT framework. 52
4.2 Detail of the Exploration phase process 55
4.3 Probability of satisfying a constraint $\overline{g_i}$ for a sampled point X_k^j. 56
4.4 Visualisation Tool Example 59
4.5 Content of the input.csv file 61
4.6 Content of the response.csv file 62

5.1 Propulsion system configuration. 71
5.2 Mission analysis method flowchart. 72
5.3 Mission Profile. 73
5.4 Scatter plots of the results of the search phase. 74
5.5 Combinations of DOH values in climb and cruise of the design points. . . 75
5.6 Exploration phase results. 76
5.7 Search phase results. 77
5.8 Unconstrained optimisation results. 79
5.9 Predicted probabilities vs actual range of values in the sets. 80

6.1 Examples of energy management definitions over a single mission phase . 83
6.2 Mission analysis method flowchart. 85
6.3 Mission Profile. 86
6.4 Propulsion system configuration. 86
6.5 Data of Experiment CC-2S. 91
6.6 Data of Experiment CC-2S with additional soft constraints. 92
6.7 Objective space of data from Figures 6.5 and 6.6. 93
6.8 Effects of parameterisation complexity. 94
6.9 Hybridisation strategies of the optimal points of Fig. 6.8a. 95
6.10 Optimal energy management strategies of Figure 6.8b. 96
6.11 Effects of mission hybridisation. 98
6.12 Pareto front of Experiment CC-L, highlighting the effects on the EMS of the trade-off between fuel consumption and NO_x emissions. 99
6.13 Optimal energy management strategies of Figure 6.11b. 101
6.14 Pareto fronts for different Take-off Mass constraints. 102
6.15 Energy management strategies for different take-off mass constraint levels. 103
6.16 Points selected for the recommended strategies. 104
6.17 Parallel coordinates visualisation of Experiment 6. 105
6.18 Parallel coordinates visualisation of Experiment 10. 106
6.19 Visualisation of Experiment 6 energy management strategies 107
6.20 Visualisation of Experiment 10 energy management strategies. 107

7.1 Missions considered for this study. 110
7.2 Definition of the Energy Management Strategy function 110
7.3 Thevenin Electric Model for a Battery Cell. 112
7.4 Battery pack geometric layout. 113
7.5 Parallel coordinates visualisation of Design condition. 115
7.7 Parallel coordinates visualisation of one day regional operations. 117
7.8 Local Pareto fronts of the one day regional operations. 118
7.9 Objective space of the one day regional operations. 119
7.10 Parallel coordinates results of commuter flight operations. 121
7.11 Objective space of the commuter flight condition. 122

8.1 Pareto front of the three objectives with different battery pack energy densities. 127
8.2 Effects of Battery Energy Density on the Energy Management Strategies (PC Visualisation). 129
8.3 Pareto front with selections. 130
8.4 Effects of Battery Energy Density on the Energy Management Strategies (Objective Space Visualisation). 130
8.5 Effects of Degradation on the Energy Management Strategies. 131
8.6 Degradation histories of extreme points for each $e_{battery}$ case. 132
8.7 Degradation histories of extreme points for each $e_{battery}$ case (Continued). 133
8.8 Aircraft performance at representative points in the Pareto front. 136
8.9 Effects of $e_{battery}$ on performance objectives. 137
8.10 EMS of the three scenarios analyzed in Section 8.4.4, compared to the lowest r_{degr} EMS. 138

8.11 Statistical modelling of future cell technology. 140
8.12 Prediction of future battery technology. 141

9.1 Example of a Bayes Net for mapping uncertainties on a design set. 156

List of Tables

4.1 Optimisation `kwarg` parameters. 58

5.1 Parameters used for this test case. 70
5.2 Efficiency values for each component. 70
5.3 Mission Parameters. 73
5.4 Effectiveness of Probabilistic Filtering 73

6.1 Definition of Experiments 82
6.2 Baseline reference values. 82
6.3 Aircraft Properties. 84
6.4 Propulsion system parameters. 86
6.5 Engine specifications at ISA-SLS. 87
6.6 Number of discarded subspaces and saved time. 89
6.7 Breakdown of the total computational time. 90
6.8 Detailed results of the optimal points where Climb and Cruise are hybridised. 95
6.9 Detailed results of the points in Figure 6.11a. 100

7.1 Baseline Data 111
7.2 Parameters of the Hybrid-Electric Aircraft 114

8.1 Input Parameters. 124
8.2 Baseline Quantities. 124
8.3 Details of the selected design points for discussion. 134
8.4 Predicted years of $e_{battery}$ technological availability. 142

A.1 Identified input parameters for the problem under analysis. 177
A.2 Identified response parameters for the problem under analysis. 178

List of Abbreviations

DOD	Depth of Discharge
DOH	Degree of Hybridisation
EMS	Energy Management Strategy
GP	Gaussian Process
HE	Hybrid Electric
LHS	Latin Hypercube Sampling
MBSE	Model-Based Systems Engineering
MDO	Multi-Disciplinary Optimisation
ML	Machine Learning
MOO	Multi-Objective Optimisation
MTOM	Maximum Take-Off Mass
NN	Neural Network
PC	Parallel Coordinates
PDOPT	Probabilistic Design and OPTimisation
QoI	Quantity of Interest
RBF	Radial-Basis Function
RQ	Research Question
SBD	Set-Based Design
SBCE	Set-Based Concurrent Engineering
TLAR	Top Level Aircraft Requirements
TOM	Take-Off Mass

Part I

Introduction and Literature Review

Chapter 1

Introduction

1.1 Problem Definition

The aeroplane, a technological innovation with a history spanning 120 years, has undergone significant evolution in its design process. During its early days, the design of aeroplanes was primarily based on intuition, previous experience, and experimentation. Designers often relied on experimental data and basic analytical tools to evaluate the feasibility of their conceptual designs before testing them in wind tunnels or on airfields.

However, with the advent of computers after the Second World War, detailed numerical models became available, allowing designers to refine their designs with greater precision. Additionally, the Cold War and the boom of airline transport introduced increasingly demanding top-level requirements (TLAR) for military and civil aeroplanes, increasing the complexity of their design. Civil aviation authorities, such as the FAA and JAA/EASA, played a crucial role in improving air transport safety, but at the cost of more stringent requirements.

Nowadays, developing a commercial aircraft from a clean sheet design is a monumental task, requiring substantial funds and human resources investments. Improving previous designs regarding operating costs, fuel consumption, and emission reduction requires an increasingly high level of refinement and optimisation in the design process. Furthermore, the introduction of new technologies is a long process, slowed by the necessity to prove their safety to the aviation authorities. To manage the complexity of these systems and avoid endless development cycles, manufacturers rely on evolving the previous generation design and outsourcing subsystem and component development to specialised companies.

In addition to these challenges, climate change urges the aeronautical industry to develop new aircraft with significantly less environmental impact, often with a target of zero emissions by 2050. Disruptive designs incorporating new technologies, such as electrification and hydrogen propulsion, are being proposed and tested. However, the traditional evolutionary approach used in the industry is inadequate for these innovative technologies. The lack of previous design knowledge and ”learnt lessons” as a starting point would likely produce conceptual designs that are not feasible or desirable and require substantial rework. This problem is caused by many interactions between the components that, without experience, often cannot be fully accounted for in their entirety. Additionally, knowledge uncertainty when searching for the design point that would meet the require-

ments should be accounted for and managed, especially if such requirements are subject to change.

In summary, a methodology capable of thoroughly exploring the design space for suitable candidate solutions of a set of requirements without explicit configuration selection rules originating from expert knowledge is needed. In fact, in the case of innovative configurations, performing configuration selection based on intuition might incur redesign iterations at later stages of development. Hence, the methodology must not use external knowledge other than what is implicitly encoded in the design model when exploring the design space. Finally, the process should be performed in a computationally acceptable time without sacrificing the use of a detailed design model.

1.2 Purpose of Study

The research presented in this dissertation attempts to develop a methodology capable of performing space design exploration where limited knowledge or experience is available. The difficulty lies in extracting information from the design model to explore the design space.

The implemented methodology is applied to understand the behaviour of hybrid-electric propulsion systems. In particular, this study focuses on optimal energy management strategies and their interaction with aircraft and batteries.

1.3 Research Questions

In light of the overall research scope, the following research questions have been identified:

RQ1 How to manage the uncertainty that arises from the decision boundary in the design space?

RQ2 How to extract the knowledge embedded in the design model to evaluate the design space?

RQ3 How to evaluate all possible alternatives in reasonable computational time?

RQ4 How can the information obtained from exploring the design space aid in understanding the system under design?

In addition to these research questions, the test cases seek to explore the problem of the optimal energy management strategy for hybrid-electric propulsion. This is related to Research Question 4, as the information gathered was included in the deliverables of Project FUTPRINT50 (3), the sponsor of this PhD.

1.4 Aims and Objectives

The content of this dissertation attempts to answer the outlined research questions by investigating the field of design space exploration and optimisation, focusing on the principles of Set-Based Design (SBD). This methodology is proposed as an alternative to

the traditional iterative approach, which commits to refining a single design solution. In contrast, SBD evaluates multiple alternatives and focuses on eliminating the unfeasible solutions. This approach avoids committing to a single design point early, reducing the risk of redesign iterations in the late stages of the development cycle. Therefore, SBD seems ideal in design problems where prior knowledge is limited.

However, the critical component of SBD is the selection criteria, which determine the decision boundary for every requirement. Often these are handcrafted based on the product under design or rely on the designer's expertise to select the correct mapping between input parameters and quantities of interest. This limitation is prominent in developing systems where no previous experience is available. This issue is reflected in RQ1, the uncertainty due to the lack of knowledge of the decision boundary. This dissertation attempts to answer this question by exploiting the Bayesian interpretation of probability and modelling this as an epistemic uncertainty.

It is also argued that, unlike the previous development of SBD methodologies (4, 5), it is unnecessary to introduce additional rules for set selection when the design model already captures the knowledge of the system's behaviour. Indeed, assuming the modelling is reasonably accurate, the relationship between input parameters and quantities of interest is already present. Extracting this information and using it in the elimination process would make SBD capable of managing problems without the need for assumptions or biases from the designer. RQ2 address this issue. Furthermore, SBD is significantly more computationally expensive than the traditional approach of developing a single design. RQ3 seeks to address this issue.

The proposed methodology attempts to answer all these questions by employing a probabilistic surrogate model trained with data extracted from the design model. Quantitative engineering requirements are expressed as inequalities over a quantity of interest in the model under design. These are converted into probabilistic statements, e.g. "How likely is the requirement to be satisfied?". The advantage of this approach is the ability to combine the different requirements using probabilistic logic. Using a surrogate model also reduces the computational cost of evaluating all the alternatives in the design space. The only overhead is generating the data to train the model, which could be significant if the number of configurations is low and the evaluation cost of the design model is high.

Configurations in the context of SBD are not a single design point but a range of input parameters. Within this range, engineers usually look for the design with the best performance or the lowest cost. Hence, to understand the behaviour of the system under design (RQ4) and recover the individual points, an optimisation problem must be introduced. However, optimisation is used for extracting as much information from each pocket of the design space rather than identifying a restricted set of optimal solutions. Local Pareto fronts are obtained from each set and aggregated into a global database of feasible and desirable solutions.

This final output contains the global Pareto front and suboptimal designs: both enable the designers to understand the system's response under design to the provided requirements. Additionally, the suboptimal points allow us to analyse scenarios with more restrictive requirements without recalculation.

The demonstration of the methodology involves multiple test cases from the field of hybrid-electric aircraft design. The goal is to test the limits of the proposed framework by gradually increasing the complexity and computational cost of the design model. This

family of problems was chosen as hybrid-electric propulsion is still a developing field within aeronautical engineering research; therefore, it features a design space lacking prior experience. These case studies focused on understanding the repercussions of the energy management strategies over the performance, emissions, and battery life of a parallel hybrid-electric regional aircraft. The framework is implemented in Python and employs robust scientific computing libraries for easy portability and maintenance. While this dissertation focuses on hybrid-electric aircraft, the code is designed as a problem-agnostic library that can be adapted to any industrial design problem.

1.5 Structure of the Dissertation

This dissertation is structured in four parts: Introduction and Literature Review, The PDOPT Framework, Case Studies and Conclusions. The following paragraphs summarise the contents of each of them.

Part I - Introduction and Literature Review. The first two chapters introduce the topic and context surrounding the research project presented in this dissertation. Research questions are described along with the objectives and objectives to answer them. Chapter 2 reviews the areas that this dissertation investigates: engineering design methods, the use of probability in engineering, multi-disciplinary optimisation and hybrid-electric aircraft design. The goal is to familiarise the reader with the fundamentals and identify the research gaps.

The first sections focus on Set-Based Design and its quantitative implementations. The review attempts to identify uses of SBD to manage epistemic uncertainty, specifically from requirements. A survey of the use of probability theory in engineering design is also provided, contrasting how the most common application is in propagating aleatoric uncertainty originating from component tolerances and operational variability. The following section presents a short review of engineering optimisation, presenting the taxonomy of the different algorithms, and discussing the use of surrogate modelling for speeding up optimisation analysis. The remaining sections of this chapter cover the state-of-the-art in hybrid-electric aircraft. The design challenges present in this research field provide excellent test cases for the methodology developed in this dissertation.

Part II - The PDOPT Framework. This part concerns the description of the methodology developed. Chapter 3 discusses the ideas which make up the backbone of the methodology. These address research questions and literature gaps from a theoretical standpoint. Focus is given to explain the use of Bayesian statistics and surrogate machine learning for extracting knowledge from the design model and using it to map requirements to the design space.

Chapter 4 presents the implementation of the methodology as a Python library, describing the libraries and computational process. Details for setting up an analysis are also presented, with an example setup included in the Appendix.

Part III - Case Studies. This part discusses the test cases developed to test and demonstrate the methodology. These concern the design of hybrid-electric propulsion systems, focusing on the interaction between the energy management strategy, the system performance, and the behaviour of the components. Each chapter is a fully-contained application of PDOPT featuring a description of the problem, the design model adopted, and a discussion of the results.

Chapter 5 tests on a simple parametric model the ability of the methodology to eliminate the unfeasible sets. The problem is finding the degree of hybridisation (DOH), battery energy density and motor power density that minimise the fuel consumption of a parallel hybrid-electric turboprop. This form of analysis is extended in Chapter 6 by introducing a parametric definition of energy management strategies (EMS) and NO_x emission modelling into the system. PDOPT is used in 12 computational experiments for exploring different EMS parametrisations and mission arrangements. Results provide recommendations on EMS definition and the relationship between different mission segments and the objectives (fuel consumption, NO_x emissions).

Chapter 7 investigates the off-design performance of the propulsion system sized with the design mission of Chapter 6. The battery pack model is enhanced with an equivalent circuit model of the cells, capturing the discharge dynamics. Results indicated a trade-off between extending operations and battery life. Similarly, Chapter 8 sought to analyse the impact of battery ageing on the system's performance. Ageing modelling was introduced on the parameters of the equivalent circuit model along with a new optimisation objective to minimise the increment of fuel consumption caused by the reduced capacity of the cells. In addition, a discussion of technological uncertainty is presented by analysing different energy density values and estimating the probability of availability in the next 20 years.

Part IV: Conclusions. The final part summarises the results of the scientific work presented in this dissertation. The research questions are reviewed alongside the discoveries and results obtained. The advantages and drawbacks of the developed methodology are presented. The final section discusses potential enhancements of the PDOPT framework. In particular, integration of a visual analytics tool to explore the produced data and introduce methods such as causal discovery to explain the behaviour of the system.

Chapter 2

Review of Literature and Related Research

2.1 Introduction

This chapter will introduce the reader to the fundamentals and state of the art of the three main topics touched on by this dissertation: engineering design, multi-disciplinary optimisation and hybrid-electric propulsion.

Section 2.2 discusses the challenges in designing complex systems, reviewing the problem of rework and design iterations. Set-Based Design is introduced as a methodology capable of handling unexpected rework and uncertainty in the design phase. A discussion of the role of uncertainty in design is also presented, along with its interaction with rework and engineering changes.

Section 2.3 surveys the fundamentals of Multi-objective and Multi-Disciplinary optimisation. These methods are increasingly adopted to make effective design decisions that satisfy the proposed requirements. By capturing the behaviour of the components and their interaction, it is possible to generate an optimal design in earlier design stages. Surrogate modelling is used for this purpose, as it can capture higher fidelity behaviour at a lower computational cost. Finally, a brief discussion of visualisation tools for complex data is provided.

Section 2.4 covers the current state of the art in hybrid-electric propulsion. These include the current technological gap and the need for new integrated methodologies to explore the design space of these new concepts.

2.2 Engineering Design

2.2.1 Challenges in Systems Engineering Design

Engineering design is a problem-solving discipline that maps a set of requirements or needs to a set of functions that are transformed into a set of decisions leading to the description of a solution (6).

The typical process most commonly taught in engineering schools follows a sequential model consisting of identifying and refining a design point through iterations. Theoretical descriptions of this philosophy are the Design Spiral by Evans (7), the Waterfall Model

by Royce (8) and the V-Model of Forsberg and Mooz (9). Generally, the design process is broken down into three different levels of detail: conceptual design, preliminary design, and detail design (10). In each of them, the requirements and specifications of the previous step are analysed and applied through design synthesis to its level of detail. For instance, conceptual design defines the system functionality and prescribes general behaviour, while detail design produces drawings and tooling for the actual components to be built and assembled. As development progresses, the degrees of freedom of the problem reduce, increasing the difficulty of introducing changes into the design if problems are discovered. Indeed, most of the cost and resources for a project are committed in the conceptual design process, about 65% in the case of aerospace programs (11). This phase introduces a considerable risk, where critical configuration decisions must be made with little or inaccurate design information. Expertise and heuristics are essential in selecting an initial configuration close to a satisficing solution, usually due to the evolution of a previous design. In fact, aerospace manufacturers often rely on design evolution rather than total innovation (12).

Iterations in the design process are therefore needed to reduce the uncertainty of knowledge. However, not all types of iteration procedures are beneficial. Costa (13) identifies three categories of iterative procedures: design iteration, behavioural iteration, and rework iteration.

The first is repeating the design activity but at a different level of detail: developing a product from a conceptual stage down to the preliminary and detailed stages. The second is the process of repeating the design activity at the same level of abstraction but in a different scope. In other words, the team moves their attention to a different subproblem. For instance, one typically progresses from analysing weight to analysing stability and control when designing an aircraft. Both types of iterations are physiological to the design process and are expected to increase the knowledge level as the project progresses.

However, the third type is defined as reiterating the design activity at the same level of abstraction and on the same scope to correct an identified mistake. This form of iteration should be avoided as it wastes resources (time, funding, personnel) and halts product development.

Wynn and Eckert (14), in their extensive taxonomy of design iterations, categorise rework as a form of corrective iteration. Their survey concludes that corrective iterations should be avoided at the source. It is also argued that a highly iterative project entails more uncertainty and is less efficient. More ad hoc procedures and supervision are required, increasing the risk of producing mistakes or overlooking important issues – often requiring more iterations to fix the mistakes. Hence, iterations have a self-reinforcing effect that can exacerbate costs and delays (15).

Indeed, engineering rework is still a significant contributor to the cost and schedule of aerospace industries (16). Furthermore, Reichelt (17) and Chucholowski (18) estimated that between 30 to 70% of the design hours have been spent on rework, with Kennedy et al. (19) reporting up to 80%.

Minimising undesired design iterations is paramount for engineering projects, especially in designing complex systems. Several approaches have been proposed to manage iterations in the product design process. One family involves identifying the chain of dependency with methods such as Eppinger's (20) Design Structure Matrix, Lindemann's (21) Multi-Domain Matrix and Rotzer et al.'s (22) Attribute Dependency Graphs.

Awareness of the causal dependency between the system properties and behaviours should guide the designer to avoid circular dependencies during the design phase. Nevertheless, these approaches still require a good starting candidate design to converge towards a satisfactory solution.

The alternative is to avoid picking a candidate solution from the start to optimise but instead to evaluate a family of design points in parallel and converging towards the desired solution. Methods based on this principle are named ”convergent” or ”set-based”, in contrast with the classical ”point-based” methods, which focus on refining a design point (23). Examples include Pugh's (24) Controlled Convergence Method (PuCC), Wheelwright and Clark's (25) Design-Build-Test cycle (DBT), Augustine's (26) Fuzzy Concept Convergence process (FCC), Zimmermann's (27) Solution Space Engineering (SSE) and Ward's (28) Set-Based Design (SBD). While all these approaches share the similarity of parallel exploration of the design space, SBD focuses on eliminating the unfeasible and undesirable solutions rather than selecting the most promising solution from the parallel evaluations. As will be argued in the next section, this feature is key in the ability of SBD to manage design uncertainty and cope with mistakes found mid-way during the product development process.

2.2.2 Principles of Set-Based Design

The first reference to a Set-Based Design approach can be traced to Allen Ward's dissertation titled ”A Theory of Quantitative Inference for Artifact Sets, Applied to a Mechanical Design Compiler” (29). Ward demonstrated a computer program solving a configuration problem by reasoning about quantitative intervals of components and operating conditions, eliminating those sets that do not meet the performance requirement under the given constraints (28). He referred to the approach as ”set-based”, as the process output is a set of satisfactory solutions rather than an optimal point within the design space. Ward later coined the term ”Set-Based Design” to describe this methodology (30). SBD is also known, especially in the project management field, as ”Set-Based Concurrent Engineering” (SBCE). This name is mainly associated with the Toyota Production System, as this manufacturer was recognised to have developed SBD principles independently (31). Despite similar names and origins, SBD is more focused on quantitative inference methods for design selection, while SBCE is a managerial practice concerned with communication between teams that develop subsystems for a project (32). Both, however, are characterised by three core principles:

1. Reasoning on sets or families of designs rather than design points;

2. Focusing on eliminating the unfeasible or undesirable solutions rather than finding the optimal design;

3. Delay the commitment to a design solution as much as possible.

The general SBD process that implements these principles is as follows. The first step is to map the design space by finding the feasible regions that satisfy each requirement. These sets are integrated by intersecting the feasible sets and identifying the solution space for all requirements. Several point concepts from this candidate solution space

are developed and evaluated in parallel, focusing on eliminating undesirable and unfeasible solutions. Convergence is achieved when the solution space has been sufficiently narrowed so that little uncertainty remains in the design. Feasibility is, therefore, established before the design team can commit to a set of solutions. Figure 2.1 exemplifies this procedure.

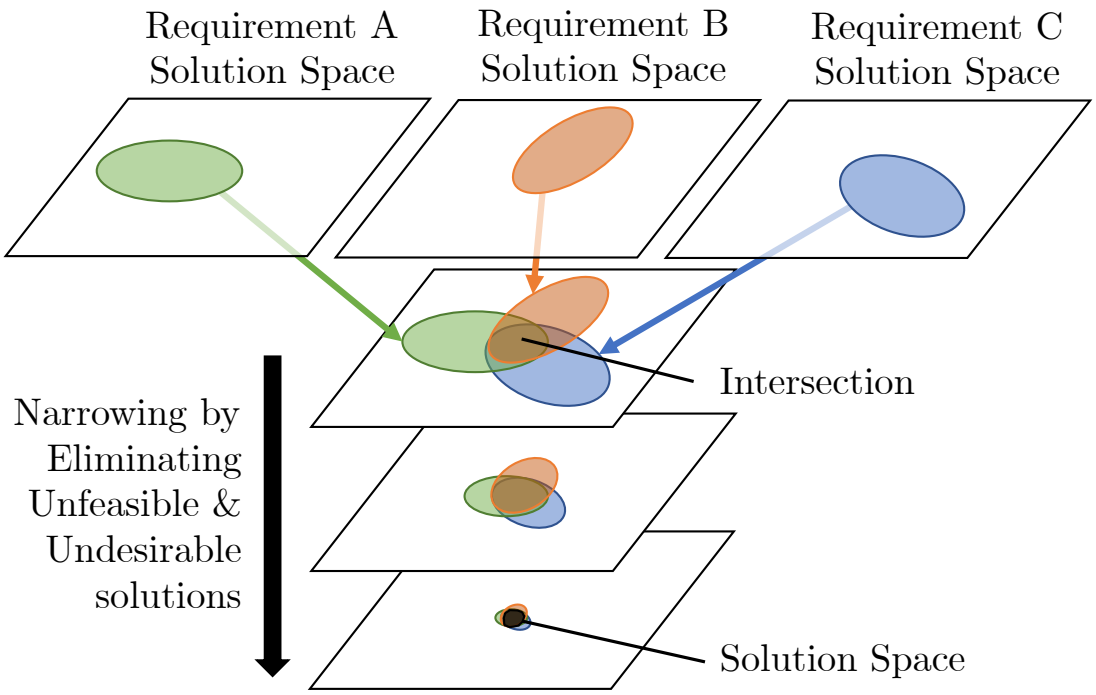


Figure 2.1: Set-Based Design process, after Bernstein (1).

These principles attempt to solve the issue of converging early on a single design point. By reasoning with large families of design points, SBD increases the likelihood of identifying a set of feasible and Pareto-optimal designs (33). Terwiesch (34) underlined how parallel development of concepts prevents the need to commit to every design decision when all information is unavailable. Alternatives are kept open until the last moment. The advantage, as pointed out by Bogus (35), is to reduce the sensitivity of downstream tasks (e.g. detail design) from upstream decisions (e.g. conceptual design).

Shallcross et al. (33) ascribe the ability to resolve epistemic uncertainty through a structured, sequential decision-making process to SBD. As shown in Fig. 2.2, critical learning points affect the success of the final design. Delaying commitment avoids premature design decisions, which may result in a selection of inappropriate designs requiring extensive and costly rework. SBD can reduce the risk of failure thanks to deferring commitment and allowing time to analyse and resolve the identified knowledge gaps in the design phase.

Costa and Sobek (13) argue that SBD is robust in preventing rework also because of its focus on eliminating unfeasibility and undesirability early on. Furthermore, McKenney (36) demonstrated how SBD is robust to change in requirements. The author identified this capability in sets, which were flexible enough to accommodate the changing requirement halfway through the design process. Similarly, Parsons (37) attributes the

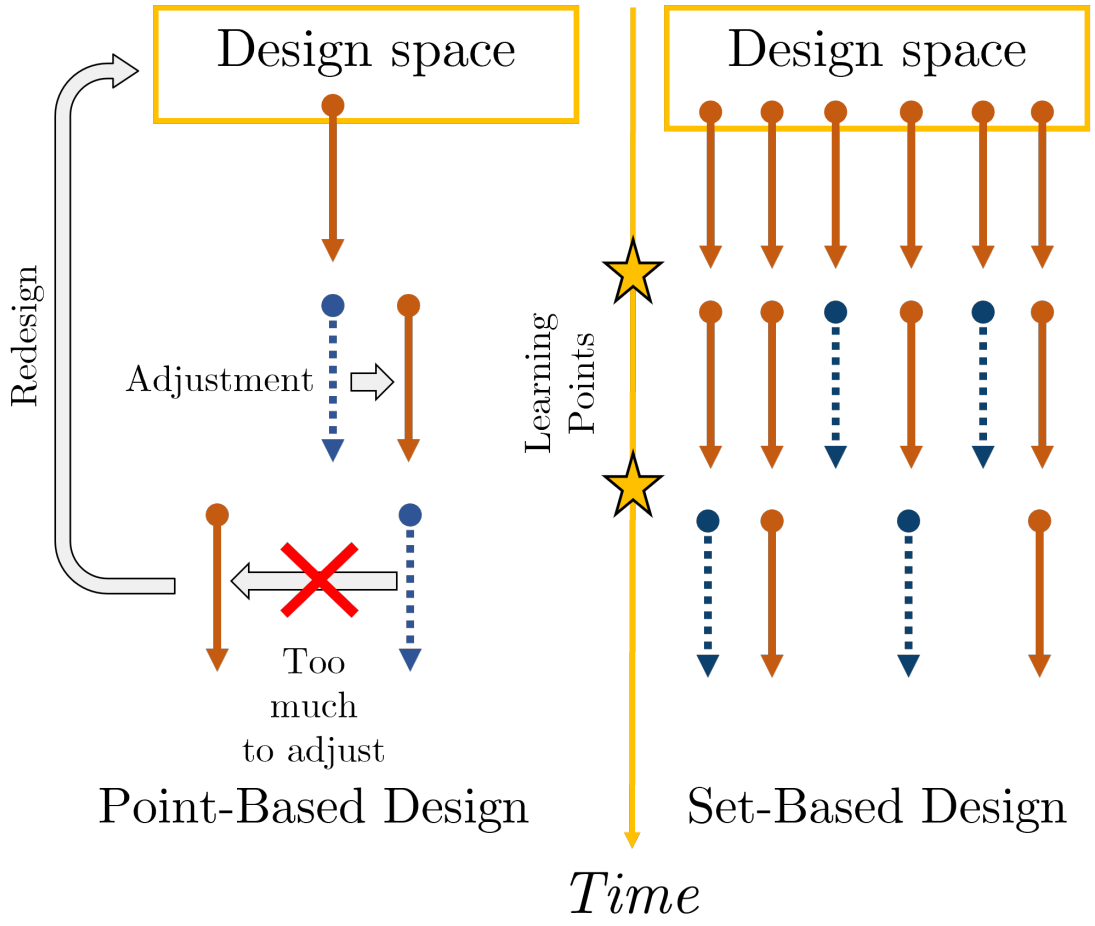


Figure 2.2: Comparison between Set-Based Design with traditional Point-Based Design.

robustness against design errors to using sets. In general, by delaying critical design decisions and maintaining a range of alternatives, more time is provided for the designer to understand the requirements and specifications fully.

From the project management perspective, Morgan (38) and Ward (39) claim that SBD, integrated with Lean production practises, increases productivity by up to four times. Khan (40, 41) argues that SBD promotes innovative thinking, creativity, and organisational knowledge. The concurrent development of systems and components is enabled by the ”ambiguity” that a set of designs has, which is communicated through ranges of input values and design models and concepts (42, 43). This benefits modern-day product development cycles, where team members are geographically dispersed (44). Specking et al. (45) demonstrated near real-time design evaluation by integrating Model-Based Systems Engineering (MBSE) data-sharing practices with SBD.

The publication of the principles of the Toyota Production System (38), a refinement of SBCE, spurred universities, institutions, and industries to investigate SBD methodologies. Examples of the adoption of SBD include the US Navy (46, 47, 43), Rolls-Royce (48, 49) and the Swedish industry (50). In addition, the EU sponsored a SBCE research programme named ”LeanPPD”, involving several academic and industrial partners from the United Kingdom, Spain, Germany, Italy and Poland (40). Research in SBD

is active in academia with several universities contributing, such as MIT, Cranfield University, University of Michigan, Chalmers University of Technology, and the University of Texas at Austin (51).

Despite the active interest in SBD, the main focus is on management principles rather than quantitative inference and selection methods. This is expected as contemporary engineering involves with multiple companies, whose coordination is often difficult in projects with long lead times (52). Qualitative approaches are preferred as can establish a starting point in design iterations before quantitative models have been defined.

Nonetheless, both Shallcross et al. (33) and Dullen et al. (51) highlight this quantitative gap in the literature. The following section will discuss this aspect in detail, surveying the state-of-the-art on quantitative approaches in design inference in SBD.

2.2.3 Quantitative Methods in Set-Based Design

When implemented, three aspects of SBD require proper quantitative methods: the procedure for mapping requirements in the design space, the integration by intersection method, and the decision-making criteria for narrowing the sets towards the desired solution.

In most cases, the same methodology is used for all three procedures, but some exceptions are present in the literature. The following approaches have been identified:

- Constraint Satisfaction Problem (CSP)
- Fuzzy Logic
- Classification Methods
- Multi-Objective Optimisation
- Utility Theory
- Markov-Decision Processes

Constraint Satisfaction Problems (CSPs) were the original methodology used for SBD. CSPs consist of a set of variables $X = X_1,..,X_n$, each associated with a domain $D = D_1,...,D_n$ of possible values that can be assigned to them. Constraints C, defined as pairs of variables and statements such as $< (X_1,X_2), X_1 \neq X_2 >$, specify the allowed combination of values. A solution is found when every variable is assigned, and all constraints are satisfied (53). Efficient and robust solvers for this type of problem have been developed in artificial intelligence (53) and operations research (54). Ward's original SBD dissertation (29, 28) used a CSP solver to perform set reduction and identify the bounds for design variables. The original implementation converted a catalogue selection problem into a CSP by "compiling" the specification language (i.e. "Motor RPM must be in the range of 1600–1800") into a set of predicates which then interacted with the mathematical modelling equations. The goal was to identify the smallest set by process of elimination, which could satisfy the model equation and the specification. Finch (55, 30) formalised the set calculus of Ward and provided a concise mathematical algorithm for identifying the set boundaries for continuous ranges of variables when subjected to algebraic equations of the type $g(\boldsymbol{x}) = 0$. With the addition of a causality table, the designer can separate

the dependent variables from the independent ones, which can be specified with a range of possible values. With this analysis, the designer can account for uncertainty in the design and identify a satisfactory solution space. Seshan (56) extended the work of Finch by allowing systems of equations to be converted into predicate constraints. Qureshi (57, 58) expanded the approach by separating uncertain parameters from controllable design ones. Solutions satisfying the quantitative relations in the controllable parameters had to be valid for any interval of the uncertain parameters. Hence, the process could eliminate the non-robust solutions of the design requirements.

Fuzzy Logic (FL) is a system of rules and procedures developed by Zadeh (59, 60) to capture the approximate reasoning in the human language. It uses "fuzzy sets", an extension of classical set theory, which models ambiguity. Although objects in classical "crisp" sets can have a membership of 0 or 1, fuzzy sets utilise membership functions $h(x)$, which map the input quantity to a value between 0 and 1. Once the fuzzy sets have been defined, logical operations can be performed. Inferences are carried out the IF-ELSE rules on linguistic operators. Finally, a defuzzifier operator converts the output membership function into a crisp output quantity. Fuzzy logic systems have found extensive application in control systems (61) and are used as an alternative to probability to model belief uncertainty. Nathm and Ishikawa (62, 63) integrated FL into SBD by constructing membership functions to represent the preference for a design decision made by an expert. The interval calculus of Finch and Ward (30) is used to propagate the preferences of the input values on the design constraints and thus restrict them to a range of feasible fuzzy solutions. The authors integrated this tool into a 3D CAD programme for mechanical design (63). Furthermore, they argued that the preference number allowed homogeneous concurrent communication between teams while accounting for design uncertainty (62). Wang and Terpenny (64) used FL to represent imprecise performance constraints in a multi-agent set-based optimisation programme. The method consisted of individual agents generating subsystem designs from the set-based pool of solutions (obtained after applying the FL relationships) and a system-level agent combining the individual solutions through a genetic algorithm procedure and evaluating them. The best-performing solution sets would be retained. Then the preference fuzzy sets would be adapted through the interactivity of the designer, and the cycle repeats until convergence. Other authors such as Singer (65) and McKenney (66) implemented similar multi-agent SBD tools to elicit design preference in the context of ship design and manage uncertainty from late-stage changes.

Classification Methods (CM) are Machine Learning (ML) algorithms that map vectors to a category point after training on a dataset of vector-label associations. In the context of SBD, CM are used to classify the design space for the requirements or preferences to identify the areas to eliminate. Commonly used algorithms include Bayesian network classifiers (BNCs) (67), Support Vector Machines (SVMs) (68), Random Forest Classifiers (69) and Neural Networks (70). BNCs are the most frequently used in SBD applications, as they inherently allow to map the design uncertainty. Sashan and Seepersad (71) used BNCs for SBD to design a spring and a UAV wing. The classification focused on separating the feasible from the unfeasible design space, with the ability to draw arbitrarily shaped boundaries. Matthews et al. (72) developed a SBD approach using BNCs for multilevel design problems, such as the propagation of material properties from the micro to the macro level. Morris et al. (73) extended the work of Seepersad by incorporating a classification based on robustness alongside the previous feasibility

classification.

Multi-Objective Optimisation (MOO) problems are composed of more than one objective function $f(x)$ to maximise/minimise and several constraint functions $g(x)$, often expressed as inequalities. Input variables x are often bounded, restricting the domain of decision variables to be searched. Objectives often compete; hence, no solution point can dominate every objective's other points. Instead, the goal is to identify the trade-off between the objectives. The Pareto front is the set of design points which identify the best available trade-off, as these points are not dominated by any other in the solution objective space. Common approaches to solve MOO problems include the weighted-sum method, epsilon-constraint method, normal-boundary intersection method, and evolutionary algorithms (74). More detailed details on algorithms for solving MOOs are presented in Sec. 2.3. Most optimisation methods are point-based; hence current research is active in extending it to set-based optimisation. Hannapel et al. (75) implemented an MDO approach to SBD, using a two-step method: a system-wide optimisation problem tasked with reducing the intervals describing the sets and local discipline-specific optimisation problems which returned to the global optimiser the optimal parameter in each interval. Moshaiov et al. (76) developed a set-based concept optimisation method to explore the design space and identify several local Pareto fronts associated with design sets. A global s-Pareto front (77) of the concepts is found through a relaxation zone around the globally non-dominated points and evolving them. Farhi (78) improved this method by replacing the relaxation zone with a Window of Interest (WoI), an area of the objective space of interest for the designer. The algorithm was modified to promote and evolve the Pareto concepts towards the WoI. Gong et al. (79) extended the NSGA-II Genetic Algorithm for evolving Set-Based concepts defined as parameter intervals. Integration of MOO methods with machine learning was carried out by Han et al. (80). The authors modified a population-based optimisation algorithm by first evaluating the feasibility of each design point and then, by using k-means clustering, evolving the solutions towards the identified target point. Georgiades et al. (5) developed a framework which uses MOO to refine the elimination process of SBD further and produce a robust set of design alternatives.

Utility Theory is a decision-making process model commonly used in economics (81). It measures the preference of an outcome x_i given a decision y_i with a function $V(x)$, called Value Function or Preference Function. If the outcome is affected by randomness, the Expected Utility of a decision is the weighted sum of the value of all the outcomes by their probability. The suggested decision is the one that maximises the expected utility. Multi-Attribute Utility Theory (MAUT) (82) extends utility theory with multiple preference criteria, including competing ones. The global value function $V(x)$ is defined as a weighted sum of the individual preference criteria functions $v_i(x)$. In the context of SBD, MAUT has been used as a mechanism to model the designer's preference in a component selection process and eliminate sets with low utility. Malak et al. (83) first implemented MAUT to represent a designer's preference in a configuration selection process, driving the narrowing of the sets based on interval calculus (30). Specking et al. (45) presented a SBD framework with integrated Model-Based Systems Engineering (MBSE) principles, where the decision-making process is driven by a framework which captures the influence of design decisions over the affordability and performance of an unmanned aerial vehicle. Details of this demonstration were presented by Small in his thesis (42, 84), focusing on the analysis of alternatives (AoA) using SBD. Fitzgerald and Ross (85) demonstrated

using artificial intelligence to identify set drivers, using a clustering algorithm to group designs based on similar features. The results allowed the identification of design sets based on value differentiation and the parameters most affected value.

Markov-Decision Processes (MDPs) are a mathematical model of decision processes where outcomes are partially random and partially under the control of the decision maker (86). MDPs are composed of states S, actions the decision maker takes to transition from one state to another A_s and a state transition matrix P, quantifying the probability of transition from one state to another. Each action yields a reward $R_{s,a}$ to the agent, a positive or negative number, with a discount factor $\gamma \in [0, 1]$ to reduce the reward of actions far into the future. The solution of a MDP is a sequence of actions known as policy π. The optimal policy is the one that maximises the return, that is, the sum of rewards. If P and R of the MDP are known, Dynamic Programming (DP) (87) is used to find the optimal π. Otherwise, Reinforcement Learning (RL) (88) algorithms are used. The use of MDPs for SBD in literature is limited, and most of the work focuses on modelling the transition between different concepts or fidelity levels. Miller et al. (89) developed a MDP model of the engineering design process as a Sequential Decision Process (SDP), describing costs as the computational time required from a certain level of analysis and actions as the increment of fidelity. Boundary models, representing the upper and lower limits of the real output, are used in conjunction with SBD for set elimination. The authors present an optimal path of set reduction and increment of fidelity after solving the MDP with dynamic programming. Rapp et al. (90) also used an MDP to model the SBD process, focusing on the cost of changing design mid-development and adding or removing options from the design space. The authors compared SBD with traditional point-based design, with a reward based on contribution to design (CTD). Chhabra and Warn (91) extended the work of Miller by solving the SDP problem with Reinforcement Learning. This approach avoided defining a mathematical model of the computational cost and set-elimination ability associated with a simulation fidelity level. This study reflects a more realistic approach of the design process, where agents usually do not have a full picture of what level of fidelity of the models is required.

Currently, there is no prevalent quantitative approach for SBD. While embracing the qualitative principles of SBD, most methods often focus on one specific aspect in their quantitative implementation. For instance, MAUT and MDPs emphasise the delay in decision-making, FL on concurrent engineering, while CSPs and CMs focus on mapping the requirements on the sets. The most common representation of design sets is input parameter intervals, originating from the interval calculus of Ward and Finch (30), mainly used in CSPs and MAUT approaches. Some methods use intervals of the quantities of interest, such as the boundary models of Miller et al. (89) or the set-based Pareto fronts of Gong et al. (79). Finally, CM and MOO methods rely on sets represented as groups of design points (such as the concept Paretos of Moshaiov et al. (76)). Some methods eliminate sets by directly eliciting the designer's preference using fuzzy logic or value functions. While intuitive and easy to implement, it is arbitrary. The alternative is using constraints on the quantities of interest of the design model, requiring, however, indicator variables in the case of vaguely defined requirements (such as affordability, availability or environmental impact). Some approaches (57, 73, 45) also introduce robustness as a set elimination criterion. Most authors cite design uncertainty as the main motivation to develop their framework (28, 57, 64, 79, 45, 73), using different quantification methods.

Intervals are the most commonly used approach, followed by fuzzy logic and Bayesian probability. The following section will discuss in detail the role of uncertainty in design, its representation, and its repercussions in the design process.

2.2.4 Uncertainty and Change in Engineering Design

Uncertainty is defined as the condition where not all information about a process or object is available. Agents must make decisions with the best available evidence and maximise their success rate. Indeed, most real-world situations involve taking action without certainty of the outcome. The causes are twofold and associated with two general forms of uncertainty: aleatory and epistemic (92, 93).

Aleatory uncertainty is caused by events that are unpredictable and random. Engineers are typically well-versed in this form of uncertainty as it affects measurements and reliability. In this context, frequentist probability theory is employed. In the context of product design, aleatoric uncertainty is present when designing against component failure or extreme events, such as strong wind and heavy snow. As this uncertainty cannot be reduced, engineers overdesign products or trade-off performance with reliability.

However, epistemic uncertainty originates from the lack of information on the state of affairs. This form of uncertainty is usually labelled as ”unknowns”, and engineers are not often aware of it. However, the design process, especially at the conceptual stage, is characterised primarily by it. Examples include imprecise requirements, technological level, operating conditions, and modelling assumptions. Epistemic uncertainty can be reduced by acquiring more information. In practice, it can be unfeasible due to time constraints, difficulty in team communication, and the bounded rationality of human beings. Han et al. (94) further divide the epistemic uncertainty into data and description uncertainty. The former refers to the inability to know some essential information due to insufficient data. An example is the level of fidelity of the model used in the design process. The latter, instead, is related to the vagueness of the information and the subjectivity of the description. Vagueness typically affects requirements, especially if they relate to qualitative needs such as flexibility in operating conditions or environmental impact. Unlike its aleatory counterpart, epistemic uncertainty is managed differently according to its specific source. The main difficulty is identifying the source of uncertainty, as designers might be unaware of the lack of knowledge (usually defined as unknown unknowns (95)). Wynn et al. (93) identify five forms of epistemic uncertainty typical in the design process: imprecision, inaccuracy, inconsistency, indecision, and instability.

Inaccuracy and imprecision are uncertainties caused by a lack of information of true value. Specifically, inaccuracy is caused by the discrepancy due to the analysis methods' fidelity level. This is often managed through experimentally validated methods, which quantify the expected error in the early stages of design. Multifidelity approaches (96) has also been proposed to reduce the gap between the low computational cost of low-fidelity methods and the high accuracy of high-fidelity analysis. However, imprecision comes from the lack of detail required to choose a precise quantity. For example, in wing design, designers leave the geometry open until loads are fully specified. Often there is an interdependency between quantities, which requires the propagation of ranges between the different disciplines. Eckert et al. (97) propose using margins in collaborative design as a mechanism for managing changes between iterations. Similarly, Antonsson

and Otto (98) presented the Method of Imprecision, which leverages fuzzy sets to reason and propagate uncertainty through the design process. Wang et al. (64) and McKennedy et al. (66) adopted fuzzy sets into an SBD framework to manage design imprecision and communicate it between engineers.

The other three forms of epistemic uncertainty, inconsistency, indecision, and instability arise from the design methodology. Inconsistency arises when duplicate descriptions of the same parameters are present, creating conflicts. Indecision is caused when the designer lacks information to select a satisfactory design point. Instability is the likelihood that the current configuration requires changes and rework. All three are associated with iterative rework (14) and changes propagating through the design (99).

Specifically, rework, engineering changes, and design instability are interconnected and may reinforce each other (14). Engineering changes are internal, such as the discovery of design mistakes, or external, like updates in the specifications or technological level (97). These have a knock-on effect on the amount of rework to be carried out, especially for complex systems (100). Maier and Langer (101) report that changes can increase costs and time up to 30-50% of the original design resources allocated. Managing them is, therefore, as important as managing iterations in the product design process.

Several approaches have been developed to predict engineering changes and mitigate its effects. Clarkson et al. (102) developed the Change Prediction Method (CPM), which assesses the risk of changes by evaluating the dependency between the components of a system with Design Structure Matrix (DSM). Each connection is quantified by a likelihood of change and the impact of the changes. A probabilistic algorithm is provided to calculate the risk of propagation taking into account also the multiple paths by which two components might influence each other. Many variants of CPM have been developed, such as Koh's et al. (103) integration with House of Quality to manage requirements change, or Hamraz et al. (104) method which assesses the probability of propagating changes between two subsystems.

Another approach is to focus on the robustness of the design, which has been advocated by Taguchi and Clausing (105). Margins are commonly used to achieve robustness of designs in the face of uncertainty (95). According to Eckert (106), margins are defined as "the extent to which a parameter value exceeds what it needs to meet its functional requirements regardless of the motivation for which the margin was included". In other words, margins occur when a property of the design has not been optimised, and therefore has a surplus of some kind. This allows us to accommodate unexpected changes without rework (97) by consuming the margins. However, margins entail performance loss and increased costs due to overspecification (107). Guenov et al. (108) proposed an interactive framework for the visualisation and management of margins, allowing qualitative evaluation of the design space of a commercial aircraft. Brahma and Wynn (107) developed a method to identify excessive margins that contribute little to the robustness of the design relative to its performance degradation.

Margins share with Set-Based Design the characteristic of reasoning with interval values. This ability allows both methods to accommodate changes with limited repercussions through the design process. However, margins are the overspecification of an individual design point, that is, designers consider variability *after the design point has been selected*. In contrast, SBD considers variability *before selecting a design point*. The margins act as a form of local insurance, accommodating changes that are within the vicinity

of the design point. Large mistakes, such as not recognising the boundaries of the desired solution space, cannot be accommodated with margins alone. Instead, SBD intrinsically explores the alternatives, providing a much better coverage of the design space. Instability is mitigated by avoiding early commitment to a design space without eliminating unfeasible solutions (32). McKenney et al. (36) demonstrated that the delay in SBD commitment is capable of handling engineering changes better than traditional point-based methods. This dissertation will focus on applying the SBD methodology to requirements with a probabilistic approach in order to accommodate changes and consider alternatives that might not 100% satisfy the requirements a priori.

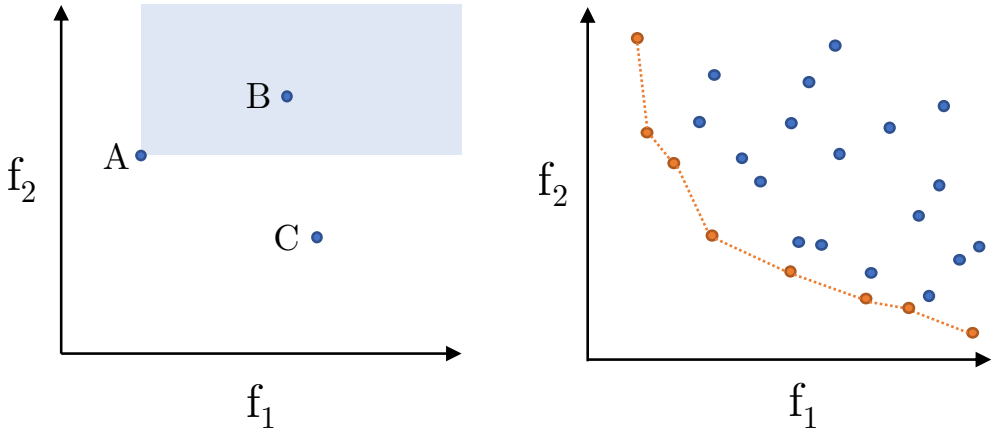
2.3 Multi-Disciplinary Optimisation

2.3.1 MDO in Engineering Design

Optimisation is the act of identifying the input value(s) x of an objective function $f(x)$ such that it is maximised or minimised globally (74). The input values are often bounded by constraints, i.e. $x \in [a,b]$, and subject to one or more constraint functions such as $g(x) \geq 0$. Furthermore, multiple objective functions could be present, often competing. Engineering design problems usually feature two or three competing objective functions, many constraints and a vector of input parameters called the design vector. The generic mathematical formulation of a multi-objective optimisation (MOO) problem is shown in (2.1). MOO problems are of engineering interest because designs are often the result of a compromise between different requirements. For instance a commercial aeroplane is designed to achieve the lowest fuel consumption and operating costs but subject to constraints of reliability and safety, or the aerodynamics of a race car need to produce enough downforce to sustain high speed bends without excessive drag. Hence, engineering optimisation is used to make product design decisions and identify the best available design options.

$$\begin{array}{lll} \text{given} & \boldsymbol{x} = \{x_1, x_2, ..., x_n\} & \\ \underset{\boldsymbol{x}}{\text{minimize}} & [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), ...] & \\ \text{subject to} & g_k(\boldsymbol{x}) \geq 0 & k = 1, 2, ..., K \\ \text{subject to} & x_{iL} < x_i < x_{iU} & x_i = 1, 2, ..., n \end{array} \tag{2.1}$$

Compared to single objective optimisation problems, the meaning of optimality is less intuitive when multiple objectives are considered. Pareto optimality is used for identifying the optimal points in MOO problems. Figure 2.3a shows three design points measured against two objectives, objective f_1 and f_2. Point A is said to *dominate* point B because it is better in both objectives. This is not the case for point C, which is better than point A in objective f_2, but not in f_1. In this case, A does not dominate C. A point is said to be *nondominated* if none of the other evaluated points dominate it. If that is the case, then the point is said to be *Pareto optimal*. The group of nondominated points is referred to as the *Pareto set*, while their evaluation in the solution space is known as the *Pareto front*. Figure 2.3b highlights the nondominated points and their approximation of the Pareto front.



(a) The shaded region highlights the area of the design space dominated by design A (After Martins and Ning (74)).

(b) Orange points are non-dominated designs approximating the Pareto set (After Martins and Ning (74)).

Figure 2.3

Engineers use the Pareto front to analyse the trade-off between different objectives, such as cost and performance, and select the most suitable design point for their scopes. Different search methods have been developed for MOO problems. The simplest is the weighted sum method (109), where each objective is assigned a scalar weight representing the relative importance to other objectives, as shown in (2.2). This effectively converts the MOO problem into a single objective optimisation problem.

$$\bar{f}(x) = \sum_{i=0}^{n_f} w_i f_i(x) \quad \text{where} \quad \sum_{i=0}^{n_f} w_i = 1 \tag{2.2}$$

However, the weights w_i prescribe a trade-off between the objectives, that is, fix the Pareto front's local slope, leaving the optimiser to just find the point closest to it. Indeed the weighted-sum objective equation specifies a line with the objectives as ordinates. Hence, for identifying the full Pareto set of solutions, the user should prescribe different distributions of w_i. Specifying the weights is one of the drawbacks of the method, as it is not as straightforward and uniform spacing of weights often leads to non-uniform Pareto sets (109). Furthermore, the weighted-sum method can only capture convex Pareto fronts.

An improvement over the weighted-sum method is the epsilon-constraint method, first described by Haimes et al. (110). The original MOO problem is converted to a single-objective problem by converting the other objectives to constraints, as shown in (2.3). The procedure is repeated by changing the value ε of the constrained objectives until each Pareto set point is found.

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f_k(\boldsymbol{x}) \\ \text{subject to} \quad & f_m(\boldsymbol{x}) \leq \varepsilon_m \qquad m = 1,2,...,M \quad \text{and} \quad m \neq k \\ & g_j(\boldsymbol{x}) \leq 0 \qquad j = 1,2,...,K \\ & x_{iL} < x_i < x_{iU} \quad x_i = 1,2,...,n \end{aligned} \tag{2.3}$$

Unlike the weighted-sum method, the ε constraints can be set more intuitively and can help explore the Pareto front interactively. In addition, non-convex portions of the Pareto front can be identified. Nonetheless, the epsilon-constraint method too does not yield a uniform spacing of the Pareto front.

In order to guarantee a uniform Pareto front, the normal boundary intersection (NBI) method was developed by Das and Dennis (111). It constructs a support plane passing through each objective function's single-objective (SO) optima. Then, points are equally spaced along the plane, defined by a weighting vector b. The MOO is then reformulated as a maximisation problem to find the point furthest from the support plane and perpendicular to the node, with the constraint that the point is consistent with the objective functions:

$$\begin{aligned} \underset{\boldsymbol{x},\alpha}{\text{maximise}} \quad & \alpha \\ \text{subject to} \quad & Pb + f^{\star} + \alpha\hat{n} = f(x) \\ & g_j(\boldsymbol{x}) \leq 0 \qquad\qquad j = 1,2,...,K \\ & x_{iL} < x_i < x_{iU} \qquad x_i = 1,2,...,n \end{aligned} \tag{2.4}$$

where $Pb + f^s tar$ is the support plane equation, defined by the vector of SO optima $f^{\star} = [f_1(x_1^{\star}), f_2(x_2^{\star}), ...]$ and the coefficients matrix $P_{i,j} = f_j(x_i^{\star}) - f_j^{\star}$, and $\hat{n}$ is the pseudo-normal in the direction of the Pareto front. This process yields a more uniform front at the cost of a more complicated methodology.

The method presented so far effectively reduces a MOO problem into a SO one so that it is possible to use gradient-based and gradient-free algorithms. In contrast, evolutionary algorithms such as Genetic Algorithms (GA) do not require this reduction because they keep track of a large population of designs at each iteration. Evolutionary methods originated in Artificial Intelligence research, with the earliest example by Barricelli in 1954 (112), and focusing on simulating the behaviour of artificial organisms (113). Eventually, these methods were applied to solving engineering optimisation problems (John Holland proposing them in 1975 (114)), as GA does not require the estimation of derivatives, allowing the use of complex evaluation functions. In addition, GAs are well suited to handle MOO problems since they track a large population of points independently of the number of objectives. The non-dominated set evolves with the population, converging towards the actual Pareto front. The most popular MOO version of GA is the Nondominated Sorting Genetic Algorithm II (NSGA-II) developed by Deb et al. (115).

NSGA-II begins with a random population P_t of n_d designs, each represented by a chromosome, a vector containing the design variables. Each solution is then evaluated and ranked based on its nondomination level. This is done through a nondominated sorting algorithm such as Kung's method (116). The lower the rank, the higher the number of points that are dominated by the set, where rank 1 is the set of the nondominated designs in the entire population.

The three steps of GA are carried out next: selection, crossover and mutation. The selection process pairs the solutions for mating: tournament selection is used in NSGA-II to increase the chances of improving the gene pool. Each population point is paired randomly, with the best point being added to the gene pool. Crossover is carried out by randomly generating two offsprings from two "parent" points by splicing the chromosome vectors and recombining the parts. A mutation is finally added by randomly modifying

the values in the chromosome vector of a few design points.

The new offspring population Q_t is then added to the parent population P_t, generating a population of $2n_d$ designs to be reduced. Reduction is carried out based on the *elitist* strategy; that is, the fittest designs are retained and passed to the next generation. First, the nondominated sorting algorithm is applied to rank the population. Then the new population is filled with the points of the best nondominated set, repeating iteratively until no more sets can be accommodated. If the last selected set has more points than available spaces in the population, a crowding-distance sorting is carried out to select the points most spaced out from those already present in the new population. By doing this, there is better coverage of the objective space, allowing for the convergence to a uniformly spaced Pareto front. For this reason, crowding distance is used as an additional criterion in the tournament selection process of NSGA-II.

Deb and Jain (117) later introduced a many-objective extension of NSGA-II. This new algorithm, NSGA-III, replaces the crowding distance sorting with reference directions. The objective space is divided into several non-dimensional vectors, each representing a non-dimensional trade-off. The population filling process follows the same nondominated sorting ranking but prioritises the reference lines that contain the least number of designs. The design point closest to the reference direction is selected to ensure better coverage of the objective space. This is particularly important when there are more than two objectives, despite random selection. A further improvement came from the Unified-NSGA-III (U-NSGA-III) algorithm developed by Seada and Deb (118). It is a modification of NSGA-III with additional tournament pressure, which allows it to perform effectively on single-objective and bi-objective problems and in many-objective ones.

Optimisation algorithms became widely used in engineering as the availability of computers increased from the 1960s onwards. The aerospace sector has been at the forefront of the application of optimisation methods (119), specifically regarding the optimisation of structures (120) in weight and stiffness, and the aerodynamic optimisation of bodies (121) and wings (122). At the same time, aeronautical engineers started investigating optimising a system by coupling more than one discipline. This approach would capture the interactions between the components as the system is optimised. This field of engineering optimisation is named Multi-Disciplinary Optimisation (MDO). Coupled models better reflect the state of the system under design and its performance once deployed in the real world, in contrast with optimising individual components with assumed boundary conditions. Furthermore, engineers can introduce in the model other relevant considerations, such as cost, human factors and environmental impact.

Another advantage is the ability to handle shared design variables, parameters that simultaneously affect more than one system or discipline. The traditional sequential approach, where each discipline optimises the design variables independently, would likely yield contradicting results. On the other hand, if only one discipline is optimised, it would likely produce a design that violates the other disciplines' constraints. These pitfalls are avoided in MDO, as optimising components simultaneously leads to designs with better performance and no constraint violations (74).

MDO originated from the efforts of Schmit (123, 120) and Haftka (124, 125) in coupling wing structural design with aerodynamic forces. The aeroservoelastic problem, in which wing design would simultaneously consider aerodynamics, structure and controls, was a significant driver of the development of MDO methods, as discussed by Holt (119)

and Livne (126). Since then, MDO has been extended to whole aircraft design as computational power increased (Kroo et al. (127), Antoine and Kroo (128), Henderson et al. (129)).

Different methodologies have been developed to handle the optimisation of complex systems. Martins and Lambe (130) extensively classify the current approaches. The first main classification, already identified by Balling and Sobieszczanski-Sobieski (131), is in the number of concurrent optimisation problems. The approaches are named "monolithic" or "single-level" and "distributed" or "multilevel". A monolithic approach features a single system-level optimiser that simultaneously identifies the system and discipline parameters. In contrast, a distributed method features discipline-level optimisers identifying the design variables for each discipline and interacting with a system-level optimiser which searches only for the system design variables. Consistency between the disciplines is maintained by communicating constraints between the local optimisation problems. Decomposition was developed originally as an attempt to exploit the structure of the problem to reduce computational time. Unlike in a monolithic architecture, where every discipline is run at every function evaluation, the decomposed sub-problems can be run asynchronously, depending on the need to update the system-level optimiser at each iteration (132). Distributed computing centres can use this to speed up the computations significantly by employing reconfigurable workflow scheduling systems. Furthermore, a distributed approach can benefit large teams divided into disciplines (133). Another advantage is the ability to propagate design objectives and constraints to the subsystems or components, reducing the number of functional evaluations when optimising a design. An example of such an approach is the Analytical Target Cascading (ATC) method developed by Kim et al. (134). In contrast, monolithic approaches guarantee a tighter coupling between the disciplines and are easier to manage conceptually, as the whole design is cast as a single optimisation problem. Since this dissertation will adopt a monolithic MDO architecture, it will receive more attention.

Monolithic approaches are categorised according to the structure of the model MDA (Multi-Disciplinary Analysis) and the number of residuals the optimiser has to manage. The layout of the MDA affects the computational cost and the coupled model's convergence rate. Indeed, it is possible to rearrange the coupled variables such that the individual solvers can be executed in parallel (Jacobi method) or in sequence (Gauss-Seidel method), minimising the number of iterations for the evaluation function (135).

The Extended Design Structure Matrix (XDSM) introduced by Lambe and Martins (135) displays the interdependency between the coupled variables and can be used for structuring the MDA model for maximising performance. An example of an XDSM of a wing design problem is shown in Fig. 2.4. The system state vector y is composed of output from the aerodynamics solver y_1 and the structural solver y_2. The MDA solver, in this case, a Gauss-Seidel type, is tasked with converging the solution for each solver iteration. The MDA can benefit from a hierarchical structure for complex systems where nested solvers calculate the state variables for local components and update them up to the system level. Gray et al. (136) developed the OpenMDAO framework to facilitate integrating different analyses within an efficient MDA structure for MDO problems.

If a MDA model is used and the optimiser does not handle any residuals, then the monolithic MDO architecture is named Multi-Disciplinary Feasible (MDF) (137). MDF is the intuitive form of MDO, where the optimisation algorithm is only exposed to design

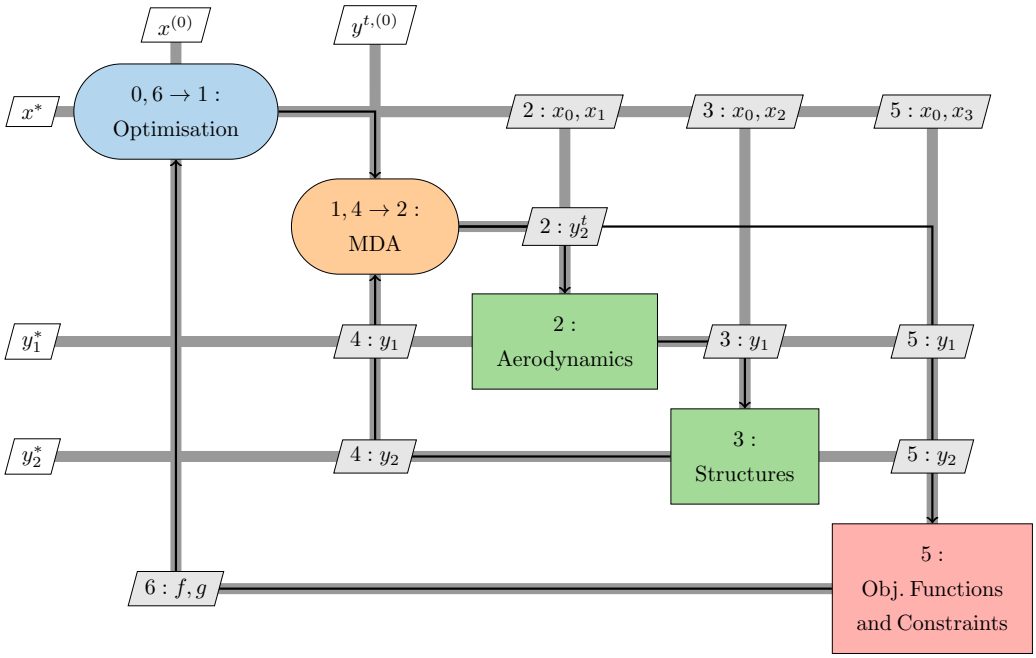


Figure 2.4: Example of a XDSM Diagram of an aerostructural MDO problem (From Jasa et al. (2)). The architecture used is of MDF type, where the optimiser is not tasked to manage the coupling between the disciplines.

variables and quantities to be optimised or constrained. The evaluation function, a MDA model, guarantees the consistency of the variables shared by the components. While being flexible, its disadvantage is in formulating and implementing the MDA. Moreover, each optimisation iteration requires a fully converged multi-disciplinary analysis. An example of MDF is the aero-structural problem shown in Fig. 2.4. The MDA block converges the analysis by fixed-point iteration, providing each solver with the updated system state.

An alternative is introducing a consistency constraint for the shared variables at the optimiser level and executing the component solvers in parallel: this architecture is called Individual Discipline Feasible (IDF) (137). It avoids the formulation of an MDA but increases the burden of the optimisation algorithm. Furthermore, the discipline coupling is not guaranteed until full convergence as the consistency constraints are in the optimisation problem. Early stopping of the solver would return solutions with their coupling unconverged. Figure 2.5 shows the IDF version of the aerostructural problem discussed above. The added constraint $g_y = 0$, where $g_y = y - y^t$, is introduced to obtain the coupling of the solvers.

Finally, the Simultaneous Analysis and Design (SAND) architecture extends the idea of IDF by moving every component state to the optimisation problem. The optimiser is now tasked to design the system to satisfy the constraints and objectives and identify the set of state variables that solve the analysis models, all expressed in a residual form. Converging the residuals towards zero would guarantee the coupling between the disciplines. This approach is also known as ”full space”, as the optimiser controls every variable. Adopting a SAND architecture requires an optimisation algorithm capable of handling many variables in a feasible computational time. In addition, the discipline solvers must

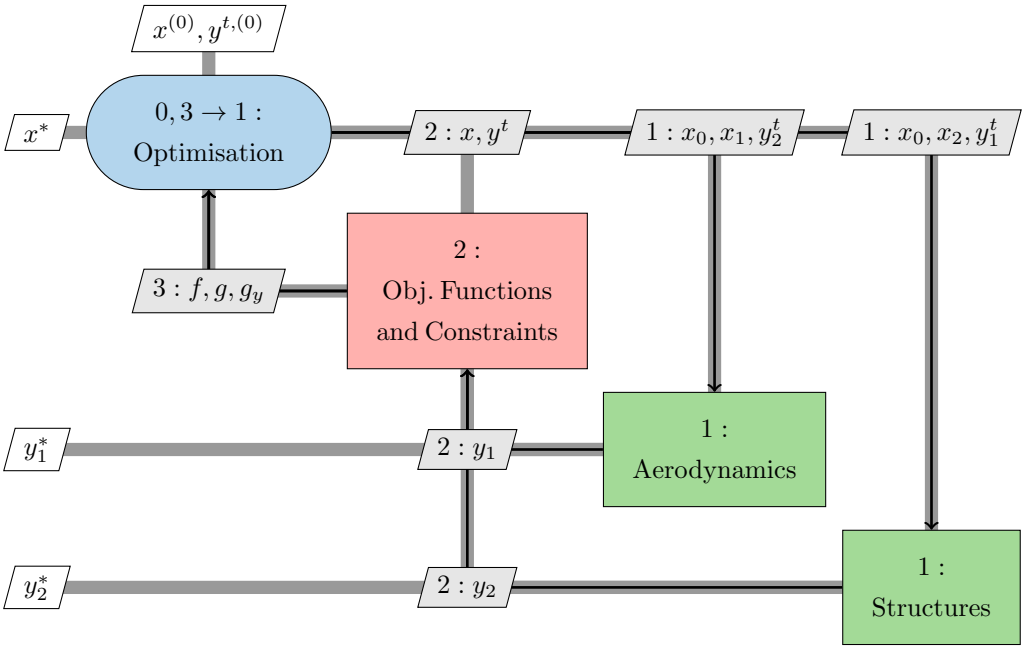


Figure 2.5: IDF version of the MDO problem presented in Fig. 2.4.

be provided in residual form, which cannot be possible for black box models. However, the advantage of SAND, and IDF, is the decoupling of the disciplines, which is beneficial for gradient-based solvers. It is easier to calculate the gradient of each discipline individually than to produce the full Jacobian matrix for the fully coupled MDA, as in the case of the MDF architecture. Figure 2.6 presents the SAND version of the example aerostructural problem. The solvers must return their values in the residual form $r(y)$, calculated as the difference between the updated state variable and the previous state vector $\hat{y}$.

Overall, the development of MDO has brought the capability to optimise a design in earlier stages of the design phase, thanks to the integration of components and the coupling between their behaviours. Engineers can simultaneously design and analyse their system and select the best design variables according to the constraints and objectives. Indeed as long as a requirement can be formulated as a quantified expression, be it an objective function or a constraint function, it is possible to identify the desirable Pareto set for that design. The optimisation problem's mathematical flexibility allows for optimisation for other parameters that are not directly related to performance or costs. For instance, a system could be designed to tolerate variability in its design parameters, which can be affected by unpredicted technological changes or production errors. Di Bianchi et al. (138) presented a framework which utilises an MDO problem with an uncertainty-propagating model. This framework enabled decision-making while considering the impact of design variability over performance.

One difficulty in large scale integration is the computational cost of the MDA. The components often require high-fidelity models unsuitable for large amounts of function evaluations. A solution is developing a surrogate model that can capture the original model's dynamic behaviour with a reduced computational cost. The following section will discuss this point. MDO generally produces a large amount of data which can be challeng-

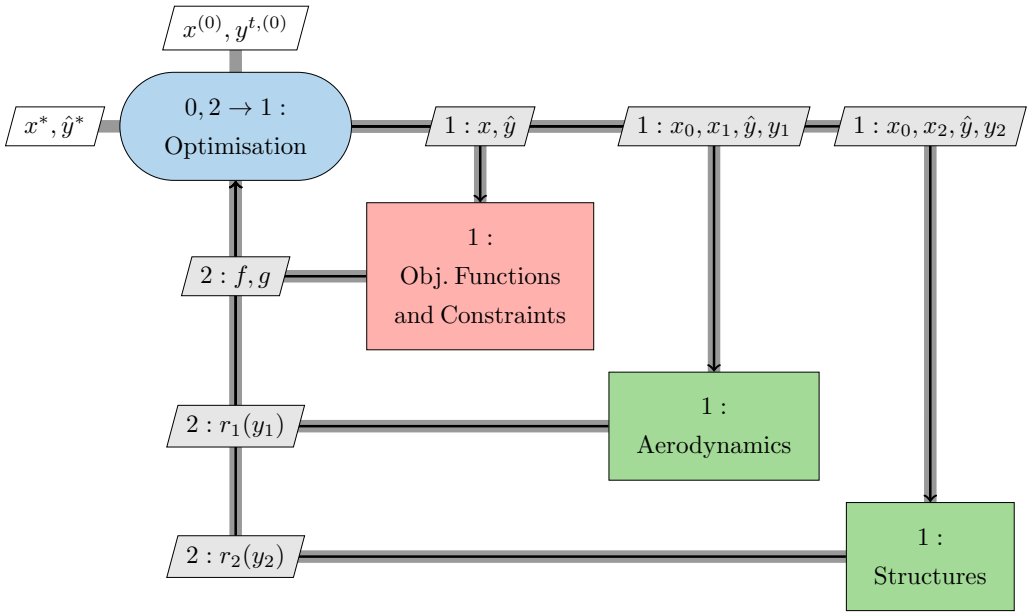


Figure 2.6: SAND version of the MDO problem presented in Fig. 2.4.

ing to interpret. Hence, suitable visualisation tools are necessary for the decision-making process. Section 2.3.3 will introduce the reader to multi-axis visualisation methods for this purpose.

2.3.2 Surrogate Modelling Principles

Surrogate models (SM), also known as response surfaces, reduced order models (ROM) or metamodels, is an approximate model of a functional output that represents an interpolation of some underlying data (74). The objective of the surrogate model is to be less computationally expensive to use than the full model but retain enough accuracy away from known data points.

Surrogate-based Optimisation (SBO) performs optimisation using a surrogate model. The SM can be used in place of the entire model or as a part of it. For instance, a SM could be trained for modelling the thermodynamic behaviour of a gas turbine and used in a mission analysis solver.

SMs are beneficial when the overhead cost of generating data and training a SM for optimisation is lower than the cost of using the full model in the optimisation problem. In addition, SMs have the advantage of smoothing noisy models and combining information from multiple sources, such as models with different fidelities or experimental data.

One potential difficulty is the curse of dimensionality, the poor scalability of SMs with the number of inputs. If the number of design variables is large, the cost of producing the training data points might outweigh the benefits of using the SM in the first place.

The SBO process consists of 3 steps: sampling the design space, constructing the surrogate and performing the optimisation. Full-fidelity data can be recovered at the end by evaluating the optimal results with the original model. Some SBO systems adopt a convergence check for evaluating the accuracy of the SM after optimisation: an infill

procedure is used if the SM is not accurate enough in some areas of the design space. Pagliuca et al. (139) published a SBO study with an active infilling method automatically switching between the surrogate and full models to maximise speed and accuracy.

Sampling produces data points from the full model for training the SM. Covering the design space uniformly is fundamental for extracting as much information as possible. The first sampling techniques were derived from Design of Experiments (DOE) and consisted of factorial sampling. These cannot be scaled for high inputs samples, as they evaluate every possible combination of input levels. Random sampling methods are instead commonly used. However, simple random sampling is not robust in covering the design space with sufficient uniformity. For this reason, the most prevalent sampling method is Latin Hypercube Sampling (LHS) (140). LHS divides the design space in N_s hypercubes, one for each sample point. Samples are then positioned such that only one point exists per dimension, forming a multidimensional Latin square. The algorithm for producing an LHS is straightforward. Each axis is partitioned in N_s evenly spaced regions, and a random number ranging from 0 to 1 is generated for each interval. Then the per cent-point function (the inverse of the cumulative distribution function) of the corresponding probability distribution is used to calculate the actual points. The uniform distribution is the most commonly used, but in principle, any is supported. Finally, the points are randomly permuted. While this approach does not guarantee optimal spacing, it works well and is simple to implement. Alternatives to LHS are the pseudo-random low discrepancy sequences. Examples are the Halton sequence (141) and Sobol sampling (142).

The training dataset is produced by evaluating the samples with the original model. The next step is constructing and training the surrogate model. SMs are classified into two categories: physics-driven and data-driven. The former has the advantage of using lower fidelity physical laws with an interpolated correction model, requiring fewer data points for the same level of accuracy (143). Nevertheless, it is not as flexible as the latter since they are limited to the specific application of the model, such as fluid mechanics or stress analysis. On the other hand, a data-driven approach can reproduce any function without any mathematical assumptions but requires more data.

Popular data-driven SMs include least squares regression, Gaussian processes (GPs) and deep neural networks (DNN). Least square regression is based on a linear regression model expressed as:

$$\hat{f} = \sum w_i \psi_i(x) \tag{2.5}$$

where w_i are weights and $\psi_i(x)$ the basis functions. These can be any function, not necessarily linear, as linearity is in the combination of the basis functions. The weights w_i are found by minimising the sum of squared errors between the surrogate function and the actual function values through quadratic programming:

$$w = (\Psi^T \Psi + \mu I)^{-1} \Psi^T f \tag{2.6}$$

where Ψ is a matrix of size $(n_s \times n_w)$, where each row contains the evaluation of each sample point in the ψ_i basis function, and f is the vector of the actual function values. μ is a regularisation weight used for smoothing the weights in case of noisy data or strong outliers. Polynomials and Radial Basis Functions (RBF) are the most commonly used basis functions. Polynomials are useful if the underlying data or physical phenomenon

suggest a trend, such as quadratic or linear. High-order polynomials are rarely used because they seldom generalise well, usually suffering from high oscillations away from the data points (overfitting). Radial basis functions are functions whose value depends on the distance from a centre point (144). For linear regression, the sampling data is used for centre points as it captures the concept that accuracy is higher the closest to the data points. A commonly used RBF is the Gaussian basis:

$$\psi_i(x) = exp\left(-\theta|x - x_i|^2\right) \tag{2.7}$$

where θ is the model parameter. RBFs and low-order polynomials can be combined to capture local and global trends.

Gaussian processes (GP), also known as Kriging, is an interpolation model capable of approximating highly nonlinear functions (145). It is based on approximating the true function by predicting the behaviour probabilistically rather than tuning a simplified model to the data points. The statistical model is constructed as $f(x) = \mu(x) + Y(x)$ where Y is a Gaussian random variable $Y(x) \sim N(0, \sigma^2)$. Usually, the deterministic value μ is constant as the random variable is effective enough in capturing the function's behaviour. The core of the statistical model is in predicting the correlation between the random variables, which is assumed to be a function of its distance. For this purpose, a multivariate Gaussian distribution is assumed to be the correlation function, as shown in (2.8), with parameters similar to the RBF. This produces a correlation matrix used in the prediction of unknown data points. Training is performed by maximising the log-likelihood function through numerical optimisation for tuning the θ_k and p_k hyperparameters. Jones (145) presents the full mathematical derivation of the GP equations for training and predicting.

$$K(x^i, x^j) = corr[Y(x^i), Y(x^j)] = exp\left(-\sum_{k=0} \theta_k |x_k^i - x_k^j|^{p_k}\right) \tag{2.8}$$

The expected value of the GP is the response used in the SM. In addition, GPs have the advantage of returning the mean square error of the prediction, which can quantify the model's accuracy. This capability is useful for infilling techniques, as it is possible to calculate the expected improvement to the optimisation from adding a new data point over the current dataset, as initially demonstrated by Jones (146). Furthermore, GPs are interpolatory and match the data points perfectly. With noisy data, it is possible to use GPs for regression by adding an identity matrix with a weight factor to the correlation matrix to be tuned with the other hyperparameters (147). The early interpolation application of GPs has been the identification of deposits of natural resources by the geostatistician Danie Krige (148), from whom the name "Kriging" is derived. Since then, GPs have been successfully used in surrogate modelling for engineering computer experiments (149).

Deep Neural Networks are graph models which can be used as universal function approximators (150). These are loosely based on biological neural networks where each artificial neuron fires, i.e. output a quantity if the weighted sum of its inputs goes above an activation level. Non-linearity of DNNs is achieved through activation functions and stacking multiple layers of neurons. Connecting neurons is equivalent to chaining multiple functions to obtain a complex one. The layout of a simple feedforward DNN is shown in Fig. 2.7. Input neurons receive the values for computation, while output neurons return the function output. Hidden layers perform the transformations on the values

and propagate them forward. A DNN is ”deep” when more than two hidden layers are present.

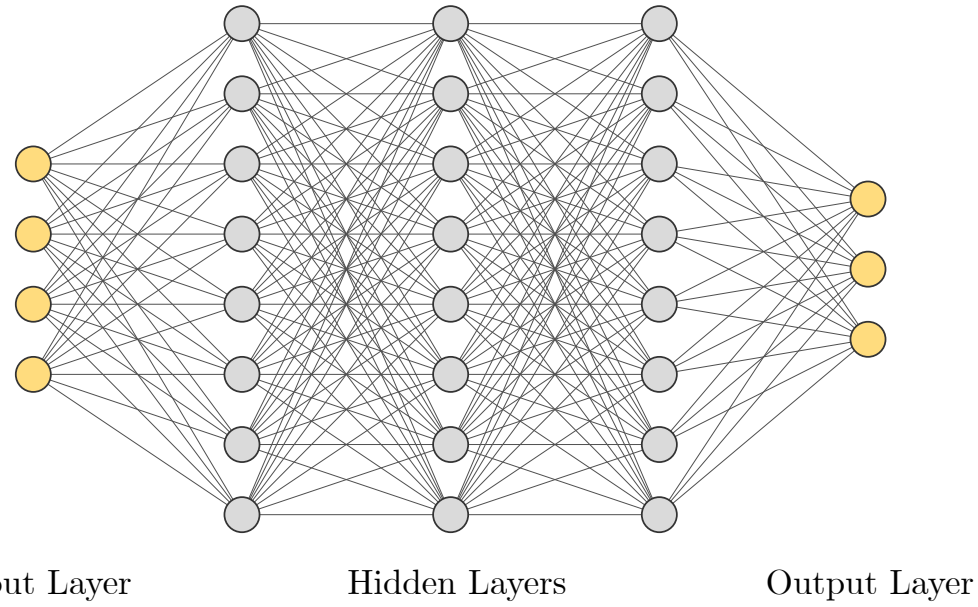


Figure 2.7: Layout of a Deep Neural Network.

Training involves identifying the combination of weights and biases that minimises the training error. The backpropagation algorithm (151) is used to find the gradient of the whole network to optimise it with a steepest descent algorithm efficiently. Goodfellow et al. (152) recently published an extensive mathematical treatment of DNNs. The advantage of DNNs over the other approximation models is their generality; no parametric model is required. However, compared to linear regression methods, much more data is required to achieve the same level of accuracy.

When the model is trained, validation must be performed to quantify the accuracy and the potential need for more data. Simple validation involves using a different dataset and calculating the error between predicted and true values. However, a more robust procedure is the k-fold cross validation (153). It shuffled the training dataset and sequentially reserved a portion for validating the model trained with a specific combination of hyper-parameters. This method enables comparing different models and ensures convergence without overfitting (154).

2.3.3 Visualisation Tools for Complex Data

Visualisation is the process of converting data into pictures for easier interpretation by human beings. This facilitates the cognitive process of acquiring information and knowledge from the dataset (155). Colours, spatial relationships, dimensions and patterns are mapped to the different features of the dataset for extracting and communicating the underlying meaning. Hence, information can be extracted from the data and synthesised into knowledge that the engineer uses to move the product design forward.

In an optimisation problem, the information that needs to be visualised is the Pareto set in the parameter space and the Pareto front in the functional space. These provide

the range of available solutions which can be selected for further analysis and development. For bi-objective problems, a scatter plot is sufficient for showing the Pareto front, with each axis representing an objective (Fig. 2.3b). One additional dimension can be represented through a hue scale or a gradient of colours on the markers.

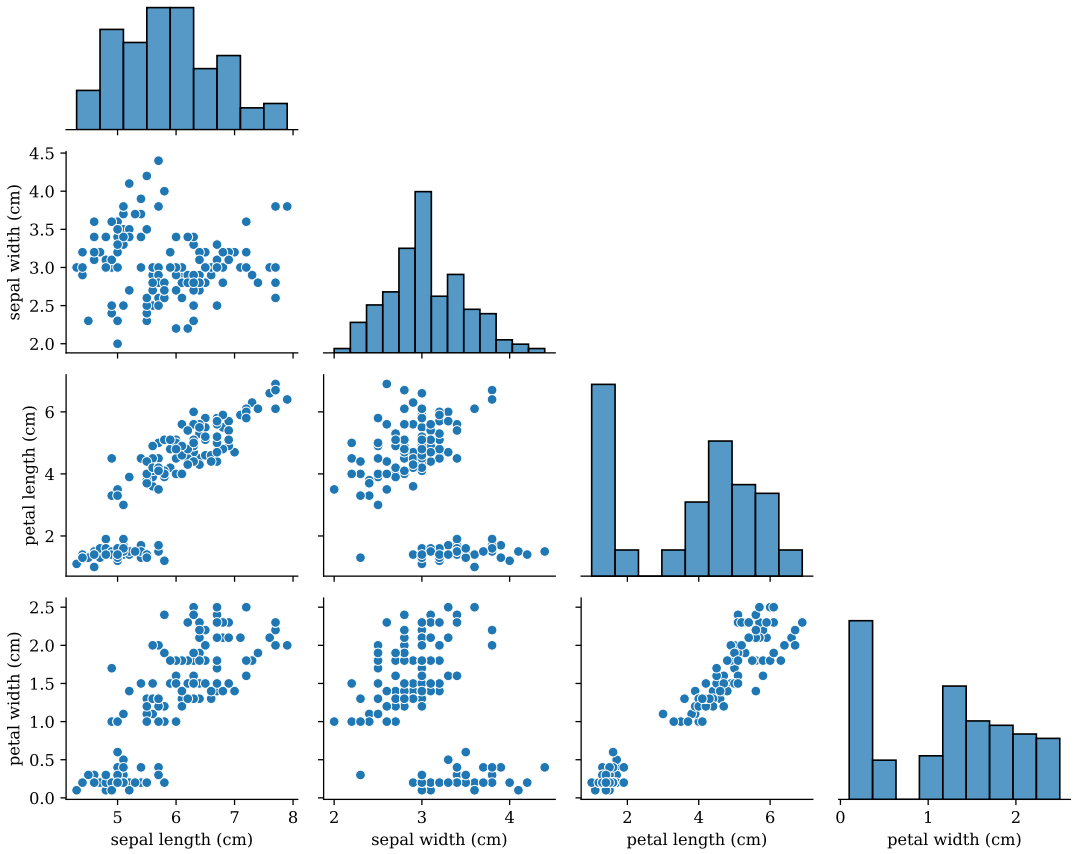


Figure 2.8: A pairwise scatter plot of the iris dataset.

Challenges arise when the number of data dimensions is greater than three. Scatter plots can ideally support up to 5 dimensions by introducing hue, saturation, marker size and marker shape scales on top of the two axes. In practice, the result would be a very cluttered and hard to read diagram.

One possible approach is combining multiple simple scatter plots in a matrix layout such that it is possible to visualise every possible pair of data axis. This diagram, named *pairwise scatter*, shows immediately the trends between each pair of data and, in the case of functional outputs, which input parameters have the most significant impact on the objectives or constraints. An example of this diagram is shown in Fig. 2.8.

Nonetheless, this diagram cannot show in a bird's eye view the effects of one dimension over all the other dimensions. This can be achieved with a *Parallel Coordinates* (PC) diagram. PCs are composed of parallel axes for each dimension of the dataset. Each data entry is portrayed with a polyline intersecting the axes at the value of each dimension. PCs can effectively display data up to 10-15 dimensions (156). Figure 2.9 shows an example of a PC plot for the same iris dataset of Fig. 2.8.

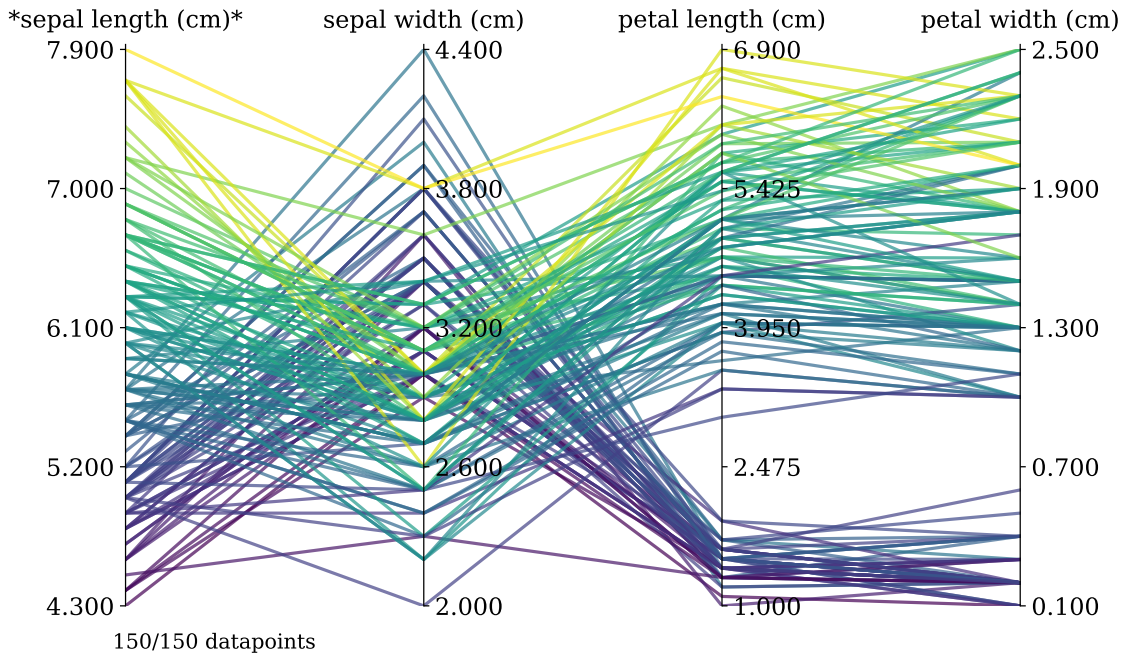


Figure 2.9: Parallel coordinates plot of the iris dataset.

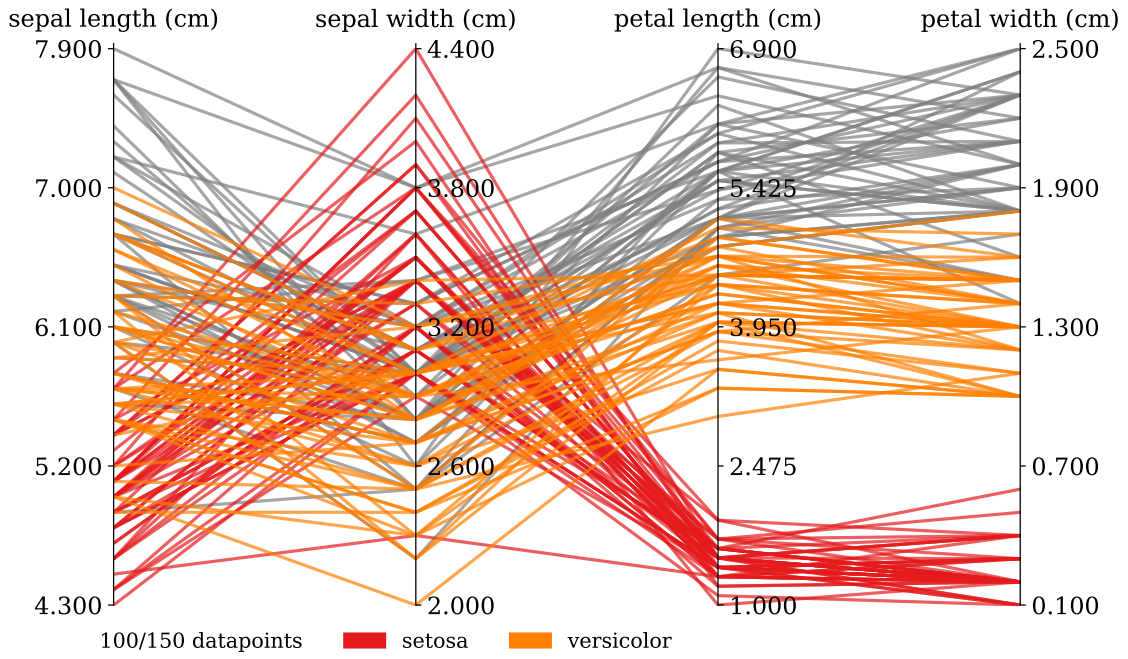


Figure 2.10: Selections in a PC plot.

Parallel Coordinate diagrams are particularly suited for pattern discovery and exploratory data analysis. Inselberg (157) famously published an article where he describes the process as a ”multidimensional detective”. In essence, PCs shine in their ability to propagate selections from one axis onto all the dimensions, enabling them to analyse data more quickly and identify patterns. For instance, in Fig. 2.10, the setosa and versicolor datapoints have been selected and highlighted. The setosa class specifically has a clear

separation in petal width and petal length. After selecting these data points by limiting the range of petal length between 1 and 2 cm, it is discovered that the sepal length for this class of flowers ranges between 4.3 and 5.8 cm. The same effect can be achieved globally by applying a colour scale reflecting the values on one axis on the lines. Figure 2.9 shows how the sepal length is correlated with the other dimensions. Besides colour and selections, PCs portray the correlation between each adjacent axis through the disposition of the lines. Figure 2.11 shows direct and inverse correlation examples as displayed in a cartesian plot and its equivalent PC plot. Inselberg's book on the topic (158) is the definitive resource for implementing and utilising PCs.

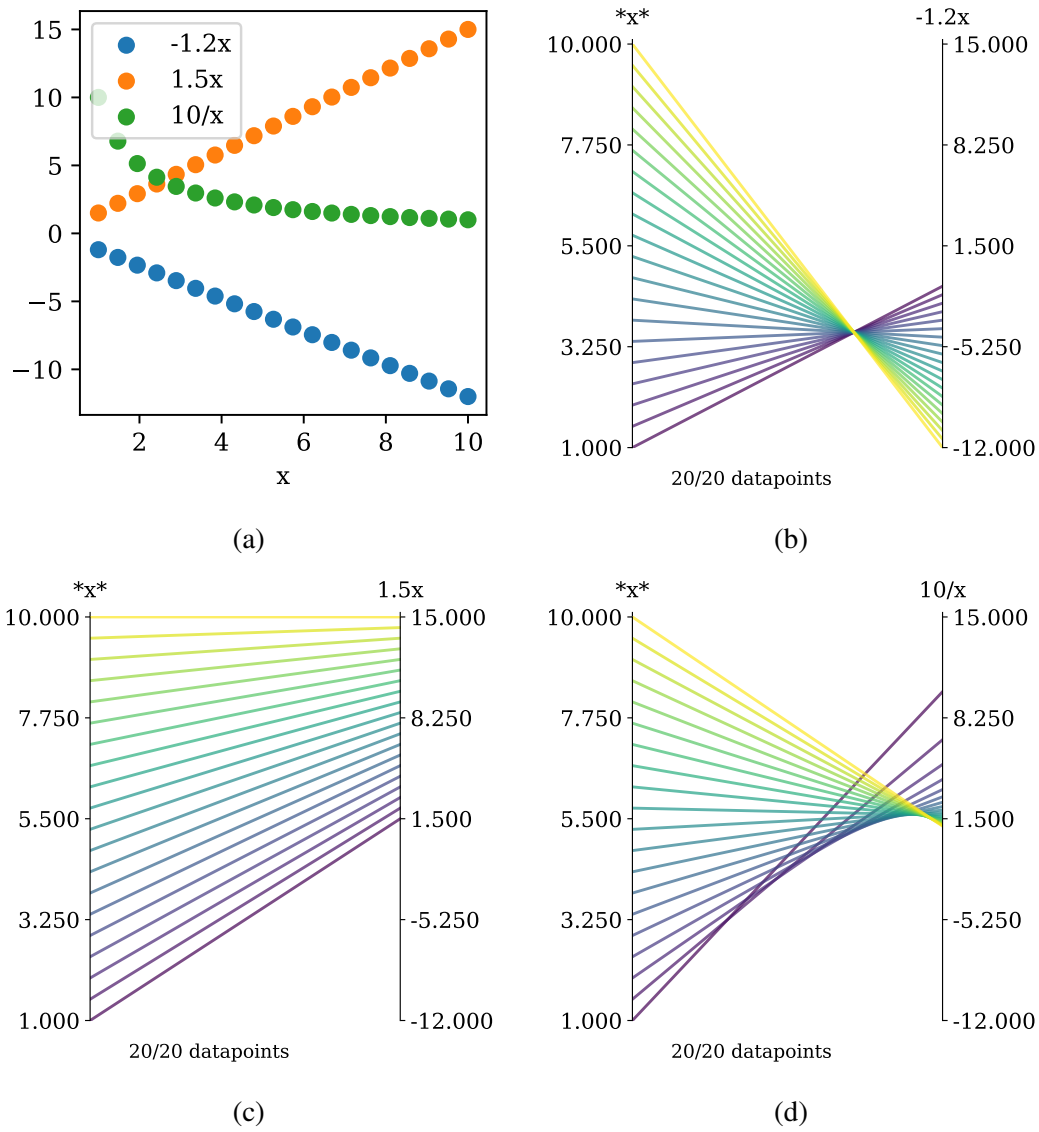


Figure 2.11: Comparison between Cartesian and PC representation of correlations.

PC plots are best suited for interactive visualisation tools. Specifically, the ability to change the order of the axis and select ranges within the axis themselves provide the user with exploratory capabilities that cannot be obtained with scatter plots alone. Kipouros et al. (159) demonstrated an interactive visualisation tool combining PC and scatter plots.

The engineer would use the PC plot as a guiding tool to isolate interesting data points and the scatter plots as a magnifying scope for understanding the correlations between two specific dimensions or visualising the Pareto front.

Interactivity has the advantage of combining the strengths of different plots (e.g. Fig. 2.8 and Fig. 2.9) and providing dynamic connectivity as the user searches within the data. Selections from the PC plot would propagate to the scatter plots and vice-versa, enabling one to extract information from the data rapidly. A further enhancement of interactive visualisation is the addition of regression and machine learning methods for extracting new information on the fly. The combination of these has been named *Visual Analytics*. Piotrowski et al. (160) presented an enhanced tool where Kriging was used to produce new data points.

As the complexity of MDO studies and the amount of produced data increases, interactive visualisation tools and visual analytics become necessary to understand the behaviour of systems under design and identify the best parameters that satisfy the requirements.

2.4 Technological Challenges of Hybrid-Electric Aircraft

Aviation contributes to 2.5% of the global anthropogenic emissions of CO_2 (161), which is expected to grow due to the expansion of airline operations and fleets. Indeed, the leading manufacturers, Airbus and Boeing, predict an annual growth rate of traffic between 3.6 and 3.8% (162, 163). Consequently, several regulatory bodies have put forth emission reduction targets to reduce the impact of the current climate change emergency. The European Commission, through the FlightPath 2050 plan, targets a reduction of CO_2 and NO_x emissions by 75% and 90% respectively compared to the levels in 2000 (164). ICAO has put forward similar provisions for the year 2050 (165).

The electrification of aircraft propulsion is seen as a potential solution for making aviation more sustainable and meeting these tight goals. Three different concepts have been proposed: full-electric, turboelectric and hybrid-electric. Their difference is in the extent to which the electrical energy source is used in flight and the arrangement of the electrical powertrain.

On one end of the spectrum, full electric propulsion envisions using batteries or fuel cells to power propellers or ducted fans. Often it is paired with exotic airframe configurations for exploiting synergies between aerodynamics, structures and propulsion (166). However, the forecasted improvement of battery pack technology cannot meet the required energy and power density in the near and mid-time frames (167).

Turboelectric propulsion falls on the opposite side, where the electrical powertrain is used for decoupling the gas turbine from the propulsive devices. While the gas turbine would operate in its ideal range in any flight condition, the overall system would be less efficient due to the losses in the electrical drive system. However, proponents of the turboelectric configuration claim these losses can be compensated by the adoption of distributed (168) or boundary-layer ingestion propulsion (169).

Hybrid-electric (HE) sits in between, as fuel and electric energy are used to power the aeroplane. The two sources of energy are complementary, one low density and high efficiency and the other high density but low efficiency. The combination of the two

can yield a less polluting propulsion system that is more efficient than a fuel-powered aeroplane alone.

The HE concept is further divided into parallel hybrid and serial hybrid. The parallel hybrid configuration features the gas turbine and the electric motors mechanically connected to the same shaft, providing power to the propulsive device in synergy. This arrangement requires fewer components; hence it is simpler and potentially less heavy than the other concepts. Furthermore, the electric motor can act as a booster to the gas turbine, which can be downsized to less power-demanding mission phases (e.g. cruise) (170). On the negative side, the mechanical coupling disrupts the operating conditions of the gas turbine, particularly if it is not downsized. Hybridisation requires less power from the engine under the same trust requirements. This causes the high-pressure shaft to operate at lower rotating speeds for the same airflow, which produces a reduced surge margin in the low-pressure compressor (171).

Nonetheless, the parallel hybrid concept has been widely explored with examples such as the NASA SUGAR Volt (171), the UTC Geared Turbofan (172), the EVE design by Rolls-Royce North America (173), and the PEGASUS design by NASA Langely (174), to name a few.

The series hybrid configuration instead completely decouples the gas turbine from the propulsors. Instead, these are electrically powered by a gas turbine-driven generator or a battery pack. Much like the turboelectric concept, the gas turbine can operate in its optimal condition independent of the fan or propeller speed. Likewise, this arrangement can accommodate distributed propulsion concepts, improving the overall aerodynamic efficiency if adequately integrated with the airframe. Despite these advantages, the lower efficiency of the electrical power chain causes a significant increment in weight, which may offset the efficiency improvements (175). For these reasons, fewer examples of series hybrid concepts can be found in the literature. Some examples are the Airbus E-Fan X (176), the Zunum Aero ZA10 (177), and the Pipistrel Panthera (178).

Despite the investigation in the last decade, HE propulsion is still far from reality. The limitations are in technology and understanding the multi-disciplinary aspects of the novel propulsion system.

In their extensive survey, Sahoo et al. (179) highlights the need for technological breakthroughs in energy storage technology, namely a significant improvement in energy density and power density and reliability under high cycle life. In addition, improvements in conducting materials and high-specific power electrical components are required. Further technological challenges are in managing high voltage, which aligns with the higher specific power requirement and weight-efficient thermal management systems for handling the waste heat from the electrical equipment (180).

HE propulsion introduces additional complexity in its system design requiring innovative design methodologies. Indeed, the sizing of the electrical components and the fuel consumption is determined by the degree of hybridisation (DOH) and the adopted energy management strategy (EMS). The added complexity in sizing arises from combining the variable-weight fuel source and the fixed-weight electrical energy source. Currently, sizing methods have been published by Pornet et al. (181), Riboldi et al. (182), Cinar et al. (183) and De Vries et al. (184). However, many of these methods do not consider the effect of the EMS on the design of the HE propulsion system.

Few studies have been published where the two are coupled (185, 186, 187, 188). Two

distinct approaches are adopted in these studies. The first consists of considering a fixed energy management strategy, usually a uniform degree of hybridisation over cruise and climb, and size the aircraft according to the energy requirements. This approach is usually selected to study and compare different HE architectures and can be found in Pornet (185) and Zamboni (186). Voskuij et al.(189) also investigate an EMS where the gas turbine operates at a constant power setting and compare it to the constant DOH strategy.

The other approach is to couple the energy management optimisation with the system sizing iteratively. Trawick (187, 190) implemented a global optimisation methodology where dynamic programming is used to construct the optimal EMS and sequentially size the system. This approach is more flexible since it does not assume any analytical form of the power schedule but calculates the optimal schedule by starting from the end state of the aircraft and searching backwards for the optimal power setting that minimises the objective function. The drawback is the high memory and computational cost, as the algorithm has to evaluate N×M2 states when searching for the optimal path, assuming a single-setting discretised M times over N time steps. Perullo and Mavris (191) proposed to couple the EMS and design with a distributed MDO approach. A system-level optimiser would search the optimal system design while a local model predictive control algorithm would set the energy management strategy to maximise range. Jinning et al. (188) also presented a MDO problem where the energy sources (fuel and batteries) were sized according to an optimal EMS.

Overall, both Sahoo et al. (179) and Brelje and Martins (192) recognise the need for more tightly integrated MDO frameworks for HE aircraft design. The challenging technological limitations caused by the electrical components require optimised designs with small margins. The traditional approach of sequentially optimising the components after fixing the system design would produce designs that require unachievable technological levels in the medium timeframe. Hence, both Set-based Design and Multi-Disciplinary Optimisation seem ideal approaches to be applied in the design of HE aircraft.

2.5 Summary of Review

This chapter provided the background of the central tenets that this dissertation covers.

The complexity of the design of modern systems has increased exponentially, requiring new methods to manage them effectively. Failure to do so causes rework and iterative procedures which correct mistakes found during the design process. These impact the cost and time of the development.

Set-Based Design is a methodology proposed to avoid rework and explore the design space effectively. It suggests delaying freezing design configurations and restricting the search by eliminating unfeasible options. While it has been demonstrated effective, there is still no definite quantitative approach to SBD design. The most common approach is reasoning with intervals and selecting the designs based on multi-attribute value theory.

Design uncertainty, caused by a general lack of information, is responsible for rework and changes mid-course. In complex systems, changes can propagate in a self-reinforcing cycle, causing more rework and more changes. As a mitigation mechanism, margins absorb the shocks caused by changes without propagating rework. SBD acts similarly to margins in accommodating changes without propagation but with the advantage that

variability is accounted for before the design is selected. In other words, alternatives are intrinsically explored.

Complexity entails many interacting components and subsystems, each with its design parameters. Engineering optimisation is a mathematical approach to finding the design parameters that satisfy the design requirements. Multi-Objective optimisation (MOO) problems are helpful in identifying the design alternatives that satisfy the same constraints and objectives while prioritising one over the others.

Multi-disciplinary optimisation (MDO) aims to apply the MOO methods to systems composed of multiple interacting components, intending to produce a design that performs better than one with components optimised sequentially. Coupled models have a significantly higher computational cost than single-discipline optimisation problems. Surrogate modelling methods are employed to alleviate this and enable higher fidelity analysis. MDO also produces high dimensional data, which can be effectively visualised with parallel coordinates and scatter plots.

Hybrid-electric aircraft is a proposed concept for enabling sustainable aviation and achieving the target emissions reduction. Many conceptual designs have been produced in the last ten years, but technological limitations hinder their practical development. Alongside the need for improved energy storage solutions, better electrical equipment and advanced thermal management, advanced design and optimisation tools are needed for successfully managing the design complexity. For example, the interaction between energy management and the system design is still not fully understood.

With these in mind, Set-Based Design and Multi-Disciplinary Optimisation are proposed as a solution to the design exploration of hybrid-electric aircraft. They are complementary: SBD can manage epistemic uncertainty and produce thorough design space exploration, while MDO generates designs that satisfy the requirements considering the interactions between the components.

Part II

The PDOPT Framework

Chapter 3

Conceptual Ideas of the PDOPT Framework

3.1 Introduction

The literature review of Chapter 2 highlighted engineering design challenges. Rework iterations and lack of knowledge are difficulties engineers face when making design decisions, especially during the conceptual stages. These are exacerbated by the pressing need to test and implement new technologies such as electrification, hydrogen storage, and urban air mobility. While labelled the ”third revolution” in aviation (193), there is still limited knowledge on the practical design of aeroplanes with these systems.

New design methodologies are required to cope with this scenario. The necessary characteristics to solve these problems are rapidity, comprehensiveness, and the ability to cope with uncertainty. Furthermore, unfruitful areas of the design space should be identified as soon as possible in the project and eliminated, reducing the pool of solutions that would lead to a ”rework trap”. Set-Based Design does provide a partial solution, but it is limited by its computational cost and the expert-driven set elimination process. This last aspect could increase uncertainty when working with new, untested technologies. Therefore, minimising the need for external assumptions is fundamental to reducing uncertainty.

This chapter will present and explain the key ideas of the thesis. These concepts cover the aims and research questions proposed in Chapter 1 and provide the theoretical backbone of the framework developed in this thesis.

3.2 Design Space exploration through elimination

Classical engineering design focuses on producing one or a few candidate solutions which are refined as much as possible to meet the requirements put forward by the customer. As discussed previously, this approach is sustainable when a previous solution is evolved and gradually adapted to new conditions. It has been the standard approach adopted in the aeronautical industry, as it is the safest from a certification perspective.

Evolutionary approaches are no longer suitable when disruptive technologies need to be introduced in a new revolutionary design. Selecting a suitable starting point without

experience is often time-consuming and prone to corrective rework. The leading cause of design failure is the inability to satisfy a requirement, often a feasibility one. For example, in a wing design context, a designer might select a guess value of L/D, which requires a very high aspect ratio or a low thickness aerofoil: both could conflict with the structural requirement of strength and stiffness. As structural design is often a preliminary design process, this issue would arise if not caught early on and require a rework loop back into conceptual design to change the original assumption. The consequence would be that other design aspects would change further, for instance, the predicted range through the Breguet equation.

Such pitfalls can be avoided by scanning the design space to identify where all requirements are satisfied sufficiently and simultaneously. This procedure requires a form of reasoning that is interval-based or set-based. As initially argued by Alan Ward, the creator of set-based design (28, 30), it is more robust and easier to solve the complementary form of this question. In other words, searching for areas of the design space that *do not satisfy all requirements*. This approach is similar to how computers check the truthfulness of conditional statements in programmes. Suppose a logical proposition contains several predicates connected by "AND" logical connectors. In that case, it is cheaper computationally to check if any of them is false rather than checking the truthfulness of every one of them.

Once these unfruitful areas of the design space have been identified, it is possible to discard them and reduce the degrees of freedom of the problem. This approach effectively reduces the uncertainty in the entire design problem. First, it was ensured that the requirements considered would be at least satisfied. Second, the total computational cost is decreased as the number of alternatives has been reduced. As discussed later in this chapter, optimisation is used to locally search the surviving design spaces and recover the design points.

From the engineering perspective, requirements are often encoded in two forms: constraints and objectives. Constraints are mathematical inequalities that must be satisfied for a successful design, hence encoding feasibility requirements. These can be external constraints due to customer or regulator needs and expectations or internal constraints due to component or subsystem compatibility. An example of the first would be the aircraft range being larger than a minimum value, while the second type could be geometric constraints on the wing aerodynamics caused by the size of a wing fuel tank. Often a feasibility requirement may manifest as a technological limitation that makes or breaks the concept under design. This is too encoded as a constraint on the QoI representing the technology, for instance, the energy density of a battery pack. Objectives are quantities the customer desires to be maximised or minimised: these encode desirability requirements. For example, aircraft fuel consumption should be minimised as much as possible to reduce operating costs and maximise environmental compatibility.

Although objectives usually come into play when defining an optimisation problem, it is possible to convert them into constraints if the stakeholders define a minimum level of satisfaction for such an objective. As an example, the TLARs of the FUTPRINT50 project state that the new HE regional aircraft should at least reduce NO_x emissions by 90% and CO_2 emissions by 75% compared to a representative aircraft from year 2000 (194). Therefore, these objectives can be cast into constraints for further reducing the design space towards better performance or, conversely, exploring if the assumptions of the design model

could ever achieve such performance. Within the context of the PDOPT framework, this type of constraint is labelled "soft" because it encodes a desirability requirement that does not determine the feasibility of the design. In contrast, feasibility-based constraints are named "hard".

This procedure of finding the promising area of the design space through elimination is called within PDOPT "exploration phase". The name highlights that this phase identifies a broad direction for the desired solution. The mathematical procedure to achieve it relies on probabilistic logic and surrogate modelling. Philosophically, these stand on knowledge representation in a simulation model and the Bayesian interpretation probability. Both aspects are discussed next.

3.3 Surrogate Models for knowledge extraction

The limitation of the ADOPT framework developed by Georgiades (4, 5) was using logical rules to eliminate undesirable and unfeasible configurations. These rules originated from expert knowledge about the system under design, a fuel system of a jet aircraft. As this design problem does not present any specific innovation, eliciting expert knowledge is reasonable and easy to implement. Nevertheless, it may be difficult for novel configurations and technologies where experts do not yet agree on best practices.

ADOPT presented two forms of knowledge: the knowledge of the rules to perform the set reduction and the knowledge that describes the system's behaviour under design contained in the evaluation function of the optimisation problem. Both pieces of knowledge originate from the description of the same physical object. It is reasonable to argue that the assumptions and limitations of the simulation model are also present in the expert-made ruleset.

Assume the design of a novel system, for instance, a hybrid-electric aircraft. Given its low TRL, the only possible source of knowledge is analogies with conventional aircraft design or the basic physics of the system components. Much like the test case of ADOPT, the assumptions must be the same, as physical laws determine the behaviour of engineering systems. However, the lack of best-practise knowledge due to the low TRL could lead to biased or incorrect expert judgements, causing incorrect selections and eliminations of the design space. Therefore, extracting the embedded knowledge in the simulation model is best as it would be closer to the truth of the system behaviour.

The knowledge that the system model contains is the mathematical relationship between the input parameters and the quantities of interest (QoI). These values are either subject to constraints or to be optimised. It represents the object's behaviour to be designed, limited to the designer's interest and the level of detail required. Capturing this information with a simpler mathematical model enables us to infer the design space. Indeed, it would allow one to invert the relationship between Inputs-QoI to evaluate constraints.

Machine learning is used to extract knowledge through data sampling and regression. Data are extracted by sampling the entire design space and producing a dataset containing N pairs of (y_{QoI}, x_{inp}). Statistical learning theory ensures that given a suitable approximation function $f(x)$, it is possible to find the combination of hyperparameters that are most probable given the observed data (53). The obtained model, therefore, "learnt" the relationship from the data and can generate new outputs from unobserved inputs. In essence,

it transferred the knowledge embedded in the design model into a statistical relationship that can be used for rapid testing.

Now, unlike surrogate-based optimisation, the exploration phase is not interested in high precision. Instead, the surrogate model should capture the uncertainty of the prediction so that it can be used with probabilistic reasoning and does not require excessive data. For this purpose, Gaussian Processes (GP) have been adopted. This mathematical object describes the distribution of possible functions consistent with the observed data (195). The assumption is that the probability distribution of observing the output values of the training set $P(f(x_1), f_(x_2), ..., f(x_N))$ is a multivariate Gaussian distribution with a mean vector $\mu(x)$ and a covariance matrix $\Sigma_{i,j} = k(x_i, x_j)$ where k is a positive definite kernel function.

Predicting a new value $f(x_*))$ consists in finding the output value that maximises the conditional probability $P(f(x_*)|x_*, x, f(x))$, which in turn is to evaluate the mean vector and covariance matrix of the multivariate Gaussian distribution:

$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} K & K_* \\ K_* & K_{**} \end{pmatrix}\right) \tag{3.1}$$

where $K_{ij} = k(x_i, x_j)$ is the covariance matrix of the training data, $K_{*,i} = k(x_*, x_i)$ is a vector containing the covariance between the new observations and the training data, and K_{**} is the covariance of the new observations. The kernel calculates the similarity between two points, as it is reasonable to assume that points close to the function domain have similar codomain values. GPs are trained by selecting a suitable kernel function and finding the hyperparameters that maximise the likelihood of the training data. The output of a GP prediction is the expected mean value of the output along the variance, which forms a credible interval. The distribution of possible predictions for the new observation x_* is assumed to be Gaussian, which will be necessary to calculate the probability of satisfying the constraints.

3.4 Bayesian statistics for requirements mapping

Implementing set elimination as discussed in Sec. 3.2 requires mapping a multivariate continuous design space to a discrete Boolean operation. Furthermore, all requirements must be satisfied to avoid set elimination. Hence, the evaluation of the requirements must be fused into a single criterion. The uncertainty present in this problem lies in the location of the decision boundary for set elimination: a form of epistemic uncertainty.

Bayesian statistics is proposed as a suitable mathematical tool for modelling this problem. Its fundamental assumption is in the interpretation of the concept of probability. Rather than quantifying the frequency of a phenomenon, probability quantifies the expectation of a state of knowledge being true. This interpretation allows one to extend propositional logic enabling them to reason with a partial or imperfect state of knowledge.

The starting point in applying Bayesian reasoning to set-based design is reformulating requirements as statements. As discussed previously, constraints expressed as inequalities encode requirements mathematically, such as:

$$g_i(x) \leq \bar{G}_k \tag{3.2}$$

where $g_i(x)$ is a function that maps the input parameters to the QoI to be constrained, and G_k is the limit value of the constraint. This inequality effectively defines a decision boundary in the input domain, whether satisfied or not. Figure 3.1 shows a representation of this in a 2D input space.

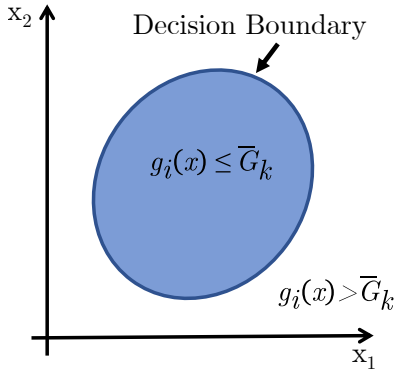


Figure 3.1: Decision Boundary of the constraint.

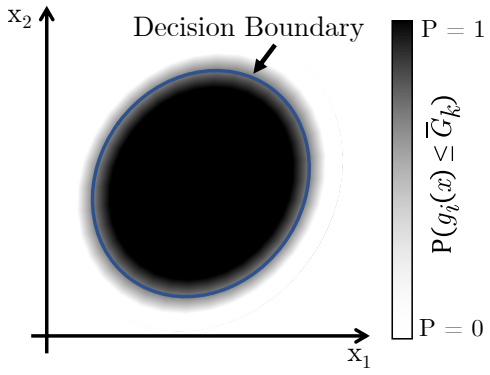


Figure 3.2: Probability of the constraint being satisfied.

This statement is now reformulated as a probabilistic proposition, answering the question, "What is the probability that the constraint is satisfied?". Mathematically, it can be expressed as:

$$P(g_i(x) \leq \bar{G}_k) \tag{3.3}$$

whose value, according to the Bayesian interpretation of probability, corresponds to the certainty that the constraint is satisfied. The advantage of this formulation is twofold. First, it maps a continuous space to a continuous quantity while expressing a discrete yes/no statement. Second, it enables synthesising different requirements under a single elimination criterion. This process utilises probabilistic logic, which encodes the logical connectors of AND, OR, and NOT through set theory. Satisfying all requirements corresponds to AND-ing all the probabilistic statements. This is done with the following expression, given a set of statements S_i:

$$P(S_1 \wedge S_2 \wedge S_3 ... \wedge S_i) = P(S_1) \cdot P(S_2|S_1) \cdot P(S_3|S_1, S_2) ... \cdot P(S_i|S_{i-1}, ..., S_1) \tag{3.4}$$

This general expression, while accurate, can be computationally intractable for a large number of statements. In particular, calculating the conditional probability of the following statements requires access to the full joint distribution. However, in most cases, the statements are conditionally independent; that is, the truthfulness of the statement is not affected by the verification of the others. Since the objective of the exploration phase is to eliminate unfeasible and undesirable candidate solutions roughly, the statements are assumed to be all conditionally independent. This assumption is known as the "naive Bayes" in the literature (53), which significantly simplifies (3.4):

$$P(S_1 \wedge S_2 \wedge S_3 ... \wedge S_i) = \prod_i P(S_i) \tag{3.5}$$

Using this formula, it is finally possible to calculate the probability that a design or a design space can satisfy (or not) all the requirements. The PDOPT framework uses this criterion for set elimination by defining a *minimum satisfaction probability* P_{sat}. A set is eliminated if its probability is below the P_{sat} threshold. The detailed calculation procedure is described in the next chapter. It suffices to say that the set probability is calculated using a Monte Carlo method: points are sampled and evaluated with the Gaussian Processes, their probability is evaluated, and the total number of satisfying points is counted over the total number of samples. This final ratio provides the P of satisfying the requirement in the set.

3.5 Optimisation for Design Exploration

Optimisation techniques are usually employed to refine a design once it has been frozen. Bounds for input parameters are provided by the previous design phases, usually in the form of margins, where the optimising algorithm searches for the best combination for marginally improving the design. In contrast, as discussed in Chapter 2, multidisciplinary optimisation enables us to optimise the design early, during the preliminary or conceptual phase. This would lead to a better system design, avoiding the limitations of sequential optimisation.

However, the utilisation of optimisation problems in PDOPT differs from the conventional one. The aim is to explore the available designs within a pocket of design space, assuming that the engineers would produce the best possible design. The result is a family of local Pareto fronts composed of many globally suboptimal points. This requires using population-based approaches such as Genetic Algorithms or Particle Swarms. Furthermore, the convergence criteria are modified to avoid early stopping caused by tolerance convergence on either the functional evaluation or the input parameters. Instead, the maximum number of functional evaluations or generations guarantees a fair comparison between the optimisation outputs from each set.

Optimisation is used to carry out computational experiments in the design space. Instead of seeking a single optimal solution, multiple points are recovered from each set. The results from each set are aggregated and stored in a database, yielding the global Pareto front with multiple suboptimal design points.

With this abundance of information, one can deduce valuable insights about the system and formulate new hypotheses regarding its design. These insights can then be utilised to modify and improve the analysis setup. Furthermore, the optimality condition makes it possible to study the trade-off between the objectives and how the input parameters and constraints affect them. Engineers design their systems with the best achievable performance; hence the decision-making process expects optimal design points. With many points, it is possible to understand the implications of each input parameter and identify the desired combination of them for the requirements set by the designers.

3.6 Conclusions: the PDOPT framework

This chapter presented the ideas that function as a backbone to the Probabilistic Design and OPTimisation (PDOPT) framework. Figure 3.3 highlights how these ideas are incor-

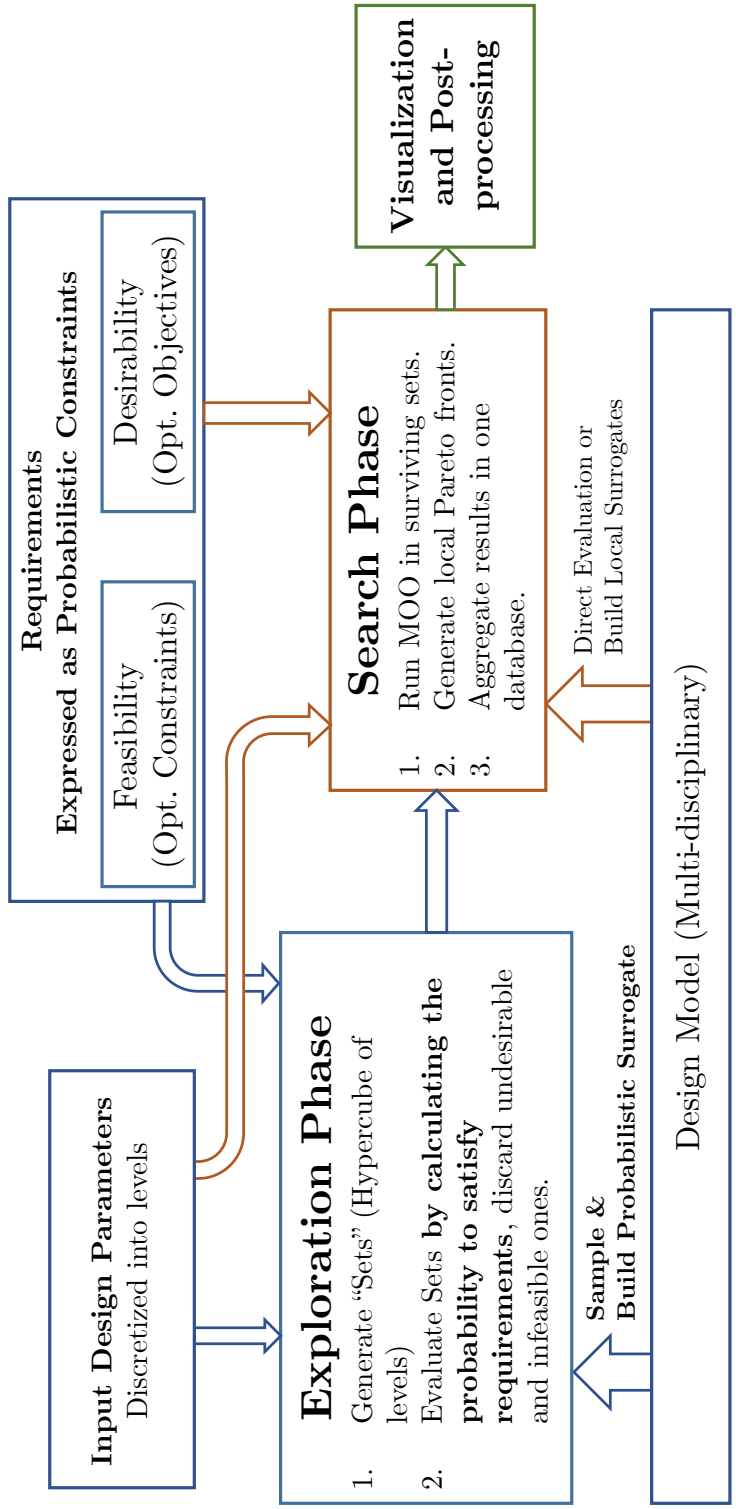


Figure 3.3: Framework Structure with ideas highlighted.

porated into the framework.

The Exploration Phase implements the SBD procedure for set generation and elimination. Input design parameters are discretised into levels and combined to generate the sets. These are pockets of the continuous design space, whose designs originate from a combination of intervals of the input parameters.

The elimination of unfeasible sets is carried out by estimating the likelihood of meeting the requirements. Here, the two key ideas of Bayesian statistics and knowledge extraction through surrogate modelling come into play. First, requirements are stated as probabilistic constraints: the confidence level that a requirement is satisfied in a set. All the optimisation constraints are converted with the addition, if present, of the ”soft constraints” on the objectives.

The next step is to build the surrogate model to calculate the probabilities. Data is sampled from the design model and used to train the Gaussian Processes, one for each quantity of interest subject to the constraints. This process extracts the knowledge from the full model and encodes it within a probabilistic surrogate. The sets are evaluated by sampling points in each of them, evaluating the mean and variance of the QoIs with the GPs, and calculating the probability that the constraints are satisfied at that point. The aggregated set probability is obtained with a Monte Carlo approach. The final probability is calculated by ANDing all the evaluations together.

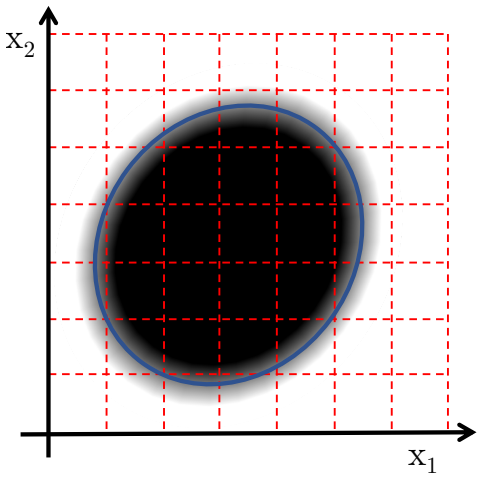


Figure 3.4: Fuzzyness and ambiguity of the decision boundary when the design space is divided into sets.

Using probabilities rather than Boolean quantities provides a fuzziness that makes it possible to handle those sets crossed by the decision boundary. As shown in Fig. 3.4, the decision boundary causes ambiguity in most sets it crosses. Rule-based set elimination cannot handle these cases, as they are based on Boolean logic. Instead, Bayesian probabilistic logic enables handling cases where requirements are partially satisfied. These edge sets would be retained for the next phase, where the optimiser could recover some design points that would otherwise be lost.

The Search Phase carries out optimisation to obtain the design points in each surviving set. The advantage is twofold: the total computational cost is reduced compared to running the optimiser in every set, and more information is extracted from the design space

compared to running a global optimisation. Results from this phase are aggregated into a database organised by set id, ready for being post-processed or visualised. Parallel coordinates diagrams, as discussed in Sec. 2.3.3, are suitable for analysing and understanding the results, as these are multi-dimensional. Using an interactive visualisation tool incorporating both parallel coordinates and scatter plots complements the framework, enabling a better presentation of the results for decision-making.

This chapter covered the key ideas behind the PDOPT framework. The next one will detail the computational implementation and outline the practical use of the tool.

Chapter 4

Framework Computational Implementation

4.1 Introduction

This chapter discusses the computational implementation of the ideas discussed in Chapter 3. The code has been developed in Python 3.9 using only well-maintained libraries in the same language. No other languages or external libraries have been used to avoid difficulties in maintaining the codebase. The code has been made available publicly on a Github repository (196), along with a manual and example files.

Figure 3.3 presents the flow chart of the code, with the section references. The PDOPT library is composed of three objects: the `DesignSpace()` data structure which encapsulates the design space divided into sets; the `ProbabilisticExploration()` object which implements the Exploration step; and the `Optimisation()` object implementing the Search step. Two input files define the input parameters and the QoIs affected by the constraints and objectives. The information flows from the `DesignSpace()` to first the Exploration phase and then to the Search phase, returning the results as Pandas data frames back to the `DesignSpace()` object. These results are then exported to the output files using the Pandas export interface.

The design of the PDOPT framework as a Python library grants a high level of flexibility in its application. The two steps can be run asynchronously and independently of each other. For instance, multiple exploration phases can be run under different requirements, for example, to study their effect on the design space. This chapter will present a canonical example, but the user can experiment with the tools provided.

In addition, `ProbabilisticExploration()` supports reading sampled data from other sources using the `samples.csv` file. This usually is automatically generated from the provided evaluation function. The user can edit it to introduce data from experiments or higher-fidelity simulations. Similarly, `Optimisation()` supports two forms of surrogate optimisation: Gaussian Process Regression and Neural Networks. The user can activate these if the original evaluation function is computationally too expensive or presents instabilities in some areas of the design space.

The structure of the chapter is as follows. The first half covers the detailed implementation of the two steps, discussing the mathematical process and the Python libraries adopted. The second part describes the case setup, including the required input files and

Figure 4.1: Computational flowchart of the PDOPT framework.

the API of the evaluation function.

4.2 Design Space Representation

PDOPT is designed using an object-oriented programming approach. Each mathematical object in the framework (parameters, objectives, constraints, etc.) is modelled using classes and objects. The file `data.py` contains their implementation. These are:

- **Parameters**. These objects model the input parameters. Two classes are available, `ContinousParameter` and `DiscreteParameter`. Both encapsulate the name of the parameter, the bounds, and the number of levels. Continuous parameters can return the level bounds with a method, while discrete parameters only contain the number of levels. It is assumed that the evaluation function would translate the discrete level into an actual internal representation (e.g. a configuration).
- **Objectives**. Representation of an objective as defined in the `responses.csv`. It encodes the name of the output, the type of objective (”minimise” or ”maximise”), and, if present, the ”soft constraint” to be used in the exploration phase.
- **Constraints**. Representation of a constraint as defined in the `responses.csv`. It contains the name of the output, the type of constraint (”greater than” or ”less than”), the constraint value, and the satisfaction probability level. Its methods provide manipulation of the information and check the constraint satisfaction.
- **OptimisationProblem**. Encapsulation for PDOPT of the `pymoo.Problem` object.
- **DesignSet**. The representation of a set in the PDOPT framework. It is defined with a unique ID, the levels of the input parameters, the probability of satisfying the constraints, the discarded status, and the optimisation problem.
- **DesignSpace**. The main data structure encapsulates the whole design space. When instantiated, it builds the hypercube of parameter levels and stores each set inside a list of DesignSet objects. Methods are provided for producing the pandas DataFrames results of both the exploration and search phase.

Details of the classes can be found in the PDOPT manual and in the code itself. The objects have interface methods that can be used to set and retrieve the information stored in them. Furthermore, the user can easily inspect the contents with `print()` for the parameters, objectives, constraints, and design set objects. An example listing of `DesignSet()` objects is shown in Listing 4.1, showing the level of input parameters, if the set has been eliminated, and its total probability.

```
>print(design_space.sets[4:7])
[DesignSet 4 [ ], Parameters: {'climb_h0': 0, 'climb_h1': 0, 'cruise_h0': 1, '
  cruise_h1': 0} P = 0.990,
 DesignSet 5 [ ], Parameters: {'climb_h0': 0, 'climb_h1': 0, 'cruise_h0': 1, '
  cruise_h1': 1} P = 0.510,
 DesignSet 6 [X], Parameters: {'climb_h0': 0, 'climb_h1': 0, 'cruise_h0': 1, '
  cruise_h1': 2} P = 0.000]
```

Listing 4.1: Example inspection of `DesignSet()` objects

4.3 Step 1: The Exploration Phase

As described previously, the exploration phase implements the set-based design ideas augmented with Bayesian probability. Its scope is to survey the entire design space and eliminate those sets that do not satisfy the requirements set by the user.

The computational process comprises three main steps: training the required surrogate models, sampling the sets, evaluating the points, and calculating the total probability of the set. Figure 4.2 shows this process in a flowchart.

Training of the surrogate processes begins with sampling the entire design space to generate the training dataset. The quasi-random Sobol sequence is used, with 128 points, to ensure optimal filling (197). For the validation dataset, Latin Hypercube Sampling (LHS) (198) is used instead, with 32 points. The `joblib` (199) parallelisation library is used to speed up the process, and the training dataset is stored in the `samples.csv` file for future use. The data is scaled to the $[0,1]$ range to improve the convergence of the training algorithm. The inputs and outputs are automatically scaled when calling the surrogate model for prediction.

The Gaussian Process implementation is from the `scikit-learn` machine learning library (200), which also contains dataset normalisation and model validation tools. After training, the user can inspect from the console output the Pearson R^2 correlation coefficient to evaluate the quality of the model. GPs are single output functions; hence one is trained per QoI. The kernel adopted is the Matern kernel with the standard coefficients of `scikit-learn`. Matern can handle more irregular data than the commonly used radial-basis kernel.

Once the GPs are trained, the probabilistic evaluation can start. Each set j is sampled with LHS, generating N_j of candidate points X_k^j (100 by default). For each constraint i, these are passed through the corresponding GP function, generating the mean value $\mu_{y,i}^k$ and variance $\sigma_{y,i}^k$ at the sample point of the QoI y_i. Given the constraint boundary $\bar{g}_i$, it is possible to calculate the probability $P^k(y_i < \bar{g}_i)$ that it is satisfied at the sampled point using the Gaussian cumulative distribution function Φ:

$$P^k(y_i < \overline{g_i}) = \Phi\left(\frac{\overline{g_i} - \mu_i^k}{\sigma_i^k}\right) \tag{4.1}$$

The equation calculates the area under the Gaussian probability distribution curve from $-\infty$ to the constraint value $\overline{g_i}$. Figure 4.3 shows this process. In case of a "greater than" constraint, the function's argument Φ in (4.1) is reversed. The framework automatically performs the necessary algebraic manipulations.

The set probability of the requirement $P_{i,j}$ is calculated using a Monte Carlo approximation. In the definition of the problem, each probabilistic constraint is defined with a minimum satisfaction probability $\bar{P}_i$. This quantity then is compared to the probability of each sampled point, obtained using (4.1). $P_{i,j}$ is finally calculated by counting how many of the sampled points satisfy the constraint ($n_{i,sat}$) and dividing it by the total number of sampled points:

$$P_{i,j} = \frac{n_{i,sat}}{N_{samples}} \tag{4.2}$$

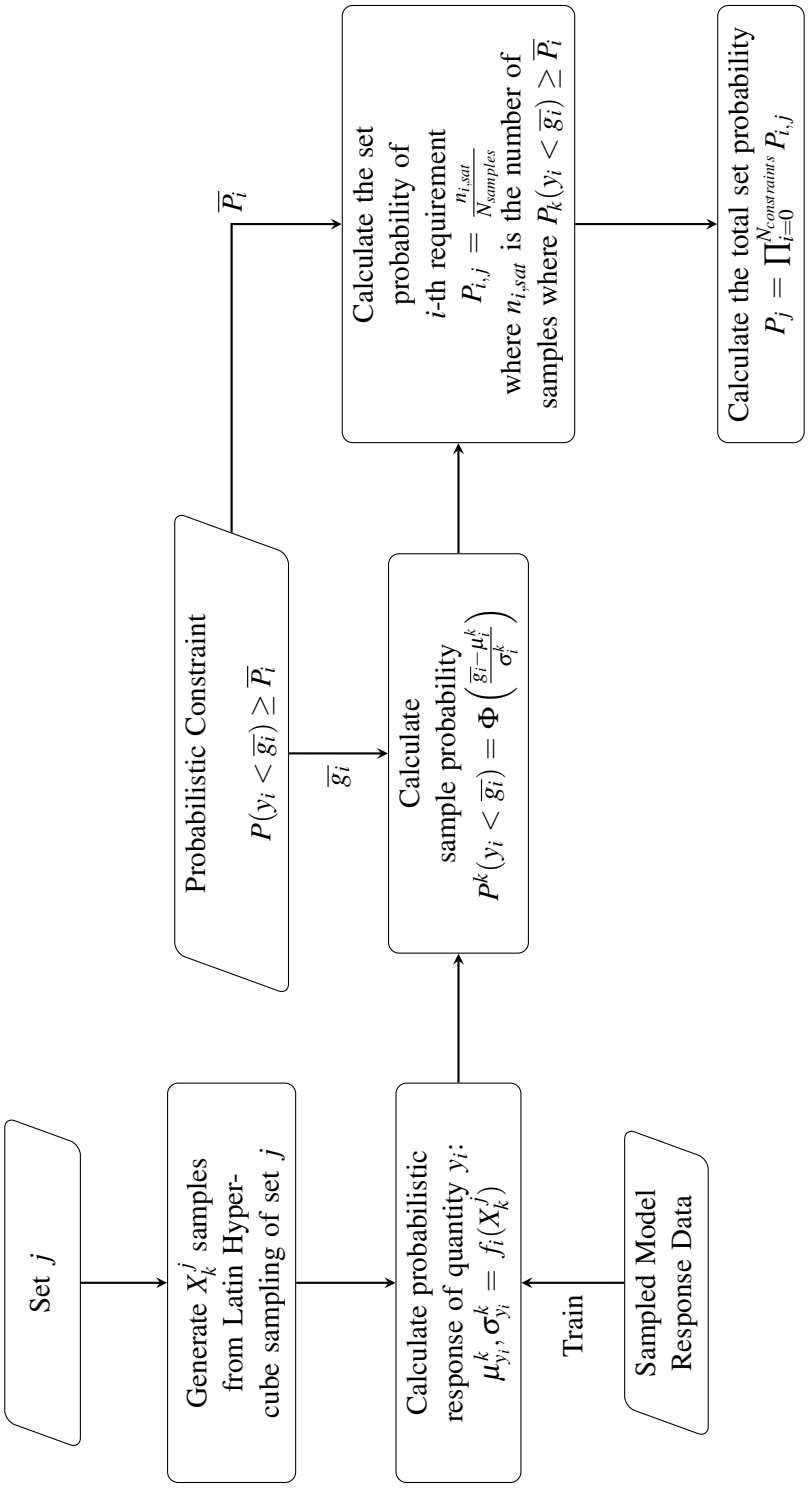


Figure 4.2: Detail of the Exploration phase process

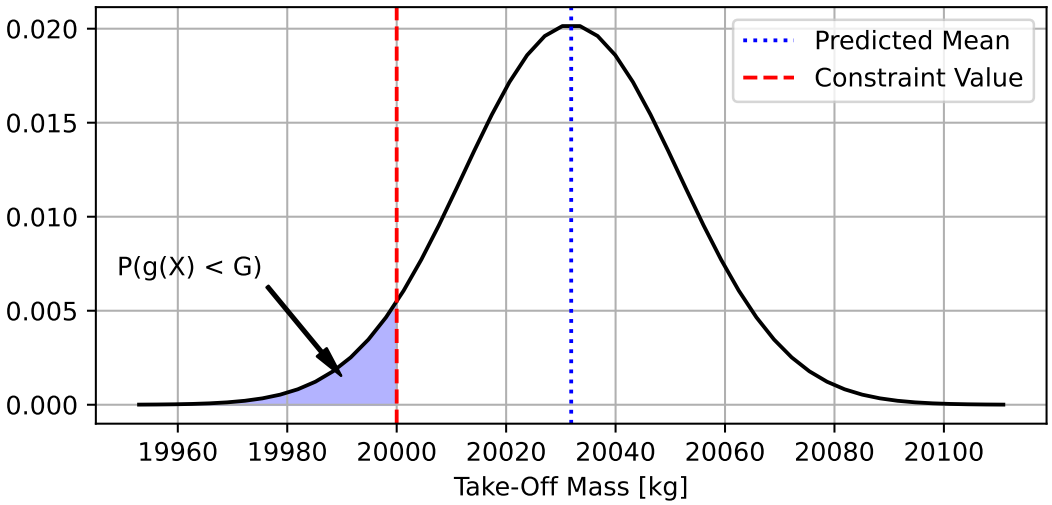


Figure 4.3: Probability of satisfying a constraint $\overline{g_i}$ for a sampled point X_k^j.

Finally, the total set probability is calculated by multiplying all the requirement probabilities. The value is normalised by a factor k such that the probabilities of all sets are rescaled between 0 and 1:

$$P_j = \frac{1}{k} \prod_{i=0}^{N_{constr}} P_{i,j} \tag{4.3}$$

Sets are discarded if the total probability falls below the required minimum. The user specifies this value at the time of execution and is independent of $\bar{P}_i$ of the requirements. Instead, it acts as a filter for the fuzziness of the constraint boundary. Setting it closer to 1 would produce a more crisp division but might eliminate sets partially crossed by the constraint boundary, potentially losing feasible data points.

After the exploration phase, the user can retrieve the results in a `pandas.DataFrame` using the method `DesignSpace.get_exploration_results()`.

4.4 Step 2: The Search Phase

The search phase implements the ”Optimisation as a Discovery Tool” idea discussed in Sec. 3.5. The surviving sets are explored with the optimisation algorithm to extract the design points.

The Python library `pymoo` (201) is adopted for this process. This library implements the interface for defining a multiobjective optimisation problem and popular and robust algorithms to solve them. For the scopes of PDOPT, the algorithm should be population-based but capable of handling problems spanning from single-objective to many-objective. Additionally, it should be gradient-free, as the evaluation function may not be differentiable, which is the typical case for multidisciplinary models. Hence the Universal Non-dominated Sorting Genetic Algorithm III (U-NSGA-III) was selected (118).

The drawback of using population-based methods is the large number of evaluations of the objective functions, which grows significantly with the number of input parame-

ters. For this reason, PDOPT presents the option to use a locally trained surrogate model when performing the set optimisation. Two types of models are supported: Gaussian Processes (Kriging) and Neural Networks. It can be activated by passing the parameter `use_surrogate=True` to the `Optimisation()` object. The algorithm recovers the actual output from the found Pareto set after the optimisation cycle when using the surrogate models.

Both are non-parametric for guaranteeing generality. The training process follows the same procedure for training GPs in the Exploration phase (Sec. 4.3): the set is sampled using the Sobol sequence for the training data and LHS for the validation data. Scaling is also introduced for better convergence.

The kernel of the Kriging model is fixed, a combination of Radial Basis Function (RBF) kernel with two constants:

$$k(x_i, x_j) = a \cdot exp\left(\frac{d(x_i, x_j)}{l^2}\right) + b \tag{4.4}$$

where a, b are constants, l is the RBF length scale and $d(x_i, x_j)$ is the Euclidean distance of the two points. During training, an optimiser finds the best combination of hyperparameters minimising the regression error. RBF has been preferred in this application as the function response is locally smooth for most problems.

On the other hand, the structure of the NN model is parametrised and tuned using a randomised search method. The algorithm generates 100 random candidates with hidden layers ranging from 1 to 4 and a number of neurons ranging from 50 to 200. The search finds the combination that minimises the mean squared error of the validation dataset.

It has been observed that the Kriging model generally has the lowest mean square error, and it should be preferred in most applications. It is the default model when `use_surrogate=True` is passed. The Neural Network model may be helpful if the response surface presents strong discontinuities that cannot be captured with an RBF kernel. If the user prefers to use the NN model, the parameter `use_NN` should be set to `True` along with the parameter activating the surrogate modelling.

The general process of the search phase is as follows. First, in each surviving set, the optimisation problem is set up. Set bounds are converted into bounds for the continuous parameters, while the discrete ones are held constant, as the value is uniform over the whole set. Constraints are converted into the standard form $G(x) < 0$, since `pymoo` evaluates constraint violations if the constrained value is greater than 0. After the setup, each problem is run sequentially. The optimiser uses the `joblib` library to parallelise the function evaluations at each iteration. The results of each problem are stored inside the `DesignSpace()` object in case the programme is interrupted before completion. As mentioned above, if a surrogate model is selected, the code automatically samples and trains the surrogate before running the optimisation problem. Finally, at the end of the process, the results are extracted from each set and aggregated into a single `pandas.DataFrame`, accessible with the method `DesignSpace.get_optimum_data()`.

It is possible to control the UNSGA-III parameters by passing Python `kwarg` arguments to the `Optimisation()` object. These are shown in Tab. 4.1 along with their default value. As discussed in Sec. 3.5, the convergence tolerances are set to a very small number to disable them effectively. The user should instead use a maximum number of generations or function evaluations to set the stopping criterion to guarantee a fair com-

Name	Description	Default Value
pop_size	Number of population points	10 + Number of Reference Directions (calculated with the Das-Dennis method)
n_partitions	Number of partitions of Pareto front	12
x_tol	Input parameter convergence tolerance	$1 \cdot 10^{-16}$
cv_tol	Constraint violation convergence tolerance	$1 \cdot 10^{-16}$
f_tol	Functional evaluation convergence tolerance	$1 \cdot 10^{-16}$
n_max_gen	Maximum number of generations	1000
n_max_evals	Maximum number of function evaluations	$1 \cdot 10^{6}$

Table 4.1: Optimisation `kwarg` parameters.

parison between the set results.

4.5 Postprocessing: Visualisation and Decision Making

The framework was developed with generic I/O to integrate it with decision-making or visualisation tools. However, a bare-bones visualisation web tool is included in the software distribution in the `visualisation.py` script. It uses the `plotly` and `dash` Python libraries to interactively visualise any.csv file outputted by PDOPT. A screenshot of the tool is shown in Fig. 4.4.

The user is presented with a Parallel Coordinates plot and five scatter plots. It is possible to select which dimension of the dataset to use for the XY axes or the colour scheme. Furthermore, selecting a set of points within the Parallel Coordinates and propagating them to the scatter plot is possible. Interactivity is vital to effective decision-making when dealing with complex datasets. Although this tool is helpful in this context, the library `matplotib` was used instead to generate the static images used in this dissertation.

4.6 Problem Definition: Files and Code Structure

PDOPT is primarily a Python library. It is designed to be stand-alone inside a test case script or incorporated into larger frameworks. The minimum amount of files required to perform a PDOPT analysis successfully is:

- A Python script to import the PDOPT library, define the evaluation function, and set up the workflow.
- A .csv input file describing the input parameters (`inputs.csv`).

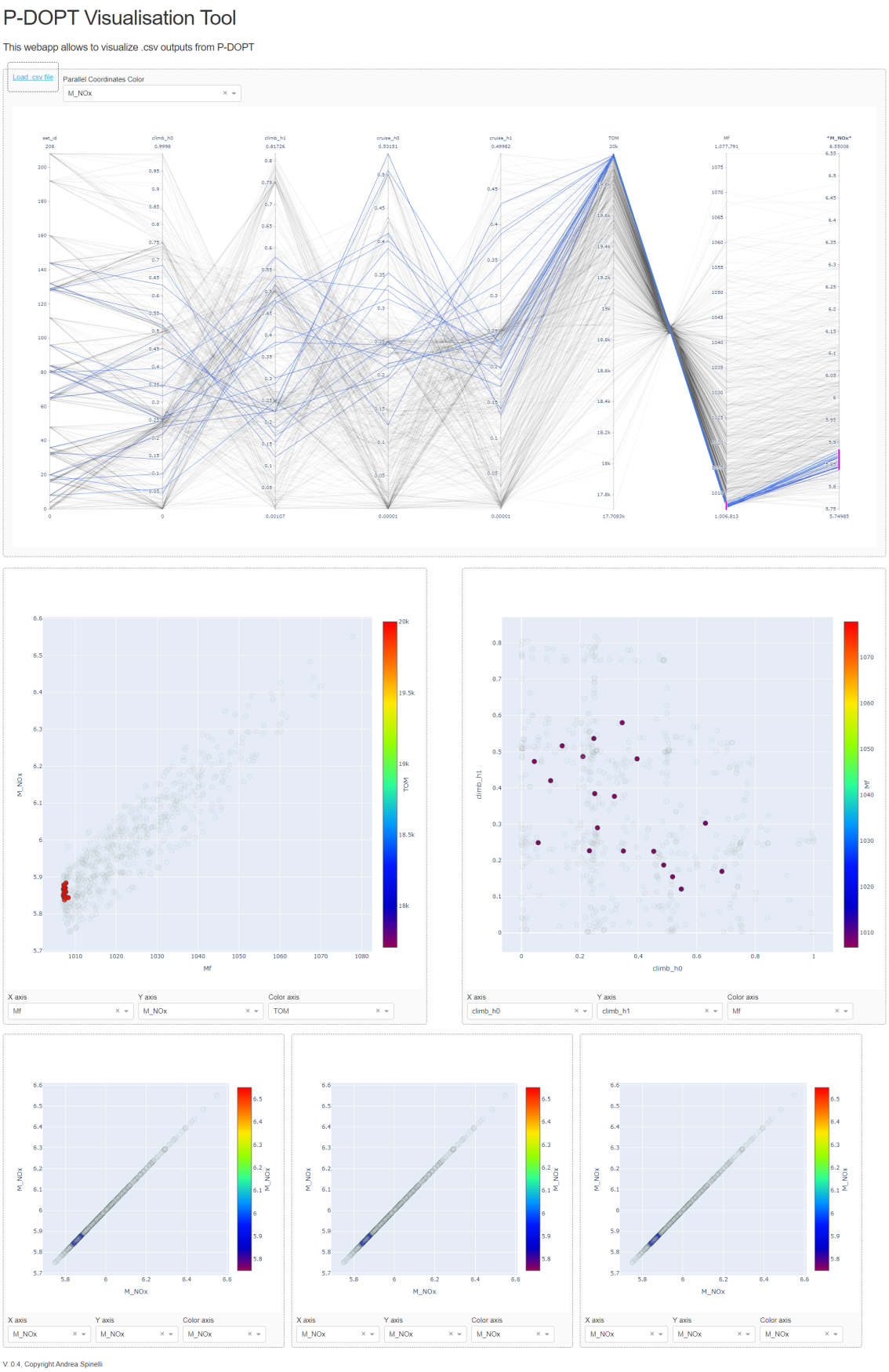


Figure 4.4: Visualisation Tool Example

- A .csv response file, describing the QoI (`response.csv`).

The structure of the script run file follows the two-step procedure outlined in Fig. 4.1. The components required to be imported from the `pdopt` library are:

```
from pdopt.data import DesignSpace, Model
from pdopt.exploration import ProbabilisticExploration
from pdopt.optimisation import Optimisation
```

As described before, `DesignSpace()` is the data object containing all the information about the problem (parameters, QoI, sets). In contrast, the other two objects implement the exploration and search phases. Given the required input files, a minimal workable example is shown in Listing 4.6, assuming the evaluation function is defined as `my_model()`.

```
# Wrapping the evaluatio function into a model object
my_model = Model(my_eval_fun)

# Definition of the design space
design_space = DesignSpace("input.csv", "responses.csv")

# Creation of the exploration phase object
exploration  = ProbabilisticExploration(design_space, my_model)

# Run exploration phase
n_exploration_samples = 100
P_exploration_sat = 0.5
exploration.run(n_exploration_samples, P_exploration)

# Get exploration results
df_exp_results = design_space.get_exploration_results()

# Creation of search phase object
optimisation = Optimisation(design_space, my_model)

# Run search phase
optimisation.run()

# Get search phase results
df_opt_results = design_space.get_optimum_results()
```

4.6.1 Definition of Input Parameters

The input parameters are defined in an ASCII .csv file (`input.csv`), one row per parameter and heading as shown in Listing 4.2. While any text editor can edit the file, a spreadsheet programme can be used to better visualise the tabular layout, as shown in Fig. 4.5.

```
name,type,lb,ub,levels,uq_dist,uq_var_l,uq_var_u
climb_h0,continous,0,1,4,nan,nan,nan
climb_h1,continous,0,1,4,nan,nan,nan
cruise_h0,continous,0,1,4,nan,nan,nan
cruise_h1,continous,0,1,4,nan,nan,nan
```

Listing 4.2: Input File

A	B	C	D	E	F	G	H
name	type	lb	ub	levels	uq_dist	uq_var_l	uq_var_u
climb_h0	continous	0	1	4	nan	nan	nan
climb_h1	continous	0	1	4	nan	nan	nan
cruise_h0	continous	0	1	4	nan	nan	nan
cruise_h1	continous	0	1	4	nan	nan	nan

Figure 4.5: Content of the input.csv file

The input parameters are distinguished as continuous and discrete. Discrete parameters are defined by N integers, mapped in the evaluation function to discrete possibilities, while continuous parameters represent a range of possible variables. The file is structured as follows:

- **name** – Name of the parameter. It must not contain spaces; use an underscore for it.
- **type** – If the variable is continuous or discrete. If discrete, the lb and ub parameters will be ignored, as only integer levels will be provided.
- **lb** – Lower bound of the continuous parameter.
- **ub** - Upper bound of the continuous parameter.
- **levels** – The number of parameter levels: it must be an integer.
- **uq_dist** – Type of UQ distribution for this parameter for uncertainty-based optimisation. It can be 'uniform', 'triangular' (triangular), or 'norm' (gaussian). Set it to 'nan' for ignoring. Uniform and Triangular distributions support asymmetric distributions, while Gaussian is only symmetric.
- **uq_var_l** – Lower variation bound for the distribution, expressed as a decimal percentile variation (i.e. 0.05 for 5% variation).
- **uq_var_u** – Upper variation bound for the distribution, expressed as the decimal percentile variation. Set it to 'nan' if the distribution is symmetric.

Each row in the .csv file must be filled in. If a variable is unused, it must be set to 'nan'. Due to the implementation of the framework, it is fundamental that the input parameters in the `input.csv` file are ordered in the same order as the evaluation function arguments. During any functional evaluation, PDOPT passes an array of inputs in the specified order assuming that it matches the expected order of the evaluation function arguments.

Some rows are used to specify an uncertainty quantification distribution to be used in uncertainty-based optimisation. While the bare bones are present in the current version of the codebase, it is not yet fully implemented. Hence, these rows are not used other than to set up the `ContinousParameter()` objects.

4.6.2 Definition of Requirements, Constraints, and Objectives)

The `responses.csv` file is used for defining the QoIs of the evaluation function that are taken into account by PDOPT. Listing 4.3 shows an example of its content. Much like input.csv, a spreadsheet programme can be used to assist with data entry and visualise the contents in a table format (Fig. 4.6).

```
name,type,op,val,pSat
TOM,constraint,lt,20000,0.5
Mf,objective,min,nan,nan
M_NOx,objective,min,nan,nan
Degradation,objective,min,nan,nan
```

Listing 4.3: Responses File

	A	B	C	D	E
1	name	type	op	val	pSat
2	TOM	constraint	lt	20000	0.5
3	Mf	objective	min	nan	nan
4	M_NOx	objective	min	nan	nan
5	Degradatic	objective	min	nan	nan

Figure 4.6: Content of the response.csv file

Unlike the input definition, the order of the entries does not affect the code's functionality. Instead, the user must use a unique name for each QoI: these have to match the keyword used in the return Python dictionary of the evaluation function (see Sec. 4.6.3). At least one entry per type (objective and constraint) must be present for the code to function, and every column must be filled. The file is structured as such:

- **name** – Name of the response from the evaluation function. These names must be used in the dict() object returned from the evaluation function.
- **type** – Type of response. It can be 'constraint' or 'objective'. These are the responses that the optimisation step will handle. In the exploration step, constraints are always active, while objectives can be set up as constraints or not (see next point).
- **op** – Operator on the response. If the type is set to 'constraint', the operator will be 'lt' (less than) or 'gt' (greater than), which represents the inequality of the response with respect to the quantity in the value column (that is, 'TOM, constraint, lt, 2000' corresponds to TOM ¡ 20000). If the type is set to 'objective', the operator is 'min' (minimise) or 'max' (maximise).
- **val** – Value which is going to be used by the operator. In the case of an objective, this value is used to set a constraint in the exploration phase with the following criteria: if 'min' is set as the operator, then it is less than the constraint; if 'max' is set as the operator, then it is greater than the constraint. This is done to drive the

exploration phase and to remove areas of the design space that might not satisfy minimum requirements. Set it to 'nan' to disable the constraint.

- **pSat** – The minimum satisfaction probability for the constraint/objective. When evaluating the sets, this is the minimum probability to which samples are tested. Samples that go under pSat are counted as unsatisfactory.

4.6.3 The evaluation function

Both the `ProbabilisticExploration()` and `Optimisation()` objects expect the simulation model to have the `.run()` method for running the evaluation function. The framework provides two approaches to connect the simulation model to the API: the class `data.Model()` and the `data.ExtendableModel()` abstract class. The first is a wrapper for the actual evaluation function. At the same time, the second is a template for defining a custom model object, useful if the simulation model requires a setup or parameters to be changed at the object creation (see the example in Listing 4.4).

```
class My_Model(ExtendableModel):

def __init__(self, model_parameters):
self.model_parameters = model_parameters

### Do something with the parameters to setup the simulation

def run(self, *args):

### Evaluation function, with variable inputs

```

Listing 4.4: Example of use of `ExtendableModel()`

In both cases, the evaluation function must be crafted so that the input and output match the format PDOPT expects. Input arguments are specified in the `input.csv` file in the order listed. Failure to do this will cause a mismatch of the input parameters. For instance, if the input file contains the parameters A, B, and C in this order, the evaluation function should be:

```
def my_eval_fun(A, B, C):

## Process the input and return the QoIs.
```

Listing 4.5: Evaluation function with hardcoded inputs.

It is possible to make the evaluation function more flexible by using `*args`. This creates a tuple object called `args` that contains the parameters passed, enabling handling a variable number of inputs. The user must hardcode the internal logic to handle the variable inputs. In the case of the previous example (Listing 4.5), the function can be rewritten using the `*args` argument:

```
def my_eval_fun(*args):

## Internal logic
if len(args) == 3:
A, B, C = args
```

```
6  else:
7  A, B, C, D = args
8
9  ## Process the input and return the QoIs.
```

Listing 4.6: Evaluation function with variable inputs.

The example unpacks the `args` object depending if it has a length of 3 or 4. The output of the evaluation function expected by the framework is a Python dictionary containing the values of the QoIs indexed by the name used in the `response.csv` file. An example is shown in Listing 4.7 assuming the QoIs are X, Y, and Z.

```
1  def my_eval_fun(*args):
2  ## Variable input example
3
4  ## Internal logic
5  if len(args) == 3:
6  A, B, C = args
7  else:
8  A, B, C, D = args
9
10 ## Process the input and gejnerate the QoIs.
11 out_dict = {'X': X,
12  'Y': Y,
13  'Z': Z}
14
15 return out_dict
```

Listing 4.7: Example of return dictionary in the evaluation function.

4.6.4 Structure of the Output

Two Pandas `DataFrame` are generated as output. The first covers the results from the exploration phase, while the second covers the results of the search phase.

The results of the exploration phase cover the probabilities of satisfaction of the requirements and the levels of the input parameters that make up each set. Each entry is a set with the following columns:

- `set_id` : The number identifying the set.
- `is_discarded` : A Boolean value of 0/1 indicating whether the set has been discarded.
- The input parameters specified in `input.csv`, each column contains the level selected for each set.
- `P`: The total calculated probability for that set.
- The probabilities for each requirement specified in the `response.csv` file such that it plays a role in the exploration phase.

For the search phase, the results present the optimal points and their QoI values. Each entry in the table is a design point with these columns:

- `set_id`: The number identifying the set to which this data point belongs.
- Columns of the input parameters as defined in `input.csv`. These values are the actual number and not a level; it is the found optimum value.
- Columns of response quantities, objective and constraints, as defined in `response.csv`.

4.7 Conclusions

This chapter covered the computational implementation of the PDOPT framework. It presented the Python data objects that encode the design space and perform the two steps described in Chapter 3. Appendix A presents an example test case definition, with detailed input and outputs, based on one of the experiments in Chapter 6.

Part III

Case Studies

Chapter 5

Parametric Analysis of Hybrid-Electric Propulsion

5.1 Introduction

This first test case is a simple parametric study of a parallel hybrid-electric propulsion system to be fitted to an ATR-42 class regional aircraft. To demonstrate the PDOPT method, a low-cost, low-fidelity model was initially selected. The focus was to prove the effectiveness of the exploration step in eliminating the unfeasible and undesirable areas of the design space. The results presented in this chapter are an update of the contents initially presented at the EASN Conference in 2021 (202).

5.2 Problem Definition

The framework is tasked with finding the combination of energy management strategy, battery energy density, and electric motor power density that minimises fuel consumption and take-off mass. Two constraints are applied: a take-off mass constraint of less than 20000 kg and an overall propulsion efficiency constraint above 0.4. The overall efficiency of the propulsive system is calculated as follows:

$$\eta_{total} = \frac{P_{tr}}{P_{total\,source}} \tag{5.1}$$

where P_{tr} is the total power required to fly the aircraft and $P_{total\,source}$ is the total power the gas turbine and battery provide before efficiency. The design space's parameters are presented in Tab. 5.1. The number of levels and parameters produces 81 different combinations to be evaluated. The degree of hybridisation DOH is defined as the ratio between the power provided by the electrical power chain and the total amount of power required, evaluated in the propeller. The energy management strategy is defined by two constant DOH values for climb and cruise.

The analytical definition of the optimisation problem is presented in (5.2). M_{fuel} is the fuel consumed in the mission and TOM is the take-off mass. The two constraints were used for the exploration phase, with a satisfaction probability $P_{sat} \geq 0.5$. Finally, to fairly compare the results of each optimisation problem, only a stopping criteria based on func-

Input Parameter	Symbol	Range	Levels
Climb Degree of Hybridisation	DOH_{cl}	0.1 - 0.9	3
Cruise Degree of Hybridisation	DOH_{cr}	0.1 - 0.9	3
Battery Energy Density	e_b	200 - 500 Wh/kg	3
Motor Power Density	p_{mg}	6000 - 8000 W/kg	3

Table 5.1: Parameters used for this test case.

tion evaluations was chosen. The optimiser would stop after 2000 function evaluations, regardless of the tolerance in the objectives or constraint violations.

$$\begin{aligned} &\text{given} && X = \{DOH_{cl}, DOH_{cr}, e_b, p_{mg}\} \\ &\underset{X}{\text{minimise}} && M_{fuel}, TOM = f(X) \\ &\text{subject to} && TOM \leq 20000\,kg \\ & && \eta_{total} \geq 0.4 \end{aligned} \tag{5.2}$$

It should be noted that the maximum take-off mass of an ATR-42-600 is 18600 kg, which is lower than the constraint used in this test case. While the difference is 7.5%, changing the MTOM would require a redesign of the airframe, hence a different empty weight. Therefore, the results obtained in this study are only for demonstrating the method and not for actual design purposes.

5.3 Simulation Model

The simulation model is composed of an iterative mission analysis solver and functions to simulate the power flow between each propulsion system component. Since this is the simplest test case, most components consist of a single efficiency value; hence there are no dynamics to simulate. Table 5.2 contains the efficiencies of each component present in the system.

The architecture is a parallel hybrid-electric power system in which the two power chains feed into the propeller through a planetary gearbox (Fig. 5.1). Energy recovery during the descent phase was not taken into account.

Propeller	η_{prop}	0.9
Gearbox	η_{gb}	0.99
Gas Turbine	η_{GT}	0.40
Electric Motor	η_{em}	0.95
Power Electronics	η_{pe}	0.94
Battery	η_{batt}	0.95
Cabling	η_{cables}	0.995

Table 5.2: Efficiency values for each component.

The flow chart in Fig. 5.2 shows the mission analysis method implemented in this model. The code starts from a guessed battery and fuel mass values and calculates the

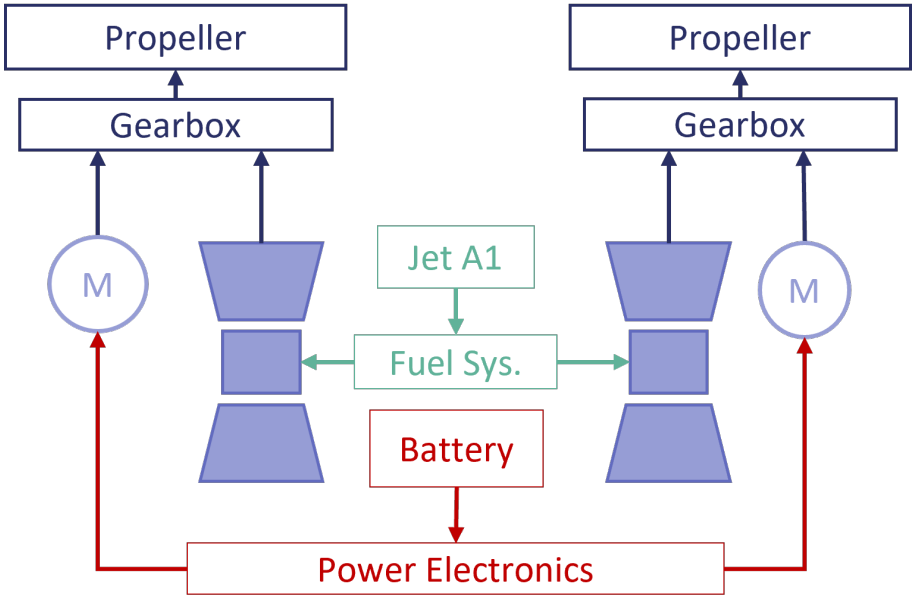


Figure 5.1: Propulsion system configuration.

take-off mass. An iterative subroutine is used to calculate the mass of the electric motors based on the amount of power required for take-off.

The next step is to calculate for each mission segment. First, it calculates the power required to fly P_{req} the segment with (5.3). The mission kinematics and aerodynamics model provide the values of flight speed V, vertical velocity V_z, total mass of the aircraft M, and lift-to-drag ratio L/D. Note that this analysis has no aircraft polar, but a constant value of L/D is assumed for each type of mission phase (see Tab. 5.3).

$$P_{req} = V \cdot \frac{Mg}{L/D} + V_z \cdot Mg \tag{5.3}$$

The value of DOH is used to divide the power between the electrical power chain and the gas turbine. Then the efficiency values are applied to obtain the power required from the sources. Specifically, the power for the electrical power chain $P_{electric}$ and fuel one P_{GT} are calculated with (5.4) and (5.5) respectively.

$$P_{electric} = \frac{P_{req} \cdot DOH}{\eta_{prop} \cdot \eta_{gb} \cdot \eta_{em} \cdot \eta_{cables} \cdot \eta_{pe} \cdot \eta_{batt}} \tag{5.4}$$

$$P_{GT} = \frac{P_{req}\,(1 - DOH)}{\eta_{prop} \cdot \eta_{gb} \cdot \eta_{GT}} \tag{5.5}$$

Finally, the mass required by the energy source is calculated by multiplying the total source power by the operating time of the segment and dividing it by the respective energy density. While the energy density of the battery e_{batt} is given by the input, the energy density of the fuel e_{fuel} is fixed at 12000 Wh/kg, the standard value for Jet A1.

The adopted mission consists of only three segments, each of one mission phase (Fig. 5.3). Details of the kinematics and L/D are presented in Tab. 5.3. The components' efficiencies and L/D were chosen as ballpark values since this study was to demonstrate the methodology.

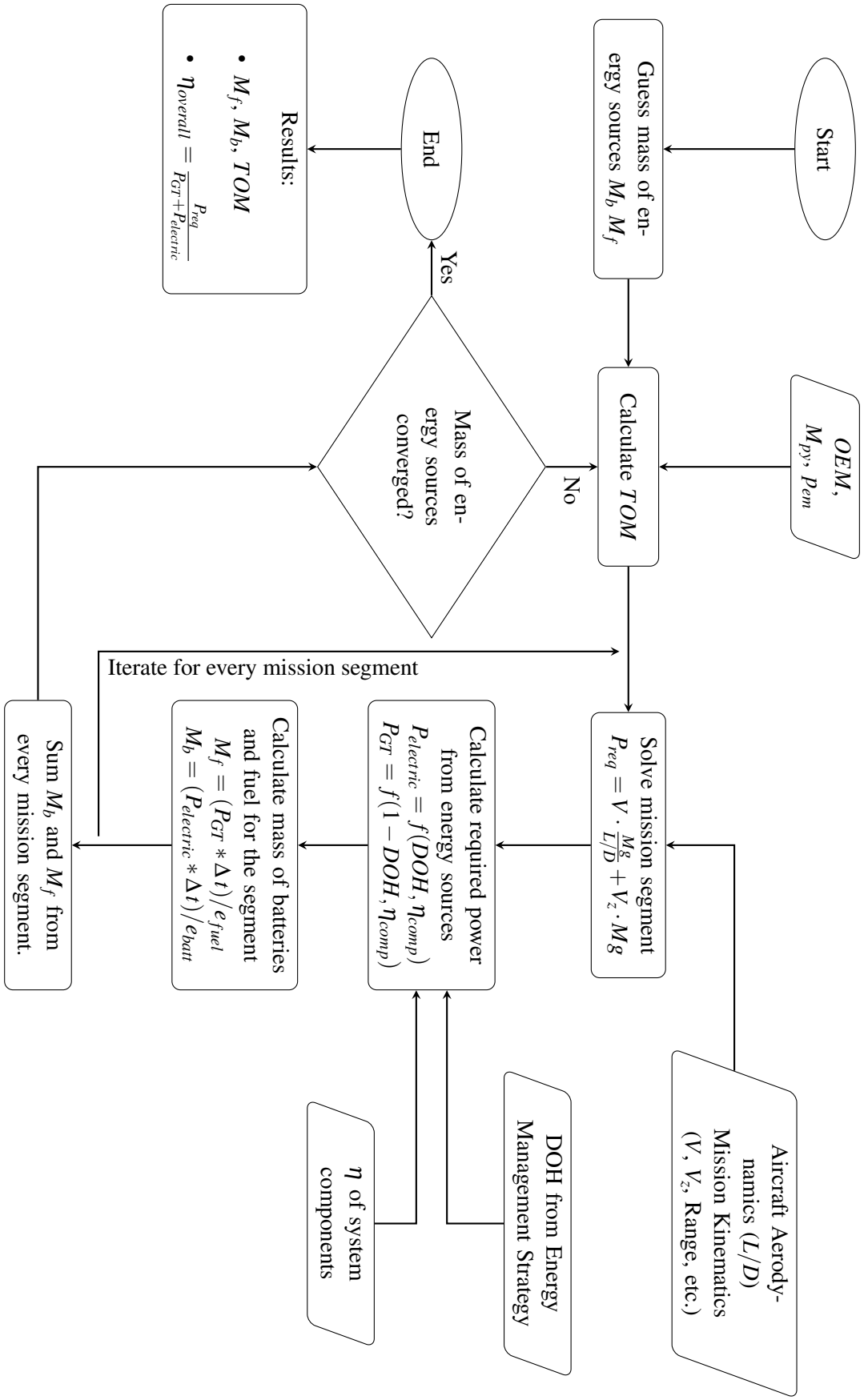


Figure 5.2: Mission analysis method flowchart.

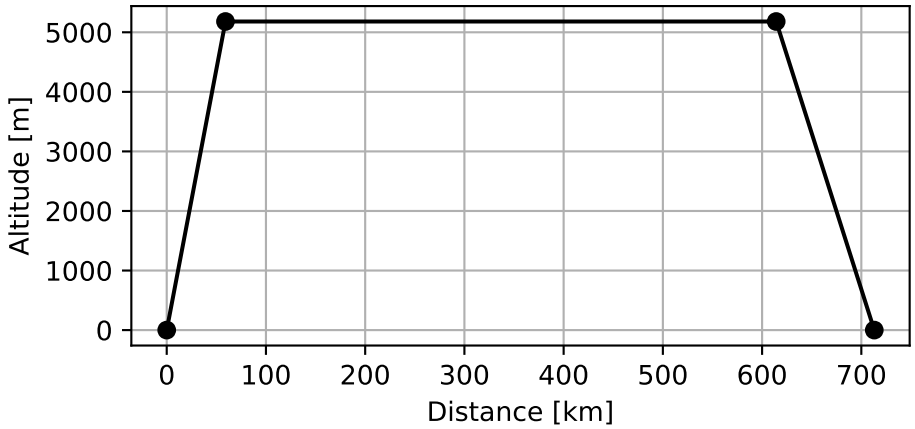


Figure 5.3: Mission Profile.

	L/D	V_{TAS} [m/s]	V_z [m/s]	Range [km]	Altitude [m]
Climb	16.00	107.44	9.4	59.21	-
Cruise	14.50	154	-	555	5181
Descent	16.00	114.48	-6	98.85	-

Table 5.3: Mission Parameters.

5.4 Results and Discussion

	With Filtering	Without Filtering	Change %
Number of Sets	18	81	-77.77%
N. of function evaluations	36036	162162	-77.77%
Exploration run-time (s)	0.195	-	-
Search run-time (s)	701.37	3052.98	-77.02%
Total run-time (s)	701.64	3052.98	-77.01%

Table 5.4: Effectiveness of Probabilistic Filtering

Two different analyses were performed to demonstrate the effectiveness of the methodology. The first performed the process outlined in Chapter 4 without modifications. In contrast, the second ignored the results of the exploration phase and carried out an unconstrained optimisation in every set. This comparison would estimate how much time was saved using the exploration phase and how well the probabilistic surrogate model captured the decision boundary of each requirement. The computational equipment used for both analyses was a laptop with a 4-core Intel i5-8265U CPU running at 1.80 GHz.

The total number of sets generated with the input from Tab. 5.1 is $3^4 = 81$. However, 18 sets were retained after the exploration phase, marking a reduction of 77% of the design space to be evaluated. Table 5.4 presents the reduction in computational cost and evaluation time, both at 77%. The exploration phase presents a minimal overhead, in this case, less than 1 second, greatly compensated by the time-saving in the search phase.

Figure 5.6 presents the results of the exploration phase for all sets. The colours of each line are the overall probability of satisfying the requirements. This value is mainly driven by the take-off mass requirement, whose satisfaction probability ranges from 0 to 0.93. In contrast, the overall efficiency requirement is mostly satisfied, as its lowest probability is 0.63. The sets composed with the highest levels of degree of hybridisation or low battery energy density are marked with low satisfaction probability, and hence they are discarded. In those scenarios, the electrical power would require a battery heavier than the 20000 kg maximum take-off mass constraint.

The surviving sets were passed in the search phase for local optimisation. Figure 5.7 presents the results, with the colours marking the amount of fuel burnt for this design point. There is a trade-off between fuel consumption and the take-off mass caused by the amount of batteries onboard. As expected, this trade-off is also present between the system's overall efficiency and fuel consumption, shown by the scatter plots in Fig. 5.4.

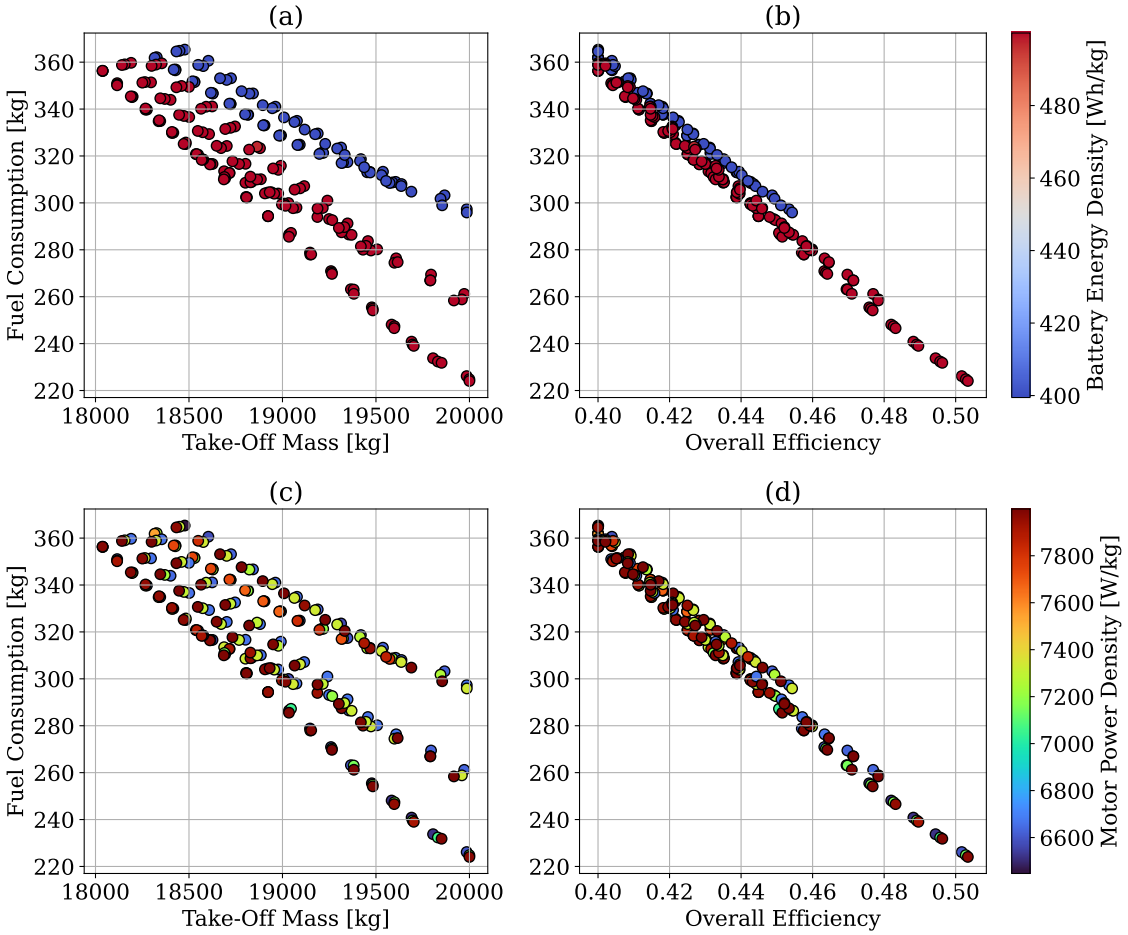


Figure 5.4: Scatter plots of the results of the search phase.

Using parallel coordinates allows one to inspect the interaction between the four inputs and the quantities of interest in the output in a single image. From the perspective of the degree of hybridisation, the cruise segment affects fuel consumption the most, with the climb having a secondary effect. Furthermore, the two parameters present an interaction where the optimiser selects low values of DOH climb and high of DOH cruise. High values of DOH climb are associated with a moderate value of DOH cruise. Figure 5.5

shows the two parameters with the resulting value of M_f. As discussed in the next chapter, the allocation of electrical power between the mission segments is determined by the objective (e.g. reduction of fuel consumption, reduction of NO_x emissions). In contrast, the maximum value is determined by the number of batteries available (limited by the maximum take-off mass).

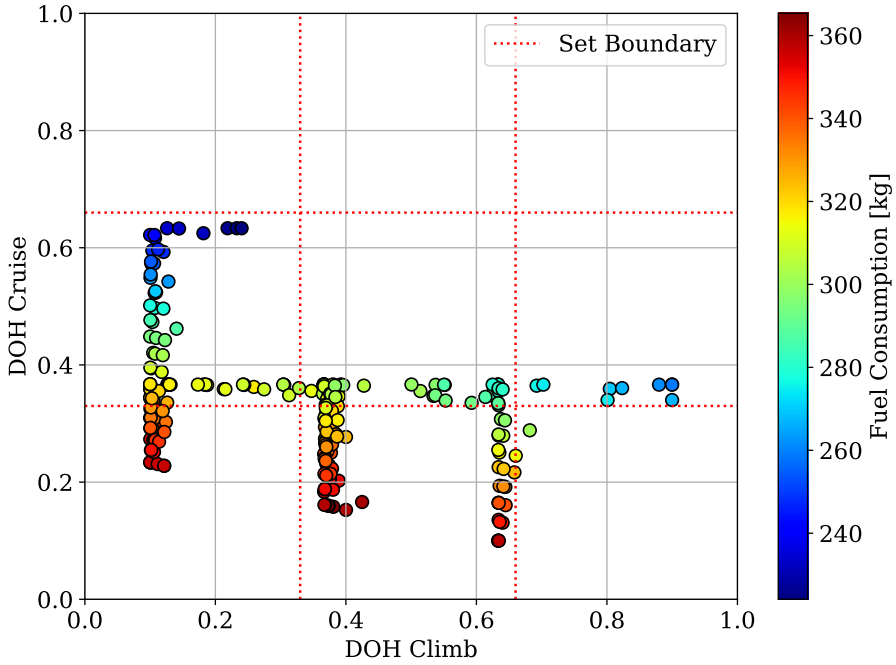


Figure 5.5: Combinations of DOH values in climb and cruise of the design points.

From the perspective of the system parameters, the optimiser is forced towards the highest value of battery energy density available in the set. Since the lowest range was eliminated, only two values remained both the upper boundary of their respective ranges (see Fig. 5.4a). Points with a higher value of e_{bt} dominate in reducing fuel consumption and reducing the take-off mass. On the other hand, the power density of the electric motors does not affect the results as much. The optimiser is still pushed to the highest available values since it produces a lower empty weight, yet the effect is secondary in the objective space. The small clusters present in Fig. 5.4b present the local dominance of the points with higher p_{mg}; however, this effect does not produce global dominance unlike e_{bt}. From the point of view of overall efficiency, a higher battery energy density enables a higher use of the electrical power chain, which is more efficient than the fuel one.

The sets that survived the exploration phase had at least a probability of 0.5 to meet the requirements (Figure 5.7). Ignoring these values and performing an unconstrained optimisation on every set reveals that the probabilistic surrogate correctly identified the boundary between the sets capable of satisfying the requirements of $MTOM$ and $\eta_{overall}$ and those that cannot do so. Figure 5.8 presents the results, with the colour indicating the total probability. The dark red lines indicate the highest probability, which is correctly found between the two constraints.

Figure 5.9 compares the actual constrained values calculated in each set with the probability that the set satisfies the requirement, the red dashed lines representing the

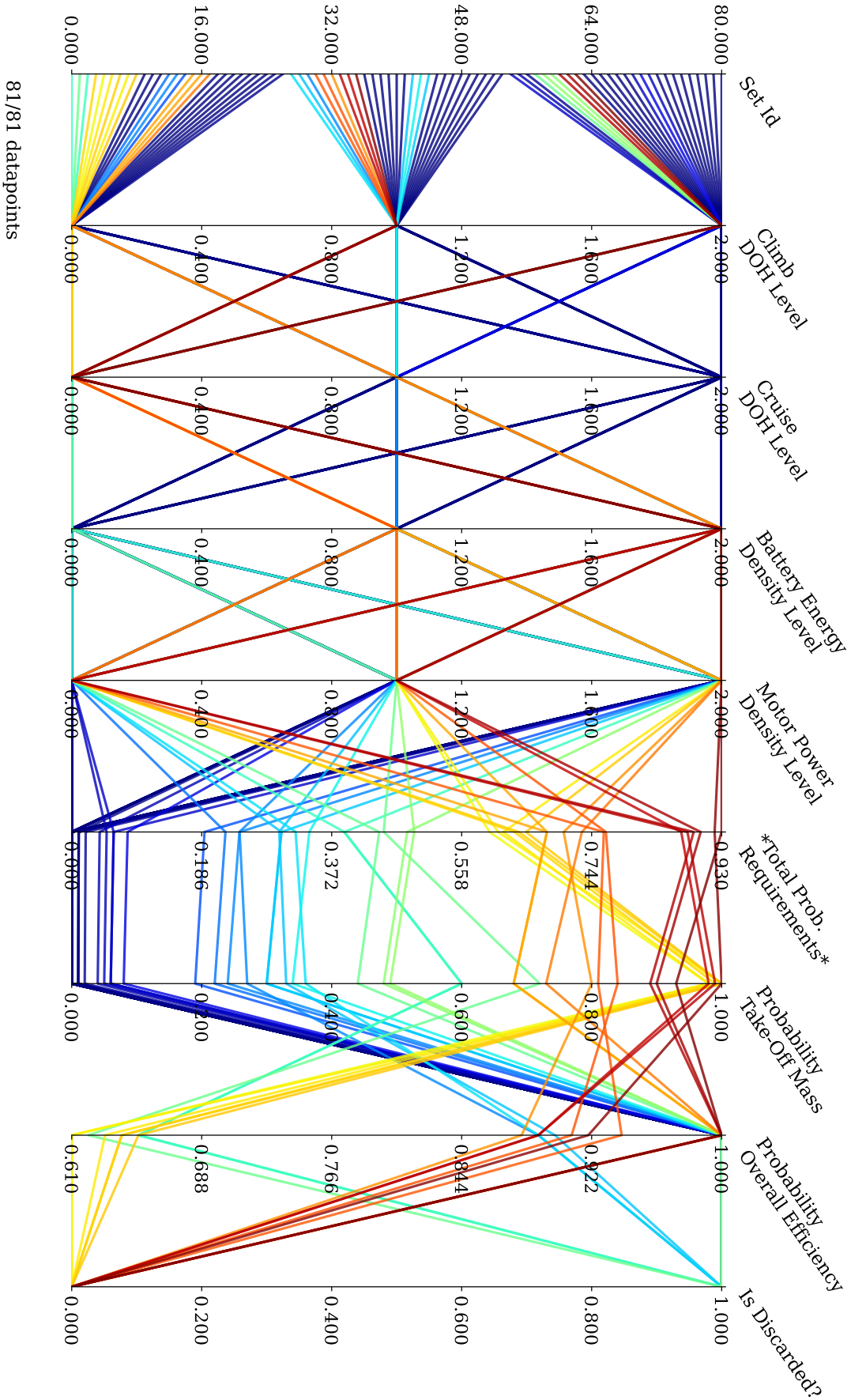


Figure 5.6: Exploration phase results.

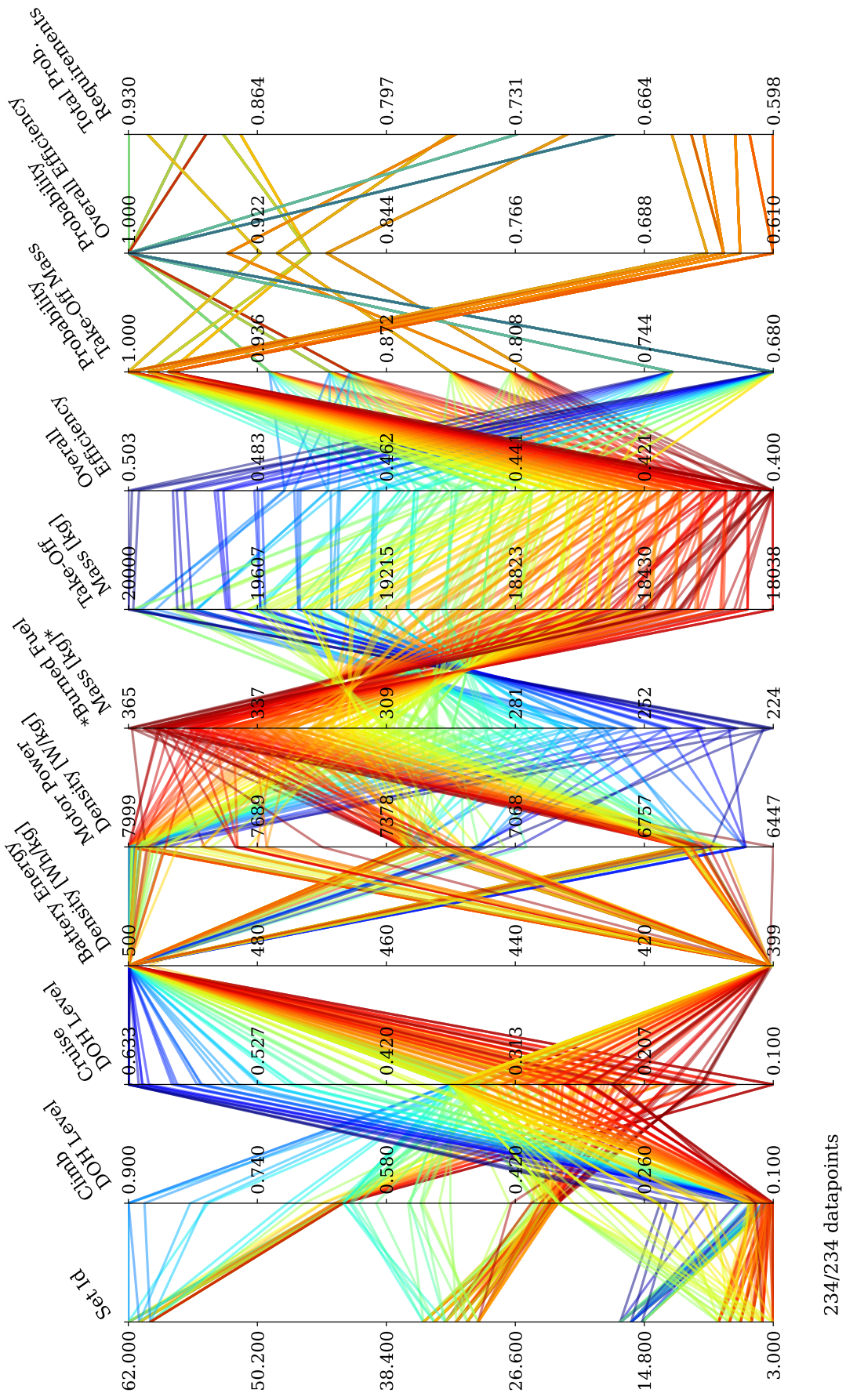


Figure 5.7: Search phase results.

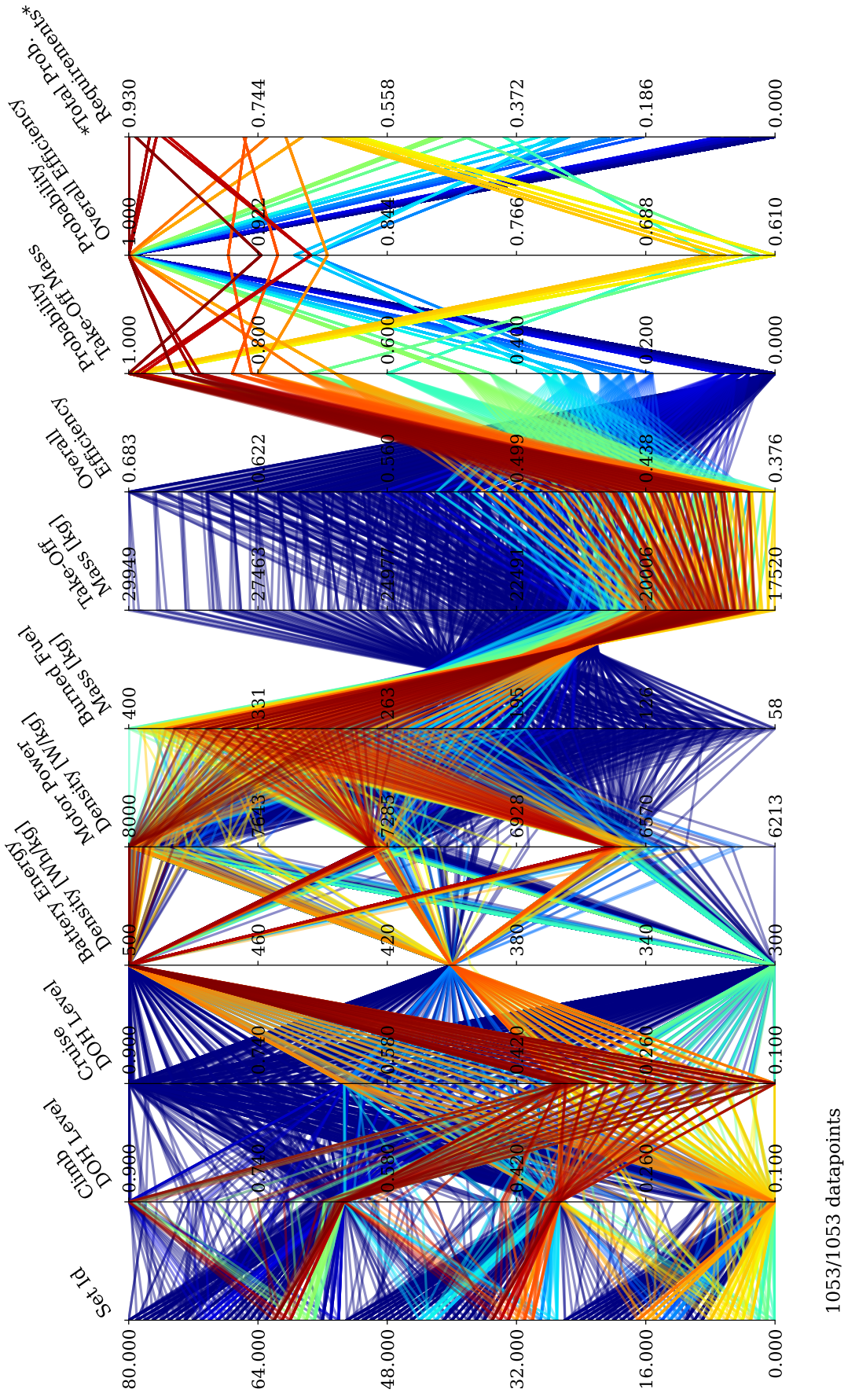
constraint boundary. Both Gaussian processes were able to quantify how likely it was possible to satisfy the constraint. As noted previously, the take-off mass constraint is the most restrictive of the two. Note that the sets that cross the constraint boundary are marked with a value between 0 and 1. Changing the satisfaction probability would determine how many sets crossing it are retained or discarded before the search phase. This parameter is, therefore, critical in correctly implementing the Set-Based Design methodology of eliminating the unfeasible and undesirable areas of the design space. For this reason, the framework was developed such that the user could inspect the results of the exploration phase and decide to reenable some areas of the design space that were initially marked as discarded.

5.5 Summary of Chapter

The test case presented in this chapter was the exploration of the design space of a parametric model of a hybrid-electric aircraft. The input parameters of the model consisted of energy management (DOH in climb and cruise) and system design (Battery energy density, electric motor power density), while the quantities of interest concern the performance of the aircraft (fuel consumption, source-to-propeller efficiency) and its take-off mass. The model used was zero fidelity to test the framework in a computationally reasonable time and prove its effectiveness. The results presented demonstrated that the exploration phase could identify sets that do not satisfy the design space and eliminate them. This allowed a reduction in the computational cost of 77%. Furthermore, thoroughly exploring the design space revealed the interaction between the input parameters and the quantities of interest. In particular, the degree of hybridisation in cruise affected the reduction of fuel consumption the most, and the maximum take-off mass limited its maximum value. These considerations will be better explored in the following chapter, where a thorough analysis of the effects of the energy management strategy is carried out, along with the introduction of a more detailed model.

5.6 Key Learning Outcomes

- The exploration phase guided by the probabilistic assessment of the design space reduced the computational cost by 77%.
- DOH of the cruise segment affected the reduction of fuel consumption the most.
- The maximum take-off value limits the amount of electric energy available onboard.



1053/1053 datapoints

Figure 5.8: Unconstrained optimisation results.

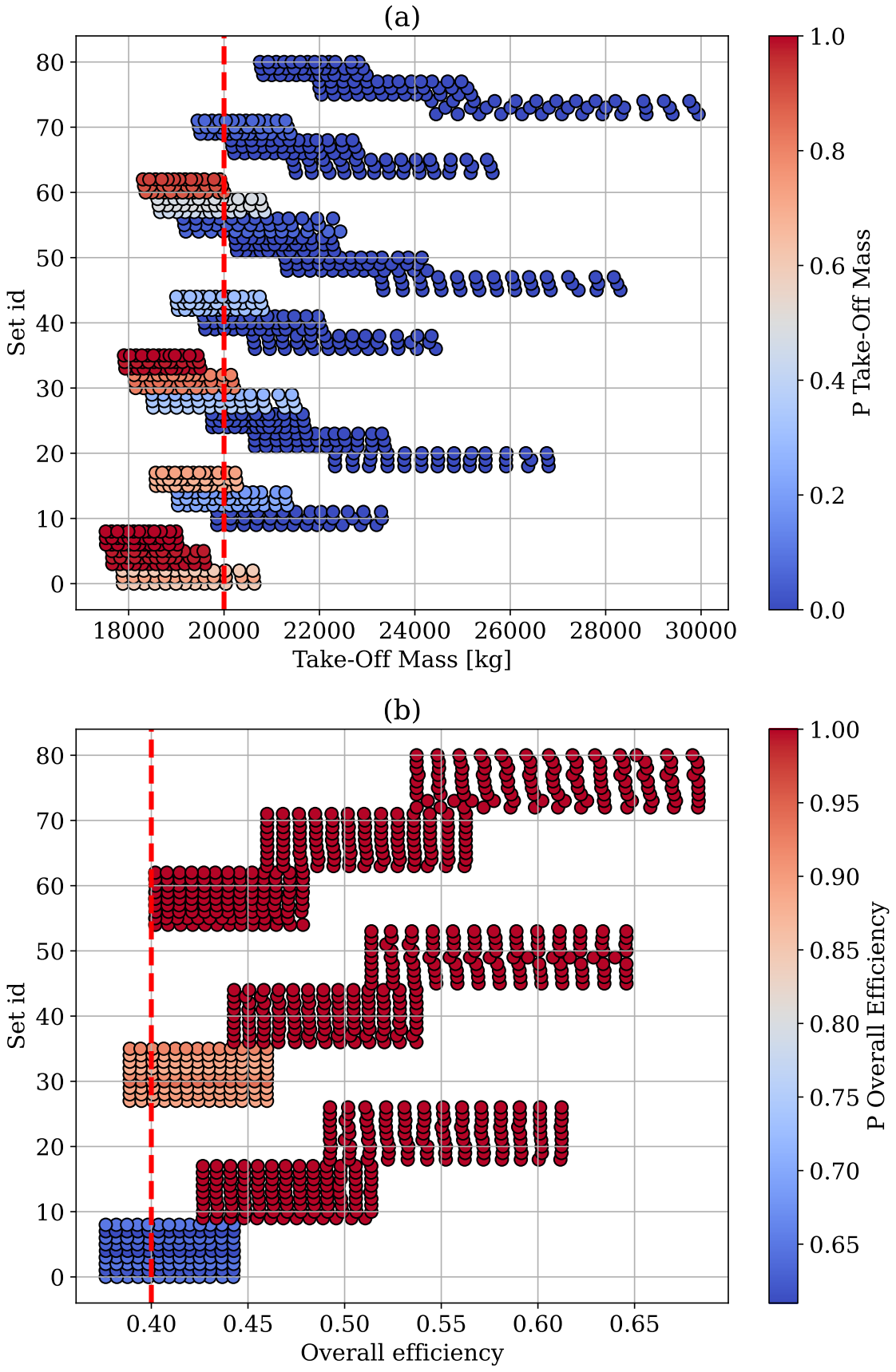


Figure 5.9: Predicted probabilities vs actual range of values in the sets.

Chapter 6

Analysis of Energy Management Strategies

6.1 Introduction

The results presented in this chapter were published initially in MDPI Aerospace in 2022 (203). As discussed in Sec. 2.4, few studies in the literature explored the interaction between the energy management strategy and the sizing of the energy sources. The test case in this chapter attempts to cover this gap by conducting a design space exploration of energy management strategies. Several computational experiments are carried out and compared by varying the topology and the number of electrified mission segments.

6.2 Problem Definition

This test case focuses on analysing the fuel consumption and emissions of hybrid-electric aircraft when subjected to different energy management strategies (EMS). These are defined by up to three linear segments applied to different mission segments (e.g. cruise, climb).

Hence, a matrix of experiments has been defined by combining different hybridised mission phases with different topologies of EMS. Each experiment is identified by a number for ease of reference during the discussion (Tab. 6.1), plus an acronym composed of the hybridised mission phase names and its parameterisation type. For instance, Experiment 6, which has Climb and Cruise hybridised phases and a Linear DOH parameterisation, would be called CC-L.

All experiments share the same optimisation problem formulated in (6.1)) and have DOH bounded between 0.1 and 1. P_{sat} is the target satisfaction probability of the constraint. If a hybridised phase is defined with more than three points, additional constraints are added to ensure that the x positions of the inner points do not cross each other (i.e. $x_2 > x_1, x_3 > x_2$). The descent phase was excluded from hybridisation since battery charging is ignored (see Sec. 6.3.3), and it is assumed that the engine is idling in this mission phase.

Table 6.1: Definition of Experiments

Hybridised Mission phases (in order)	Uniform DOH	Linear DOH	Two segments DOH	Three segments DOH
Cruise	ex. 1 C-U	ex. 5 C-L	ex. 9 C-2S	ex. 11 C-3S
Climb, Cruise	ex. 2 CC-U	ex. 6 CC-L	ex. 10 CC-2S	ex. 12 CC-3S
TO, Climb, Cruise, Final	ex. 3 TCC-U	ex. 7[a] TCC-L	-	-
Climb, Cruise, Climb to Alternate, Cruise to Alternate	ex. 4 CCA-U	ex. 8 CCA-L	-	-

[a] DOH for TO and Final phases are kept uniform as they are very short.

$$\begin{aligned} &\underset{X}{\text{minimise}} \quad M_{fuel}, M_{NO_x} \\ &\text{subject to} \quad M_{takeoff} \leq 20000\,\text{kg}\,(P_{sat} \geq 0.5) \end{aligned} \tag{6.1}$$

These experiments are grouped into test cases to discuss the results in Sec. 6.4. However, given the many results, the most relevant ones for each test case discussed were selected and presented. These cases are as follows:

- **Parametrisation Complexity**: Study the effects of increasing the complexity of the energy management parametrisation. The experiments considered are 1, 2, 5, 6, 9, 10, 11 and 12.
- **Mission Hybridisation**: Understand which and how many mission phases should be hybridised and the effects of their inclusion in the energy management system. The experiments considered are 1, 2, 3, 4, 5, 6, 7 and 8.
- **Take-Off Mass Constraint**: Understand the impact of the take-off mass constraint on the performance of the propulsive system. The experiments considered are 6 and 10.

The baseline values of the reference aircraft (ATR-42-600) are calculated with the methods presented in Sec. 6.3.2 and shown in Tab. 6.2. These quantities are used to calculate the improvement of the energy management strategies and normalise the scatter plots.

Table 6.2: Baseline reference values.

Take-Off Mass	17,568	kg
Burned Fuel (M_f)	1018.22	kg
NO_x Emissions (M_{NO_x})	11.179	kg

6.3 Simulation Model

6.3.1 Energy Management Strategy Definition

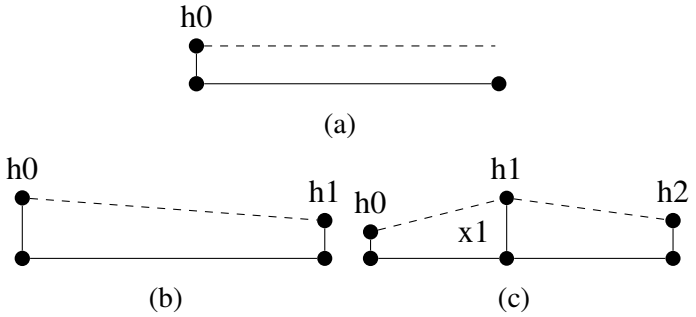


Figure 6.1: Examples of energy management definitions over a single mission phase

Energy management strategies are defined as a continuous piecewise linear degree of hybridisation throughout the mission, with values ranging from 0 to 1. DOH is set to zero when a mission phase is powered only by the gas turbine. Otherwise, the DOH is defined with a combination of two parameters: the degree of hybridisation itself h and its relative distance position x throughout the mission phase. These two quantities are defined as follows:

$$\begin{aligned} h_i &= \frac{P_{re}}{P_{rt}} \in [0,1] \\ x_i &= \frac{d}{L} \in [0,1] \end{aligned} \tag{6.2}$$

where P_{rt} is the total power (in Watts) required to sustain flight across the mission phase, of which P_{re} is the amount to be provided by electric propulsion, d is the phase length spanned over the total phase length L (in kilometers). Within each phase, a number of points are defined, where h is set and linearly interpolated between them. This approach allows modelling both constant, linear or piecewise-linear energy management strategies. Figure 6.1 shows some possible hybridisation strategies over a single mission phase. If a single or two h values are provided, their respective positions x are omitted, as the code automatically assumes a constant or linear interpolation over the phase.

6.3.2 Mission Analysis Methods

The mission analysis code is similar to what has been used in Chapter 5. The flow chart in Fig. 6.2 shows the procedure carried out by the model. It is a low-fidelity simulation code for sizing the mass of the batteries with a power flow approach, which allows for fast function evaluations but with sufficient accuracy to study these trade-offs. However, changes have been made to gas turbine modelling to calculate emissions and improve fuel consumption accuracy.

Burnt fuel is calculated by estimating the fuel flow from the table of gas turbine data with a radial-basis function interpolation and multiplying it by the step time. These

data were previously calculated with the Cranfield gas turbine performance tool TURBOMATCH (204), spanning all possible combinations of altitude, Mach number, and shaft power required for the selected mission. The steps adopted for altitude, Mach number, and power are 300 m, 0.1 and 100 kW, respectively. The maximum power for each altitude and Mach number combination is limited by the maximum allowable TET (Turbine Entry Temperature).

While the required fuel, which is consumed during the mission, can be used as an indicator of CO_2 emissions, NO_x emissions should be calculated by estimating the emission indices EI of the engine. EI is the ratio between the mass of a pollutant (e.g. NO_x) and the mass of burnt fuel. In this test case, the Boeing FuelFlow2 method (205) has been adopted, a simplified P3T3 method. It calculates a sea-level reference EI using the fuel flow value alone rather than the post-combustion chamber's pressure and temperature levels. These values are then corrected by air density and temperature. While it is a simple model, data points describing the relationship between the reference EI and the fuel flow are required. These were collected from Filippone and Bojdo (206), which fitted data from several turboshaft and turboprop engines with surface response methods.

A single mission is selected among all test cases presented. It is the maximum range reference flight mission defined by FUTPRINT50 (194), whose profile is shown in Fig. 6.3. It is divided into two stages, a main flight stage of 800 km distance and a flight to an alternate airport stage of 95 km with a 30-minute holding pattern. The average climb rate is 996 ft/min (5.059 m/s), while the cruise speed is 268 kt (137.78 m/s).

Maximum Take-Off Mass	20,000 kg
Operating Empty Mass	11,550 kg
Payload Mass	5000 kg
L/D Climbout	10.5
L/D Climb/Descent	16
L/D Cruise	14.5
L/D Final	7.5

Table 6.3: Aircraft Properties.

6.3.3 Aircraft and Propulsion Model

The selected aircraft is a retrofit of the ATR 42-600, where the original Pratt & Whitney Canada PW-127 gas turbine is kept, and electric propulsion is introduced. Batteries are assumed to be replaceable and, much like fuel, limited to the amount needed to fly a specific mission.

The test aircraft's maximum take-off mass (MTOM) was increased to 20,000 kg, a 7.5% increase over the nominal maximum take-off weight of the ATR 42-600 (207). This was done to accommodate more battery mass without sacrificing the payload, which is kept constant.

However, it should be noted that changing an aircraft's MTOM would require a rescaling of the empty masses, as the airframe must withstand higher flight loads. Since this test case focuses on the propulsive system and not the airframe, scaling was neglected.

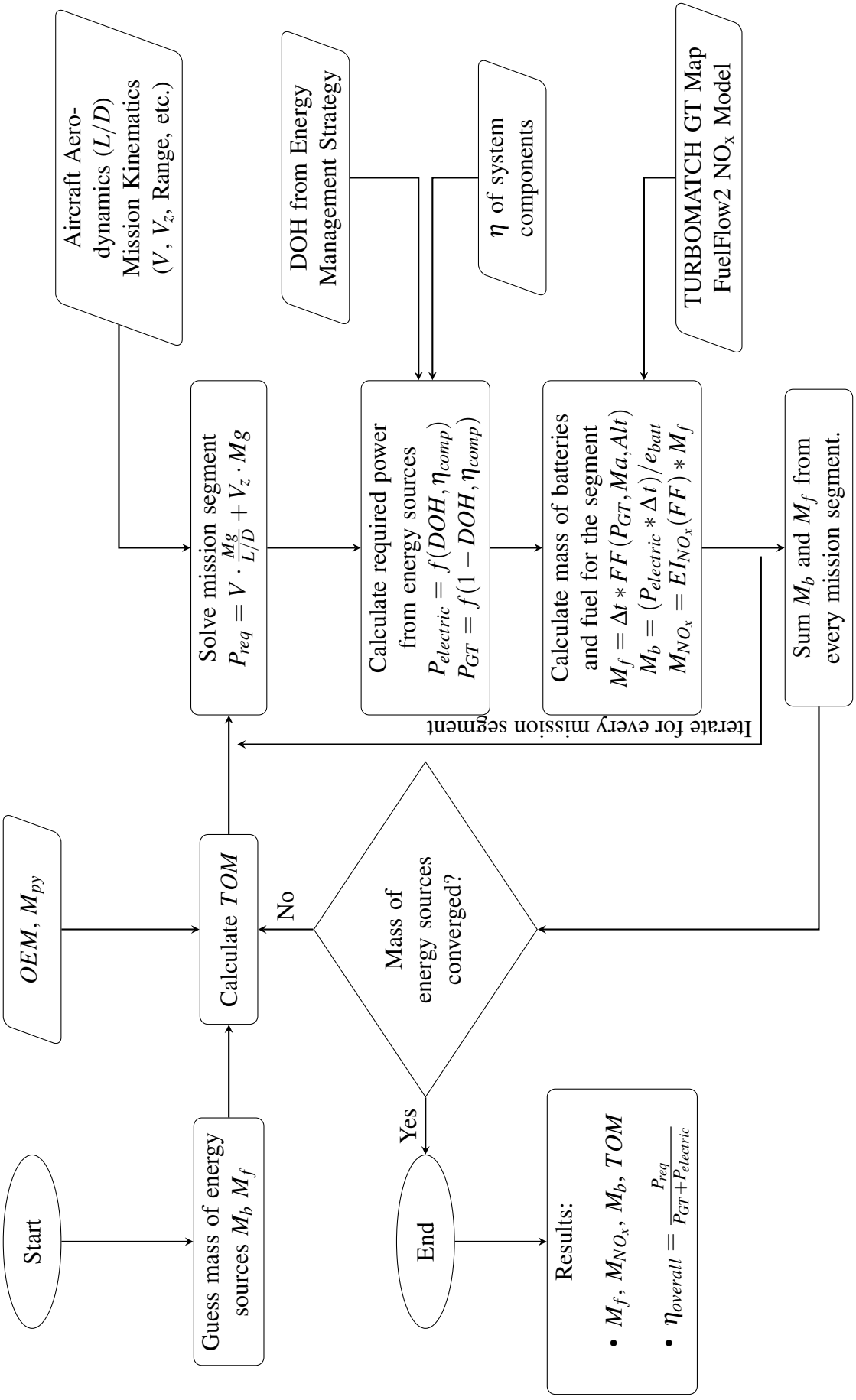


Figure 6.2: Mission analysis method flowchart.

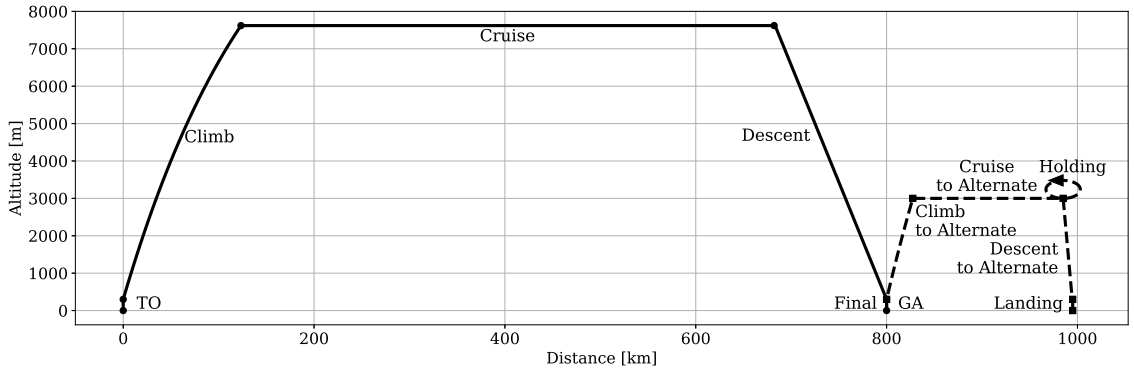


Figure 6.3: Mission Profile.

The results of this study attempt to understand and compare different energy management strategies based on their performance rather than designing a hybrid-electric aircraft.

Figure 6.4 presents the structure of the selected parallel hybrid propulsion system. Through a planetary gearbox, the gas turbine and the electric motor generate the power supplied to the propeller. Table 6.4 and Tab. 6.5 present the assumed properties of the engines, electrical components, and propeller.

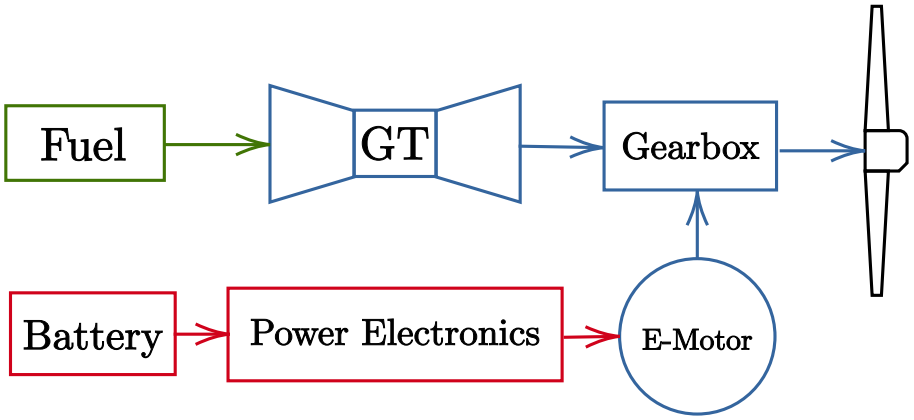


Figure 6.4: Propulsion system configuration.

Table 6.4: Propulsion system parameters.

Propeller Efficiency (Take-Off)	0.64
Propeller Efficiency (Climb)	0.73
Propeller Efficiency (Cruise)	0.86
Propeller Efficiency (Other)	0.8
Gearbox Efficiency	0.99
Electric Motor Efficiency	0.95
Power Electronics Efficiency	0.94
Cabling Efficiency	0.995
Battery Efficiency	0.95
Battery Energy Density [Wh/kg]	500

Table 6.5: Engine specifications at ISA-SLS.

OPR	14.9	HPT cooling flow to core flow ratio	0.12
Mass flow (kg/s)	8.4	LPT cooling flow to core flow ratio	0.05
TET (K)	1250	LPC efficiency	0.87
Net power (kW)	1698.6	HPC efficiency	0.88
Residual thrust (N)	712.1	HPT efficiency	0.89
Efficiency	0.35	LPT efficiency	0.89
SP (MWsec/kg)	0.20	Combustor pressure loss	0.05

The propulsion system model ignores the effects of auxiliary components such as the thermal management system (TMS). This aspect was not included as it is secondary to the interaction between EMS and HEPS. Furthermore, the efficiency of the components is assumed to be constant, except for the gas turbine.

The selected gas turbine engine is thermodynamically similar to PW127 used on the ATR-72-600. The engine is a three-spool gas turbine, where the gas generator is comprised of a high-pressure turbine (HPT), which drives a high-pressure compressor (HPC), and a low-pressure turbine (LPT), which drives a low-pressure compressor (LPC). The free turbine (FT) drives the propeller through the gearbox, as shown in Fig. 6.4. The same baseline engine is kept for different studies by de-rating the engine for different DOHs. This is performed by running the engine at a lower operating point that does not impact the compressors' operability. By de-rating the engine, the operating point of the compressors moves along their regular operating line.

The figure of battery energy density was selected considering a 2035 technological scenario (208). Electrical component efficiencies are estimated as the average of the most widely adopted for power electronics and electrical machines, as found in the literature (209). These values may vary depending on the design requirements, such as the type of machine and its specific power (210). However, for the model in this study, some of the efficiency values were chosen based on experience. Similarly, the propeller's efficiency for different missions is calculated from in-house simulations.

Finally, for this study, the recharging of the battery during flight has been ignored. Electrical power always flows from the battery to the propeller. Energy harvesting has been shown to be marginally beneficial if the airframe is specially designed for it (**?**).

6.4 Results and Discussion

The results are divided between analysing the methodology's effectiveness and understanding the energy management system. Aspects such as EMS topology, affected mission segments, and take-off mass limitations are discussed with the two objectives of M_f and M_{NO_x}.

6.4.1 Effectiveness of Design Space Exploration Methodology

This section presents the computational cost for running each experiment and the effectiveness of the probabilistic constraints and estimates how much time was saved with the

presented methodology. All the calculations presented in this study were performed on a 16-core machine (Two Intel E5-2620 v4 CPUs) and 128 GB of memory. Table 6.6 and Tab. 6.7 present the total number of design subspaces, the discarded percentile, and the time required to execute the entire calculation for each experiment.

Furthermore, Tab. 6.6 compares the actual computational time with the estimated one if the framework is searched in every subspace, ignoring the results of the exploration phase. Even with the overhead of the sampling, Gaussian process training, and probabilistic exploration, the total time required by the framework is less than would have been needed to search each subspace. This corroborates the findings in the previous chapter.

However, the amount of time saved is directly proportional to the number of subspaces discarded. The best practice is to have relaxed probabilistic constraints and fewer parameters for a first run, then increase the probabilistic constraints to drive the exploration towards the desired response. For instance, after identifying where the Pareto front is, the user can add a probabilistic soft constraint on the objectives such that the framework discards those areas of the design space far from the global Pareto front. This method is consistent with the "convergent approach" to design.

In addition to the experiments presented in Tab. 6.1, an additional experiment was performed by adding soft constraints on the objectives to the Climb-Cruise with Two Segments case (experiment 10). These additional soft constraints, shown in (6.3), were selected from the inspection of the Pareto fronts (see Sec. 6.4.2 and Sec. 6.4.3) to discard further subspaces that most likely do not contain global optimum points.

$$\begin{aligned} M_{fuel} &\leq 940\,kg \quad (P_{sat} \geq 0.5) \\ M_{NO_x} &\leq 10.25\,kg \quad (P_{sat} \geq 0.5) \end{aligned} \tag{6.3}$$

Figure 6.5 and Fig. 6.6 present the data points of the original case and the one with additional constraints, respectively. These results show that the probabilistic constraints effectively limit the search to those subspaces within the desired constraints, especially hard ones, such as the take-off mass constraint. Additionally, Fig. 6.5 shows that the probability constraint and the value are correlated. Therefore, the exploration phase effectively discards areas of the design space that would not satisfy the hard constraints, as demonstrated in the previous chapter.

Figure 6.7 presents the overlapped objective space of the two cases, indicating that the soft constraints were able to restrict the search away from areas with low objective improvement. However, as was required, the soft constraints did not strongly limit near the Pareto front, with some optimal points above the constraints. Second, the relatively low total probability in Fig. 6.6 suggests that the system under design does not have the same sensibility to each constraint.

While this "spillover" effect requires further investigation and testing, several causes can be suggested: the number of discrete levels of the input parameters, the number of sampled points in each subspace, the sensibility of the model responses to the input parameters and the satisfaction probability. Indeed, if the subspaces are too coarse, they might cover areas that include both globally optimal solutions and the least desirable solutions. These subspaces, therefore, include the probabilistic constraint boundary and, if not adequately sampled, are discarded, eliminating some possible optimal points. Conversely, if they are kept, the suboptimal solutions are retained, and spillover is visible in the objective space. The model sensibility would also play a role in the shape and steep-

Table 6.6: Number of discarded subspaces and saved time.

Experiment	$Level^{N_{par}}$	N_{par}	$N_{subspaces}$	**% Discarded**	**Total Run Time (ch)**	**Estimated Runtime without Discarded Subspaces (ch)**	**% Saved Time**
C-U	4^1	4	4	50.0%	0.75	1.48	49.09%
CC-U	4^2	4	16	75.0%	3.85	15.35	74.90%
TCC-U	4^2, 3^2	7	144	79.2%	29.17	138.01	78.86%
CCA-U	4^4	4	256	94.5%	16.08	293.48	94.52%
C-L	4^2	2	16	62.5%	5.57	14.76	62.22%
CC-L	4^4	4	256	85.1%	43.95	295.79	85.14%
TCC-L	3^6	6	729	90.1%	72.00	728.48	90.12%
CCA-L	3^8	8	6561	98.9%	86.90	7796.80	98.89%
C-2S	3^4	4	81	61.7%	30.07	78.44	61.67%
CC-2S	3^8	8	6561	90.8%	604.46	6553.42	90.78%
C-3S	3^6	6	729	84.2%	111.80	708.43	84.22%
CC-3S	3^{12}	12	531,441	99.9%	268.49	530519.84	99.95%

Table 6.7: Breakdown of the total computational time.

Experiment	Sampling Time (s)	Training Time (s)	Exploring Time (s)	Search Time (s)	Total Run Time (h)
C-U	2.644	0.025	0.357	166.30	0.047
CC-U	3.543	0.037	0.018	863.16	0.241
TCC-U	9.512	0.193	1.166	6552.60	1.823
CCA-U	4.529	0.058	1.174	3611.16	1.005
C-L	9.058	0.058	0.019	1245.18	0.348
CC-L	9.603	0.073	0.857	9878.86	2.747
TCC-L	10.745	0.127	0.673	16188.48	4.500
CCA-L	10.204	0.025	24.367	19518.74	5.431
C-2S	9.560	0.162	0.460	6754.59	1.879
CC-2S	11.005	0.042	25.138	135967.70	37.779
C-3S	7.357	0.064	3.329	25144.75	6.988
CC-3S	9.673	0.171	5819.547	54580.23	16.780

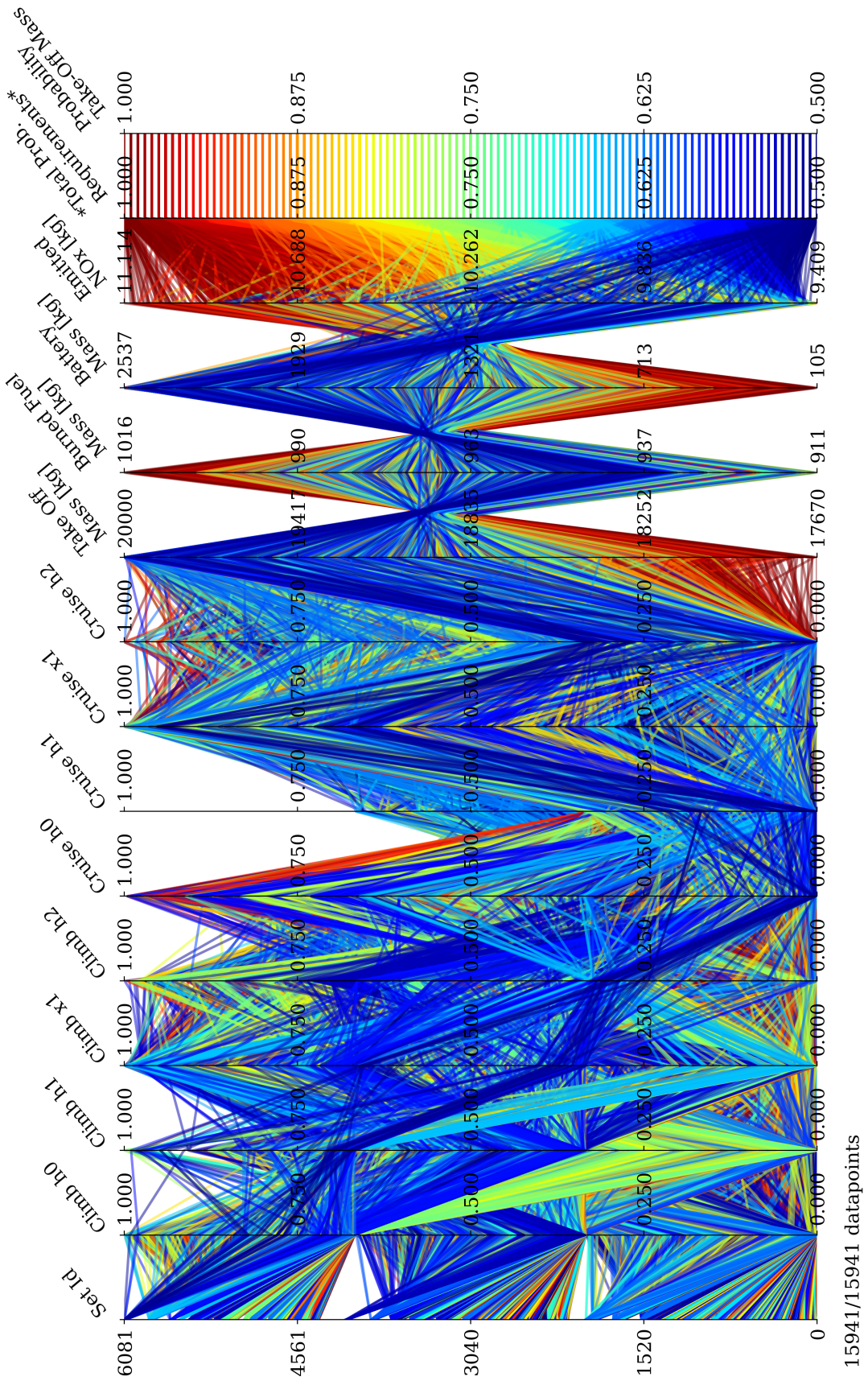


Figure 6.5: Data of Experiment CC-2S.

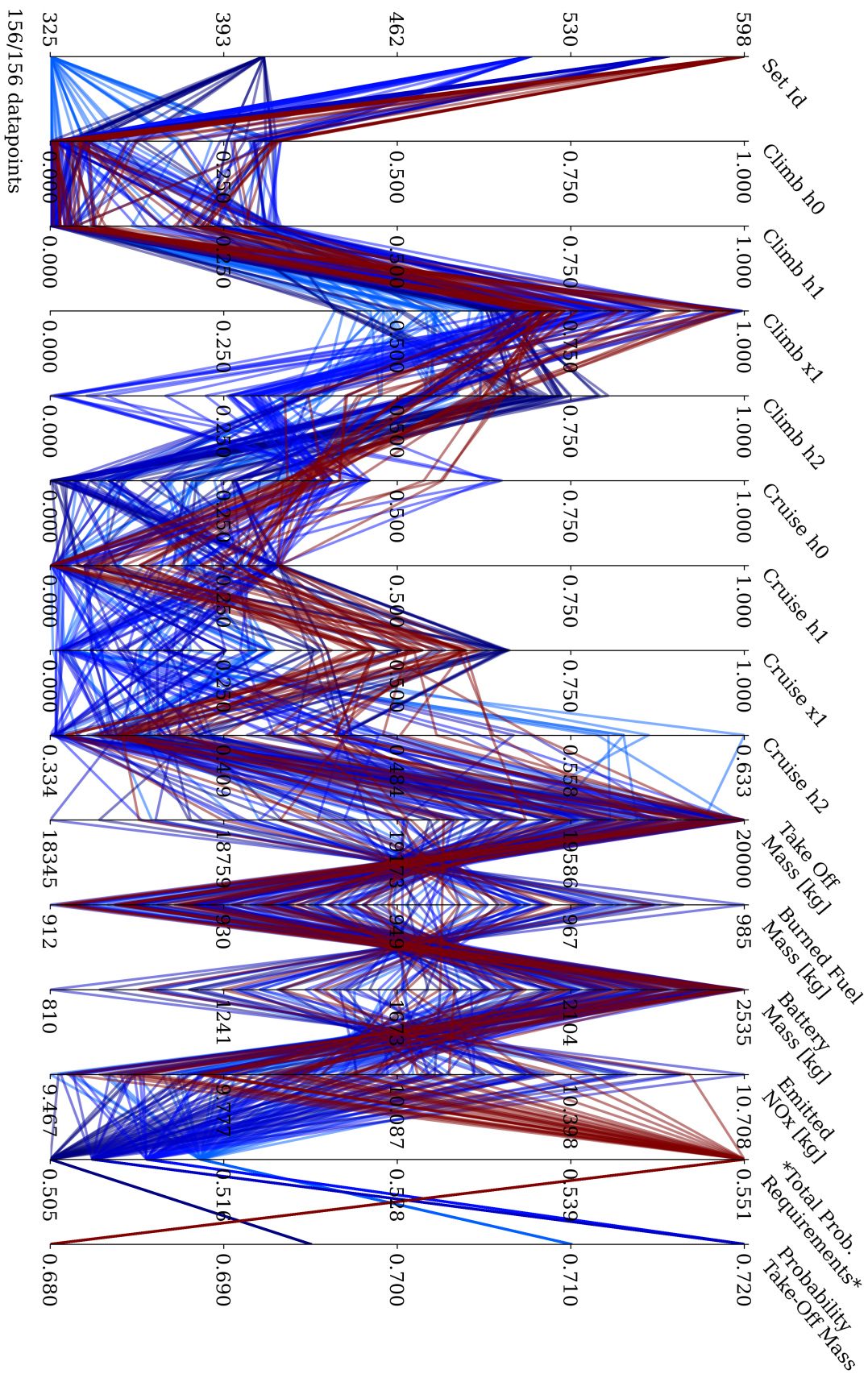


Figure 6.6: Data of Experiment CC-2S with additional soft constraints.

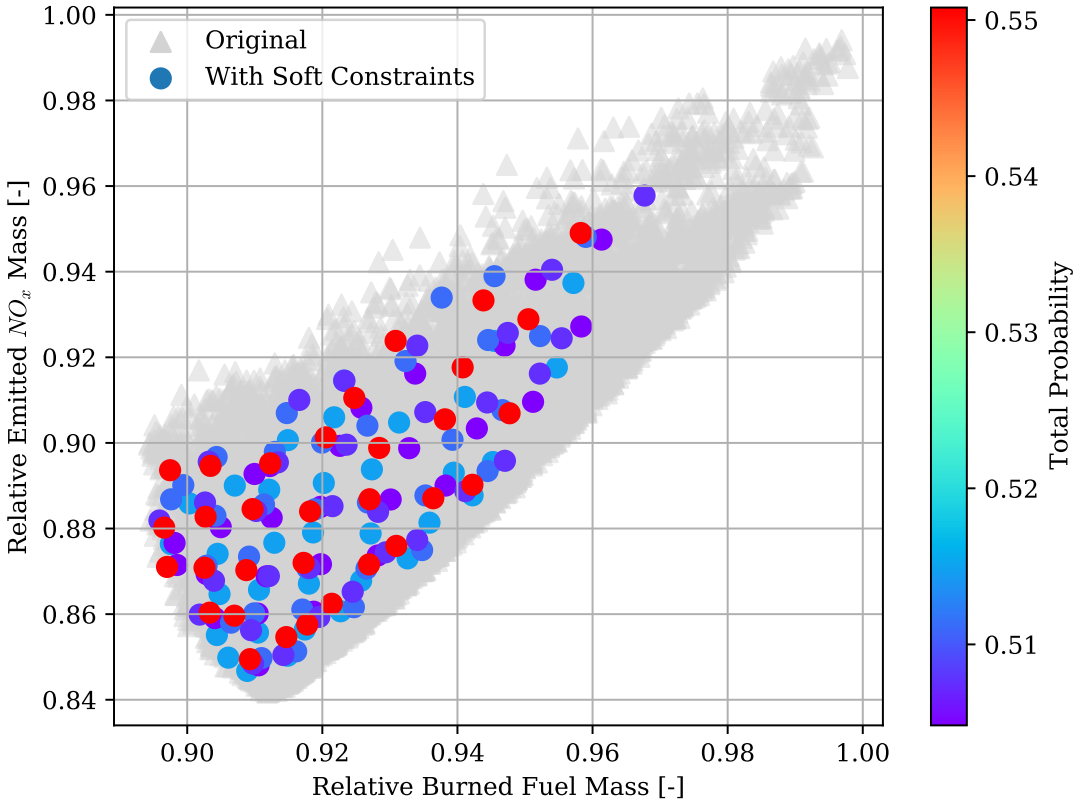


Figure 6.7: Objective space of data from Figures 6.5 and 6.6.

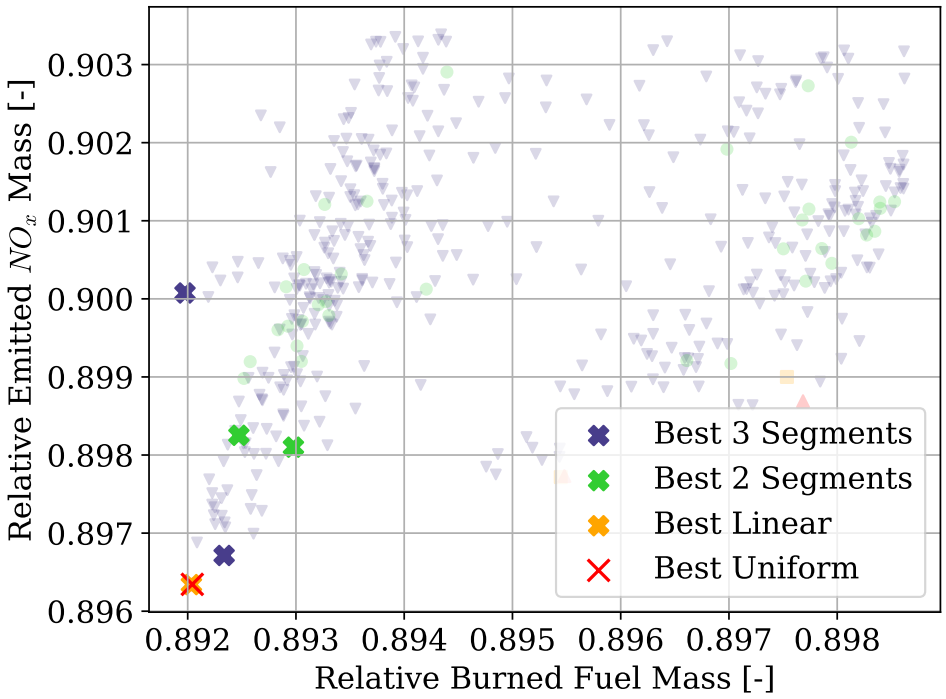
ness of the probabilistic constraint boundary. However, unlike the two other parameters, it cannot be controlled. Finally, the satisfaction probability P_{sat} determines the threshold for a single sample to be accepted and, therefore, how much the hypervolume of a design subspace can satisfy the probabilistic constraints.

In summary, the inclusion of soft constraints in the exploration phase defines an 'exploration-exploitation' problem, where the discretization of the continuous parameters, the sampling and the probability of satisfaction determine if the accepted design space is restricted to the most desirable solutions (exploitation) or relaxed to accept sub-optimal solutions (exploration).

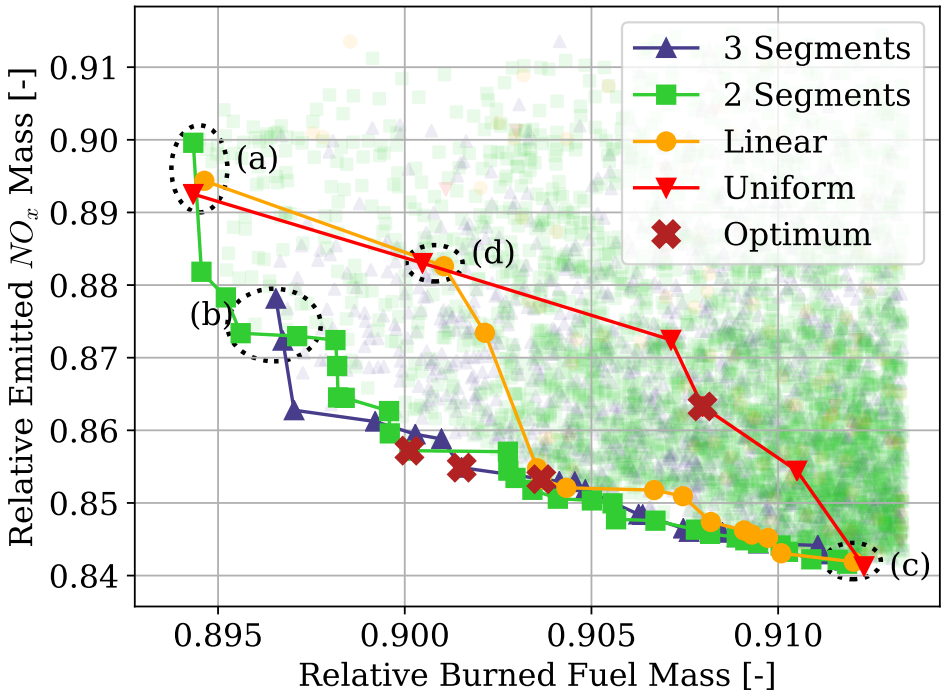
6.4.2 Parametrisation Complexity

The experiment matrix of Tab. 6.1 has four columns, each representing an increase in the complexity of the EMS parameterisation. This section will compare the first and second rows of experiments to identify the effect of parameterisation complexity on aircraft performance. The objective spaces of the two sets are shown in Fig. 6.8.

In the first group of experiments (Fig. 6.8a), there is no Pareto front between M_f and M_{NO_x}. Instead, the constant and linear parameterisations overlap with a single optimal point, whereas the two-segment and three-segment strategies have two close points. Tracing the shape of each EMS (Fig. 6.9) shows that they all converge toward constant parameterisation. In particular, three-segment strategies degenerate into a two-segment



(a) Optimal points of cruise-only EMS.



(b) Pareto fronts of climb and cruise EMS.

Figure 6.8: Effects of parameterisation complexity.

one because their first point overlaps the beginning of the mission phase.

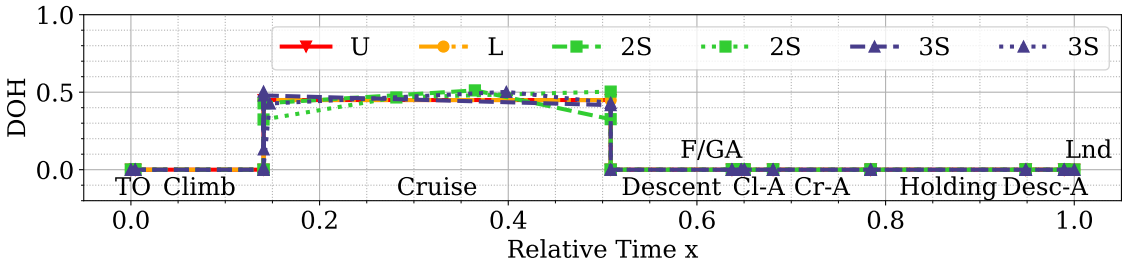


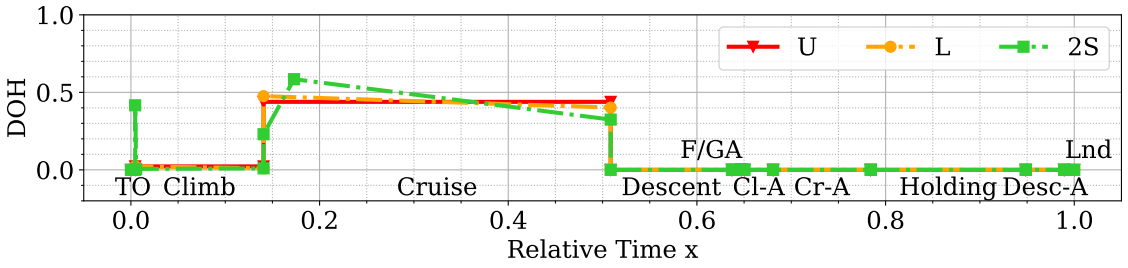
Figure 6.9: Hybridisation strategies of the optimal points of Fig. 6.8a.

The results of the second group of experiments are presented in Fig. 6.8b, where a Pareto front is present. The more complex parametrisations have more points and include some strategies dominating simpler ones. Figure 6.10 shows the optimal energy management strategies found, corresponding to the marked points in Figure 6.8b. Points close to each other in the objective space present very similar topologies, regardless of parameterisation. This is particularly true for the extremities of the Pareto fronts (Fig. 6.10a and Fig. 6.10c) and the highlighted points marked as "b" and "d" (Fig. 6.10b). In Fig. 6.10d, the optimiser found a linearly increasing h strategy for the climb phase, which allows these points to dominate the original uniform strategy. The details of these optimal points are presented in Tab. 6.8.

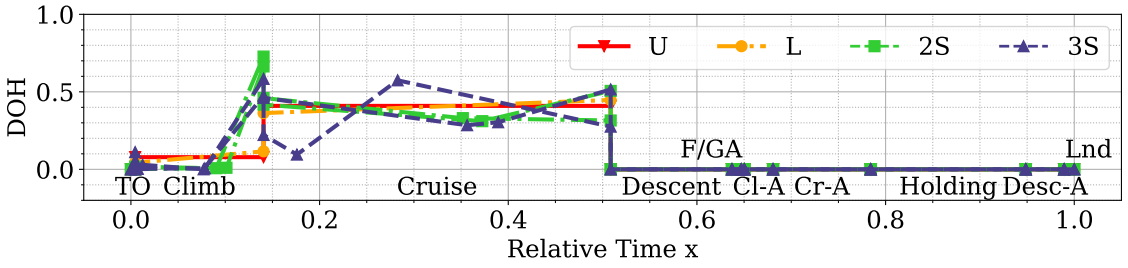
Point	TOM [kg]	M_{fuel} [kg]	$M_{battery}$ [kg]	M_{NO_x} [kg]	η_{total}
Baseline	17568	1018.22	0	11.179	0.299
CC-U, $min(M_f)$	19998	910.63	2537.65	9.978	0.316
CC-U, $min(M_f, M_{NO_x})$	19992	924.52	2518.23	9.651	0.319
CC-U, $min(M_{NO_x})$	19999	928.93	2520.33	9.405	0.326
CC-L, $min(M_f)$	19981	910.94	2520.08	10.053	0.316
CC-L, $min(M_f, M_{NO_x})$	19994	920.12	2524.12	9.594	0.322
CC-L, $min(M_{NO_x})$	19998	928.64	2519.55	9.467	0.326
CC-2S, $min(M_f)$	19997	910.63	2536.66	10.112	0.316
CC-2S, $min(M_f, M_{NO_x})$	19999	915.15	2590.40	9.588	0.321
CC-3S, $min(M_{NO_x})$	19999	928.46	2520.66	9.464	0.325
CC-3S, $min(M_f)$	19997	910.63	2536.66	10.112	0.316
CC-3S, $min(M_f, M_{NO_x})$	19999	915.15	2590.40	9.588	0.321
CC-3S, $min(M_{NO_x})$	19999	928.46	2520.66	9.464	0.325

Table 6.8: Detailed results of the optimal points where Climb and Cruise are hybridised.

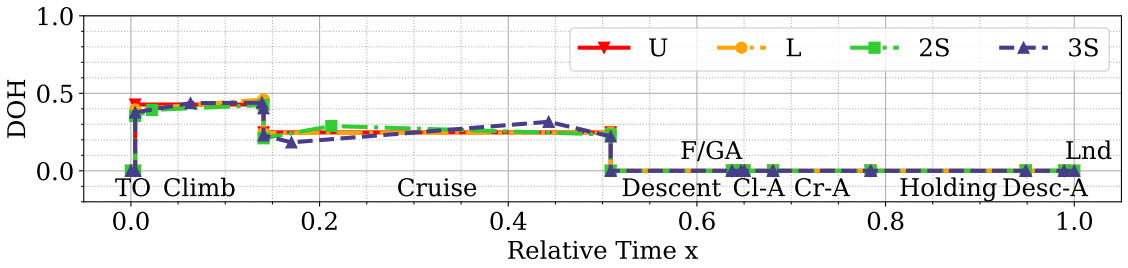
These results indicate that complex parametrisations beyond two segments do not produce significant improvements over the objectives. According to the analysis, the linear and two-segment strategies are flexible enough to identify the Pareto front in the objective space. However, it should be noted that optimal points of the two-segment type present a 2% extra reduction in NO_x emissions relative to the linear counterparts toward the upper left area of the front (where the burnt fuel mass is low). This indicates that



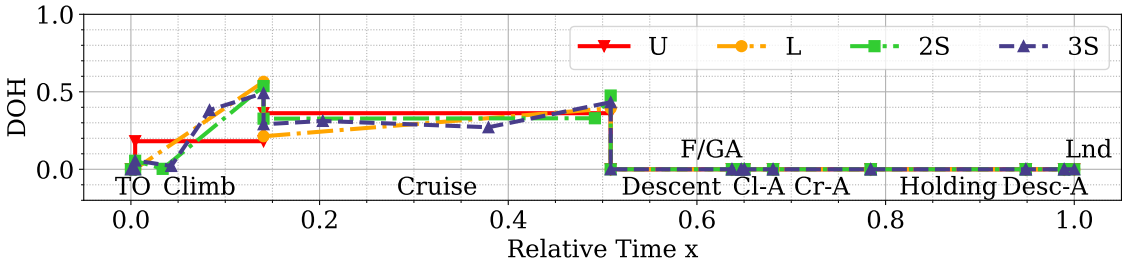
(a) Selection (a), Mininum M_f



(b) Selection (b) and (d)



(c) Selection (c), Mininum M_{NO_x}



(d) Point closest to origin, marked with "X" in Figure 6.8b

Figure 6.10: Optimal energy management strategies of Figure 6.8b.

the topology and parameterisation of the energy management strategy affect more the emissions of NO_x than the fuel consumption (and therefore the emissions of CO_2).

Linear parameterisation has a lower computational cost in design exploration. However, it cannot capture some of the solutions towards the area of the Pareto front with lower fuel consumption (Fig. 6.8b). In this region, the two-segment EMS dominates over the linear solutions. The cause is a late deployment of electric power during the climb phase, which is impossible with linear and constant strategies (Fig. 6.10b). Nevertheless, linear might be enough for quick design space exploration. Indeed, regardless of the number of degrees of freedom in the parameterisation, the shape of the EMS is sim-

ilar for points in the same region of the objective space, as was shown in both groups of experiments. In fact, the Pareto front between the objectives is mainly independent of the chosen parametrisation of the EMS. Only when at least two mission segments were hybridised the Pareto front was present. This aspect will be explored further in the next section.

6.4.3 Hybridisation of Mission Segments

This section compares the first two columns of the experiment matrix (Tab. 6.1) to understand the effects of mission hybridisation on aircraft performance. The first group comprises uniform strategies only, while the second group comprises linear strategies. The results are presented in Fig. 6.11.

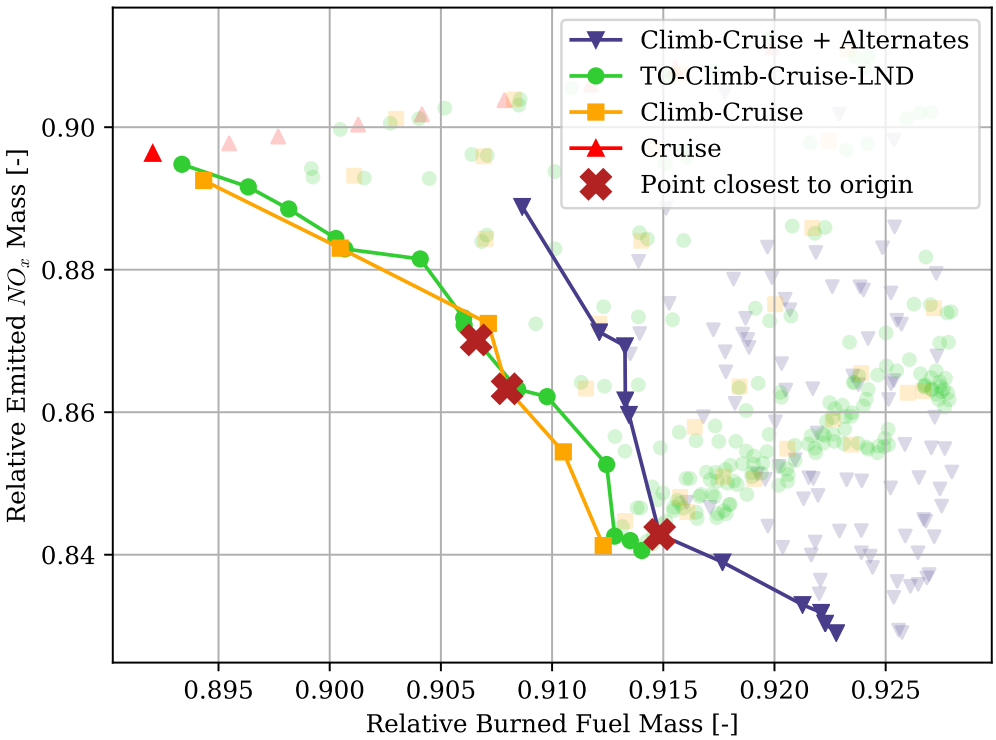
Both groups present similar trends in the comparison between the Pareto fronts. First, introducing electric power in the terminal phases of the mission (take-off and landing) does not bring about any significant change. As shown in Fig. 6.11, the points with only hybridisation of climb and cruise slightly dominate those points which also hybridise take-off and landing. The discrepancy between the two is, on average, 0.5% in the mass of burnt fuel and 1% in the emissions of NO_x. Although these parts of the mission play a marginal role in overall emissions, they are essential when considering the impact of low-altitude operations on communities living near airports. Introducing electric power at take-off has been proposed to reduce airfield noise (194, 165). The addition of electrical power in the alternate portion of the mission produced a shift of the Pareto front toward lower NO_x emissions at the cost of higher fuel consumption.

Finally, as pointed out previously, there is no trade-off between the objectives if electrical power is introduced only in the cruise segment. Parameterisation does not affect the general behaviour of the system, as the Pareto fronts of Fig. 6.11a and Fig. 6.11b are similar in shape; instead, there is an improvement in performance due to the greater flexibility of the electric power distribution along the mission phases.

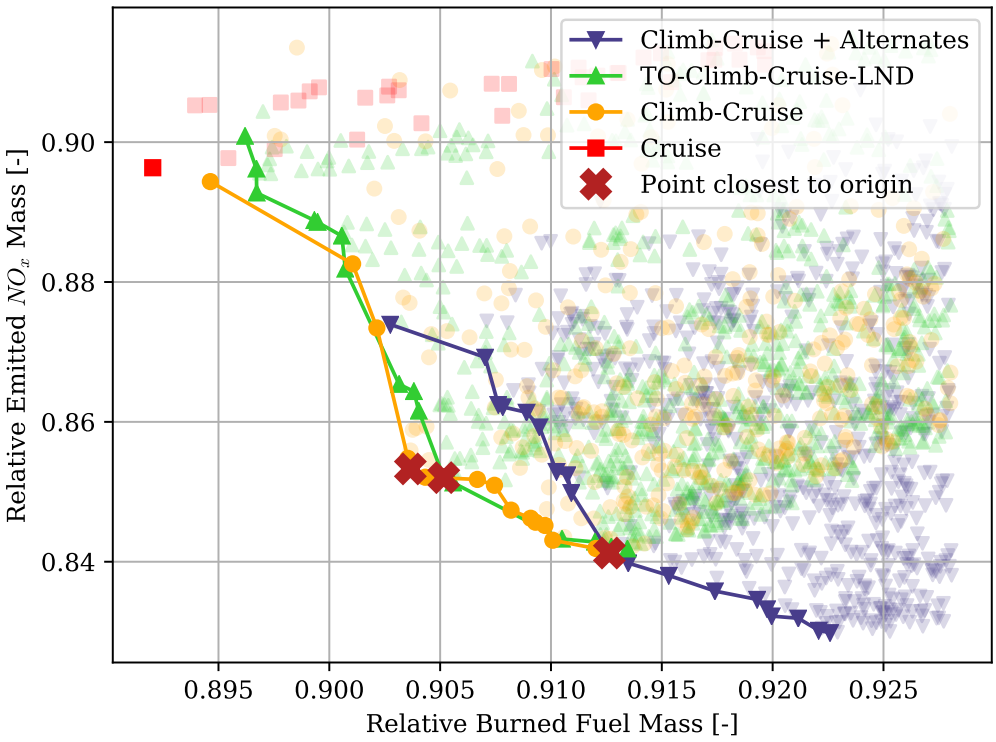
In conclusion, the highest fuel consumption reduction is achieved by prioritising the use of electrical power in the longest mission segments, cruise and climb. The lowest fuel consumption was obtained when all electrical power was allocated in the cruise segment, which is an extreme case of the trade-off when allocating electrical power between climb and cruise. The parallel coordinates plot in Fig. 6.12 can elucidate this aspect. These points are the Pareto front points of the CC-L experiment (Climb and Cruise with linear DOH). EMS that achieves a lower NO_x have more electrical power in the climb segment. As expected, the lowest fuel consumption is achieved when more electrical power is applied in the cruise segment.

Table 6.9 reports the detailed results for each of the optimal points identified in Fig. 6.11a. The overall efficiency η_{total} (calculated with (5.1)) is positively correlated with the reduction of NO_x. In contrast, fuel consumption increases with higher efficiency. This apparent paradox is caused by the constraint in take-off mass, which limits the mass allocated to batteries and fuel. When more phases of the mission are hybridised, the overall efficiency increments but the amount of electrical energy allocated to the longest mission phase is reduced. Hence, cruise has a lower degree of hybridisation, increasing fuel consumption.

Moreover, the climb segment is associated with higher power settings; consequently, more NO_x is generated due to the higher combustion temperature in the gas turbine. In-



(a) Pareto fronts of uniform strategies with different hybridised mission phases.



(b) Pareto fronts of linear strategies with different hybridised mission phases.

Figure 6.11: Effects of mission hybridisation.

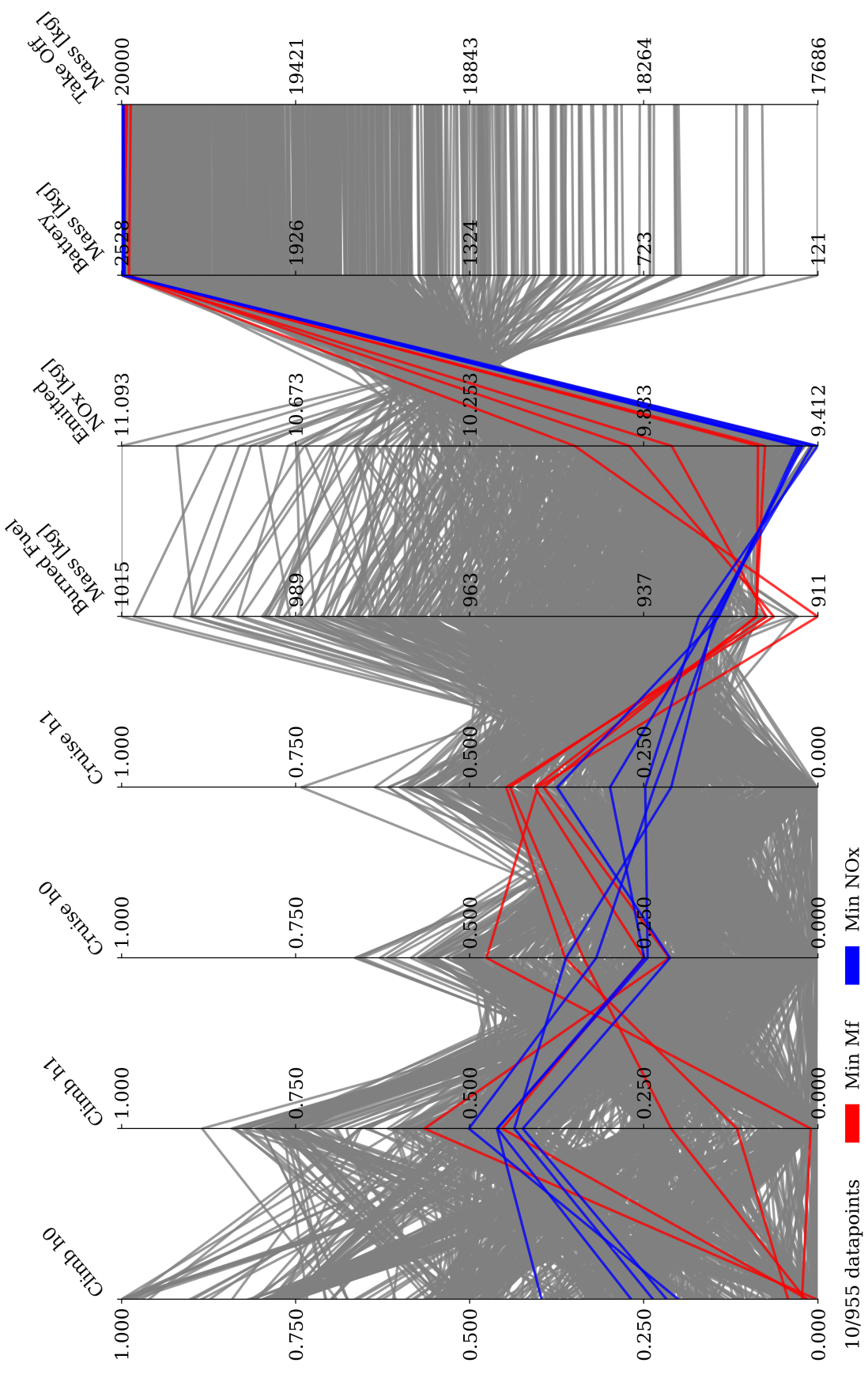


Figure 6.12: Pareto front of Experiment CC-L, highlighting the effects on the EMS of the trade-off between fuel consumption and NO_x emissions.

Point	TOM [kg]	M_{fuel} [kg]	$M_{battery}$ [kg]	M_{NO_x} [kg]	η_{total}
Baseline	17,568	1018.22	0	11.179	0.299
C-U, $min(M_f, M_{NO_x})$	19,998	908.30	2539.75	10.021	0.316
CC-U, $min(M_f)$	19,998	910.63	2537.65	9.978	0.316
CC-U, $min(M_f, M_{NO_x})$	19,992	924.52	2518.23	9.651	0.319
CC-U, $min(M_{NO_x})$	19,999	928.93	2520.33	9.405	0.326
TCC-U, $min(M_f)$	19,998	909.63	2538.67	10.004	0.320
TCC-U, $min(M_f, M_{NO_x})$	19,995	923.11	2522.46	9.728	0.322
TCC-U, $min(M_{NO_x})$	19,999	930.69	2519.03	9.398	0.332
CCA-U, $min(M_f)$	19,984	925.21	2508.95	9.937	0.328
CCA-U, $min(M_f, M_{NO_x})$	19,998	931.51	2516.56	9.423	0.328
CCA-U, $min(M_{NO_x})$	19,999	939.59	2510.19	9.269	0.342

Table 6.9: Detailed results of the points in Figure 6.11a.

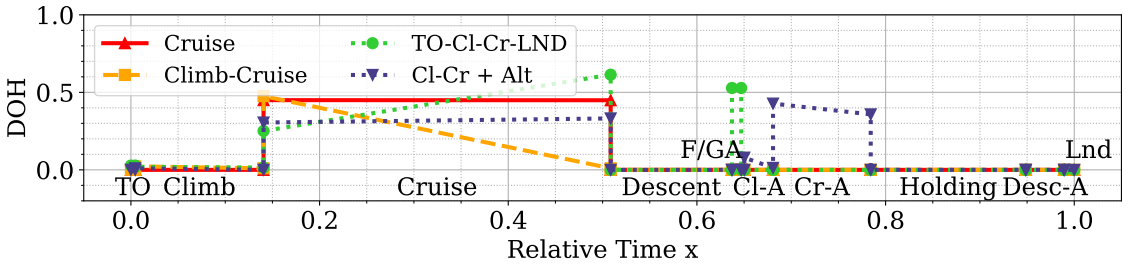
troducing electrical power in this phase relieves the gas turbine, which can operate at a lower power setting, reducing its emissions. This fact is ultimately the cause of the NO_x-M_f trade-off that has been observed when climb and cruise are hybridised. Prioritising NO_x over fuel consumption corresponds to distributing more electrical energy in the climb segment over the cruise segment. It is also visible by directly inspecting the energy management strategies, as shown in Fig. 6.13.

6.4.4 Effects of Maximum Take-Off Mass

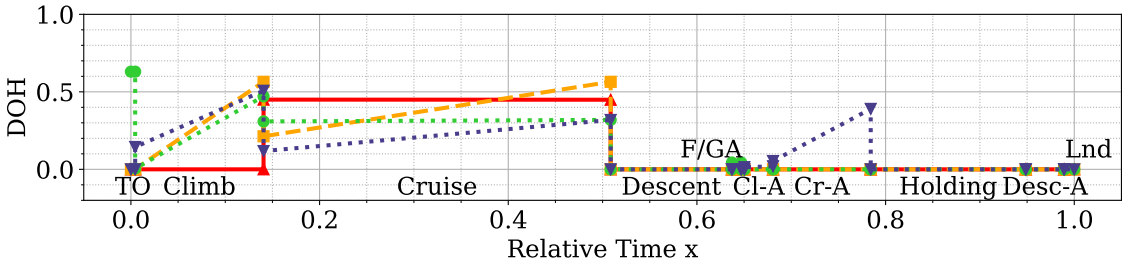
The data produced by P-DOPT contains suboptimal points and the global Pareto front. Therefore, it is possible to explore scenarios with lower values of maximum take-off mass. Figure 6.14 presents a family of Pareto fronts using data from experiments 6 and 10, where each front is the optimal energy management strategy available for a given take-off mass constraint. The fronts shrink and collapse into a single point as the maximum take-off mass limit is reduced. Figure 6.15 shows the energy management strategies associated with these Pareto fronts. The average degree of hybridisation is higher when the maximum take-off mass is greater, as more battery mass is available.

The maximum take-off mass prescribes the mass allocated to the battery pack. Since sizing is carried out to satisfy the EMS, the optimiser finds the highest possible degree of hybridisation for the same battery amount, minimising the fuel to be carried on board. Consequently, if the energy density of the pack was high, more electrical energy could be carried for the same mass, enabling a higher average DOH. Going the opposite direction, limiting the take-off mass reduces the feasible solution space the optimiser can operate. In this situation, the only possible solution would be trading the payload with batteries to recover the Pareto front.

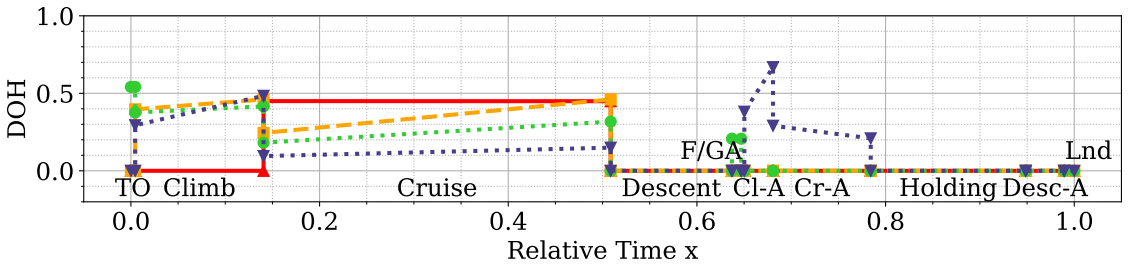
Given the trend, it might be assumed that increasing the maximum take-off mass limit of the aircraft would always reduce fuel consumption and overall emissions. However, it should be kept in mind that the model assumes a constant value of empty operating mass (OEM), as this study did not excessively increase the maximum take-off mass over the



(a) Mininum M_f



(b) Point closest to origin, marked with "X" in Figure 6.11b



(c) Mininum M_{NO_x}

Figure 6.13: Optimal energy management strategies of Figure 6.11b.

reference aircraft. Incrementing the take-off mass would require scaling the empty mass accordingly because the aircraft structure has to sustain higher loads in flight. Therefore, this trend is expected to have diminishing returns, as more batteries would require a heavier aircraft with higher production costs and more power to fly.

6.4.5 Suggested Strategies

According to the discussion of the results, four groups of optimum strategies were identified. Two for the Climb-Cruise with Linear parametrisation and two for the Climb-Cruise with Two Segments parametrisation case. These selections of groups of solutions are shown along the trade-off in Fig. 6.16 and also in the corresponding Parallel Coordinates plots in Fig. 6.17 and Fig. 6.18. These selections highlight the effects of the trade-off between the two objectives.

The shapes of each respective energy management strategy are shown in Fig. 6.19 and Fig. 6.20, respectively. Average trends were identified from these selected points, shown in Fig. 6.19 and Fig. 6.20 with a black dashed line. These average strategies have been simulated and drawn in the objective space along their original samples (see the crosses

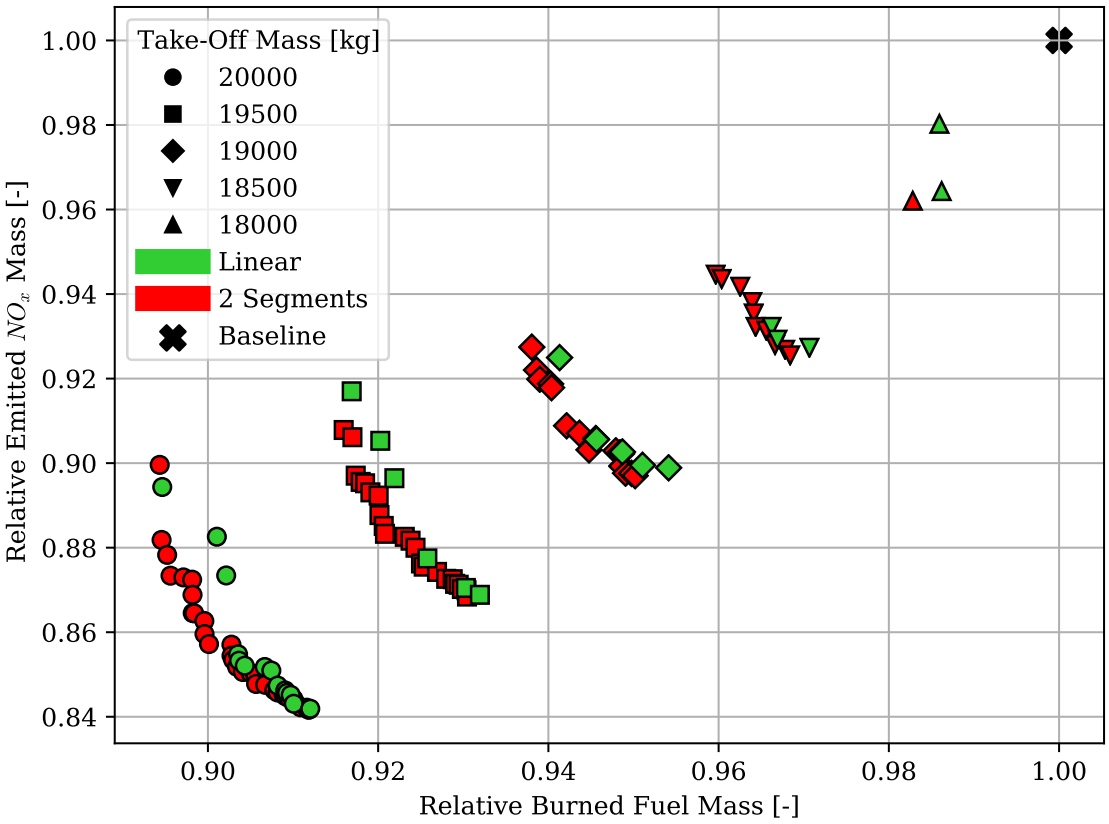


Figure 6.14: Pareto fronts for different Take-off Mass constraints.

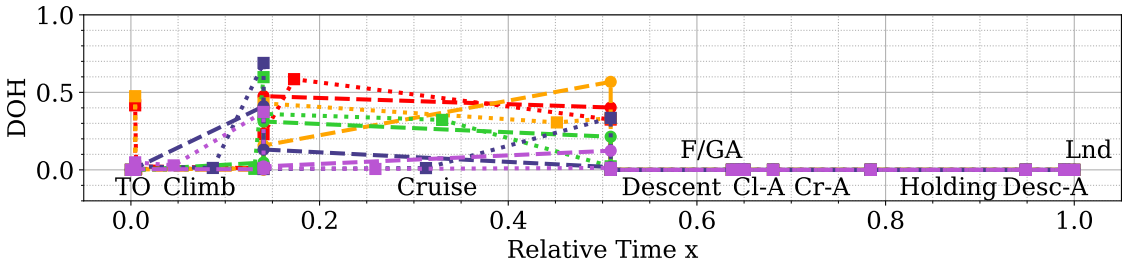
in Fig. 6.16).

While some of the averaged strategies do not fall within the sampled group due to the nonlinearity of the simulation, these points are nonetheless close to the original selection and follow their respective trends within 1% of discrepancy for both objectives.

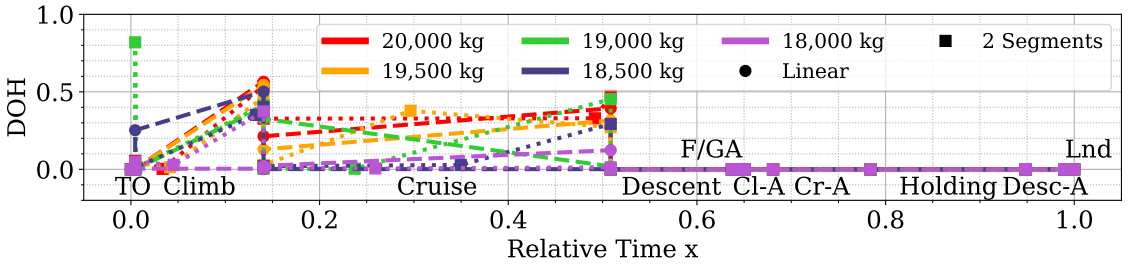
Three possible general shapes are indicated for the energy management strategy. As is evident from Fig. 6.16, the points of selection 1 of Experiment 6 and selection 2 of Experiment 10 fall within the same target area. Their respective energy management strategies are similar, except for an impulse at the beginning of the climb phase (Fig. 6.20b), an artefact of the optimisation algorithm. Another point of note is the steep climb after a late introduction of electric power in the climb phase of Selection 1 of Experiment 10 (Fig. 6.20a). This behaviour allows the two-segment strategies to expand the Pareto front in this region and dominate linear strategies. Improvement in M_{fuel} results from increased battery usage during the cruise phase. Finally, the proposed general shapes in detail are:

1. Linearly increasing DOH in climb, starting from zero, and constant DOH in cruise (Fig. 6.19a and Fig. 6.20b). This strategy is balanced between the two objectives and represents a good compromise.

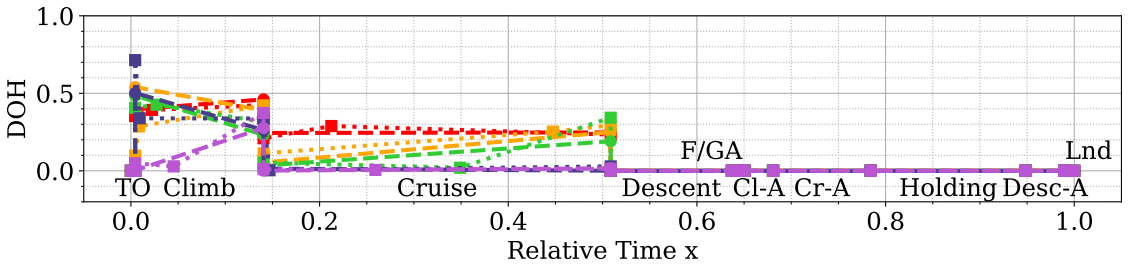
2. Late deployment of electric power in climb, with a steep linear increase in DOH, and moderately increasing DOH in cruise (Fig. 6.20a). This strategy prioritises the reduction of fuel consumption over emissions of NO_x, as more battery capacity is allocated on cruise.



(a) Mininum M_f



(b) Central point of the Pareto Front

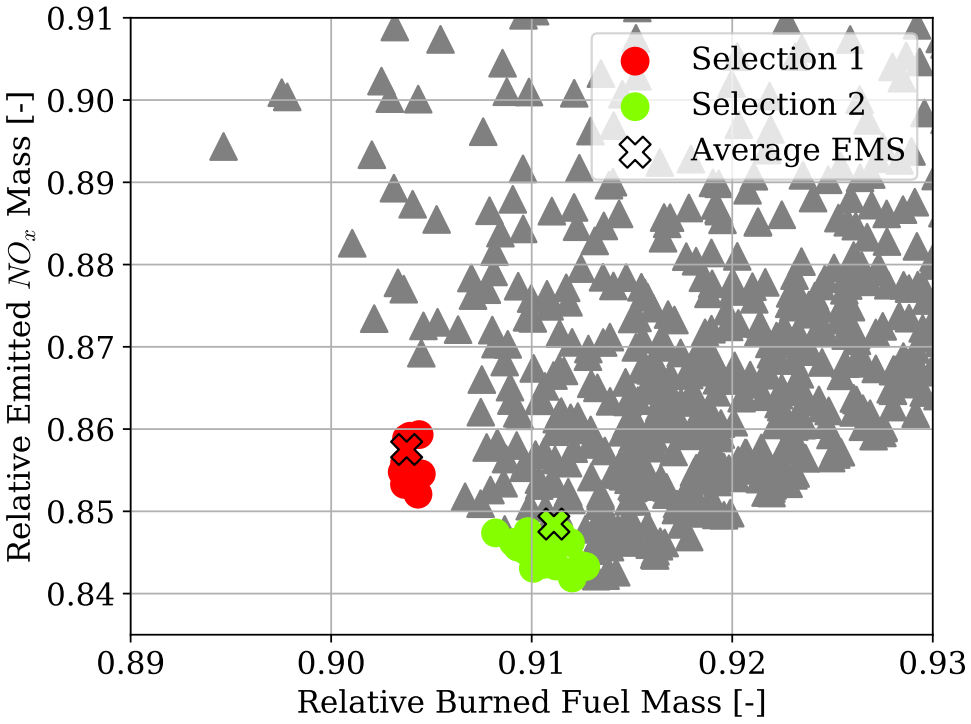


(c) Mininum M_{NO_x}

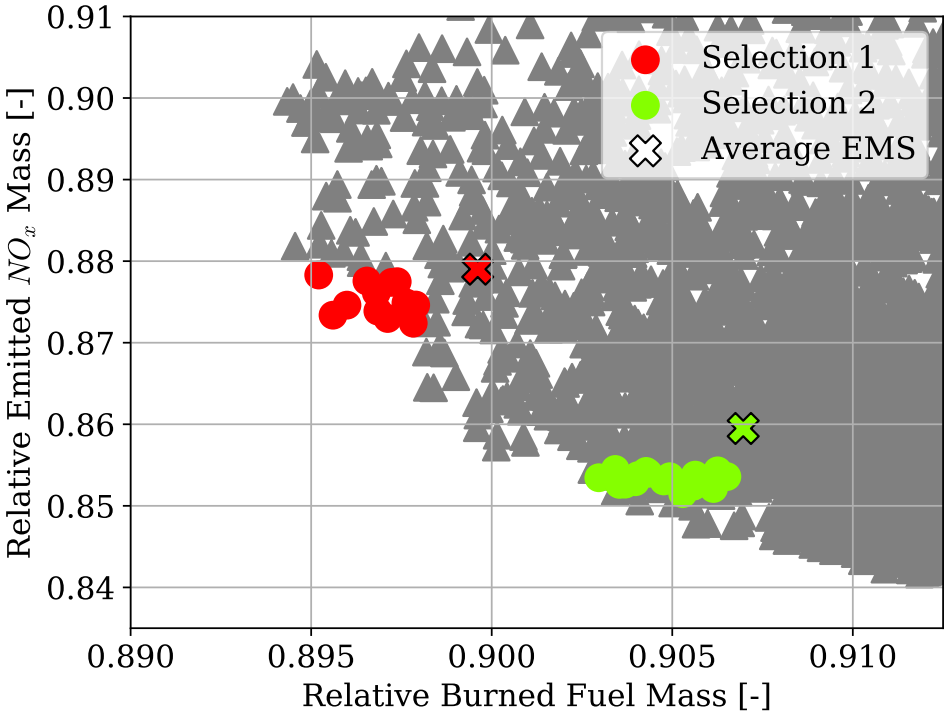
Figure 6.15: Energy management strategies for different take-off mass constraint levels.

3. Moderate increase in DOH in climb, starting from a nonzero value, and moderate increase in DOH in cruise (Fig. 6.19b). This strategy prioritises the reduction of NO_x over fuel consumption, as more battery capacity is allocated in climb.

No specific degree of hybridisation values are provided for these strategies because, as discussed in the previous section, the amount of battery mass available determines the average degree of hybridisation. The DOH values of these strategies are determined by the 20000 kg take-off mass limit adopted in this study. Nevertheless, it is possible to summarise two general trends. First, the shapes of the DOH functions are either constant, increasing, or a mixture of the two. Secondly, the trade-off between NO_x emissions and burnt fuel directly correlates with the climb or cruise hybridisation. Finally, the results indicate that the best parametrisation is the two segments piece-wise linear one since it can cover the entire Pareto front with the least number of parameters.



(a) Experiment 6.



(b) Experiment 10.

Figure 6.16: Points selected for the recommended strategies.

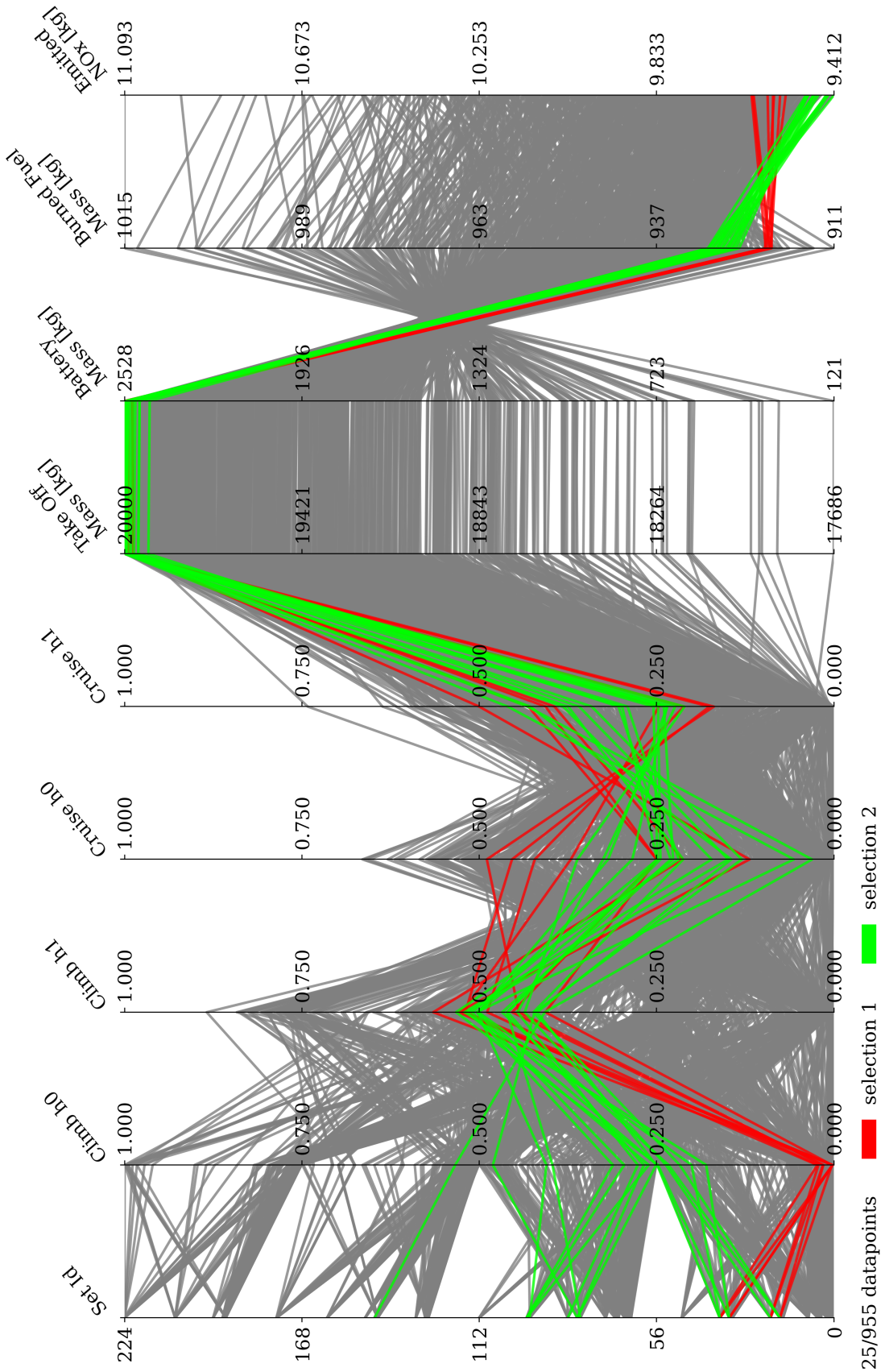


Figure 6.17: Parallel coordinates visualisation of Experiment 6.

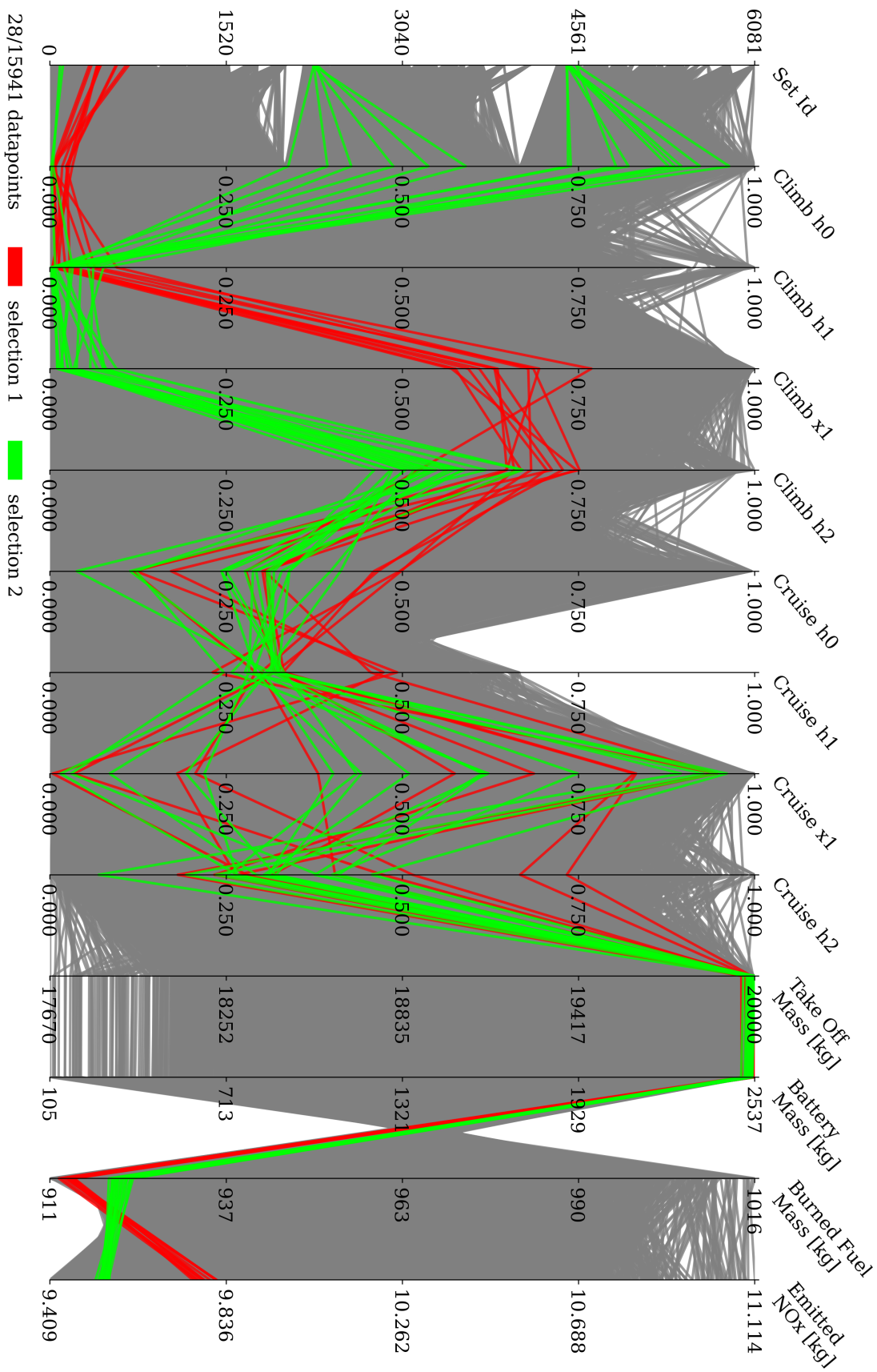
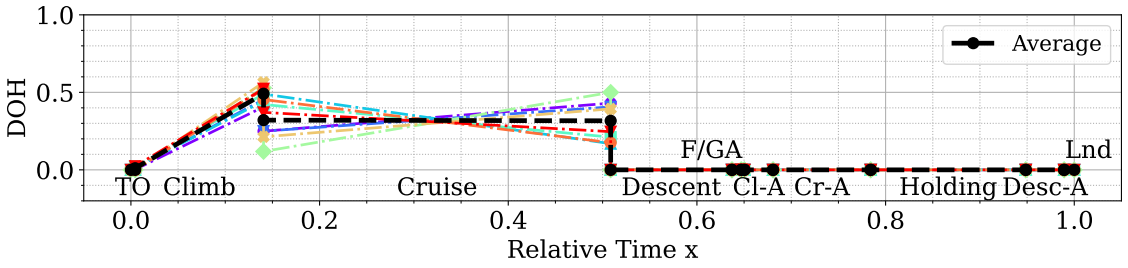
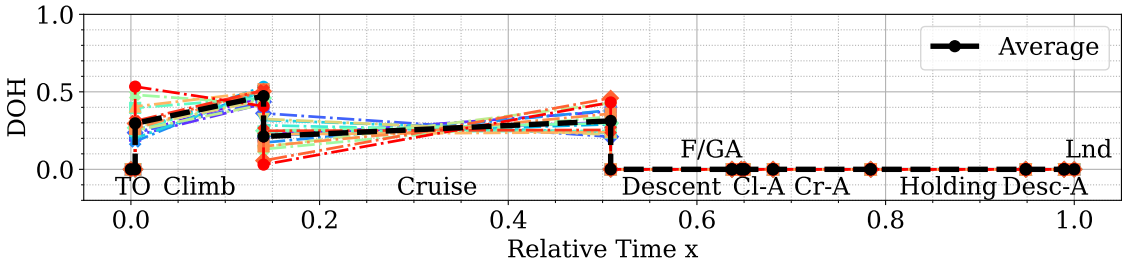


Figure 6.18: Parallel coordinates visualisation of Experiment 10.

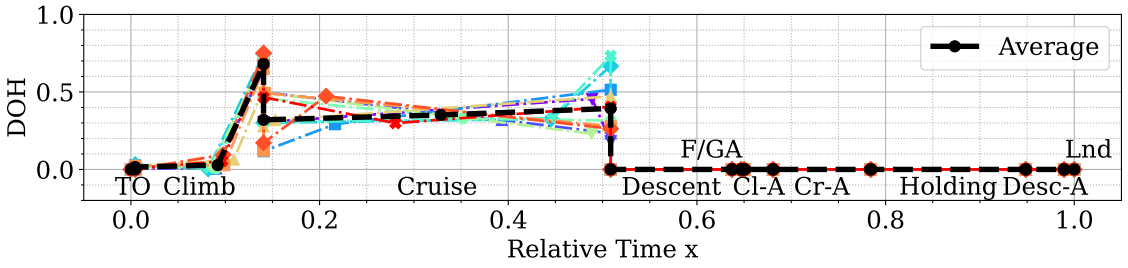


(a) Selection 1

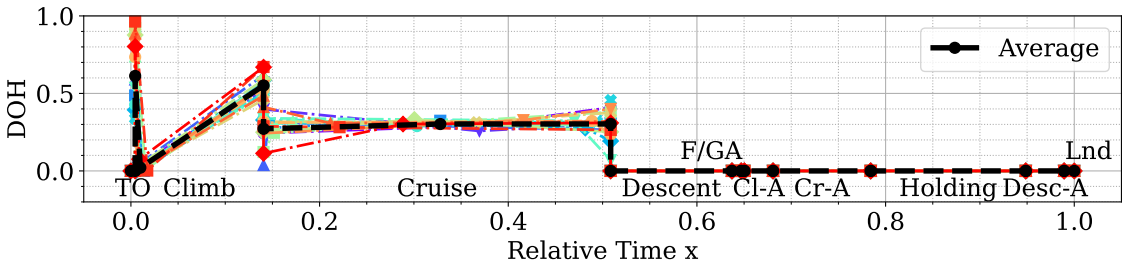


(b) Selection 2

Figure 6.19: Visualisation of Experiment 6 energy management strategies



(a) Selection 1



(b) Selection 2

Figure 6.20: Visualisation of Experiment 10 energy management strategies.

6.5 Summary of Chapter

The test case presented in this chapter carried out a thorough design space exploration of the energy management strategy of a hybrid-electric regional aircraft. The goals were to understand the interaction between EMS and aircraft performance and to test the P-DOPT framework on a more complex problem. Multiple computational experiments were

performed and compared, first by changing the number of parameters of the EMS and then by changing the number of hybridised mission segments.

As demonstrated previously, the framework could reduce the total computational time required to perform the design space exploration analysis by 75%. Introducing "soft constraints" on the objectives allowed further refinement of the search but also pointed out the need for tuning studies to understand the best setting for the methodology hyperparameters, specifically, the number of levels per parameter and the minimum satisfaction probability in the exploration phase constraints.

A flexible EMS parameterisation based on linear segments was adopted for searching the design space. It was found that the parametrisation does not affect the results directly, as EMS with higher degrees of freedom converged towards the same topology. Indeed, points that were close in the objective space had similar EMS shapes regardless of the parametrisation. Therefore, two linear segments per mission phase were deemed sufficient to describe any EMS, although a single linear segment would be sufficient for most cases in practice.

The analysis also brought about a better understanding of the distribution of electrical energy across the mission phases. A trade-off between the two objectives, fuel consumption and NO_x emissions, was present when at least two phases were hybridised. Furthermore, the relative weights corresponded to the repartition of electrical energy between climb and cruise. Providing more electrical power on cruise would yield lower fuel consumption, while more power distributed in climb, and other minor segments, produces less NO_x pollution. In addition, the maximum take-off mass defines the size of the available solution space, limiting the amount of mass allocated to batteries on board.

Finally, with the available data, some suggestions were made on the topological layout of the EMS. The following chapters will extend this work by considering multiple operating conditions and the degradation of the battery pack.

6.6 Key Learning Outcomes

- Two linear DOH segments are sufficient to define the EMS for a mission phase.
- The parameterisation complexity does not affect the objectives significantly: designs close to the objective space have similar EMS topologies.
- The highest reduction in fuel consumption is achieved by concentrating the electrical energy in the longest mission segment (cruise).
- On the contrary, the highest reduction in NO_x emissions is obtained by distributing the electrical energy in the most power-consuming phases, especially in climb.
- The trade-off between the two objectives is present when more than one mission segment is hybridised: climb and cruise.
- Maximum take-off mass affects the solution space; the lower it is, the smaller the average DOH.

Chapter 7

Regional Aircraft Operations

7.1 Introduction

The battery pack model used in the previous two test cases could be defined as a ”rubber” model; that is, it would ”stretch” to precisely the required amount of energy the EMS sets. Battery packs are composed of cell modules specifically designed to meet the system's required capacity and voltage. Consequently, merely changing the energy needed from the battery could lead to a significant change in mass.

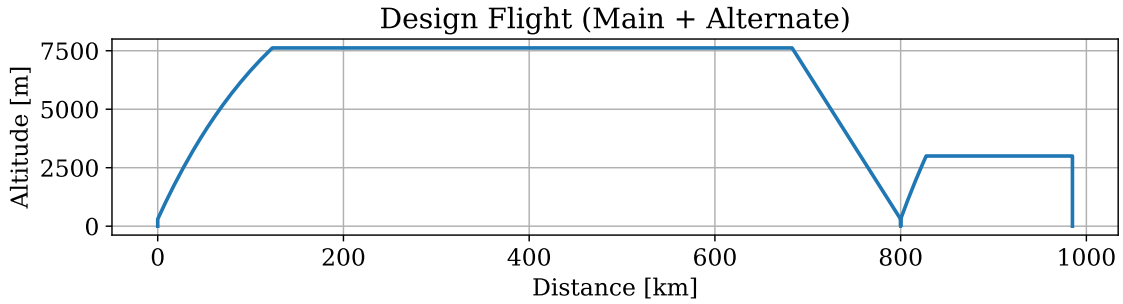
Additionally, the previous cases only considered a single design mission when analysing the propulsion system's performance. However, commercial aircraft operate in disparate missions, which may differ from the intended design condition. In the case of a hybrid-electric aircraft, the battery pack is sized with a certain distance in mind, but the operator could fly longer or shorter missions. For example, short-haul operators fly multiple flights on the same day with short stopovers. Since this dissertation focuses on regional hybrid-electric aircraft, it is reasonable to analyse a typical use case.

The test case in this chapter attempts to overcome both limitations. The battery pack model is updated, introducing cell dynamics and a more granular sizing method. Then three different mission scenarios are analysed: a design mission which sizes the battery pack and two operating conditions (one with turnarounds and one without) where the battery pack is fixed. The results presented in this chapter were originally published at the EASN 2022 conference (211).

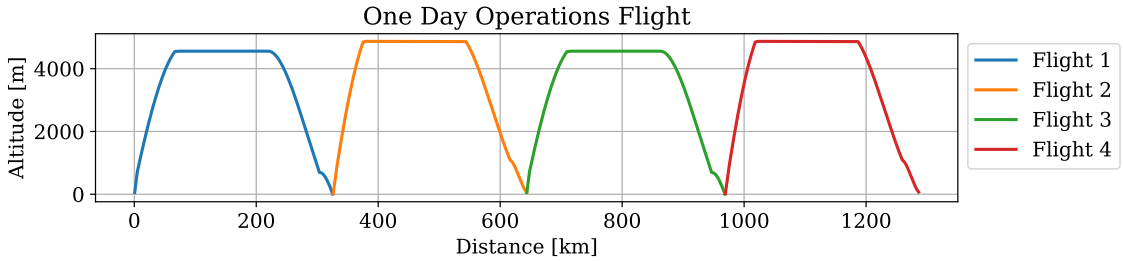
7.2 Problem Definition

This test case includes three missions: a design mission to size the battery pack and two operating missions to explore feasible energy management strategies under off-design conditions. Figure 7.1a presents the flight profile of the design mission, the same as that used in the previous chapter (taken from (194)). For the scope of battery pack sizing, only the main portion is hybridised with electrical power.

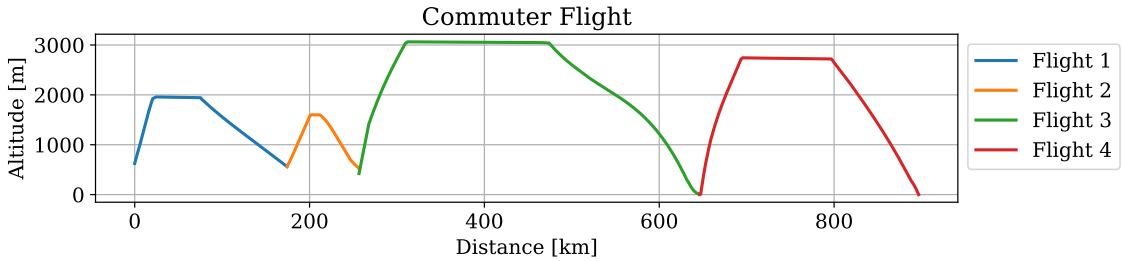
The mission presented in Fig. 7.1b is modelled after a one-day operation of an ATR-42 between Aberdeen and Sumburgh, operated by Loganair in Scotland. Two sets of return trips are carried out, one in the morning and one in the afternoon. Two scenarios will be analysed for this mission: operating with a full overnight recharge only and allowing



(a) Design Mission



(b) One-Day Operations Mission

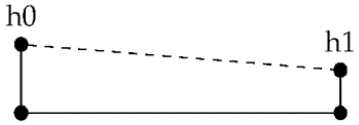


(c) Commuter Flight Mission

Figure 7.1: Missions considered for this study.

a partial recharge after the morning flights. Refuelling is allowed because the aircraft is stopped for a minimum 30-minute turnaround.

Finally, the mission presented in Fig. 7.1c is modelled after a commuter flight with multiple stops in Portugal, between Bragança and Portimão, operated by the airline Sevenair. In this scenario, the aeroplane does not stop for a turnaround but embarks and disembarks passengers. Hence, no refuelling is permitted during the stops, and the battery is recharged only at the end of the day.



(b) Linear DoH

Figure 7.2: Definition of the Energy Management Strategy function

Given the results of the previous chapter, the energy management strategy is defined

as a linear function of the degree of hybridisation applied only to the climb and cruise phases. Therefore, the EMS function is defined by four parameters, as shown in Fig. 7.2. For multiple flights on the same mission, the same EMS is used for all flights.

Equation (7.1) presents the optimisation problem formulation. The framework's objective is to find the energy management strategies that produce the lowest fuel consumption and NO_x emissions, given the constraints on the take-off mass (TOM) and depth of discharge (DOD) of the battery pack. Although most of the problem is similar to that in Chapter 6, the additional constraint on DOD was introduced to avoid over-discharging the battery, which would significantly degrade its capacity over time. The selected probability for the exploration phase is 0.5, while the input parameters were discretised with six levels each, producing a total of 1296 different subspaces.

$$\begin{aligned} &\text{given} && X = \{DOH0_{cl}, DOH1_{cl}, DOH0_{cr}, DOH1_{cr}\} \\ &\underset{X}{\text{minimise}} && M_{fuel}, M_{NO_x} = f(X) \\ &\text{subject to} && TOM \leq 20100\,kg \\ & && DOD \leq 0.8 \end{aligned} \tag{7.1}$$

The results of the objectives are presented relative to the baseline, which is the emitted NO_x and fuel consumption of the aircraft without electric propulsion. These values can be found in Tab. 7.1 for the three missions.

	Fuel Consumption [kg]	NO_x Emitted [kg]	Take-Off Mass [kg]
Design Mission	1242.43	8.59	17792
One Day Operations	1421.21	11.09	16892
Commuter Flight	774.24	4.72	17330

Table 7.1: Baseline Data

7.3 System Model

The mission analysis method of the previous chapter has been modified, replacing the calculation of the battery pack mass using the energy density with a subroutine that sizes the geometry of the battery pack. Furthermore, a simulation mode is introduced where the battery pack geometry is given, and the code discharges the cells returning the final state of charge, indicating if the EMS is feasible or not.

Figure 7.4 shows the geometrical layout of the pack. The cell model is a Thevenin equivalent circuit, whose dynamics are described by (7.2). The characteristic equations of the lumped components V_{OC}, R_0, C_t, R_t have been provided by Ceraiolo et al. (212), based on experimental data on high capacity lithium-ion cells.

$$\begin{aligned}
&\frac{dV_C}{dt} = \frac{I_{cell}(t)}{C_t} - \frac{V_C(t)}{R_t\, C_t} \\
&\frac{dDOD}{dt} = \frac{I_{cell}(t)}{E_{cell}} \\
&U_{cell} = V_{OC} - V_t - I_{cell}\, R_0 \\
&P_B = U_{cell}\, I_{cell}\ \ (N_{series} \cdot N_{modules})
\end{aligned} \tag{7.2}$$

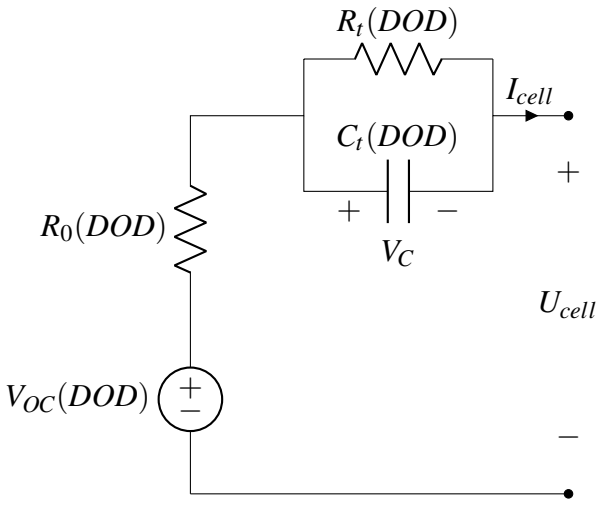


Figure 7.3: Thevenin Electric Model for a Battery Cell.

In order to match the expected pack energy density of 500 Wh/kg, the cell's mass has been scaled by calculating the equivalent cell energy density. The scaling equation is as follows:

$$M_{cell} = E_{cell} \cdot e_{cell} = E_{cell} \cdot k \cdot e_{pack} \tag{7.3}$$

where E_{cell} is the nominal cell energy in kW, k is the pack-to-cell mass ratio that accounts for the mass overheads and e_{pack} is the pack energy density. The relationship between pack energy density and cell energy density can be derived starting from the formula for calculating the battery pack mass:

$$M_{pack} = k \cdot N_{cells} \cdot M_{cell} \tag{7.4}$$

where N_{cells} is the number of cells in the battery pack. By replacing the definition of energy density $e = E/M$ and remembering that the electric energy of the pack is equal to $E_{pack} = N_{cells} \cdot E_{cell}$, Equation 7.3 is obtained with a few algebraic steps.

Sizing consists in finding the number of cells and their geometric layout, shown in Fig. 7.4. The cells are first connected in series to meet the system's voltage demand: these subunits of N_{series} cells are called modules. Then, the modules are connected in parallel such that their capacity meets the mission requirements: $N_{modules}$ in total. The product of the two quantities is the total number of cells. The complete procedure is as follows:

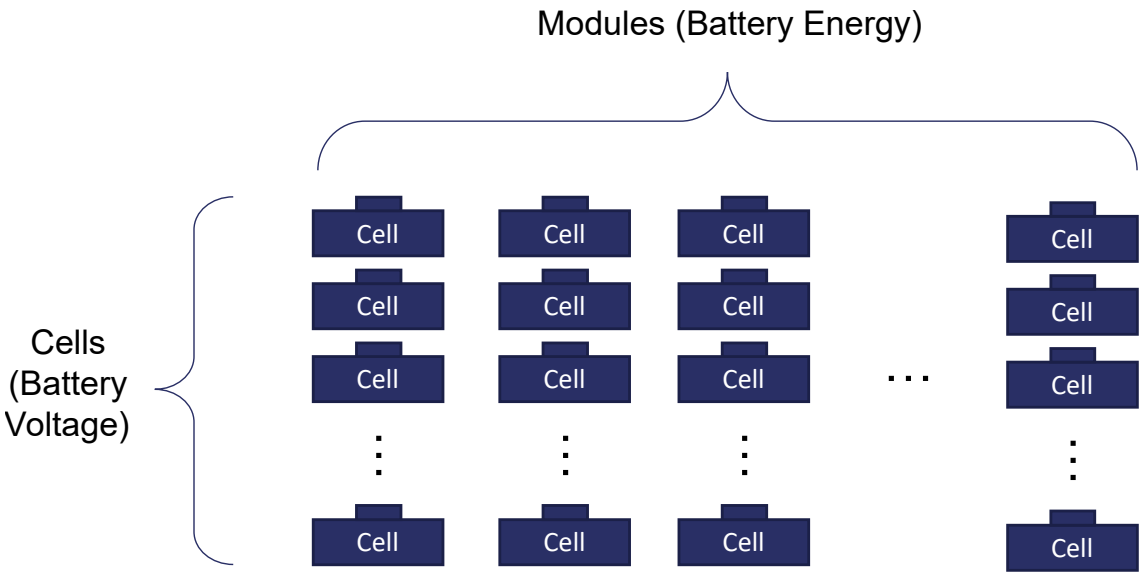


Figure 7.4: Battery pack geometric layout.

1. Calculate the number of cells in series for each module to meet the nominal system voltage: $N_{series} = \frac{V_{sys}}{V_{cell}}$
2. Increment the energy required to fly the mission by a guessed factor m: $E_{total} = m \cdot E_{mission}$
3. Calculate the number of modules to meet the required energy: $N_{modules} = \frac{E_{total}}{E_{cell}}$
4. Simulate the cell discharge by solving the ODE system in Eq. 7.2 after dividing the pack power by the total number of cells $P_{cell} = \frac{P_B}{N_{series} N_{modules}}$
5. Repeat steps 2-4 by changing m, until the pack depth of discharge matches the target of 80%.
6. Calculate the total mass of the pack by multiplying the number of cells by the cell mass (see (7.4)). This value is then incremented by the pack-to-cell mass factor k to account for the mass overheads caused by the wiring and casing.

With the battery pack adequately sized, the circuit model is simulated to evaluate the efficiency of the battery pack and its discharge characteristics. The pack-to-cell mass factor k has been set at 1.5, as has been used in previous studies (213). Estimating this value is difficult as most electric aeroplanes of this size are just conceptual studies. This factor is related to the safety of the battery pack as the overheads account for the case design, which must manage thermal runaways from the cells and contain any fire in case of faults.

The aircraft model is similar to the previous test cases, a parallel hybrid-electric 50-passenger turboprop. Table 7.2 presents the details of the system's parameters, including information on the cells and the battery pack. The data used to model the gas turbine are the same TURBOMATCH surrogate model adopted in the previous chapters.

This test case is carried out in two steps. First, the battery pack is sized with the design mission, and then the optimal pack layout is selected from the available results. Then,

Table 7.2: Parameters of the Hybrid-Electric Aircraft

Maximum Take-Off Mass [kg]	21000	Electric Motor Efficiency	0.98
Operating Empty Mass [kg]	11550	Power Electronics Efficiency	0.985
Payload Mass [kg]	5000	Cable Distribution Efficiency	0.996
Gearbox Efficiency	0.99	Pack-to-Cell Mass Ratio k	1.5
L/D (Climb)	16	Pack Energy Density [Wh/kg]	500
L/D (Cruise)	14.5	Cell Total Capacity [Ah]	7.2
Propeller Efficiency (Climb)	0.73	Cell Nominal Voltage [V]	3.7
Propeller Efficiency (Cruise)	0.86	System Nominal Voltage [V]	500

the optimisation framework finds the optimal energy management strategy in missions different from the size one. In this situation, the battery pack will operate in an off-design condition, and the EMS is tasked with recovering as much optimal performance as possible without over-discharging the cells. Three scenarios in total are analysed: the Scottish regional aircraft mission with and without recharging and the commuter flight in Portugal.

7.4 Results and Discussion

7.4.1 Design Point

Figure 7.5 presents the full results of the sizing analysis. Several local Pareto fronts have been identified, as shown in Fig. 7.6a. Most share the same battery pack charge values, with the same number of cells and layout (see Fig. 7.6b). Since the system voltage is fixed, the difference between the layouts is in the number of modules composed of 136 each. Despite the different EMS, the Pareto front points share the same battery pack layout (309 modules). This arrangement is selected and maintained for the subsequent operating conditions. The total capacity of the selected battery is 2225 Ah with a tension of 500 V.

A point of note is the lower objective improvements compared to the results in Chapter 6. Indeed, it is evident from comparing the Pareto front of Fig. 7.6a to that of Fig. 6.8b. The more detailed battery pack model reduced the improvement in fuel consumption from 9% to 4%, and likewise, the reduction in NO_x emissions went down from 14% to 3.5%. One possible explanation is the lower mass of the pack in this test case (2239 kg) than in the previous one (2524 kg), hence less energy available for hybridisation. This limitation is likely caused by the dynamics of the cells, where the depth of discharge has been limited to 80% to avoid accelerated damage and degradation at the cost of losing some of the available capacity.

Consequently, the EMS have topologies different from those seen in Sec. 6.4.5. In climb, the DOH decreases rather than increases, while in cruise, it is the opposite. Only the EMS that prioritises reduction in fuel consumption presents an increasing climb in DOH.

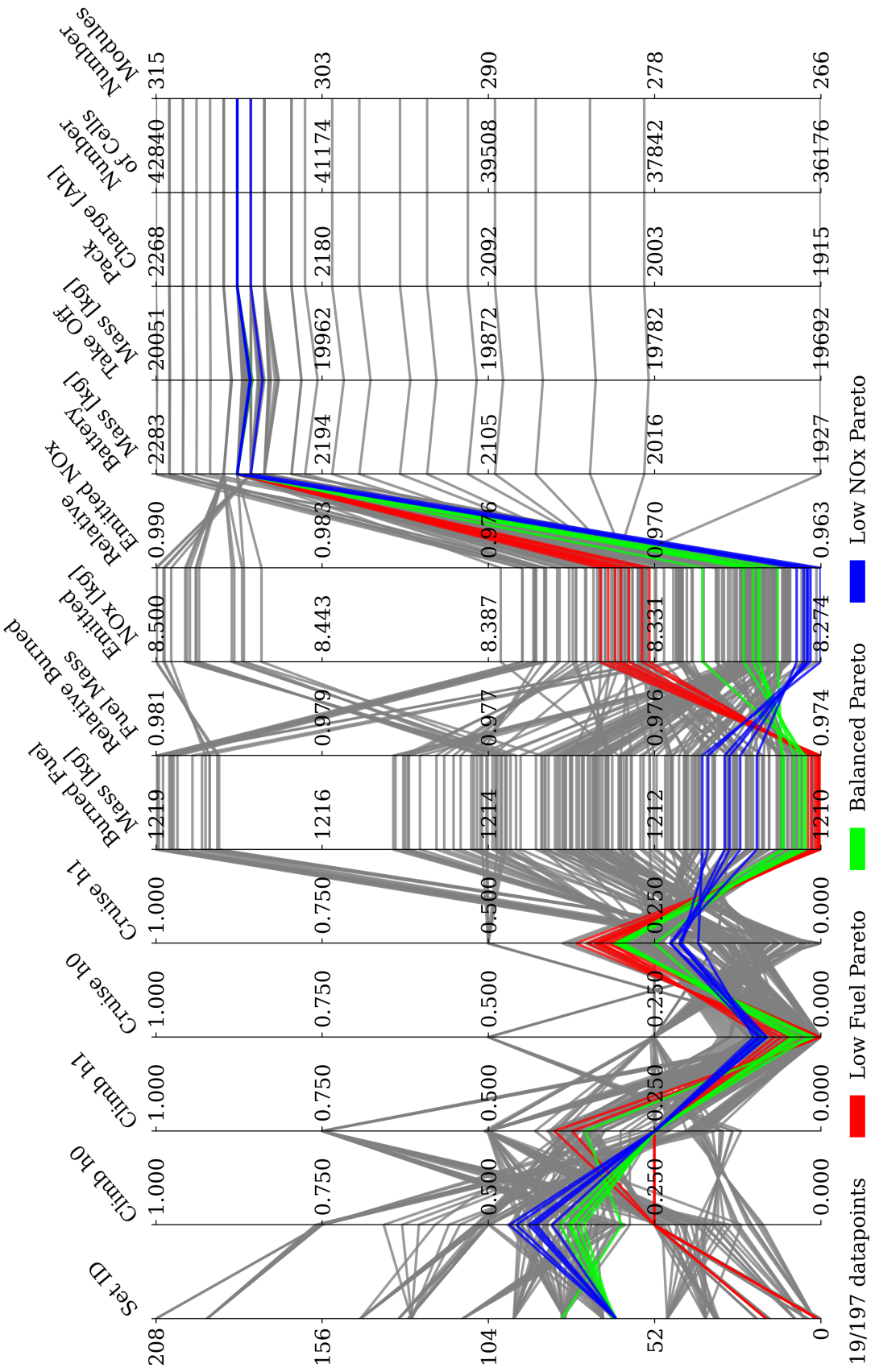
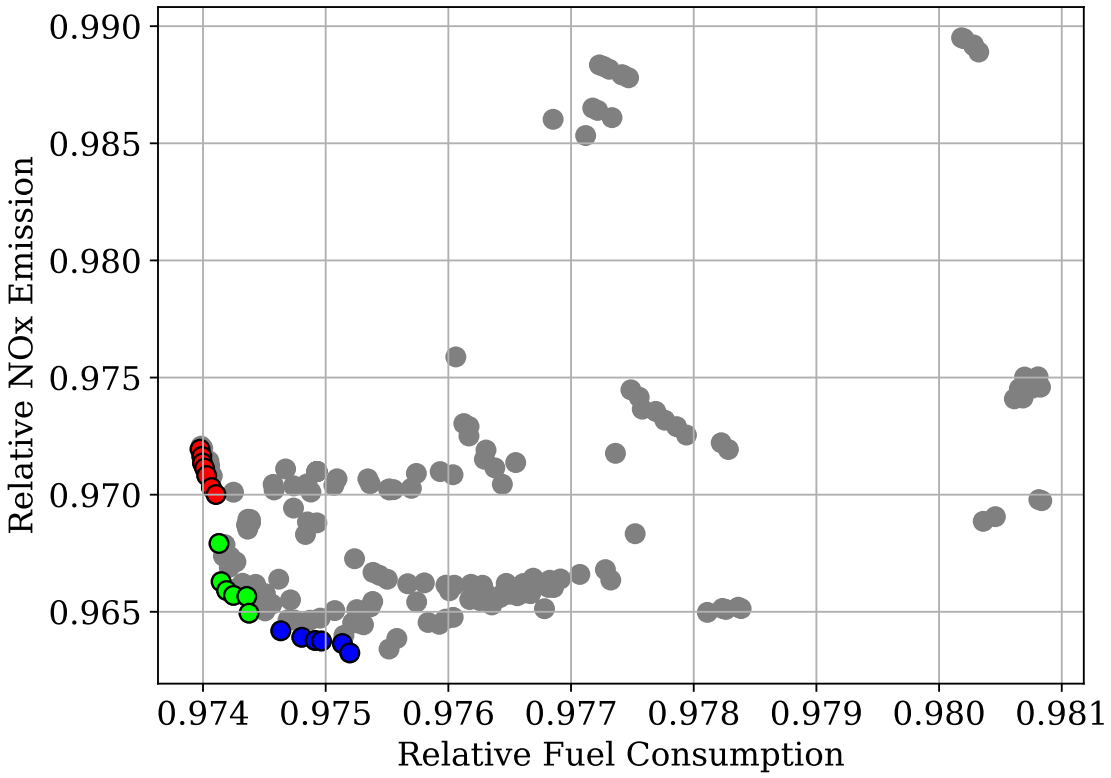
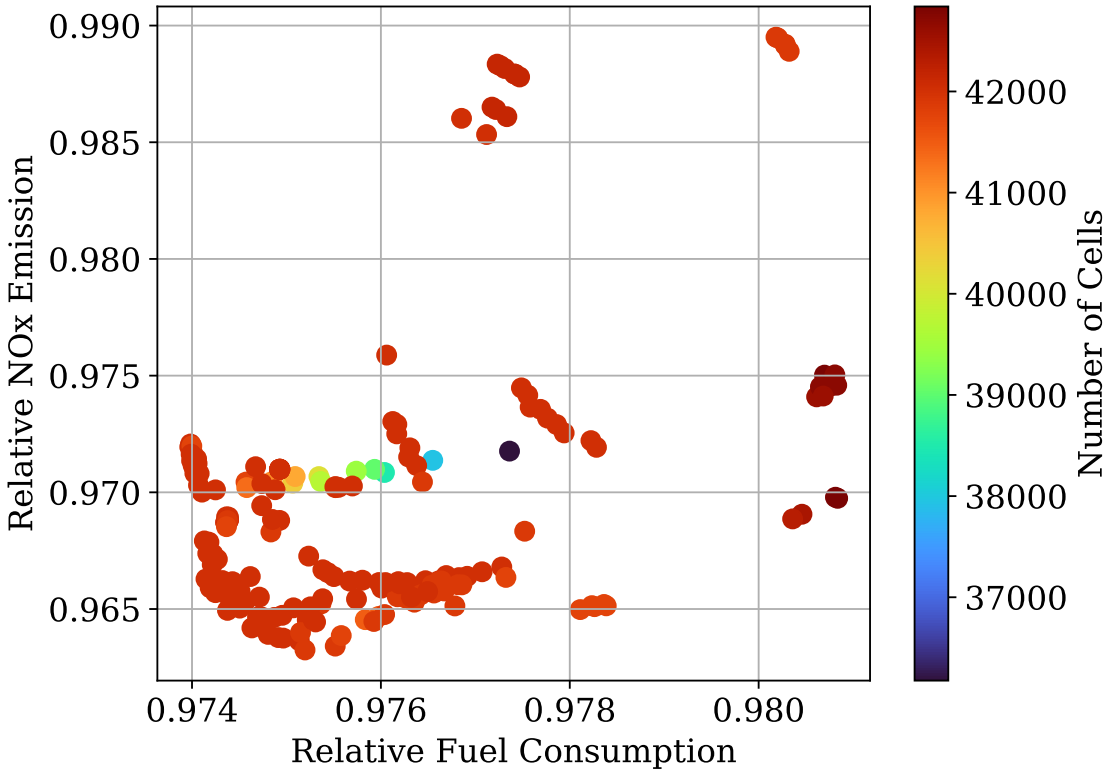


Figure 7.5: Parallel coordinates visualisation of Design condition.



(a) Objective space of the Design condition with Pareto front.



(b) Number of cells for the Design condition points.

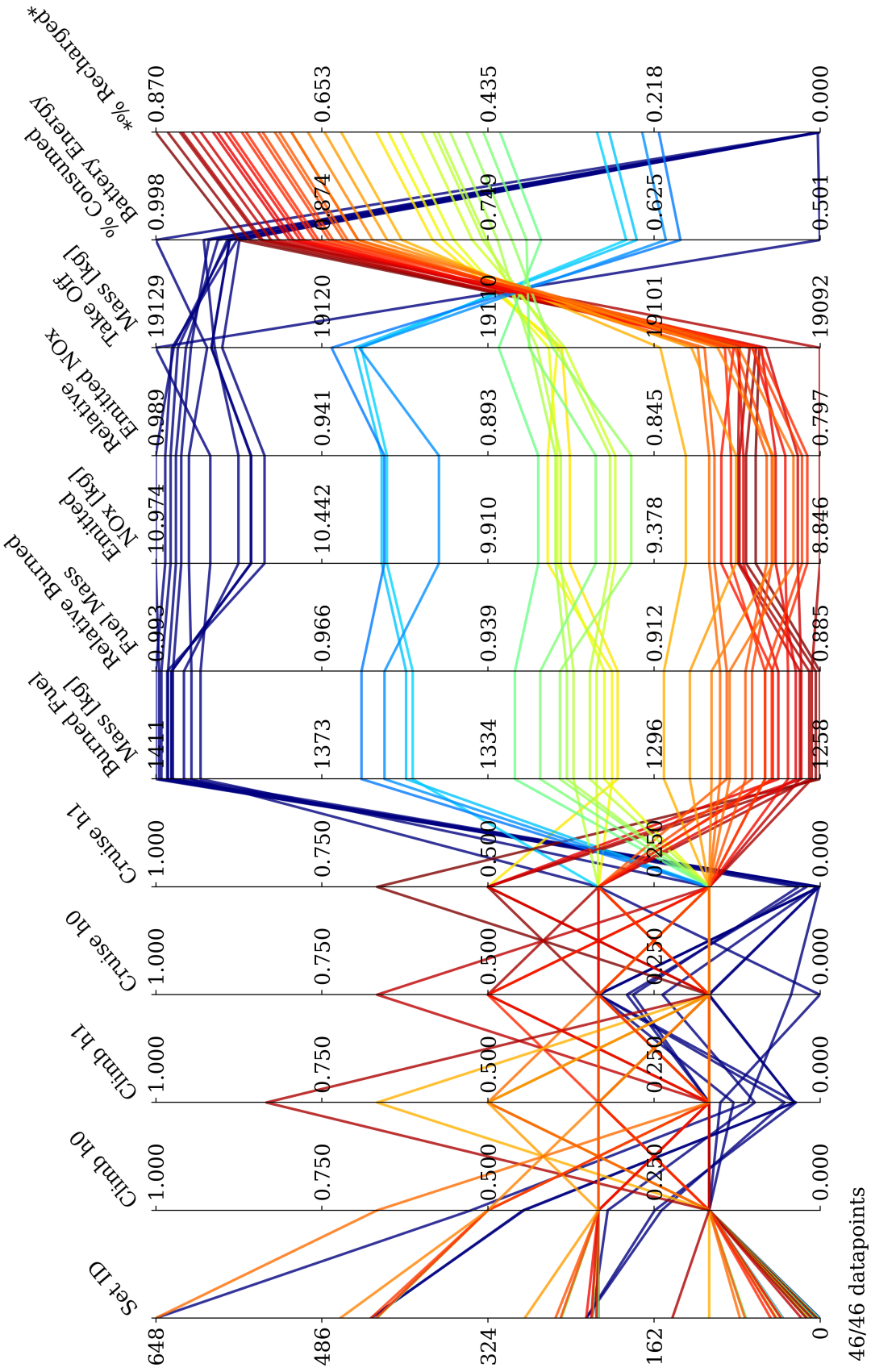


Figure 7.7: Parallel coordinates visualisation of one day regional operations.

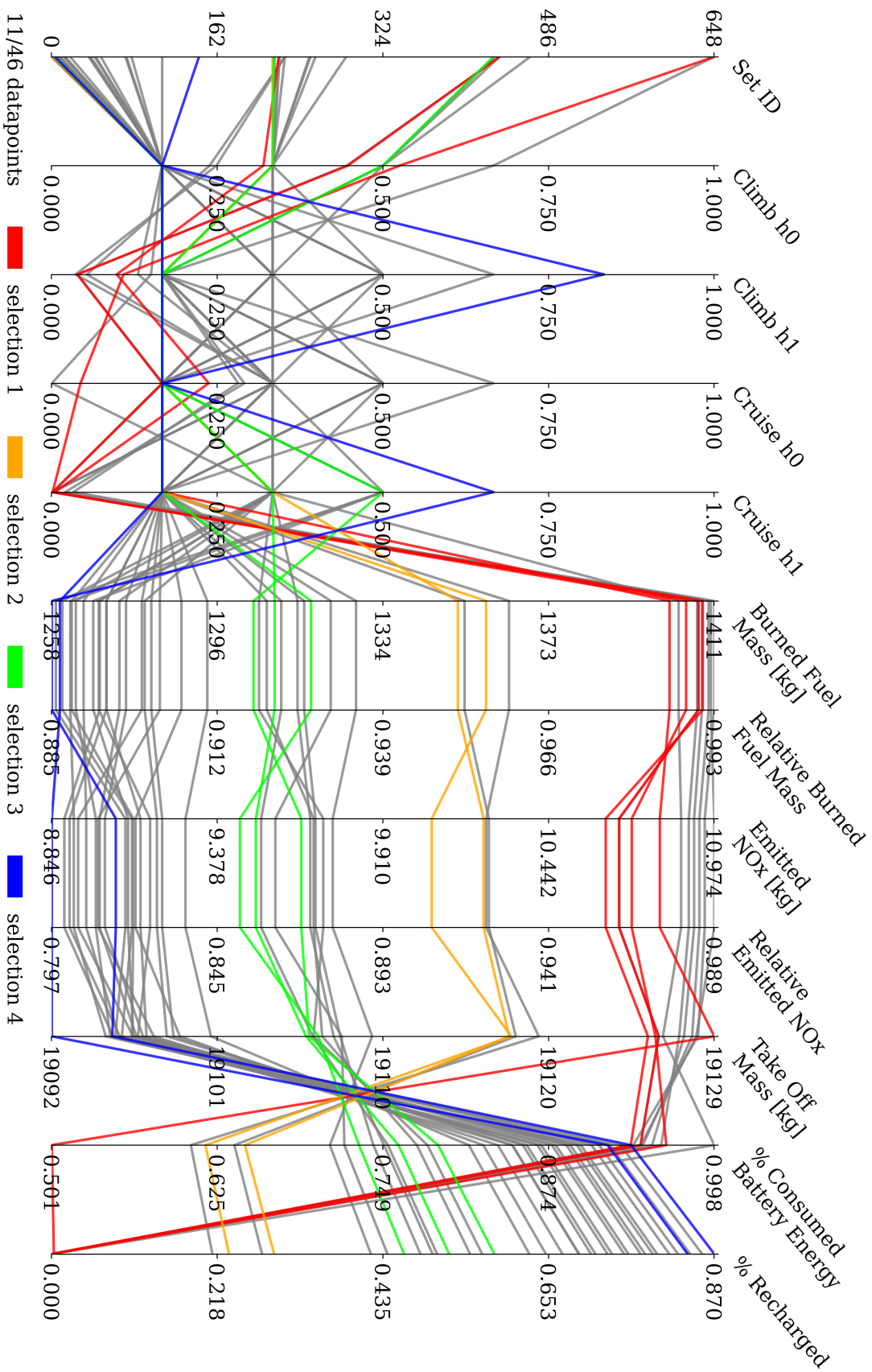


Figure 7.8: Local Pareto fronts of the one day regional operations.

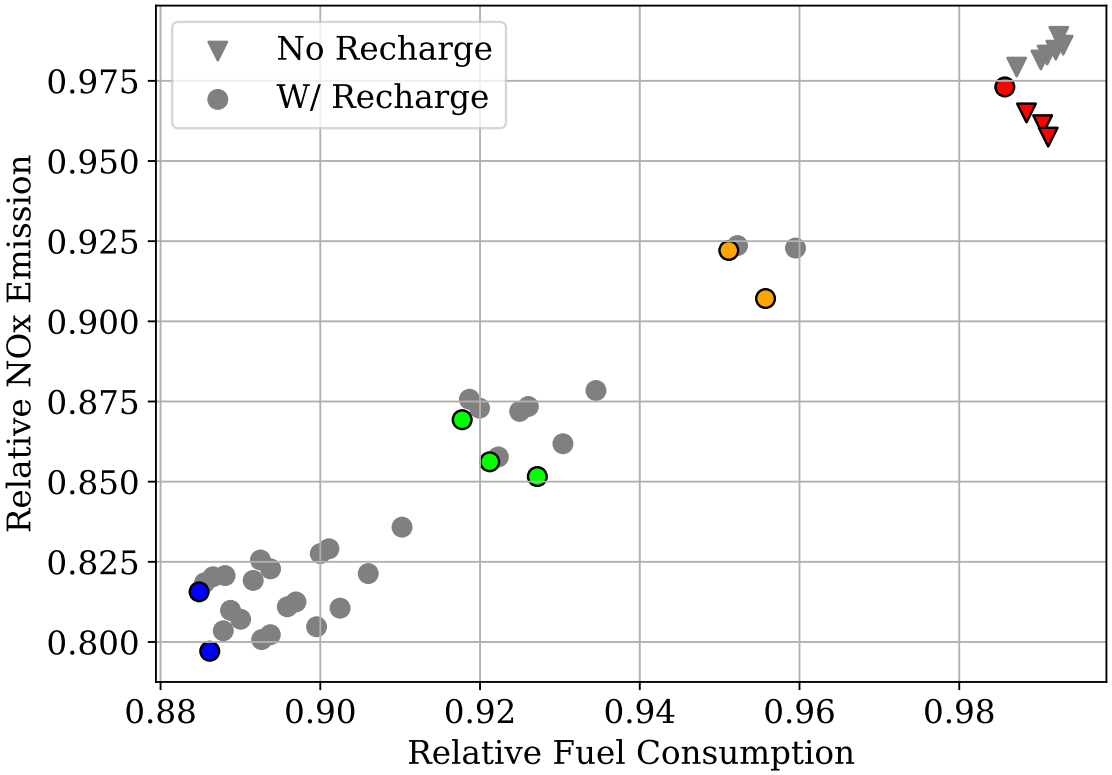


Figure 7.9: Objective space of the one day regional operations.

7.4.2 One-Day Operations

The parallel coordinates plot of Fig. 7.7 combines the results of the EMS capable of carrying out the four regional flights without recharge and those that allow partial recharge after the first two. The objective space of Fig. 7.9 shows them. Few points, represented by triangle markers, can carry out this mission since the combined length of the four flights is longer than the design mission. The improvement over baseline is limited, with 3% on NO_x emissions and 1.5% on fuel consumption.

This limitation can be overcome by partially recharging the battery after morning operations (the first two flights). The amount of energy reintroduced is required to complete the last two flights, ending the day with a fully discharged battery. The newly identified operating condition performs better in emissions than the previous ones: for instance, restoring 40% of the total battery energy reduces fuel consumption and the NO_x emissions by 6% and 12%, respectively.

However, charging and discharging cells from a partial state of charge increases the ageing of the battery pack (214), which would shorten the lifespan of the electric propulsion system. This condition is also exacerbated by the large amount of current that moves through the cells at every cycle. Further investigation of the degradation effects is required with an extended battery model to understand this potential trade-off between the pack's durability and performance. Chapter 8 will explore this aspect further.

Figure 7.8 presents the local Pareto fronts for each group of solutions identified in the objective space plot (Fig. 7.9). Fixing the battery pack geometry and analysing a mission longer than the design one produced a small set of available solutions. From the "Set

Id" axis, it is visible that most of the design space has been eliminated (97%) since those EMS inputs could not complete the mission. The Pareto fronts are small, consisting of a few points, and all the solutions are arranged in a diagonal direction. All indicate that removing the battery pack sizing limited the freedom available to the optimiser to find feasible EMSes. As a consequence, most of these solutions have the same DOH. These align with the trends seen previously in Sec. 7.4.1, with high hybridisation values at the beginning of the climb and decreasing as cruise altitude is reached.

7.4.3 Commuter Flight

Figure 7.10 presents the operating points that satisfy the mission of the commuter flight. In this scenario, the aircraft improves over the baseline when at least 55% of the battery is used. Hence, a partial recharge is necessary to carry out the return flight with the same EMS.

Upon examining the objective space ((Fig. 7.11), it is evident that the decrease in NO_x emissions surpass the decrease in fuel consumption. This can be attributed to the mission requirements, where a significant portion of time is spent in the climb phase - a power-intensive stage that largely contributes to the NO_x emissions (see Sec. 6.4.3). The maximum improvement (-11% in fuel consumption and -25% in NO_x) is higher than the design case and the previous off-design condition as the mission is shorter.

Despite the greater number of feasible solutions, the Pareto front comprises only two points in this case. Similarly to the mission in Sec. 7.4.2, the solutions in the objective space are laid out on a diagonal where the greater the improvements on the objectives, the greater the battery energy consumed. Again, this is a consequence of the inability to tailor the battery pack to the mission.

7.5 Summary of Chapter

This chapter aimed to study the behaviour of the hybrid-electric system in the off-design condition. The battery pack is kept fixed, and the optimisation framework would have to identify the optimal EMS without the ability to tailor it. In addition, the battery model was updated, introducing cell dynamics and mass prediction based on the pack layout.

The scenarios studied were a typical day of regional flights and a non-stop commuter flight, which can be expected to be operated by turboprop regional aircraft. The design mission that fixed the battery pack layout was the same as in Chapter 6. This condition proved to be a limitation when trying to carry out the one-day operation scenario without any partial recharge during midday, achieving only an improvement of 3% in NO_x emissions and 1.8% in fuel consumption, instead, introducing recharge after morning operations allowed for significantly improved emissions. For example, the reduction of NO_x emissions improved to 12% with a partial recharge of the battery by 40%. However, the battery lifespan would be affected, as cycling it at the partial state of charge would age the cells faster. More detailed analysis is required to explore a potential trade-off between performance and battery life.

Both off-design conditions had small Pareto fronts composed of few points. Furthermore, the feasible solutions were placed diagonally in the objective space plot, indicating a limited variance from the input parameters. The feasible solution space was drastically

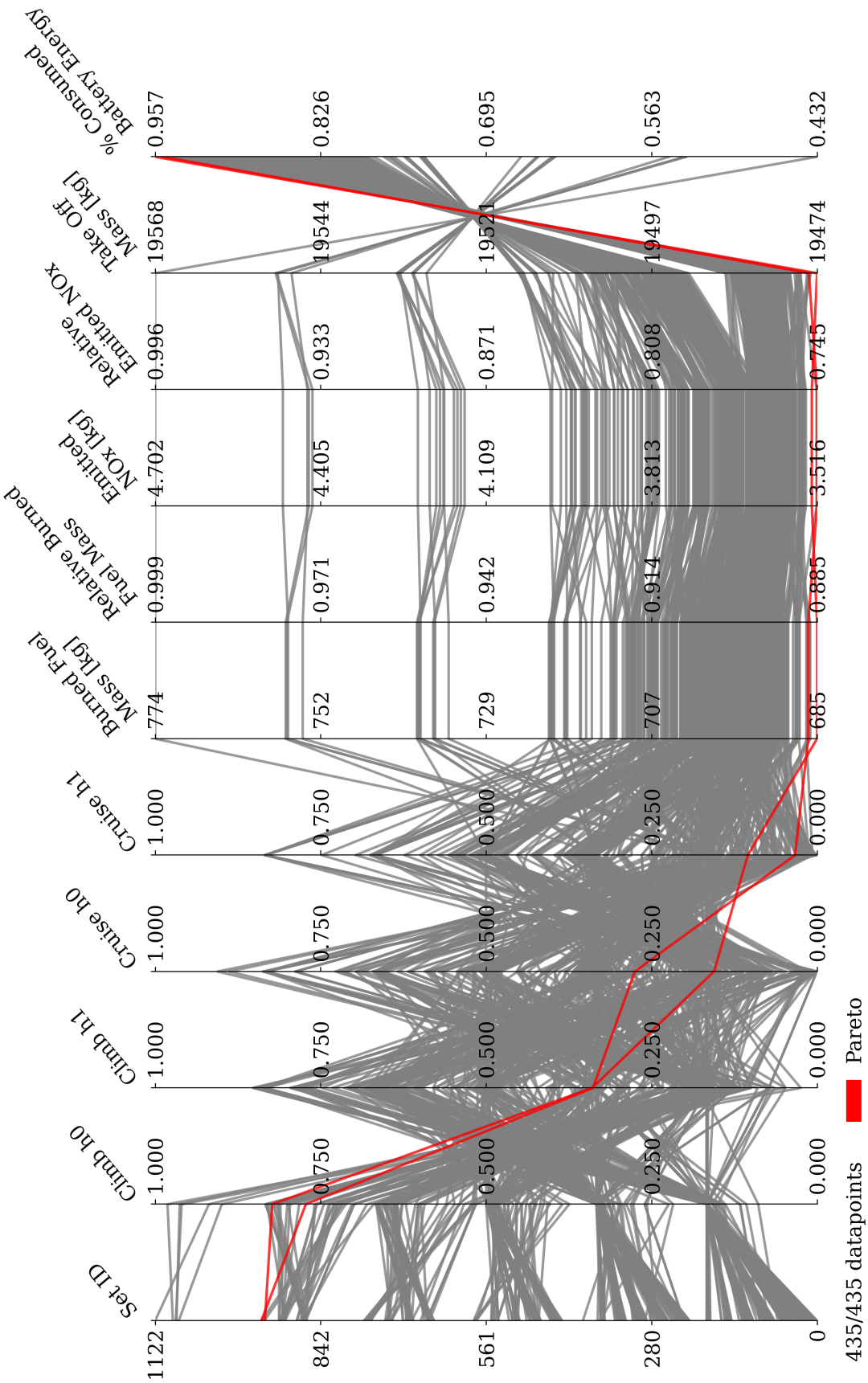


Figure 7.10: Parallel coordinates results of commuter flight operations.

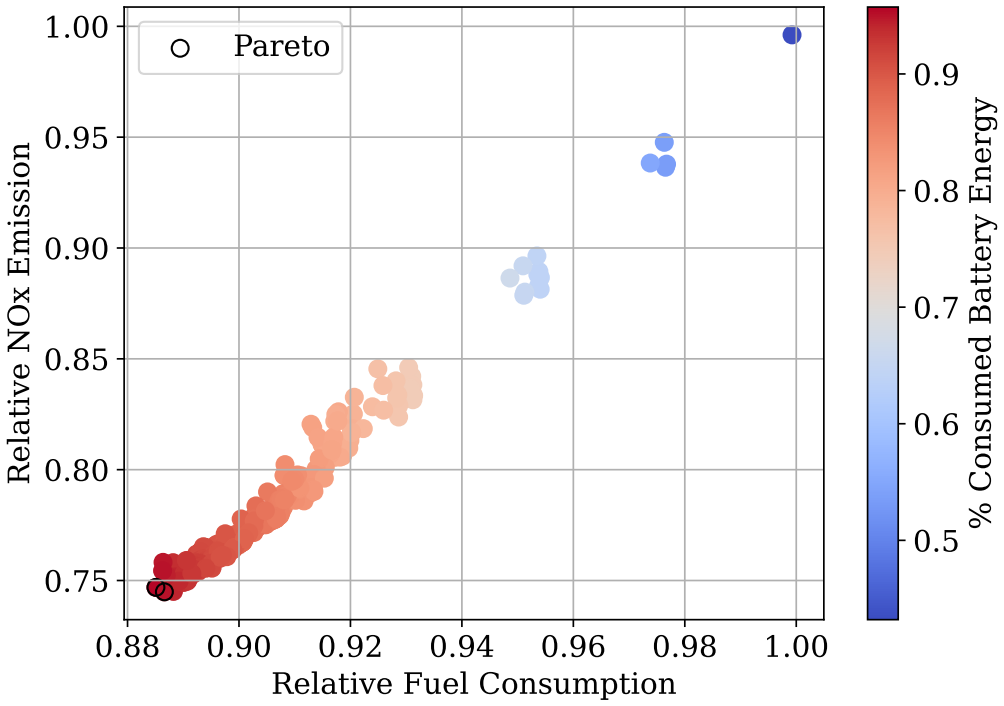


Figure 7.11: Objective space of the commuter flight condition.

reduced without the ability to change the battery pack and tailor it to the selected EMS. The EMS had more electrical power allocated in the climb segment, which reflected a considerable improvement in NO_x emissions, which reflected the nature of these short-haul flights. The aircraft spends most of its time climbing rather than cruising, which is typical for regional aircraft.

The limitation of battery capacity might be overcome if the EMS is set differently for each flight, tailored to the remaining energy after the previous flight. This arrangement might improve overall emissions without requiring a partial recharge, although the last flight would likely have no hybridisation. The next chapter will introduce the effects of ageing on the battery pack to analyse the effects of cell degradation on the aircraft's overall performance.

7.6 Key Learning Outcomes

- With a fixed battery layout, a limited range of feasible EMS can be chosen.
- Regional hybrid-electric operations most likely require fast charging to be economically feasible.
- Fast charging might implicate a shorter battery life.
- Regional operations spend more time climbing than cruising, so most energy is spent there.

Chapter 8

Battery Pack Technology

8.1 Introduction

The final test case presented in this dissertation examines the effects of battery ageing on hybrid-electric aircraft performance. In addition to minimising fuel consumption and NO_x, the optimisation problem attempts to minimise a degradation parameter r_{degr}, that is, the ratio between the fuel consumption after one year of operations and the original value with a fresh battery. The goal is to study the trade-off between performance and battery durability and determine which EMS would favour one.

Another point was evaluating the effect of different pack energy densities on the performance objectives and the EMS. This was carried out by adding the pack energy density as an input parameter to the optimisation study, ranging from 350 to 550 Wh/kg. As this is a critical technological assumption, a statistical model was constructed to estimate the year of availability. Two scenarios were included, a conservative one with linear growth and an optimistic one with exponential growth, reflecting a hypothetical increase in research funding. Additionally, the effects of the packaging (i.e. the extra mass on top of the cells) were considered since their contribution to the total battery mass is not yet precisely quantified.

The results presented in this chapter were originally published in the AIAA SciTech conference in 2023 (215) and extended to be published in the AIAA Journal (216).

8.2 Problem Definition

The optimisation problem formulation is shown in (8.1). For this study, we consider linear energy management segments applied to the climb and cruise phases and the energy density of the battery pack. Table 8.1 presents the input values. The selected figures of merit are the total fuel burnt during the mission M_f, the mass of the emitted NO_x, and the ratio of the mass of burnt fuel after one year of operation to the original amount with a fresh battery pack.

$$\begin{aligned} &\text{given} && X = \{e_{battery}, h0_{cl}, h1_{cl}, h0_{cr}, h1_{cr}\} \\ &\underset{X}{\text{minimize}} && M_{fuel}, M_{NO_x}, r_{degr} \\ &\text{subject to} && M_{TO} \leq 23000\,\text{kg}\,(P_{sat} \geq 0.5) \end{aligned} \tag{8.1}$$

Parameter		Values
Pack Energy Density [Wh/kg]	$e_{battery}$	[350, 400, 450, 500, 550]
Start Climb DOH	$h0_{cl}$	(0,1) divided into 4 levels
End Climb DOH	$h1_{cl}$	(0,1) divided into 4 levels
Start Cruise DOH	$h0_{cr}$	(0,1) divided into 4 levels
End Cruise DOH	$h1_{cr}$	(0,1) divided into 4 levels

Table 8.1: Input Parameters.

The last objective is to study the effects on system performance of aggressive hybridisation when the battery pack is fresh. While it is not a battery pack parameter, it indicates how much the cell's capacity has degraded over one year of operation. Indeed, to meet the same 80% DOD target, the energy management system must use more power from the gas turbine powertrain, leading to higher fuel consumption and higher emissions.

The take-off mass M_{TO} is restricted to the maximum take-off mass of ATR-72 and introduced as a probabilistic constraint to the exploration step, with a minimum satisfaction probability P_{sat} of 50%. Areas of the design space that fall below this threshold are discarded and not considered for optimisation.

The results are compared to the baseline, the same ATR-72 model without electric propulsion and loaded with the same payload. Table 8.2 contains the baseline values.

Take-Off Mass	17,792 kg
Burned Fuel	1242 kg
NO_x Emission	8.59 kg

Table 8.2: Baseline Quantities.

8.3 System Model

This test case introduced the system model used in Chapter 7, a simulation of the effects of ageing on the battery pack. After sizing the battery pack, the code simulates its age for one year of operations. Since the original capacity is expected to decrease, the entire EMS is automatically scaled to match the 80% DOD constraint at the end of each flight. A second addition is a parallel statistical model to estimate the feasibility of the e_{pack} required by the system. This enables one to estimate a year when a HE regional turboprop could potentially be developed and commercialised. Compared to the previous aircraft model, the maximum take-off mass was increased to 23,000 kg, the MTOM of the ATR-72 (217). The payload has been kept at 50 passengers, assuming the aeroplane has been refitted and some of the available payload mass has been allocated to the battery pack.

8.3.1 Battery pack ageing model

The holistic ageing model developed in (218) was adopted to model the cell's loss of energy and capacity and the growth of its internal resistance as it is put into operational use. Equation (8.2) are used to update the equivalent circuit model parameters (cell capacity, capacitance, and resistance) with the contributions from calendar ageing (parameter α)

and cycle ageing (parameter β). The first is proportional to the number of days of use t, while the second is proportional to the total lifetime charge Q that has passed through the cell.

$$\begin{aligned} C &= C_0 \cdot \left(1 - \alpha_c \cdot t^{0.75} - \beta_c \cdot \sqrt{Q}\right) \\ R &= R_0 \cdot \left(1 + \alpha_r \cdot t^{0.75} + \beta_r \cdot Q\right) \end{aligned} \tag{8.2}$$

The updating procedure is performed after one iteration of the ageing simulation, where the results of the equivalent circuit dynamics (current, voltage, and depth of discharge) are used to calculate the ageing model parameters. Specifically, calendar ageing is affected by storage temperature T and storage voltage V (Equation (8.3)). In contrast, cycle ageing is proportional to the root mean square voltage V_{rms} and the difference in the depth of discharge of the operating cycle ΔDOD (Equation (8.4)). This study's storage temperature is constant, assuming the thermal management system maintains it in the ideal range.

$$\begin{aligned} \alpha_c &= (7.543 \cdot V - 23.75) \cdot 10^6 \cdot e^{-\frac{6976}{T}} \\ \alpha_r &= (5.270 \cdot V - 16.32) \cdot 10^5 \cdot e^{-\frac{5986}{T}} \end{aligned} \tag{8.3}$$

$$\begin{aligned} \beta_c &= 7.348 \cdot 10^{-3} \cdot (V_{rms} - 3.667)^2 + 7.600 \cdot 10^{-4} + 4.081 \cdot 10^{-3} \cdot \Delta DOD \\ \beta_r &= 2.153 \cdot 10^{-4} \cdot (V_{rms} - 3.725)^2 + 1.521 \cdot 10^{-5} + 2.798 \cdot 10^{-4} \cdot \Delta DOD \end{aligned} \tag{8.4}$$

The adopted step in the ageing simulation is five days, which was selected as an adequate trade-off between computational cost and error. The simulation lasts one year, assuming two flights per day, seven days per week.

8.3.2 Statistical model for e_{pack} technology forecasting

Since e_{pack} is a critical technological parameter for the feasibility of the proposed HE designs, a method is adopted to estimate the probability of availability.

Data has been extracted from the Battery 2030+ Roadmap report (208) from 2010 onwards, including target energy density values up to 2030. The procedure consists in constructing a linear regression model using Ordinary Least Squares (OLS) of the cell energy density data (219). Equation (8.5) are used to estimate the coefficients of the linear model $y_{e_{cell}} = b_1 \cdot x_{year} + b_0$ and the regression variance σ.

$$
\begin{aligned}
\bar{x} &= \frac{1}{n}\sum_{i=0}^{n} x_i \\
\bar{y} &= \frac{1}{n}\sum_{i=0}^{n} y_i \\
S_{xx} &= \sum_{i=0}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=0}^{n} x_i^2\right) \\
S_{xy} &= \sum_{i=0}^{n} y_i x_i - \frac{1}{n}\left(\sum_{i=0}^{n} y_i \sum_{i=0}^{n} x_i\right) \\
b_1 &= \frac{S_{xy}}{S_{xx}} \\
b_0 &= \bar{y} - b_1 \cdot \bar{x} \\
\sigma &= \sum_{i=0}^{n} (y_i - \bar{y})^2 - b_1 \cdot S_{xy}
\end{aligned}
\tag{8.5}
$$

Once the coefficients are found, it is possible to predict the mean and variance of the cell energy density for a specific year X_{year} using Equation (8.6). These statistical parameters describe a Gaussian distribution centred in $\mu_{e_{cell}}$ and standard deviation $\sigma_{e_{cell}}$.

$$
\begin{aligned}
\mu_{e_{cell}}(X_{year}) &= b_1 \cdot X_{year} + b_0 \\
\sigma_{e_{cell}}(X_{year}) &= \sqrt{\sigma^2 \left[1 + \frac{1}{n} + \frac{X_{year} - \bar{x}}{S_{xx}}\right]}
\end{aligned}
\tag{8.6}
$$

To predict the pack energy density, the relation between e_{pack} and e_{cell} must be found through the pack-to-cell mass ratio k, which is defined as follows:

$$
k = \frac{M_{pack}}{N_{cells} \cdot M_{cell}} \tag{8.7}
$$

where N_{cells} is the number of cells in the pack, M_{cell} the mass of a single cell, and M_{pack} the mass of the entire battery pack. By replacing the definition of energy density $e = E/M$ and remembering that the pack electric energy is equal to $E_{pack} = N_{cells} \cdot E_{cell}$, the following scaling equation is obtained with a few algebraic steps:

$$
e_{pack} = \frac{e_{cell}}{k} \tag{8.8}
$$

which states that the effective energy density of the pack is lower than that of the cell one. Finally, it is possible to estimate the probability that a specific pack energy density is available in a year X_{year} using:

$$
P_{e_{pack}}(X_{year}) = 1 - \Phi\left(\frac{k \cdot e_{pack} - \mu_{e_{cell}}(X_{year})}{\sigma_{e_{cell}}(X_{year})}\right) \tag{8.9}
$$

where Φ is the Gaussian cumulative distribution function. Two scenarios were developed from the same data set. The first, labelled "conservative", assumes a linear improvement of the cell energy density and hence no change in the rate of technological development. In contrast, the second scenario, "optimistic", hypothesises the increment

of R&D funding towards energy storage; hence the regression model is exponential. The same equations (8.5) and (8.6) can be used using the following regression model:

$$log(y_{e_{cell}}) = b_1 \cdot log(x_{year}) + b_0 \tag{8.10}$$

In addition, two pack-to-cell ratios are used in both scenarios to estimate the impact of battery packaging technology on the availability year. The adopted values are 1.5, which is commonly used in the published literature for the design of electrified aircrafts (213), and 2.0 as a more conservative alternative.

8.4 Results and Discussion

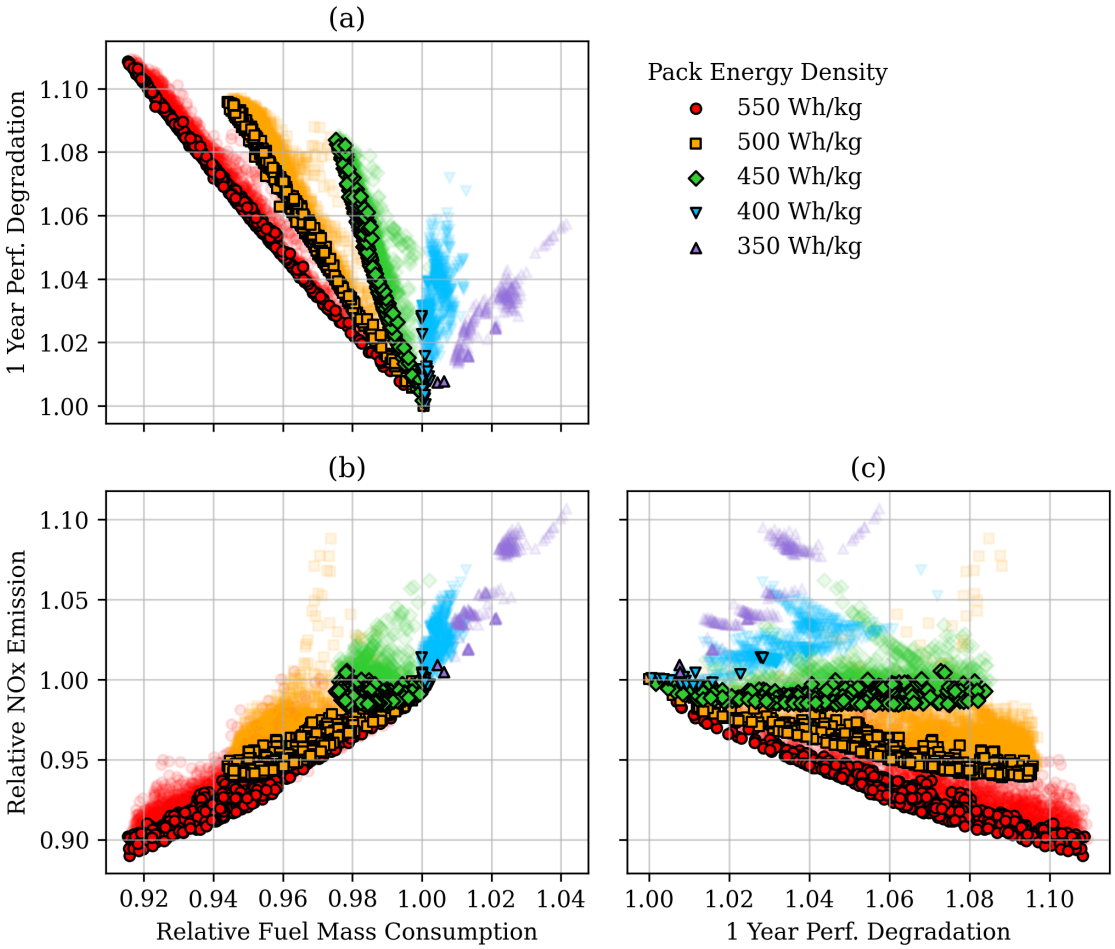


Figure 8.1: Pareto front of the three objectives with different battery pack energy densities.

Figure 8.1 presents the Pareto front solutions produced by the design space exploration framework. Three main results are clear from this graph. First, a high $e_{battery}$ produces higher values of r_{degr} overall. However, when comparing equal values of fuel burn reduction, the battery ages slower when the specific energy is higher (Fig. 8.1a). As will be discussed later, this is caused by the mass of the cells and not by cell ageing (Fig. 8.6 and Fig. 8.7). With more energy per unit of mass, fewer cells are required to achieve

the same reduction in fuel consumption. The aeroplane is less heavy; hence, when the capacity fades over time, the dead weight of the batteries has a reduced impact on fuel consumption. Instead, when all the mass available for batteries is used, the degradation rate is higher because of the higher extra dead weight.

Secondly, the rate of degradation and the other two objectives compete, most evidently, with the reduction of fuel consumption. This trade-off is essential for airline operators, where the cost of fuel and the cost of battery maintenance would compete.

Finally, the results indicate that the pack energy density should be greater than 400 Wh/kg to reduce emissions and fuel consumption above the baseline (Fig. 8.1b). This value sets a technological requirement of a minimum 600 Wh/kg energy density for individual cells, which will be achieved no earlier than 2040, as discussed in Sec. 8.4.6.

The following subsections explore the data in detail, analysing the interaction between $e_{battery}$, r_{degr}, EMS, and battery life alongside the aircraft's performance compared to the baseline.

8.4.1 Effects of $e_{battery}$ on Energy Management Strategies

Figure 8.2 and Fig. 8.4 present the optimal energy management strategies resulting for each level of battery energy density. Since the strategies are all linear segments, the input variables have been decomposed into an average value, the midpoint of the segment, and a discrepancy, the slope of the segment, as shown in (8.11).

$$\begin{cases} \mu_{DOH} = \frac{h_1+h_0}{2} \\ \Delta DOH = \frac{h_1-h_0}{2} \end{cases} \tag{8.11}$$

Higher energy density values allow for higher values of average DOH, both in climb and cruise. The electrical source can provide more flight power at the same maximum take-off mass limit. The climb segment is more hybridised than the cruise segment. Regarding the slope of the segments, the cruise phase presents a positive slope directly correlated with the energy density. However, the climb segment presents a negative slope with some exceptions, at the highest $e_{battery}$ of 550 Wh/kg. In conclusion, more specific energy in the battery pack enables more hybridisation and more flexibility in the slope of the segments because it can store more electrical power onboard for the same amount of maximum take-off mass.

8.4.2 Effects of r_{degr} on Energy Management Strategies

Figure 8.5 presents for three different levels of battery energy density the effect of the degradation parameter r_{degr} on the other variables. Battery life was calculated as the number of days after which the electrified aircraft matches the fuel consumption of the baseline conventional aircraft (see Fig. 8.6a). It is correlated with r_{degr}, where the battery lasts the longest with low degradation EMS. All three technological scenarios feature similar correlations between r_{degr} and the average values of DOH: the higher the electrical power demand, the faster the cell degradation. On the other hand, ΔDOH is concentrated around zero when r_{degr} is the lowest. It spreads without a specific trend at values of high degradation. The average electrical power requirement drives the degradation of the cell rather than the specific value over time.

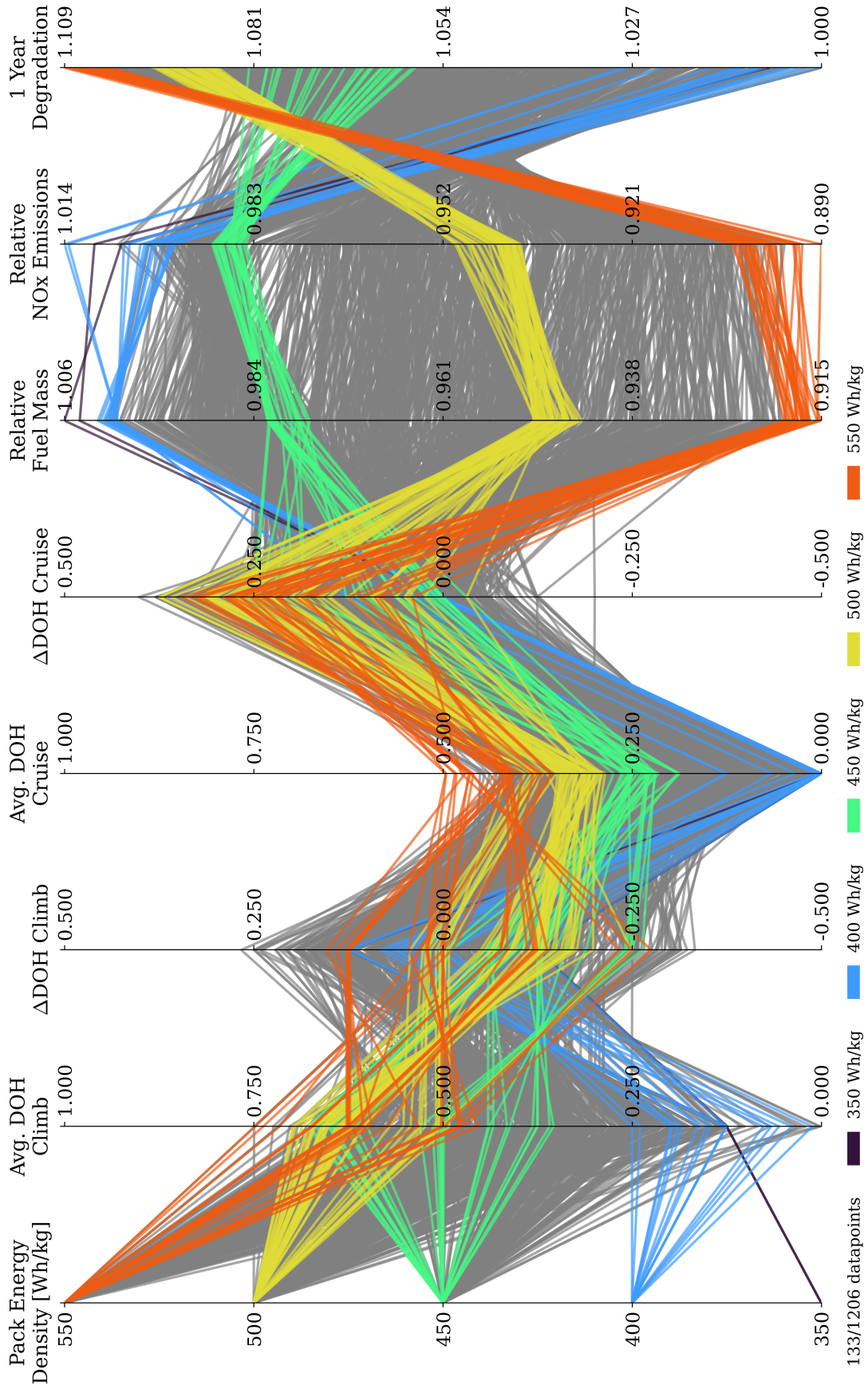


Figure 8.2: Effects of Battery Energy Density on the Energy Management Strategies (PC Visualisation).

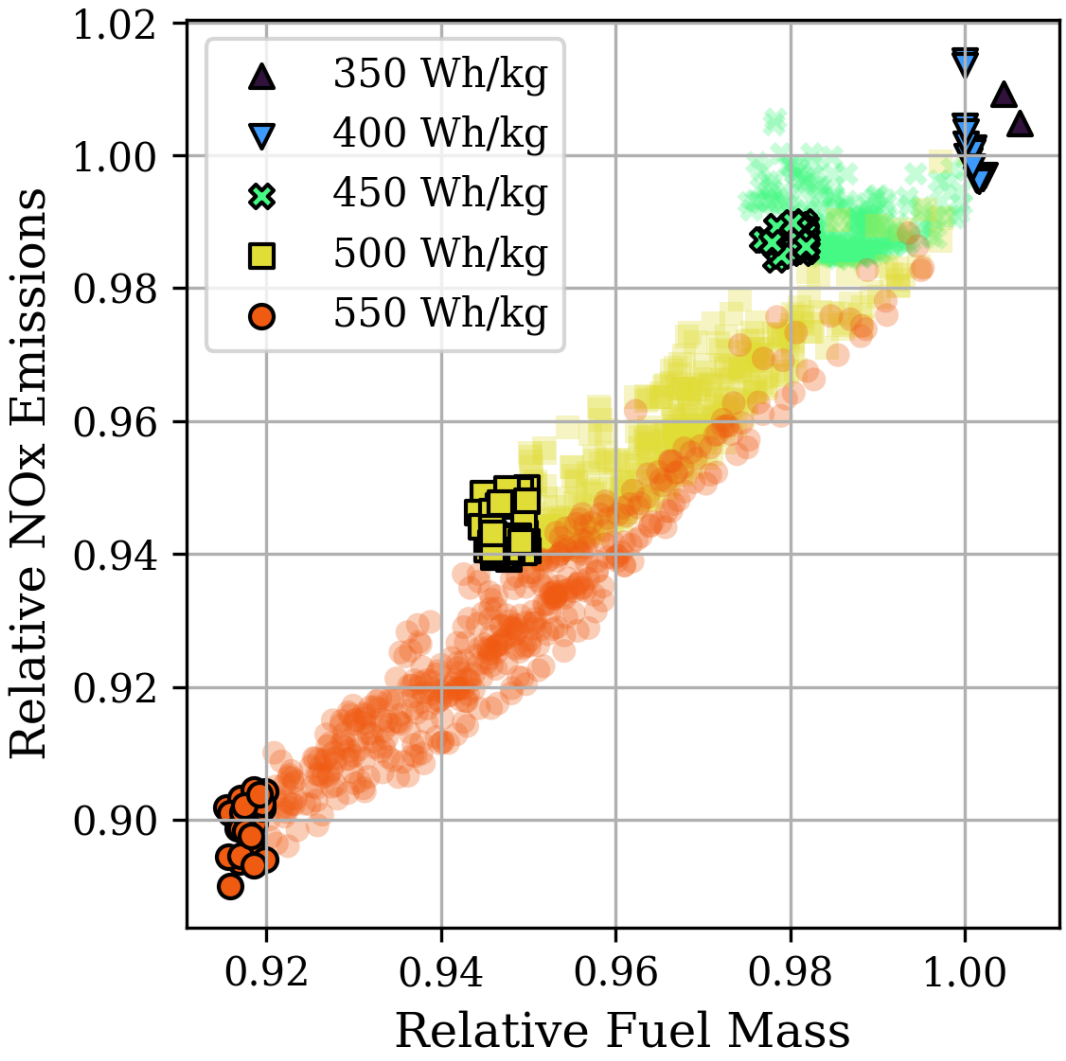


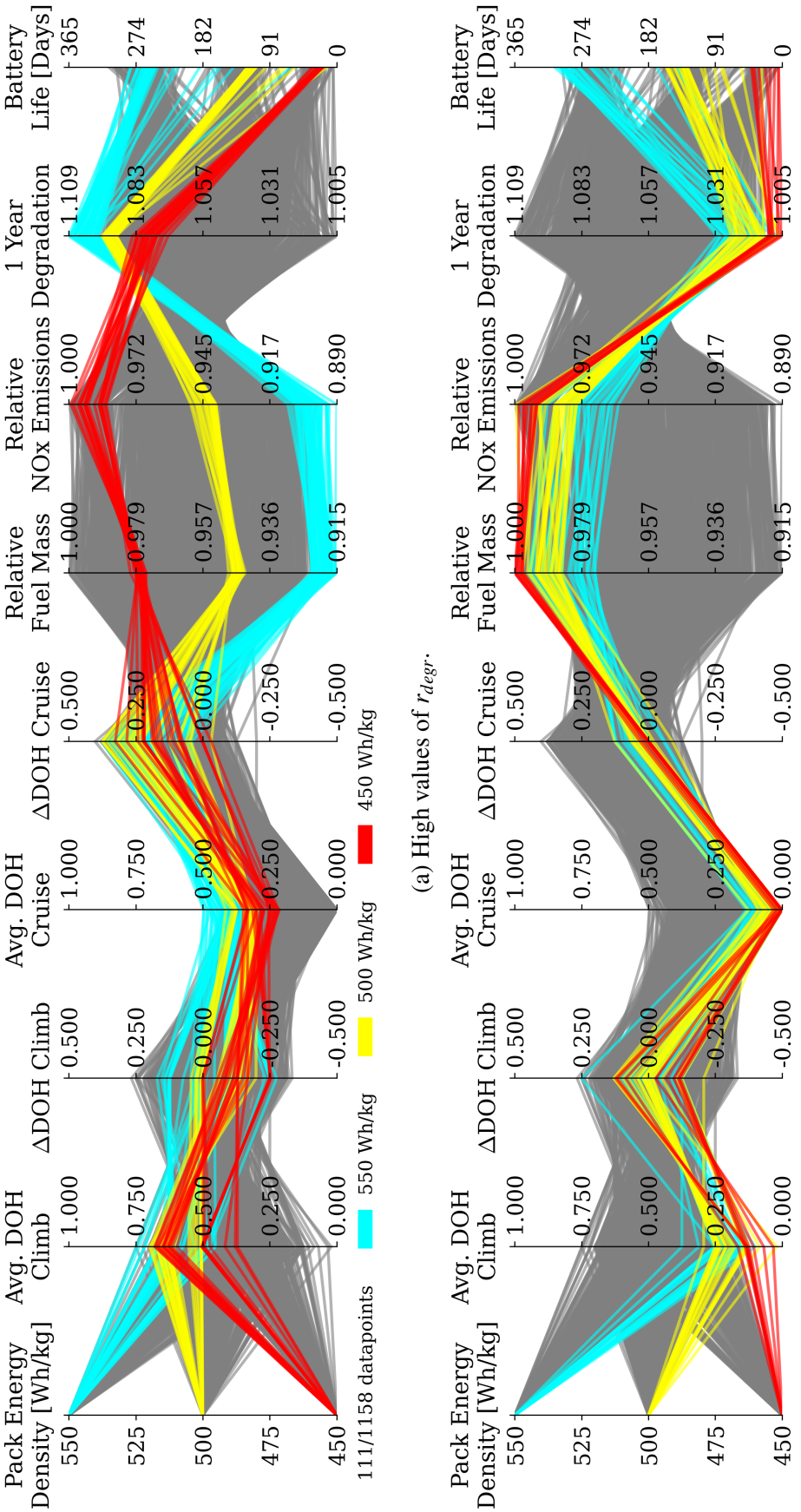
Figure 8.3: Pareto front with selections.

Figure 8.4: Effects of Battery Energy Density on the Energy Management Strategies (Objective Space Visualisation).

8.4.3 Effects of $e_{battery}$ on r_{degr} and Battery Life

Six specific points, two per $e_{battery}$ value, were selected to analyse the history of battery degradation over one year of operation. Each pair comprises designs with the lowest fuel consumption at zero days and the most extended battery life, as defined in Sec. 8.4.2. Figure 8.6 and Fig. 8.6 present these results on two scales: fuel consumption relative to the conventional baseline and relative to the day zero condition.

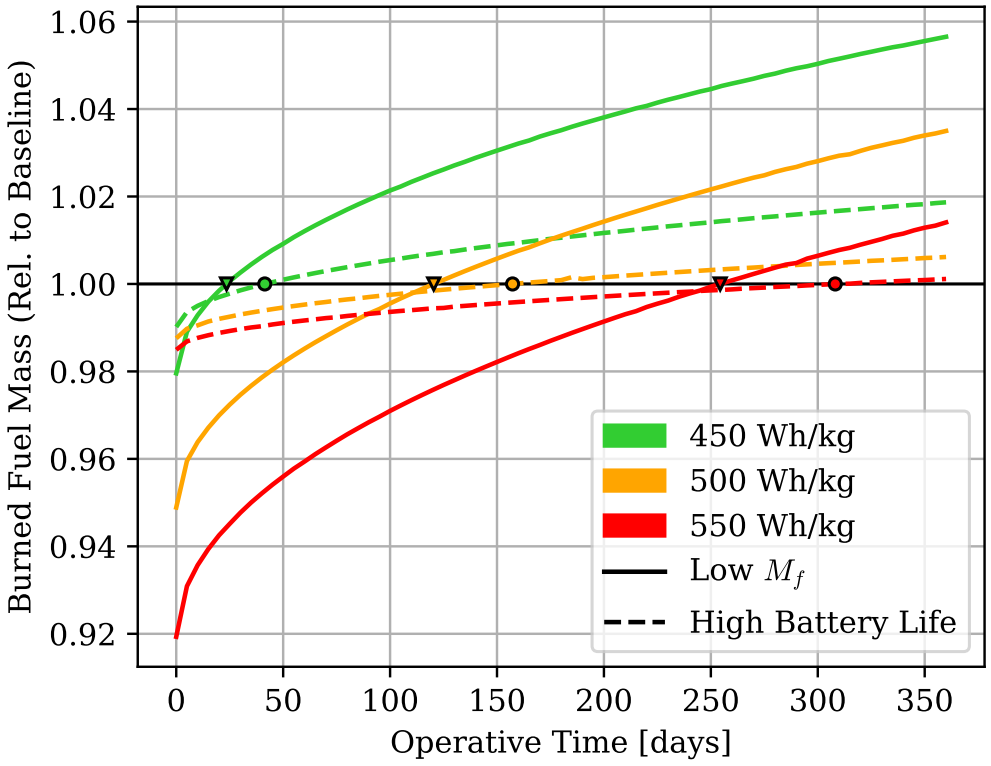
As shown in Fig. 8.6a, the battery lifetime is longer with higher $e_{battery}$ and, respectively, the gap from the lowest degradation to the highest is more prominent for each case. On the contrary, Fig. 8.6b shows that r_{degr} grows faster when $e_{battery}$ is greater, and the degradation is high; the opposite is true when the degradation is low. Despite all these



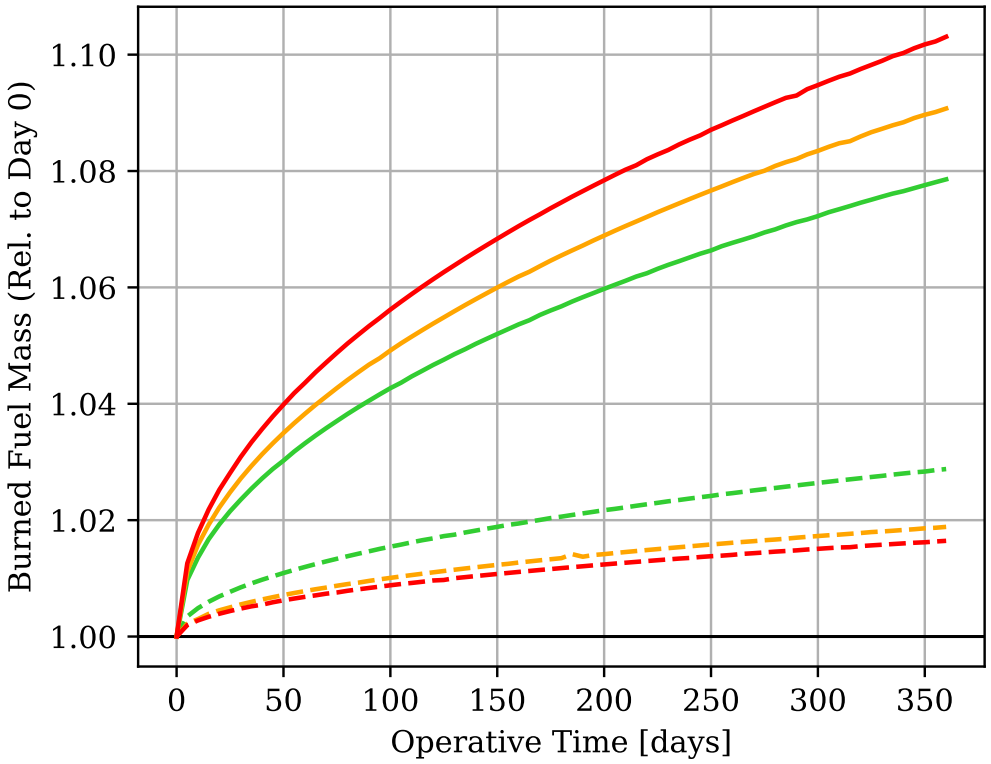
(a) High values of $r_{degr.}$

(b) Low values of $r_{degr.}$

Figure 8.5: Effects of Degradation on the Energy Management Strategies.

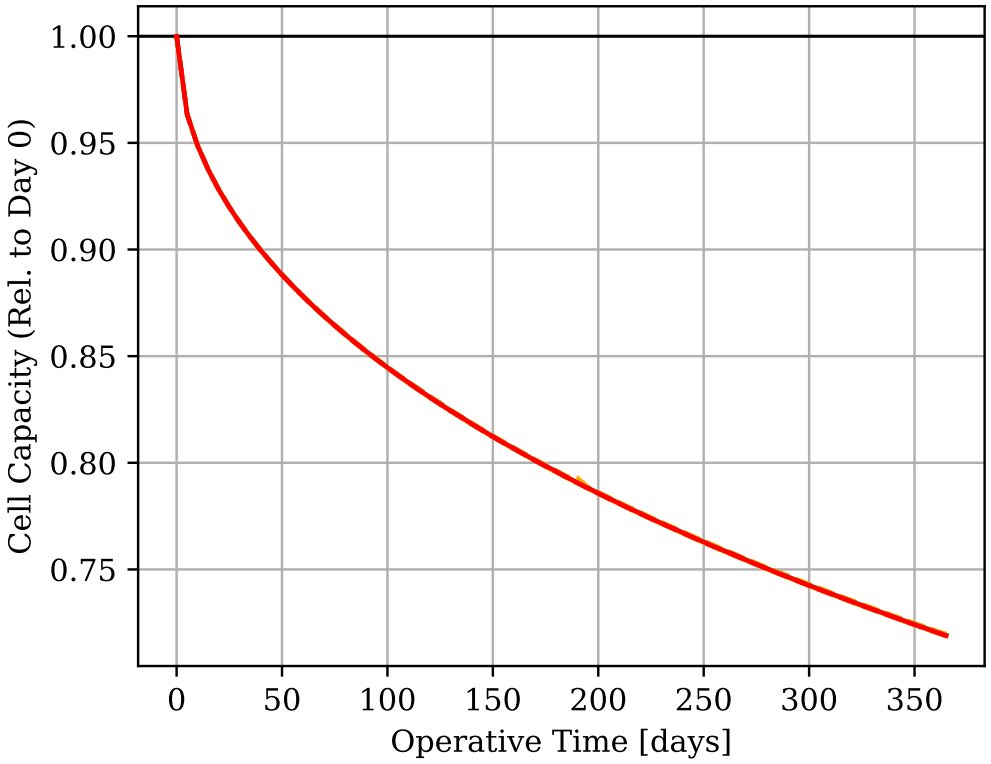


(a) Time history of relative M_f.

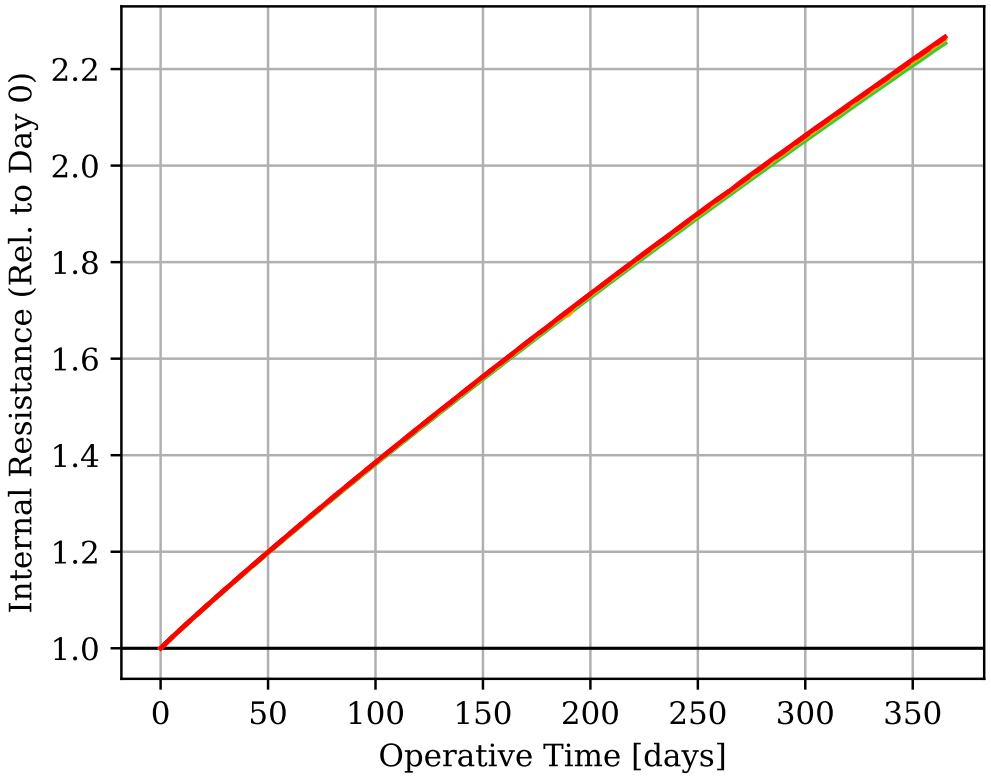


(b) Time history of r_{degr}.

Figure 8.6: Degradation histories of extreme points for each $e_{battery}$ case.



(a) Time history of cell capacity fade.



(b) Time history of internal resistance growth.

Figure 8.7: Degradation histories of extreme points for each $e_{battery}$ case (Continued).

$e_{battery}$ [Wh/kg]	Pack Life [Days]	r_{degr}	N_{cells}	N_{series}	$N_{modules}$	M_{fuel} [kg]	$M_{battery}$ [kg]	TOM [kg]
450	24	1.0838	44880	136	663	1211.2	5181.8	22943
500	121	1.0957	46784	136	344	1172.6	5248.1	22970
550	271	1.1085	112608	136	828	1136.9	5302.5	22989
450	48	1.0299	32096	136	236	1228.7	1839.8	19618
500	158	1.0197	21080	136	155	1225.7	1085.2	18860
550	310	1.0168	18224	136	134	1222.9	851.9	18624

Table 8.3: Details of the selected design points for discussion.

differences, the actual degradation of the cell (capacity depletion and growth of internal resistance) is identical for all six cases (Fig. 8.7a and Fig. 8.7b). Indeed, the parameters affecting cell degradation, such as the temperature, initial state of charge, and depth of discharge, are identical for all the cases. This would change if partially re-charging the aeroplane after one flight is introduced in the analysis, a scenario suggested by the authors for regional aircraft operations (211). More flights could be carried out per day at the cost of using the battery from a partial state of charge, accelerating the ageing of the cells.

Comparing the geometry of the battery packs of these six cases (Tab. 8.3) shows that the number of cell modules is correlated with the parameter r_{degr} of Fig. 8.6b: the higher this quantity, the faster the rate of degradation. While each cell ages at the same rate, the aeroplane carries more weight. Therefore, with less electrical power compensating for the inefficiency of the extra weight, fuel consumption grows faster as the battery ages. In summary, the effective specific energy of the pack decreases with cell ageing, eroding the benefits calculated on day zero, with a leveraging effect caused by the size of the battery pack. Therefore, more specific energy from the start enables the battery to last longer before servicing.

8.4.4 Effects of $e_{battery}$ on Aircraft Performance

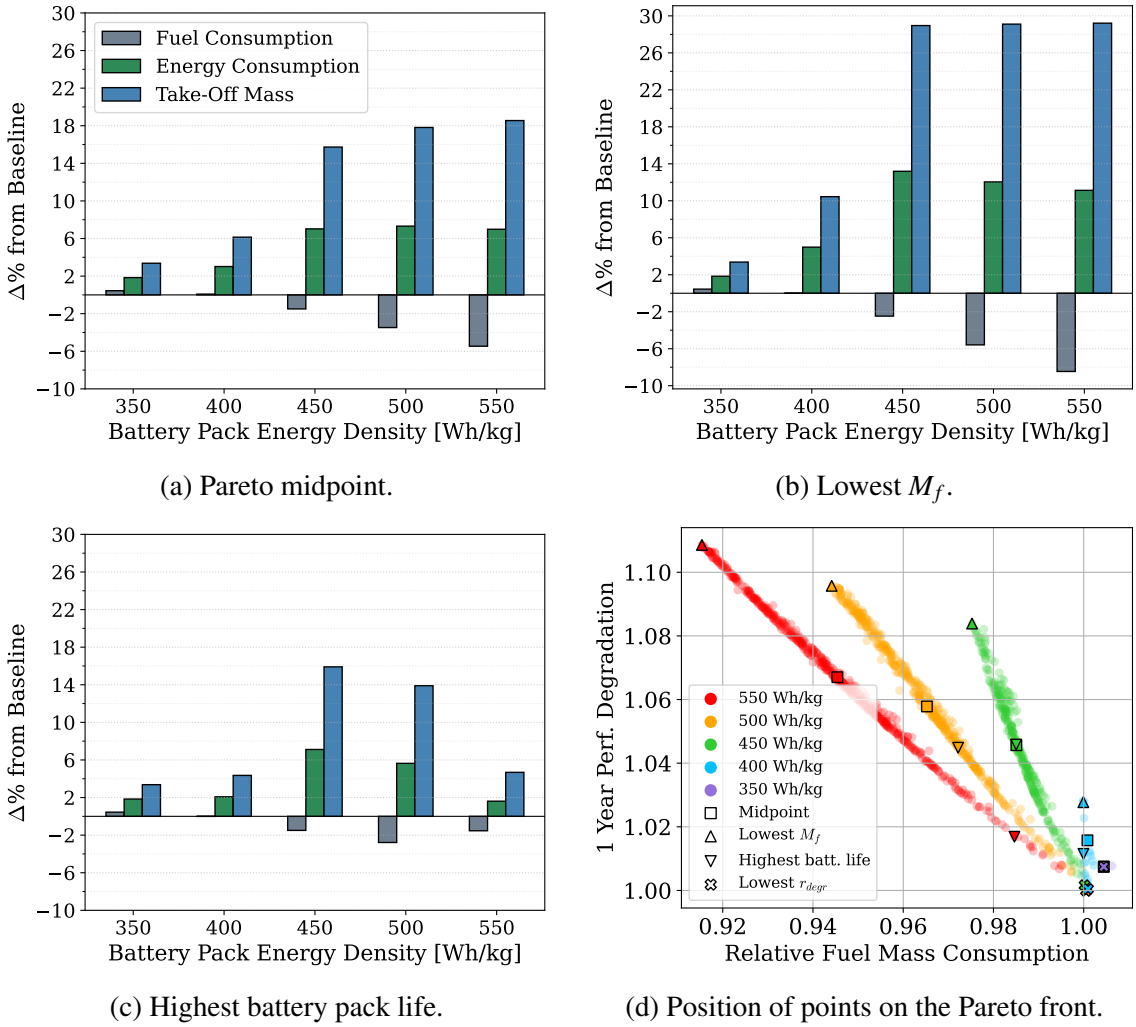
Figure 8.8 presents the percentile change in fuel consumption, energy consumption, and take-off mass relative to the baseline. It contains three Pareto front selections for each battery pack technology level. The values refer to the day-zero condition.

All investigated cases are less efficient than the baseline, requiring more energy to fly the same mission. The specific energy of the fuel is at least 20 times higher than that of the batteries, increasing the take-off mass when one source is traded for the other. Hence, heavier aeroplanes require more energy to fly, despite the lower fuel consumption. When the take-off mass is constrained, the energy consumption decreases as the specific energy increases (Fig. 8.8b). In this situation, the higher efficiency of the electrical power chain compensates for the power required to fly heavier aircraft.

All three conditions present a reduction in fuel consumption when $e_{battery}$ is above 400 Wh/kg. However, the condition with the highest battery life has no linear reduction with specific energy. Instead, it is constant around 2%, with a reduction in the take-off mass. As pointed out in the previous section 8.4.3, this design takes advantage of the increased specific energy to reduce the number of cells in the battery pack, which contains the increase in fuel consumption after the battery is aged. However, this set of points does not correspond to the lowest r_{degr}. Minimising only that objective would entail not including any battery pack in the design. The fuel consumption would stay identical as the object causing its increment is missing. (See Fig. 8.10d).

In general, the Pareto midpoint balances the two extreme cases with an increase in energy consumption of 7.5%. The three scenarios show that when the battery technology is fixed, the choice of EMS affects the efficiency of the aircraft via the increase in battery pack mass. Finally, as cells age, more fuel will be required, increasing the inefficiency in energy consumption.

The final point is investigating the performance increment with the $e_{battery}$ increase. Figure 8.9 shows the best-observed improvement for each value of the pack energy density, adding an extra extreme case of $e_{battery} = 750$ Wh/kg. Two trend lines, one linear



(a) Pareto midpoint.

(b) Lowest M_f.

(c) Highest battery pack life.

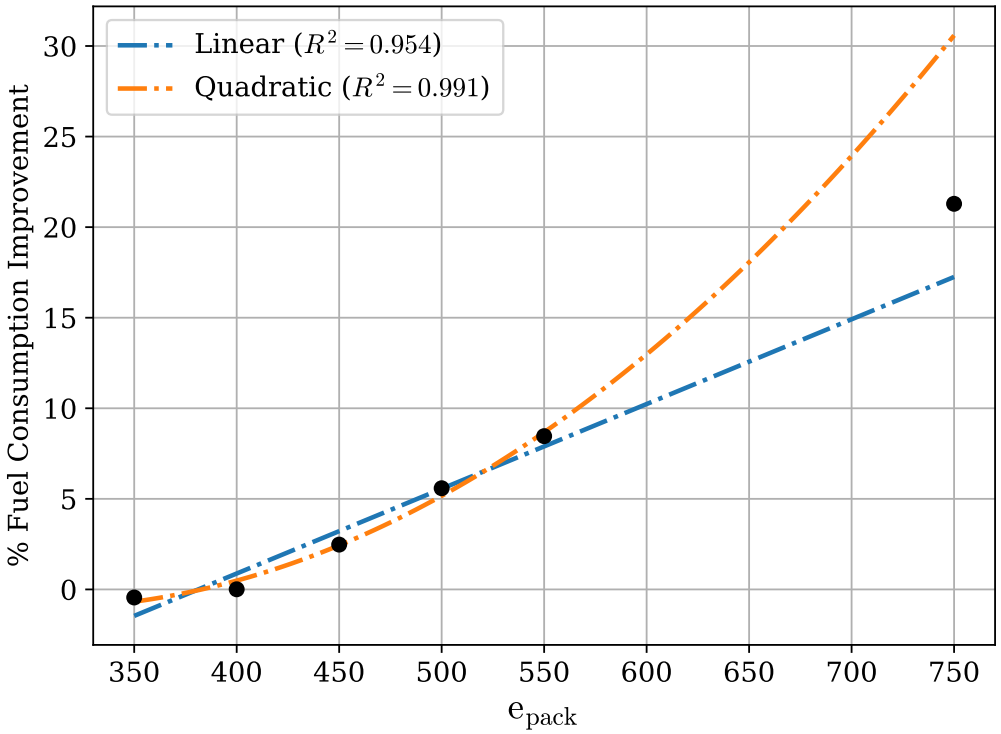
(d) Position of points on the Pareto front.

Figure 8.8: Aircraft performance at representative points in the Pareto front.

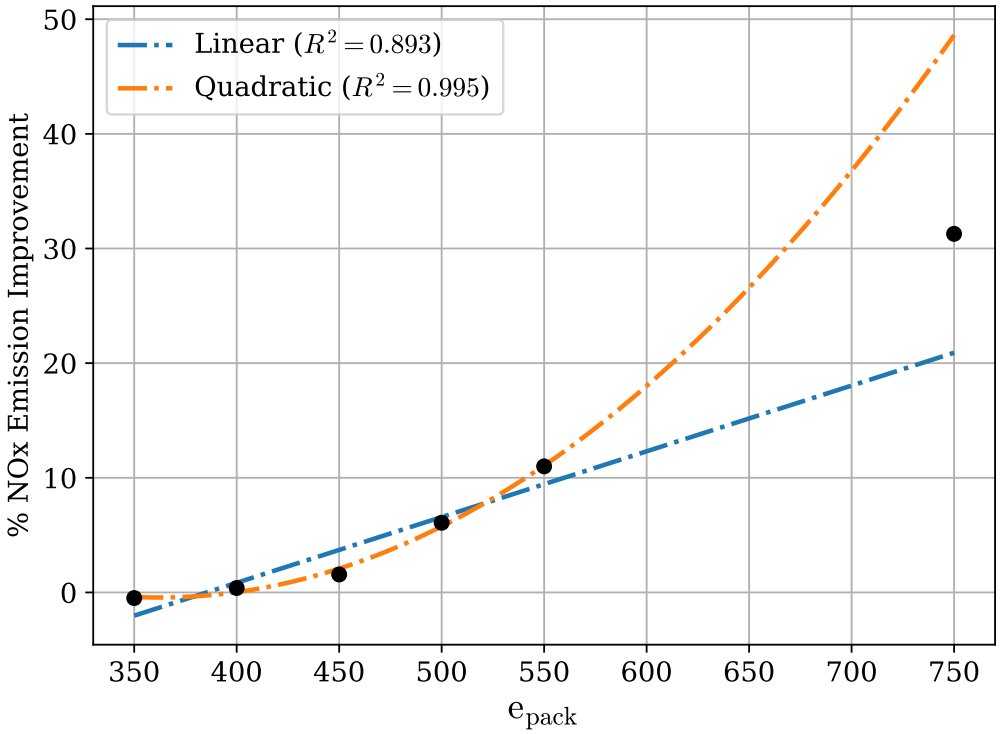
and one quadratic, were fitted to the original data set (350-550 Wh/kg). The rate of improvement is non-linear but sub-quadratic, as the new data point falls between the two lines. The nonlinearity stems from the ability of the optimiser to use more electrical energy, which is more efficient and cleaner than fuel, with less weight penalty. Therefore, investing in the improvement of battery technology would have significant returns. This aspect will be explored in discussing the technological availability of Sec. 8.4.6.

8.4.5 Suggested Energy Management Strategies

From the knowledge obtained in the previous sections, it was found that the average value of DOH for each segment is correlated with the energy density of the battery and the amount of maximum take-off mass. More DOH is allocated in the climb segment than in the cruise segment, while the slope increases in the cruise and is ambiguous for the climb. Because in this study, the battery pack is sized to match the EMS at day zero, the average value of DOH drives the increase in fuel consumption as the battery ages through the dead weight mechanism.

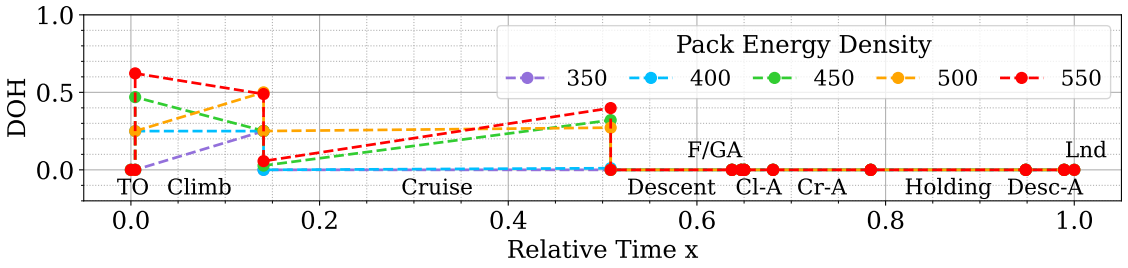


(a) Fuel Consumption.

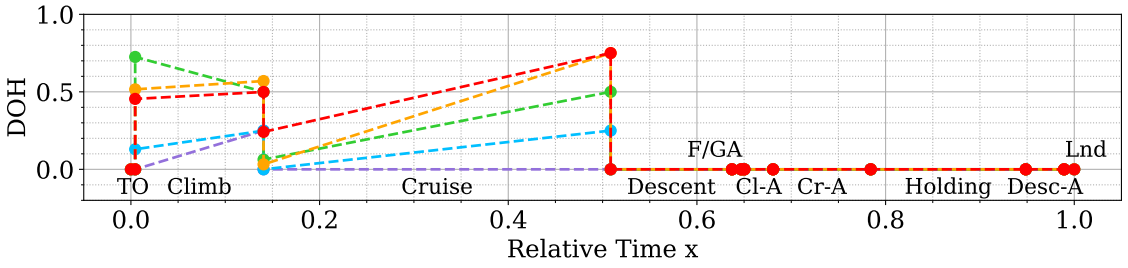


(b) NO_x Emissions.

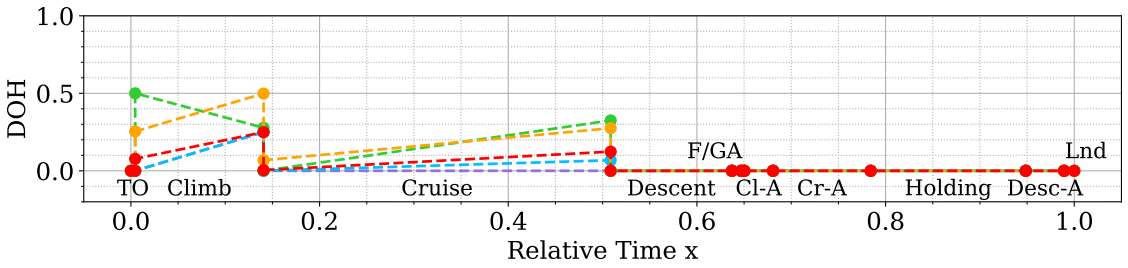
Figure 8.9: Effects of $e_{battery}$ on performance objectives.



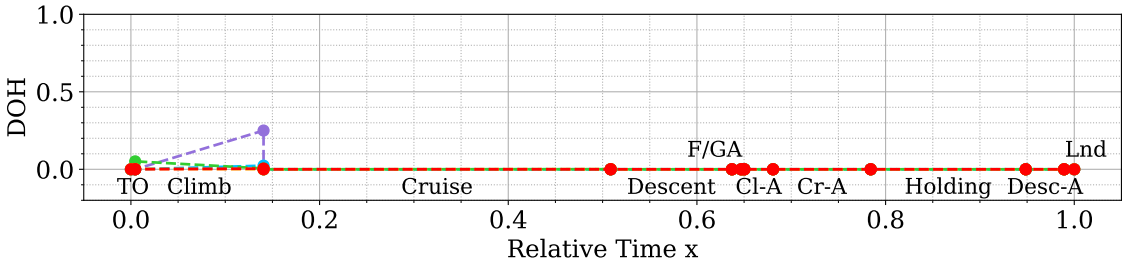
(a) Pareto midpoint.



(b) Lowest M_f.



(c) Longest battery life.



(d) Lowest r_{degr}.

Figure 8.10: EMS of the three scenarios analyzed in Section 8.4.4, compared to the lowest r_{degr} EMS.

Figure 8.10 presents the EMS of the three scenarios analysed in the previous section plus the EMS with the least r_{degr}. They present the trends described so far, which show no clear preference for the slope of the DOH function in the climb. Higher $e_{battery}$ allows for higher average DOH throughout the mission when fuel consumption is the priority (Fig. 8.10b). In contrast, when battery life is maximised, the higher energy density is used to reduce the amount of battery mass by reducing DOH (Fig. 8.10c). As pointed out, this condition does not correspond to the EMS of minimum degradation. Instead, the minimum r_{degr} is obtained when no hybridisation is used (Fig. 8.10d). Although, in

practice, this is an undesirable result, it further exhibits the relation between the battery mass and the increment of fuel consumption as the battery pack ages: no increment is present if no batteries are used.

In summary, the presented EMS are a general recommendation considering the identified interactions between pack energy density, battery ageing, and environmental requirements. They are limited by the selected design mission, the aircraft design selected as the starting point, and the figures of merit selected for the analysis. Consequently, designers should perform their optimisation study for their specific application and use the presented results as a sanity check.

8.4.6 Probability of achieving the expected technological requirements

The discussion presented so far underlined the importance of $e_{battery}$ for the feasibility of the scenarios presented in this study and hybrid-electric propulsion. However, most studies do not quantify the uncertainty of their timelines; instead, they assume a year far in the future to justify their assumption. This last section estimates the probability of technological availability with a regression analysis of current and expected specific cell energy densities. The intention is to quantify when it is reasonable to expect the required values $e_{battery}$ given the history of cell technology development.

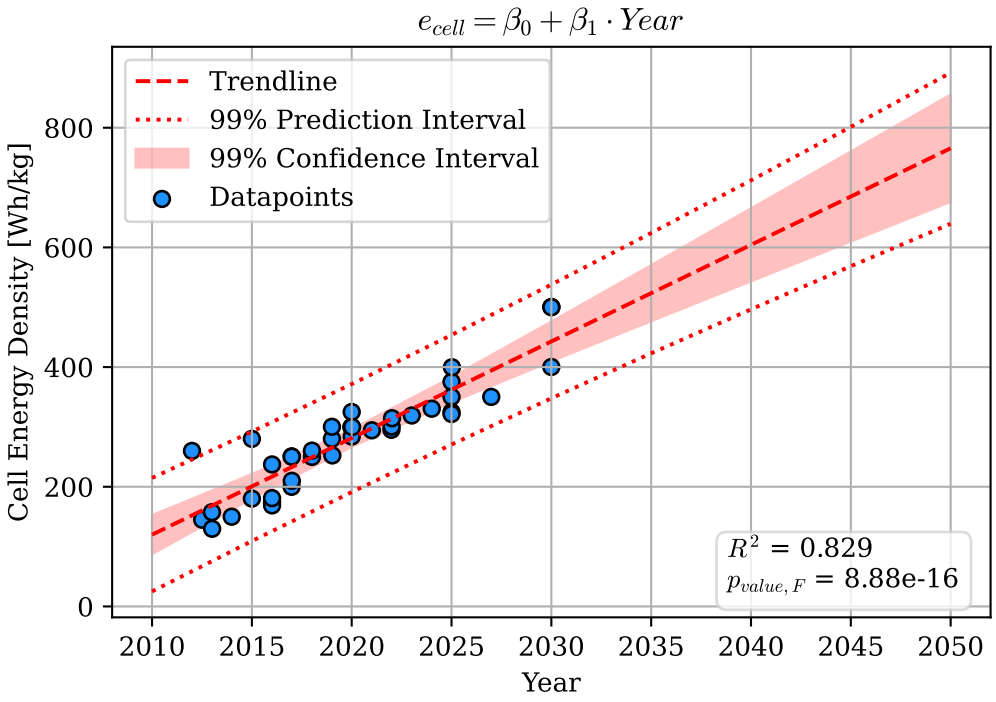
Data has been extracted from the Battery 2030+ Roadmap report (212) from 2010 onwards. Two scenarios are presented, one conservative and one optimistic. The first assumes that the cell energy density will improve linearly over time (Fig. 8.11a). The second scenario assumes it will improve exponentially (Fig. 8.11b). Ordinary Least Squares (OLS) have been used to construct the regression models using standard procedures. Both models have a satisfactory correlation ($R^2 = 0.83$ and $R^2 = 0.78$, respectively) and good modelling (the P values of the F test are lower than 1^{-10} in both cases).

The energy density of battery packs is lower than that of cell packs because of the inert mass added by the cooling system, packaging, and control electronics. Two values of the pack-to-cell ratio, 1.5 and 2.0, have been used to evaluate the impact of this technological uncertainty on the final pack energy density. Figure 8.12a and Fig. 8.12b present these results for the values of $e_{battery}$, which could match or improve on the baseline.

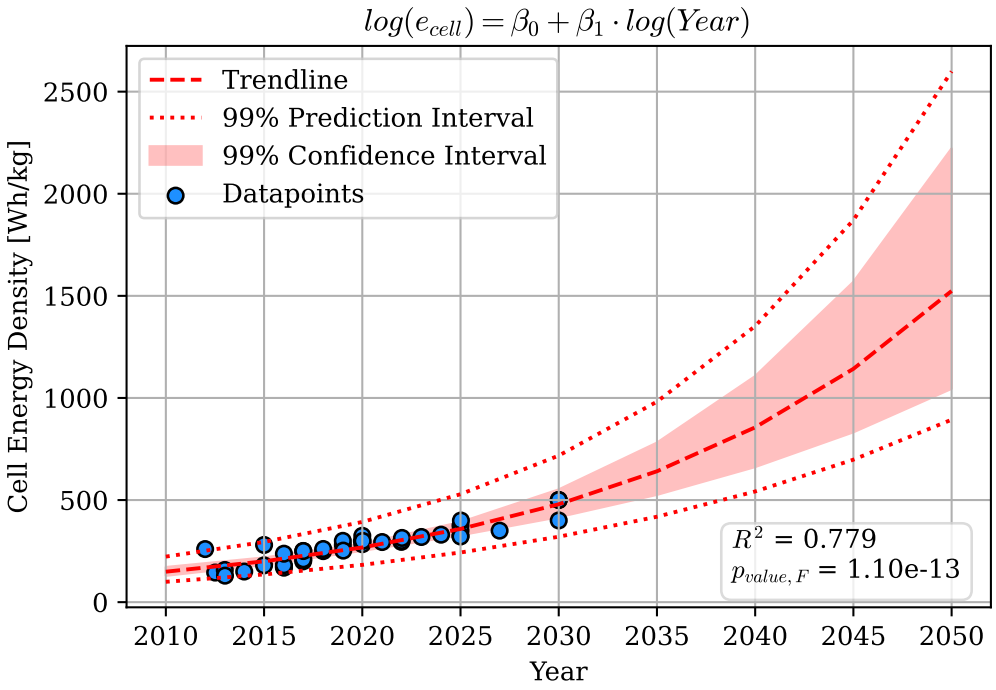
The two scenarios are considerably different, even for the same value of k. Under the conservative scenario and $k = 1.5$, 2040 is the earliest date at which a 400 Wh / kg battery pack is expected to be available with confidence 55%. On the contrary, the optimistic scenario estimates 2034 for the same value and confidence. This difference is evident for higher energy densities. In the conservative scenario, in 2050, there is a 60% chance that a battery pack with 500 Wh/kg is available, while in the optimistic scenario, the year 2038 is indicated. For a very high value of 750 Wh/kg, the availability is predicted far in the future, with the earliest date being 2045 on the most optimistic conditions. This energy density value might be achievable earlier with other technologies, such as fuel cells.

Furthermore, the assumed pack-to-cell ratio significantly affects the predicted availability. In the conservative scenario, the technological gap is, on average, 15 years, up to 18 years, for the highest energy density value. This reduces to 5 years if the optimistic scenario is assumed. The full range of predictions is presented in Tab. 8.4.

As noted above, this analysis leads to different conclusions on the feasibility of hybrid-electric aircraft. While the conservative scenario is the most likely, the automotive and

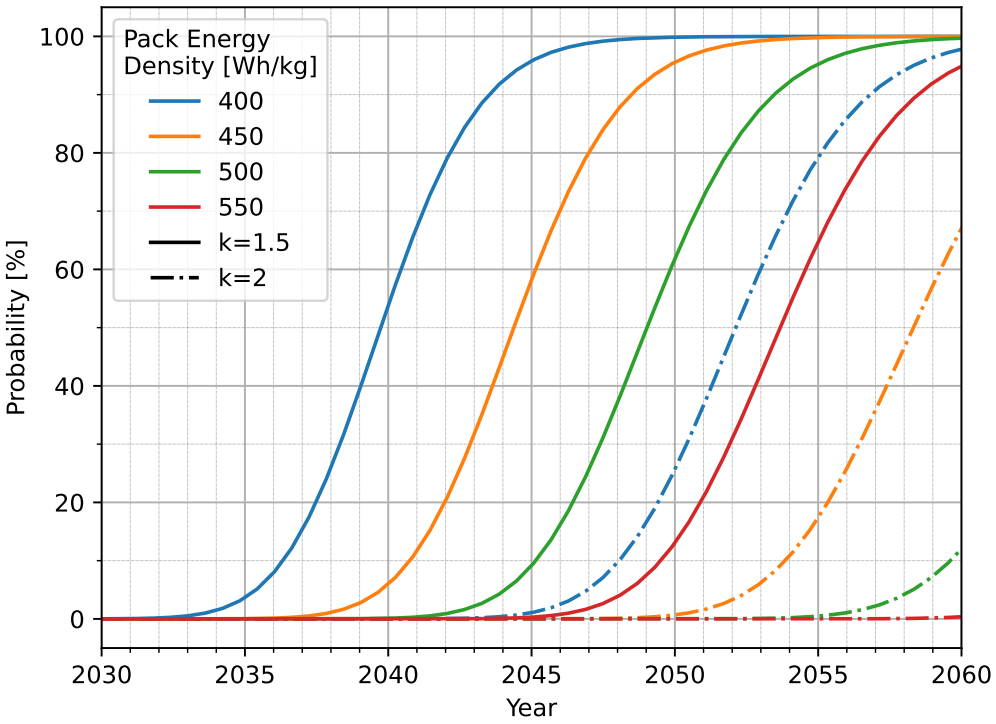


(a) Conservative trendline of Cell energy density.

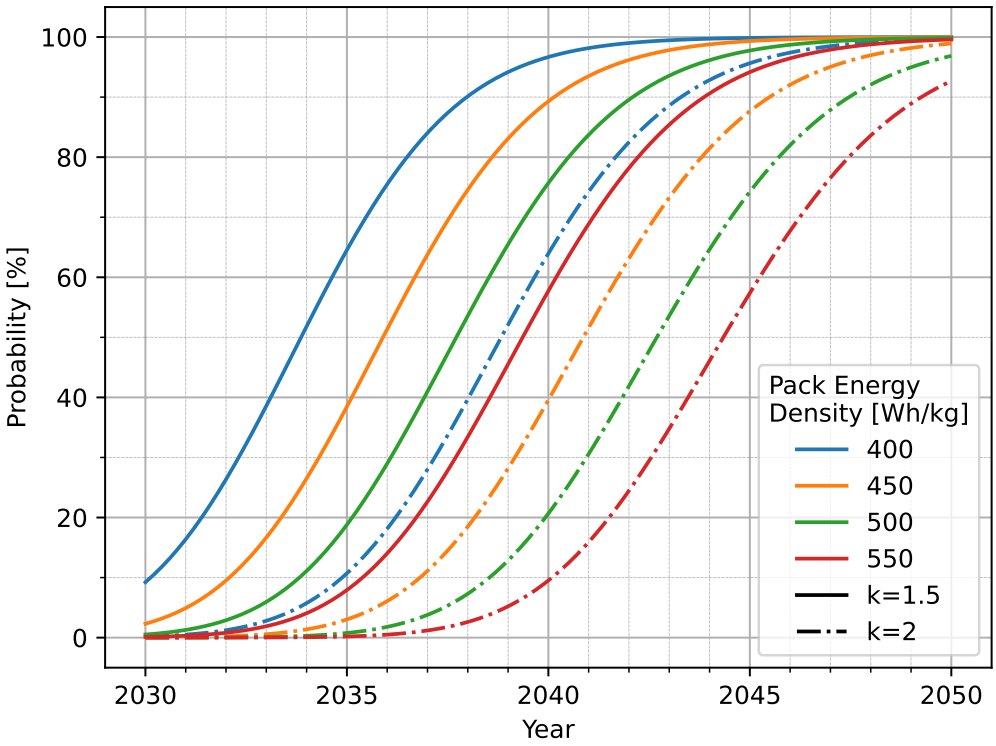


(b) Optimistic trendline of Cell energy density.

Figure 8.11: Statistical modelling of future cell technology.



(a) Conservative probability of pack availability. Solid line is $k = 1.5$, dashdot line is $k = 2.0$.



(b) Optimistic probability of pack availability. Solid line is $k = 1.5$, dashdot line is $k = 2.0$.

Figure 8.12: Prediction of future battery technology.

Table 8.4: Predicted years of $e_{battery}$ technological availability.

		Conservative				Optimistic			
	k	1.5		2.0		1.5		2.0	
	P	50%	90%	50%	90%	50%	90%	50%	90%
e_b [Wh/kg]	400	2039	2043	2052	2057	2034	2038	2039	2043
	450	2044	2048	2058	2063	2036	2040	2041	2045
	500	2049	2053	2064	2070	2037	2042	2043	2047
	550	2053	2058	2071	2077	2039	2044	2044	2049
	750	2072	2079	2096	2105	2045	2050	2049	2055

renewable sectors have been heavily investing in technology development, which could accelerate the rate of development. Indeed, the comparison shown in Fig. 8.9 suggests the potential for significant performance improvement, which is shown to be nonlinear; hence targeting the development of cells with higher energy density would have a better return on investment.

Furthermore, availability is affected by the packaging factor k, which cannot yet be accurately predicted, as it is affected by both material technology and the safety requirements in the battery pack. However, most of the technological timeframes in the literature are unrealistically optimistic unless a significant breakthrough in storage technology materialises.

8.5 Summary of Results

This chapter focused on analysing technological aspects of the battery pack, specifically the ageing and energy density. These have implications for the energy management strategy and the performance of the HE aircraft.

The specific energy of the batteries was identified as a key driver in the achievable reduction of fuel consumption and emissions, setting the maximum achievable amount of DOH in the EMS. This parameter affects the life of the battery pack, increasing its effective use before the aeroplane's fuel consumption is identical to the baseline. Fuel consumption and NO_x emissions were found to improve with specific energy at an increasing rate. Investing in research that significantly improves energy storage technology would yield better returns. Finally, 400 Wh/kg was identified as the minimum specific energy required to improve baseline emissions. It translates into a requirement of cells with 600 Wh/kg specific energy, expected to be available between 2034-2040, depending on the technological scenario. Battery packs with higher specific energy, required to make hybrid-electric propulsion feasible, are predicted to be available in the 2039-2050 time window. However, this prediction is conditioned by the current trend of technological improvement and a fixed 1.5 pack-to-cell mass weight ratio. Higher values of pack-to-cell mass ratios would shift this scenario further up to 15 years.

Including cell ageing in the battery model introduced additional considerations in the

definition of EMS. While striving for the highest DOH possible is beneficial for the immediate reduction of emissions, it shortens the operational life of the battery pack. The mechanism of this phenomenon was identified not in the cells' ageing rate, which was identical in every case studied, but in the number of batteries employed in each specific case. In fact, as the batteries aged, the effective energy density of the pack decreased, requiring the aircraft to consume more fuel. This effect is more substantial with heavier battery packs. Therefore, when trying to minimise degradation effects, the optimiser identified the designs with the fewest cells, taking advantage of the higher energy density when possible. As a result, EMS should be moderated to avoid quickly degrading zero-day performance.

8.6 Key Learning Outcomes

- The energy density of the pack is an important driver in reducing fuel consumption, NO_x emission, and the maximum possible degree of hybridisation.
- Larger increments of energy density generate greater improvements than smaller increments, indicating a higher return on technological investment.
- The high energy density of the pack mitigates the performance degradation caused by cell ageing.
- The increase in fuel consumption caused by cell ageing comes from the effective reduction of energy density, so the aircraft has more inert mass to move.
- Battery durability and zero-day performance are contrasting objectives, and mitigating DOH would improve durability.
- The pack-to-cell mass ratio affects the pack energy density and is defined by battery safety requirements.
- The minimum pack energy density required to match the baseline is 400 Wh/kg.
- With a linear improvement of cell technology, the availability of 400 Wh/kg battery packs is expected from 2039-2050, much higher for higher energy densities.
- Assuming an exponential improvement, reduces the technological availability to 2034-2040.
- With the higher pack-to-cell mass ratio, the availability timeline shifts to 15 years.

Part IV

Conclusions

Chapter 9

Discussion and Conclusions

9.1 Introduction

This dissertation aimed to develop a methodology capable of performing design space exploration of complex systems and identifying promising designs when prior experience is limited. A framework based on integrating set-based design and multidisciplinary optimisation was proposed to solve this problem. SBD's strengths are mapping the design space to the system requirements and eliminating the unproductive areas of the design space before deepening the analysis to specific design points. However, the mapping procedure is often handcrafted and based on expert judgement, which limits the application to systems of limited complexity and is well understood in the literature. Furthermore, the computational cost might grow exponentially as the number of input parameters increases.

The suggested solution to both issues was the combination of probabilistic surrogate modelling with Bayesian probability. This procedure enabled mapping the design space to the requirements without an explicit rule between input parameters and quantities of interest. This was demonstrated in a series of increasing complexity test cases investigating some aspects of hybrid-electric aircraft design. In fact, after surveying the literature, there is a lack of uncertainty-based tools for the design of HE aircraft and, from the system's perspective, a limited understanding of the coupled design of energy management strategies and the propulsion system. The results obtained by applying the framework revealed new information that addresses these gaps.

The following sections summarise and discuss the results obtained in Chapters 5–8, condensing the knowledge into key topics. Research questions are reviewed with their respective answers, and contributions to knowledge and their limitations are presented. Finally, further developments are suggested at the end of the chapter.

9.2 Summary and Discussion of Research Results

This section summarises several key learning points covered in Part III, dividing them into topics. Three of them cover the novelty and capabilities of the framework, while the latter two summarise the results regarding the energy management strategies of hybrid-electric aircraft.

9.2.1 Identification of promising areas of the design space

The dissertation aims to identify the most desirable area of the design space without an explicit mapping between inputs and quantities of interest, without requiring expert knowledge to be encoded in rules. This was solved by implementing a probabilistic exploration system, combining Bayesian probability and Set-based design approaches.

Specifically, the target area of the design space would emerge after eliminating the unfeasible and undesirable ones from the pool of possible combinations of the input parameters. This process would be carried out by constructing a probabilistic surrogate model capable of returning the mean response and variance of the quantities of interest to be optimised or constrained and then used to sample and infer the probability of satisfying the requirements. Using probabilistic logic, it was possible to combine all probabilities into a single value, quantifying the feasibility of the analysed design space. Those that did not meet a specific value are eliminated. This procedure enables surveying the design space without requiring explicit rules except for the system model and the specification of the requirements as constraints.

Chapter 5 demonstrated this capability with the toy example. The satisfaction probability was correlated with how much of the set could satisfy the constraints, as shown in Fig. 5.9. The remaining sets became candidates for further analysis by eliminating the low-probability ones. Local optimisation was used to convert the intervals into design points, which deepened the analysis.

This process can be augmented by introducing ”soft constraints”. These are additional constraints in the exploration phase, which map to objective quantities in the search phase. These are used to restrict further the design space towards the more desirable solutions on the global Pareto front. Section 6.4.1 introduced soft constraints on the objectives. It resulted in fewer surviving solutions, with a smaller spread towards the suboptimal solutions than the original feasibility-only problem. However, some points violated soft constraints, leading to a series of hypotheses on the observed behaviour. The most probable cause was the input discretisation's granularity and the model outputs' sensibility to the probabilistic constraints. Indeed, increasing the number of levels would capture the decision boundary of the constraint more precisely, reducing the number of sets with non-desirable points. However, since the exploration phase is computationally inexpensive, the user can tune the parameters before proceeding to the search phase.

9.2.2 Reduction of computational cost

One of the goals of the methodology developed in this dissertation is the ability to explore the design space without incurring exponential computational costs. Set-Based Design, while capable of producing multiple alternative designs due to its interval-based exploration of the design space, is limited in its application due to the high number of computational evaluations required. The cause lies in the combinatorial explosion of the number of input parameters and the intervals analysed. Most of these might not be desirable or feasible, wasting time and computational resources.

Machine learning was applied to identify those unfruitful areas of the design space such that the optimiser could skip them. This was achieved using Gaussian Processes and a Bayesian reasoning approach, calculating the probability of satisfying the constraints in each design space. It was successfully demonstrated to reduce the number of function

evaluations and the computational time required. The test cases presented in Chapter 5 and Chapter 6 were used to report the figures.

In the first, the reduction in computational cost was 77% for both time and number of function evaluations. The design space was also reduced by the same amount. The longer test case of Chapter 6 shows that the percentile of computational cost reduction increased as the number of input parameters increased, as the number of total combinations increased. Most of the design space would not be able to satisfy the take-off mass constraint, despite the greater degrees of freedom of the energy management strategy. According to Tab. 6.6, the time saved ranges from 49% to 99.95%, with an average of 80%.

Despite the reduction in design space, the framework did not lose optimal solutions. Indeed, as shown in Fig. 5.8, those points that best met the constraint and global optimality had the highest satisfaction probability.

The bulk of the computational cost is in the search phase. This could be reduced by adopting different MDO algorithms which are not of the monolithic type, such as Analytic Target Cascading (ATC) or with dynamically reconfigurable workflows (see Sec. 2.3.1). However, the exploration phase of PDOPT ignores the structure of the design model, since it treats it as a black box. Even if the design model is hierarchical and suitable for distributed MDO approaches, it would need to be encapsulated to a single interface suitable to PDOPT, that is all the coordination between the model subcomponents would be hidden from the algorithm. Thus, no difference would be present in the exploration phase, and PDOPT would not be able to take direct advantage of the structure of the design model. The main saving in computational cost would still come from avoiding running an MDO problem in the areas of the design space that are of no interest. Nonetheless, the computational cost of the search phase could be further improved if the design model supports distributed MDO approaches and the optimisation algorithm is suitably adapted for it.

9.2.3 Scenario analysis: robustness to changing requirements and technological uncertainty

The abundance of data points produced in the search phase enables the framework to cope with changing requirements. This ability stems from running local optimisation problems in sub-areas of the design space; hence lots of suboptimal designs are generated. However, when constraints are made more restrictive, the suboptimal points become the new Pareto front since the original became unfeasible.

The discussion in Sec. 6.4.4 did not require recomputation, as the suboptimal points were already available. A new Pareto front was identified when the take-off mass was restricted to a lower value. In this way, inferring new information from the same dataset was possible. This process can be extended to cases where requirements are made more restrictive than expected. In most scenarios, this is the case, as projects often set optimistic values for their requirements, leading to overestimating the objectives' improvement.

For requirements that are instead loosened, a quick assessment can be done by using the exploration phase with changed constraint values. This would provide the user with a qualitative answer on the changes in the feasible design space. It should be noted that changes in requirements that need a new quantity of interest cannot be captured without

a change in the design model.

Another use of the framework is to evaluate different technological scenarios, as was carried out in the test case of Chapter 8. In this situation, the system model must have modifiable parameters related to the technology. Each scenario can then be further analysed by integrating information on the assumptions and inputs used. The results can be used to justify planning and strategies or to build alternatives if an original plan does not work.

The example developed in Chapter 8 presents the integration of scenarios with different pack energy densities with a statistical analysis of the likelihood that technology will be available. Two regression models, reflecting different possible technological developments, were used, and it was possible to calculate the probability that a specific energy density value would be available for a specified year or calculate the estimated year of availability for a selected level of technological uncertainty.

This capability of "what-if" analysis can be further extended by running multiple parallel cases with different requirements to account for the uncertainty of requirement instability or produce new hypothetical market scenarios.

9.2.4 Energy Management Strategies: Topology and Systems Interaction

The test case developed in Chapter 6 sought to explore the ideal topology of the EMS using a flexible parameterisation. Complexity ranged from a constant hybridisation to three linear segments, defined by six parameters. Furthermore, electrical energy was introduced to different mission phases, from cruise up to take-off and final.

The results show that the Pareto front was captured with two segments per mission phase, and additional complexity did not produce better results. Instead, the topology presented similar shapes when the design points overlapped the objective space. For instance, the three linear segment parametrisation had a shape similar to a single linear segment one. This behaviour was independent of the number of mission phases with electrical energy.

The number and type of mission phases affected the objectives in different ways. Having just one did not produce a Pareto front but yielded a single optimal point. The trade-off was present at the introduction of electrical power to climb: the distribution of electric power affected the preference for reducing fuel consumption or NO_x emissions. Specifically, the average distribution of DOH affected NO_x emissions, while concentrating electrical power in the longest mission phases produced the highest reduction in fuel consumption. The other mission segments, such as take-off and final, did not significantly affect the objectives to justify their inclusion. Still, these low-altitude operations might require electrical energy to reduce noise near airfields and low-altitude pollution. The trade-off was also affected by the take-off mass constraint. This, in essence, limited the amount of battery mass available, limiting the total electrical energy for hybridisation.

Chapter 7 extended the problem by looking into off-design operations. With a more detailed model, the battery pack would be sized in the design mission and then evaluated in two different operating scenarios, both relevant to regional aircraft operations. A potential trade-off was identified in enabling partial recharge after half-day operations: more electric power could be used in the afternoon flights, but the battery would suffer with an

increased degradation rate.

9.2.5 Interaction between EMS and Battery Pack Ageing

Chapter 8 focused on the effects of battery ageing on the performance of EMS and HE aircraft. The battery model was updated with capacity fade and internal resistance growth caused by calendar and cycle ageing. The optimisation problem included a third objective that accounts for the effects of battery degradation on aircraft performance. It is defined as the fuel consumption ratio after one year of use to the original on day zero.

This objective was in contrast to the other two (NO_x and fuel consumption reduction). Improving the performance would yield high degradation and vice versa. The DOH in the EMS affects the pack ageing rate, which would require more batteries to be loaded onto the aircraft. Once their capacity fades, the plane will be heavier and require more fuel. EMS that minimises degradation presents a low battery mass, enough to improve over the baseline. Higher pack energy density can help in this aspect, as more energy is stored for the same battery mass.

9.3 Proposed solutions to Research Questions

Given the methodology developed and the results in the previous chapters, the following answers are proposed as solutions to the research questions formulated in Sec. 1.3.

RQ1 *How to manage the uncertainty arising from the decision boundary in the design space?*

The uncertainty is caused by the lack of information regarding the position of the decision boundary and how it maps to the input parameters. Since this is a form of epistemic uncertainty, Bayesian probability was adopted to model it. The constraints on the quantities of interest are recast as the probability of the constraint being satisfied, where the probability associated with it is the level of confidence in the truth of the proposition. Regarding the mapping procedure, the design space is sectioned into small areas by dividing the input parameters into discrete ranges. Sampling in each area with a statistical surrogate model enables one to calculate the probability of satisfying the constraints and find the likely position of the decision boundary in the input design space.

RQ2 *How to extract the knowledge embedded in the design model to evaluate the design space?*

One of the critical aspects of SBD is allowing the elimination of unfeasible solutions. As discussed in Chapter 2, most set selection methods are based on a form of expert elicitation or hand-crafted knowledge of the system under design. However, the mathematical model of the system contains the knowledge and assumptions of the system's behaviour under design. The mechanism of extracting this knowledge

and using it for set selection comes from developing a surrogate model that replicates the system's dynamics. Bayesian probability is then used to reason on the design space, identifying the promising areas without explicit elicitation of rules.

RQ3 *How to evaluate all possible alternatives in a reasonable computational time?*

The disadvantage of SBD is the increased number of design evaluations compared to the traditional point-based approach. Specifically, time can be wasted when evaluating areas of the design space that cannot produce feasible results. Therefore, marking them and quickly eliminating them would reduce computational costs without sacrificing the number of alternatives. The probabilistic approach described in this thesis allows one to carry out this reduction effectively thanks to the fast surrogate model and the Bayesian approach. Knowledge in the surviving areas of the design space is extended by running a local optimisation problem and producing individual design points.

RQ4 *How can the information obtained from exploring the design space aid in understanding the system under design?*

The results from the local optimisation problems encompass the entire feasible (and desirable) design space, not just the global optimal results. This abundance of information can be utilised to infer the sensitivity of the input parameters on the quantities of interest and the trade-off between them. Interactive visualisation tools such as parallel coordinates plots aid in this process. In this dissertation, the framework helped understand the effects of the energy management strategy and cell ageing on battery sizing and HE aircraft's environmental performance.

9.4 Contributions to Knowledge

The knowledge contributions contained in this dissertation can be divided into two categories: design methodology and hybrid-electric aircraft design.

A new set-filtering approach based on probabilistic logic and statistical surrogate models was proposed in the design methodology. This method sought to overcome the limitations of eliciting expert knowledge to build a set of rules or a value function for filtering. The hypothesis was that the simulation model already contains the information describing the system so that expert elicitation could introduce bias in the selection process. Surrogate modelling was used to construct a response model that encoded the relationship between input and quantities of interest, which was used to sample the probability of satisfying the constraints in the design space. Using probabilistic logic, it was possible to synthesise into a single figure of merit the likelihood that a portion of design space would be feasible. Furthermore, this filtering approach can reduce the computational cost of designing space exploration without sacrificing the coverage and depth of the results.

Another contribution is demonstrating the SBD method for managing technological uncertainty and requirements. This is achieved through the probabilistic approach and thorough design space exploration. A large amount of output data can be used for what-if scenarios in evaluating more restrictive requirements. Both Chapter 6 and Chapter 8

took advantage of this to reach conclusions about the system's behaviour. Specifically, the effects of the take-off mass constraint on the objectives and the energy density of the pack on the system performance.

In the topic of hybrid-electric aircraft design, the contribution focused on understanding the optimal energy management strategies and their impact on battery pack size, fuel consumption, and NO_x emissions. The topology of the EMS was found not to affect the objectives after a certain level of complexity; that is, two linear segments of DOH were enough to capture the Pareto front. On the contrary, the number of hybridised mission segments determined the trade-off between fuel consumption and NO_x emissions. Climb and cruise were the most important, the first affecting NO_x and the second affecting fuel consumption. The take-off mass constraint affected the number of design points: the more restrictive, the more limited the range of solutions.

Two other test cases were analysed to extend these results. The first (Chapter 7) covered off-design missions in which a regional aircraft might operate. The results suggested a trade-off between fast charging to prolong operations and battery durability. Furthermore, it was found that the most predominant mission segment is climb, especially for short mission phases. The second one (Chapter 8) looked into the effects of cell ageing and technological assumptions on system design. The pack energy density strongly determined the reduction of emissions and also allowed the effect of battery degradation on fuel consumption to be mitigated. The mechanism causing it was found to be correlated with the mass of the battery pack. As the cells age, the effective energy density decreases, leaving a heavier aeroplane that requires more fuel to fly. Finally, a statistical regression of expected cell energy density determined that the required technology level to enable HE turboprop would be available, most likely in the 2040s.

9.5 Impact of Research Findings

The framework developed in this dissertation has been used in the FUTPRINT50 project to better understand the design of regional hybrid electric aircraft, as shown in the project deliverables (220, 221). The focus was on understanding the interaction between energy management strategies and system performance.

Of the results of the test cases, the technological feasibility analysis of Chapter 8 has implications on the potential benefits of investing in energy storage technology with high specific energy. As the rate of reduction in emissions and fuel consumption was shown to be non-linear, there are more benefits in leaping in technology. Furthermore, a faster rate of technological development would move the entry into service in the late 2030s rather than in the estimated 2050s.

Despite these test cases, PDOPT is intended to be a problem-agnostic framework. It can be adapted for the design space exploration of other complex systems, such as those in the automotive and space industries. These sectors are characterised by many components and subsystems where PDOPT would be beneficial in the early design stages, alongside a multidisciplinary model encompassing the aspects the designers are interested in exploring.

The innovations introduced by the methodology would provide two improvements to the industry. First is the ability to map requirements onto the design space without explicit rules. The exploration phase of PDOPT enables the carrying out of different

experiments, where decision-makers can play with different values of requirements and understand the broad area of the design space which can satisfy their needs. Thanks to surrogate modelling, it is possible to provide a complex simulation model encompassing all the aspects of interest and perform scenario analysis.

The second is reducing the time required for design space exploration helps reduce the time required to reach convergence in the design. In fact, thanks to PDOPT focusing on eliminating unsatisfactory designs, engineers can avoid rework pitfalls that could not be otherwise discovered until late in the development cycle.

In summary, the framework would provide the ability to quickly and harmlessly ”fail”, so the pitfalls can be overcome and corrected before the more detailed design stages.

9.6 Limitations of Study

Despite the successful results of this work, the reader should be aware of its limitations, both in the testing of the methodology and in the modelling used in the test cases.

Although the framework was conceptualised to handle many input parameters, the largest number tested in this dissertation was 12 in the experiment CC-3S of Chapter 6. Results suggest the ability to reduce the computational cost significantly; however, multiple input parameters would still require many core-hours to produce the full results. Additionally, the test cases did not have sufficient complexity in the input parameters to require such a large number. Another aspect that requires more testing is the soft constraints in the exploration phase. The conclusions drawn in Sec. 6.4.1 should be examined with an experimental setup.

Similarly, the test cases did not include more than two requirements in the set elimination process. Testing with many requirements is necessary as the implemented approach may introduce distortions depending on the sensitivity of the requirement probability to the input space. The decision is based on a single probability resulting from fusing the probabilities of each requirement, as each of them has to be satisfied for a successful solution. The threshold for acceptance of a set is independent of the minimum satisfaction probability of each requirement (see Sec. 4.3). Distortions may arise if one requirement has lower probabilities than the others, biasing the global satisfaction probability. While the ”fusing” approach was successful in the test cases, this eventuality may be present.

This issue can be mitigated by replacing the decision-making on a single fused probability with parallel evaluations. Each requirement would be evaluated independently after calculating the probability of being satisfied and comparing it to its minimum satisfaction probability. A set would, finally, be eliminated if one of these requirements is not satisfied. Here the AND-ing process would happen with Boolean logic rather than a probabilistic one. The two approaches should be compared and evaluated in a dedicated case study.

Some potential capabilities of the framework were not fully implemented or tested. One would be the reduction of the input complexity in the search phase, which was present in the previous ADOPT (4). Theoretically, the same statistical surrogate could be employed to check which inputs have the highest sensibility to the objectives, such as fixing those input parameters in each local optimisation problem that do not sufficiently affect the results.

Another capability, which was encoded but not tested, is to perform uncertainty-based optimisation in the search phase. A local variation through the distribution definition

could catheterise input parameters. These would be propagated in the search phase in the local optimisation problems, cast as robust-based optimisation.

The aircraft modelling presented several limitations. The aerodynamics model used was zero-fidelity, consisting of constant values of L/D for each mission segment. An improvement would be using an aircraft trimmed polar constructed using established low-fidelity methods, such as ESDU datasheets. A second limitation was in the fixed empty mass of the aircraft when the maximum take-off mass was scaled. Although the increase was small, up to 14% compared to the reference ATR aeroplane, the empty mass should be recalculated based on the need for a heavier structure. Changing the MTOM would change the critical design loads in the aircraft structure.

9.7 Future Work

The framework presented in this dissertation is open to different extensions that open possibilities for future research.

More testing is necessary to understand the ability to capture the decision boundary, especially under many input parameters and when using soft constraints. As pointed out in Sec. 9.6, testing should include many requirements to check the limits of the framework when fusing the requirements probabilities in the set elimination process. A study should also compare the set selection method currently used in PDOPT with the proposed alternative based on independently evaluating the probabilities of requirements.

The current implementation of the exploration phase assumes implicitly that the input parameters have a uniform probability distribution. In other words, there is no agreement on which areas of the design space are preferred before performing the analysis. Expert judgement could be added to expedite the elimination of bad areas of the design space in situations with more familiarity with the design. This could be implemented by using Bayes' rule and prior distributions on the ranges of the input parameters. The posterior probability for each set would blend the probability that the design is feasible as calculated by the exploration phase and the prior expert judgement. Similarly, technological confidence could be included in the exploration phase process, with a prior encoding of the likelihood that a technology is available.

A Bayesian Network, or Bayes Net, could graphically represent the relationship between the different uncertainties. An example is shown in Fig. 9.1 reflecting the contributions to set feasibility from requirements, technology, and market desirability. The connections represent cause-effect interactions and how the probability distribution of feasibility is structured. With a Bayes Net, inferences could be made in both directions. For instance, a desired outcome would have a high probability of feasibility: this could be propagated back to the network to find which inputs would generate such a result.

Another aspect worth exploring is interactive visualisation tools for complex data. Given the large amount of information produced, its interpretation might be difficult using only visualisation tools. Producing a conceptual map that qualitatively describes each input parameter that influences the other quantities could benefit the user. A possible approach could be to use Causal Discovery, a branch of machine learning which attempts to learn casual relationships from data (222). It would also benefit complex systems design, where multiple subsystem models would interact.

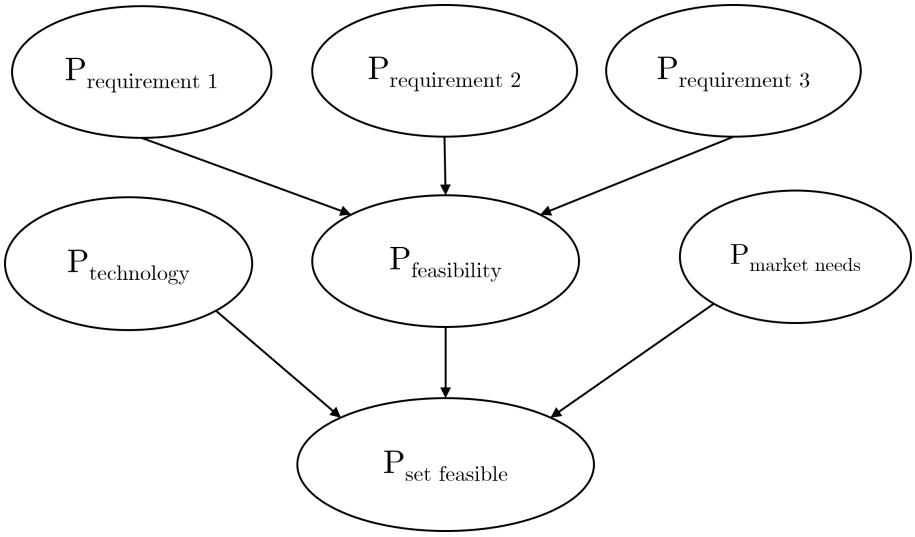


Figure 9.1: Example of a Bayes Net for mapping uncertainties on a design set.

An additional valuable development involves integrating the framework with interactive visualisation tools. Visual analytics, that is, machine learning methods with interactive visualisation, could be employed for post-processing the data and presenting the results. An interesting development would be exploiting the probabilistic surrogate of the exploration phase for interactive analysis through input from a visualisation tool, for instance, activating or turning off a requirement or modifying its constraint level. The visualisation would propagate these options and show their effect over the input design space. This could also integrate with the proposed alternative set elimination process of Sec. 9.6: the feasible space of each requirement could be shown independently and then intersected to show the overall feasible design space. The multi-dimensionality of the input space would make visualisation challenging. Effective technical communication would require a careful selection study of visualisation techniques and the development of ad-hoc interactive software.

Finally, the set-based mapping used in the exploration phase could be adapted in a multilevel design approach. Rather than using a monolithic design model, there would be mappings between top-level system requirements to subsystem design requirements, down to component-level design. This idea would mimic the distributed MDO approach.

In Chapter 2 a distinction between SBD and SBCE was made regarding the quantitative-qualitative application of the set-based principles. The methodology presented in this thesis mainly focused on a quantitative reduction of the design space, further developing the SBD methodology. Despite this, the industrial practice gives equal importance to the qualitative approaches, especially for technical communication between companies in the extended enterprise context. For instance, an airframer has to provide technical requirements to a supplier which may have contracts with the competition. Hence communication has to be fuzzy and qualitative to safeguard industrial secrecy. In this context, the exploration phase of PDOPT could be adapted for fuzzy communication between enterprises. A system-level enterprise would have a system model and utilise the exploration phase to identify the general area in the design space that the supplier company should use as requirements. Furthermore, the results of the full PDOPT analysis could be used jointly

between the system-level enterprise and the component-level contractors for requirement analysis by ranking the design concepts with respect to the needs of the project.

Another prerogative of SBD is the handling of changing requirements, especially as they mature during a project. While this aspect has been mentioned in this thesis, PDOPT only partially supports changing requirements which do not require a rewrite of the computational model (i.e. quantitative changes). Furthermore, recomputation is required if the change involves a constraint relaxation. The proposed Bayesian Networks could be used to fully support quantitative changes in the requirements in any direction (tightening or relaxing), but are still limited to the data extracted from the design model. Requirements that change the specification need more careful analysis before they can be represented in the PDOPT framework. A suggested approach may allow the user to introduce new information either by algebraically manipulating the generated data or by adding a causal relationship in the Bayesian Network where possible. Nonetheless if the change in requirement entails a new quantity of interest that cannot be obtained through algebraic manipulation a re-write of the design model is required. Integration with Model-Based System Engineering (MBSE) may help introduce modularity that would mitigate this aspect.

From the perspective of the test cases, two aspects can be explored in more detail. The first is studying off-design flight conditions with the addition of battery pack ageing, thus combining the work of Chapter 7 with Chapter 8. The tradeoff identified when looking at recharging batteries at noon could be analysed to highlight the effects of this operating condition.

The second one is introducing the thermal management modelling in the system and the requirements to cool the electronics during operations. These affect the electronics' power output and the aircraft's weight. In fact, thermal management is one of the challenging aspects of HE propulsion that limits its implementation, along with battery technology.

9.8 Dissemination of Research

The following documents have been published as a result of the research presented in this dissertation:

> **Andrea Spinelli**, Luchien Anderson, Hossein Balaghi Enalou, Bahareh Zaghari, Timoleon Kipouros, and Panagiotis Laskaridis. *Application of Probabilistic principles to Set-Based Design for the optimisation of a hybrid-electric propulsion system.* 11th EASN Conference (Online) September 2021. Published in IOP Conference Series: Materials Science and Engineering, Vol. 1226, IOP Publishing, 2022, p. 012064.
>
> **Andrea Spinelli**, Hossein B. Enalou, Baraheh Zaghari, Timoleon Kipouros, and Panagiotis Laskaridis. *Application of Probabilistic Set-Based Design Exploration on the Energy Management of a Hybrid-Electric Aircraft.* Aerospace 2022, 9, 147.
>
> **Andrea Spinelli**, Gustavo P. Krupa and Timoleon Kipouros, Boris Berseneff, and Sebastien Fiette. *Investigation of the operational flexibility of a regional hybrid-*

electric aircraft. 12th EASN Conference (Barcelona) October 2022. Published in Journal of Physics: Conference Series, IOP Publishing, 2023.

Gustavo P. Krupa, **Andrea Spinelli**, and Timoleon Kipouros. *An interactive framework to facilitate probabilistic set-based multidiscplinary design optimisation studies.* 12th EASN Conference (Barcelona) October 2022. Published in Journal of Physics: Conference Series, IOP Publishing, 2023.

Andrea Spinelli, Gustavo P. Krupa, and Timoleon Kipouros. *Set-Based Design Space Exploration to Investigate the Effect of Energy Storage Durability on the Energy Management Strategy of a Hybrid-Electric Aircraft.* AIAA 2023-0837, AIAA SCITECH 2023 Forum, January 2023.

Andrea Spinelli, Gustavo P. Krupa, Timoleon Kipouros, Panagiotis Laskaridis and Boris Berseneff. *Battery pack technological considerations for hybrid-electric regional aircraft feasibility.* Journal of Aircraft (in preparation).

Andrea Spinelli and Timoleon Kipouros. *PDOPT: A Python library for Probabilistic Design space exploration and OPTimisation.* Journal of Open Source Software (submitted).

Timoleon Kipouros and **Andrea Spinelli**. *D2.2: Design Methodology Report.* Project FUTPRINT50 Deliverable, September 2022.

Felix Brenner, Dominik Eisenhut, Hossein Balaghi Enalou, **Andrea Spinelli**, Jatinder Goyal, Jonas Mangold, Nicolas Moebs, Robert Nederlof, Felipe Reyes Barbosa, and Tomas Sinnige. *D4.2: Integrated tool environment for trade-off studies and optimisation.* Project FUTPRINT50 Deliverable, June 2023.

References

1. Bernstein JI. Design methods in the aerospace industry: looking for evidence of set-based practices. PhD Thesis. Massachusetts Institute of Technology; 1998.

2. Jasa JP, Hwang JT, Martins JR. Open-source coupled aerostructural optimization using Python. Structural and Multidisciplinary Optimization. 2018;57:1815-27.

3. FutPrInt50: Collaborating towards sustainable regional aviation. European Energy Innovation. vol. 1; 2022. p. 12. Available at: `https://www.europeanenergyinnovation.eu/OnlinePublication/Spring2022/index.html#p=13`.

4. Georgiades A. Set-based design and optimisation of aircraft systems.. PhD Thesis. Cranfield University; 2018.

5. Georgiades A, Sharma S, Kipouros T, Savill M. ADOPT: An augmented set-based design framework with optimisation. Design Science. 2019;5:e4.

6. Mavris D, Jimenez H. Systems Design. Architecture and Principles of Systems Engineering. New York: Auerbach Publications; 2016. p. 337-58.

7. Evans JH. Basic design concepts. Journal of the American Society for Naval Engineers. 1959;71(4):671-8.

8. Royce WW. Managing the Development of Large Software Systems: Concepts and Techniques. Proceedings of the 9th International Conference on Software Engineering. ICSE '87. Washington, DC, USA: IEEE Computer Society Press; 1987. p. 328–338.

9. Forsberg K, Mooz H. The Relationship of Systems Engineering to the Project Cycle. Engineering Management Journal. 1992;4(3):36-43.

10. Kundu AK, Price MA, Riordan D. Chapter 2: Aircraft Familiarity, Aircraft Design Process, Market Study. Conceptual Aircraft Design: An Industrial Approach. John Wiley & Sons; 2019. .

11. Roskam J. Airplane Design Part 8: Airplane Cost Estimation: Design, Development, Manufacturing and Operating. vol. 8 of Airplane Design. Corporation D, (ed.);.

12. Riaz A, Guenov MD, Molina-Cristobal A. Set-Based Approach to Passenger Aircraft Family Design. Journal of Aircraft. 2017;54(1):310-26.

13. Costa R, Sobek DK. Iteration in engineering design: inherent and unavoidable or product of choices made? International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. vol. 37017; 2003. p. 669-74.

14. Wynn DC, Eckert CM. Perspectives on iteration in design and development. Research in Engineering Design. 2017;28:153-84.

15. Love PE, Mandal P, Li H. Determining the causal structure of rework influences in construction. Construction Management and Economics. 1999;17(4):505-17.

16. Tan J, Otto K, Wood K, et al. A comparison of design decisions made early and late in development. DS 87-2 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 2: Design Processes, Design Organisation and Management, Vancouver, Canada, 21-25.08. 2017; 2017. p. 041-50.

17. Reichelt K, Lyneis J. The dynamics of project performance: benchmarking the drivers of cost and schedule overrun. European management journal. 1999;17(2):135-50.

18. Chucholowski N, Langer S, Ferreira MGG, Forcellini FA, Maier A. Engineering change management report 2012: Survey results on causes and effects, current practice, problems, and strategies in Brazil. Universidade Federal de Santa Carina; 2012.

19. Kennedy BM, Sobek DK, Kennedy MN. Reducing rework by applying set-based practices early in the systems engineering process. Systems Engineering. 2014;17(3):278-96.

20. Eppinger SD, Browning TR. Design structure matrix methods and applications. Cambridge, MA: MIT press; 2012.

21. Lindemann U, Maurer M, Braun T. Structural complexity management: an approach for the field of product design. Berlin: Springer Science & Business Media; 2009.

22. Rötzer S, Schweigert-Recksiek S, Thoma D, Zimmermann M. Attribute dependency graphs: modelling cause and effect in systems design. Design Science. 2022;8:e27.

23. Liker JK, Sobek DK, Ward AC, Cristiano JJ. Involving suppliers in product development in the United States and Japan: evidence for set-based concurrent engineering. IEEE Transactions on Engineering Management. 1996;43(2):165-78.

24. Frey DD, Herder PM, Wijnia Y, Subrahmanian E, Katsikopoulos K, Clausing DP. An Evaluation of the Pugh Controlled Convergence Method. vol. Volume 3: 19th International Conference on Design Theory and Methodology; 1st International Conference on Micro- and Nanosystems; and 9th International Conference on Advanced Vehicle Tire Technologies, Parts A and B of International Design Engineering Technical Conferences and Computers and Information in Engineering Conference; 2007. p. 193-203.

25. Wheelwright SC, Clark KB. Revolutionizing product development: quantum leaps in speed, efficiency, and quality. New York: Simon and Schuster; 1992.

26. Augustine M, Yadav OP, Jain R, Rathore APS. Concept convergence process: A framework for improving product concepts. Computers & Industrial Engineering. 2010;59(3):367-77.

27. Zimmermann M, Königs S, Niemeyer C, Fender J, Zeherbauer C, Vitale R, et al. On the design of large systems subject to uncertainty. Journal of Engineering Design. 2017;28(4):233-54.

28. Ward AC, Seering WP. Quantitative inference in a mechanical design 'compiler'. Journal of Mechanical Design. 1993 03;115(1):29-35.

29. Ward AC. A Theory of Quantitative Inference for Artifact Sets Applied to a Mechanical Design Compiler. PhD Thesis. Massachussets Institute of Technology; 1989.

30. Finch WW, Ward AC. A Set-Based System for Eliminating Infeasible Designs in Engineering Problems Dominated by Uncertainty. Proceedings of the ASME 1997 Design Engineering Technical Conferences. Volume 3: 9th International Design Theory and Methodology Conference.; 1997. .

31. Sobek II DK, Ward AC, Liker JK. Toyota's principles of set-based concurrent engineering. MIT Sloan Management Review. 1999.

32. Sobek DK, Ward AC. Principles from Toyota's set-based concurrent engineering process. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. vol. 97607. American Society of Mechanical Engineers; 1996. p. V004T04A035.

33. Shallcross N, Parnell GS, Pohl E, Specking E. Set-based design: The state-of-practice and research opportunities. Systems Engineering. 2020;23(5):557-78.

34. Terwiesch C, Loch CH, Meyer AD. Exchanging preliminary information in concurrent engineering: Alternative coordination strategies. Organization Science. 2002;13(4):402-19.

35. Bogus SM, Molenaar KR, Diekmann JE. Strategies for overlapping dependent design activities. Construction Management and Economics. 2006;24(8):829-37.

36. McKenney TA, Kemink LF, Singer DJ. Adapting to changes in design requirements using set-based design. Naval Engineers Journal. 2011;123(3):67-77.

37. Parsons MG, Singer DJ, Sauter JA. A hybrid agent approach for set-based conceptual ship design. 10th International Conference on Computer Applications in Shipbuilding (ICCAS); 1999. p. 207-21.

38. Morgan JM, Liker JK. The Toyota Product Development System: Integrating People, Process, and Technology. New York: Productivity Press; 2006.

39. Ward AC, Sobek II DK. Lean Product and Process Development. Cambridge, MA: Lean Enterprise Institute; 2014.

40. Khan M, Al-Ashaab A, Doultsinou A, Shehab E, Ewers P, Sulowski R. Set-based concurrent engineering process within the LeanPPD environment. Improving complex systems today: proceedings of the 18th ISPE International Conference on Concurrent Engineering. Springer; 2011. p. 433-40.

41. Khan MS, Al-Ashaab A, Shehab E, Haque B, Ewers P, Sorli M, et al. Towards lean product and process development. International Journal of Computer Integrated Manufacturing. 2013;26(12):1105-16.

42. Small C. Demonstrating set-based design techniques-a UAV case study. MSc Thesis. University of Arkansas; 2018.

43. Doerry N, Koenig P. Implementing set-based design in DOD acquisitions. 16th Annual Acquisition Research Symposium. Monterey, California, 8-9 May 2019; 2019. .

44. Burrow J, Doerry N, Earnesty M, Was J, Myers J, Banko J, et al. Concept Exploration of the Amphibious Combat Vehicle. SNAME Maritime Convention; 2014. p. D011S001R007.

45. Specking E, Parnell G, Pohl E, Buchanan R. Early design space exploration with model-based system engineering and set-based design. Systems. 2018;6(4):45.

46. Singer DJ, Doerry N, Buckley ME. What is set-based design? Naval Engineers Journal. 2009;121(4):31-43.

47. Mebane WL, Carlson CM, Dowd C, Singer DJ, Buckley ME. Set-based design and the ship to shore connector. Naval Engineers Journal. 2011;123(3):79-92.

48. Al-Ashaab A, Golob M, Attia UM, Khan M, Parsons J, Andino A, et al. The transformation of product development process into lean environment using set-based concurrent engineering: A case study from an aerospace industry. Concurrent Engineering. 2013;21(4):268-85.

49. Dobrovolskyte N. Preliminary design enhancement by incorporating set-based design principles and a navigator. MSc Thesis. Cranfield University; 2015.

50. Raudberget D. Practical Applications of Set-Based Concurrent Engineering in Industry. Strojniški vestnik - Journal of Mechanical Engineering. 2010;56(11):685-95.

51. Dullen S, Verma D, Blackburn M, Whitcomb C. Survey on set-based design (SBD) quantitative methods. Systems Engineering. 2021;24(5):269-92.

52. Isaksson O, Kossmann M, Bertoni M, Eres H, Monceaux A, Bertoni A, et al. Value-Driven Design – A methodology to Link Expectations to Technical Requirements in the Extended Enterprise. INCOSE International Symposium. 2013;23(1):803-19.

53. Russell S, Norvig P. Artificial Intelligence A Modern Approach. 4th edn. Hoboken, NJ: Pearson; 2021.

54. Brailsford SC, Potts CN, Smith BM. Constraint satisfaction problems: Algorithms and applications. European journal of operational research. 1999;119(3):557-81.

55. Finch WW. Predicate Logic Representations for Design Constraints on Uncertainty Supporting the Set-Based Design Paradigm [PhD Thesis]; 1997.

56. Seshan AV. Extension of Set-Based Inference Mechanism for Predicate Logic Design Constraints with an Application to Automotive Power Window Design.. MSc Thesis. Massachussets Institute of Technology; 2000.

57. Qureshi AJ, Dantan JY, Bruyere J, Bigot R. Set based robust design of mechanical systems using the quantifier constraint satisfaction algorithm. Engineering Applications of Artificial Intelligence. 2010;23(7):1173-86.

58. Qureshi AJ, Dantan JY, Bruyere J, Bigot R. Set-based design of mechanical systems with design robustness integrated. International Journal of Product Development. 2014;19(1-3):64-89.

59. Zadeh LA. Fuzzy sets. Information and control. 1965;8(3):338-53.

60. Bellman RE, Zadeh LA. Decision-making in a fuzzy environment. Management science. 1970;17(4):B-141.

61. Ross TJ. Fuzzy logic with engineering applications. John Wiley & Sons; 2009.

62. Nahm YE, Ishikawa H. Representing and aggregating engineering quantities with preference structure for set-based concurrent engineering. Concurrent Engineering. 2005;13(2):123-33.

63. Nahm YE, Ishikawa H. A new 3D-CAD system for set-based parametric design. The International Journal of Advanced Manufacturing Technology. 2006;29:137-50.

64. Wang J, Terpenny J. Interactive evolutionary solution synthesis in fuzzy set-based preliminary engineering design. Journal of intelligent manufacturing. 2003;14:153-67.

65. Singer DJ. A hybrid agent approach for set-based conceptual ship design through the use of a fuzzy logic agent to facilitate communications and negotiation. PhD Thesis. University of Michigan; 2003.

66. McKenney TA, Gray AW, Madrid C, Singer DJ. The use of a fuzzy logic set-based design tool to evaluate varying complexities of late-stage design changes. International Journal of Maritime Engineering. 2012;154(A4).

67. Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Machine learning. 1997;29:131-63.

68. Gunn SR, et al. Support vector machines for classification and regression. ISIS technical report. 1998;14(1):5-16.

69. Ho TK. Random decision forests. Proceedings of 3rd international conference on document analysis and recognition. vol. 1. IEEE; 1995. p. 278-82.

70. Zhang GP. Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 2000;30(4):451-62.

71. Shahan DW, Seepersad CC. Bayesian Network Classifiers for Set-Based Collaborative Design. Journal of Mechanical Design. 2012 06;134(7).

72. Matthews J, Klatt T, Morris C, Seepersad CC, Haberman M, Shahan D. Hierarchical Design of Negative Stiffness Metamaterials Using a Bayesian Network Classifier1. Journal of Mechanical Design. 2016 03;138(4).

73. Morris C, Seepersad CC, Haberman MR. A Set-Based Method for Multilevel Design of Materials and Structures With Manufacturing Uncertainty. vol. Volume 2B: 42nd Design Automation Conference of International Design Engineering Technical Conferences and Computers and Information in Engineering Conference; 2016. .

74. Martins JRRA, Ning A. Engineering Design Optimization. Cambridge University Press;.

75. Hannapel S, Vlahopoulos N, Singer D. Including principles of set-based design in multidisciplinary design optimization. 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference; 2012. p. 5444.

76. Moshaiov A, Snir A, Samina B. Concept-based evolutionary exploration of design spaces by a resolution-relaxation-Pareto approach. 2015 IEEE Congress on Evolutionary Computation (CEC). IEEE; 2015. p. 1845-52.

77. Mattson CA, Messac A. Concept selection using s-Pareto frontiers. AIAA journal. 2003;41(6):1190-8.

78. Farhi E, Moshaiov A. Window-of-interest based multi-objective evolutionary search for satisficing concepts. 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE; 2017. p. 3705-10.

79. Gong D, Sun J, Miao Z. A set-based genetic algorithm for interval many-objective optimization problems. IEEE Transactions on Evolutionary Computation. 2016;22(1):47-60.

80. Han H, Chang S, Kim H. Multiple target exploration approach for design exploration using a swarm intelligence and clustering. Journal of Mechanical Design. 2019;141(9).

81. Fishburn PC. Utility theory for decision making. McLean, VA: Research Analysis Corp.; 1970.

82. Keeney RL, Raiffa H. Decisions with Multiple Objectives: Preferences and Value Trade-Offs. Cambridge University Press; 1993.

83. Malak Jr RJ, Aughenbaugh JM, Paredis CJ. Multi-attribute utility analysis in set-based conceptual design. Computer-Aided Design. 2009;41(3):214-27.

84. Small C, Parnell GS, Pohl E, Goerger SR, Cilli M, Specking E. Demonstrating set-based design techniques: an unmanned aerial vehicle case study. The Journal of Defense Modeling and Simulation. 2020;17(4):339-55.

85. Fitzgerald ME, Ross AM. Artificial intelligence analytics with multi-attribute tradespace exploration and set-based design. Procedia Computer Science. 2019;153:27-36.

86. Feinberg EA, Shwartz A. Handbook of Markov decision processes: methods and applications. vol. 40. Springer Science & Business Media; 2012.

87. Bellman R. The theory of dynamic programming. Bulletin of the American Mathematical Society. 1954;60(6):503-15.

88. Sutton RS, Barto AG. Reinforcement learning: An introduction. MIT press; 2018.

89. Miller SW, Simpson TW, Yukish MA. Design as a sequential decision process: a method for reducing design set space using models to bound objectives. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. vol. 57076. American Society of Mechanical Engineers; 2015. p. V02AT03A020.

90. Rapp S, Chinnam R, Doerry N, Murat A, Witus G. Product development resilience through set-based design. Systems Engineering. 2018;21(5):490-500.

91. Chhabra JP, Warn GP. A method for model selection using reinforcement learning when viewing design as a sequential decision process. Structural and Multidisciplinary Optimization. 2019;59:1521-42.

92. Kumar P, Tandon P. Classification and mitigation of uncertainty as per the product design stages: framework and case study. Journal of the Brazilian Society of Mechanical Sciences and Engineering. 2017;39(11):4785-806.

93. Wynn DC, Grebici K, Clarkson PJ. Modelling the evolution of uncertainty levels during design. International Journal on Interactive Design and Manufacturing (IJIDeM). 2011;5:187-202.

94. Han X, Li R, Wang J, Ding G, Qin S. A systematic literature review of product platform design under uncertainty. Journal of Engineering Design. 2020;31(5):266-96.

95. McManus H, Hastings D. A framework for understanding uncertainty and its mitigation and exploitation in complex systems. INCOSE international symposium. vol. 15. Wiley Online Library; 2005. p. 484-503.

96. Kontogiannis SG, Demange J, Savill AM, Kipouros T. A comparison study of two multifidelity methods for aerodynamic optimization. Aerospace Science and Technology. 2020;97:105592.

97. Eckert C, Clarkson PJ, Zanker W. Change and customisation in complex engineering domains. Research in engineering design. 2004;15:1-21.

98. Antonsson EK, Otto KN. Imprecision in Engineering Design. Journal of Mechanical Design. 1995 06;117(B):25-32.

99. Jarratt T, Eckert CM, Caldwell NH, Clarkson PJ. Engineering change: an overview and perspective on the literature. Research in engineering design. 2011;22:103-24.

100. Li W, Moon YB. Modeling and managing engineering changes in a complex product development process. The International Journal of Advanced Manufacturing Technology. 2012;63:863-74.

101. Maier A, Langer S. Engineering change management report 2011: survey results on causes and effects, current practice, problems, and strategies in Denmark. Glasgow, UK; 2011.

102. Clarkson PJ, Simons C, Eckert C. Predicting Change Propagation in Complex Design . Journal of Mechanical Design. 2004 10;126(5):788-97.

103. Koh EC, Caldwell NH, Clarkson PJ. A method to assess the effects of engineering change propagation. Research in Engineering Design. 2012;23:329-51.

104. Hamraz B, Hisarciklilar O, Rahmani K, Wynn DC, Thomson V, Clarkson PJ. Change prediction using interface data. Concurrent Engineering. 2013;21(2):141-54.

105. Taguchi G, Clausing D. Robust quality. Harvard business review. 1990;68(1):65-75.

106. Eckert C, Isaksson O, Earl C. Design margins: a hidden issue in industry. Design Science. 2019;5:e9.

107. Brahma A, Wynn DC. Margin value method for engineering design improvement. Research in Engineering Design. 2020;31:353-81.

108. Guenov MD, Chen X, Molina-Cristóbal A, Riaz A, van Heerden AS, Padulo M. Margin allocation and tradeoff in complex systems design and optimization. AIAA journal. 2018;56(7):2887-902.

109. Marler RT, Arora JS. The weighted sum method for multi-objective optimization: new insights. Structural and multidisciplinary optimization. 2010;41:853-62.

110. Haimes YY, Lasdon LS, Wismer DA. On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. IEEE Transactions on Systems, Man, and Cybernetics. 1971;SMC-1(3):296-7.

111. Das I, Dennis JE. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. SIAM journal on optimization. 1998;8(3):631-57.

112. Barricelli NA, et al. Esempi numerici di processi di evoluzione. Methodos. 1954;6(21-22):45-68.

113. Fogel DB. Evolutionary Computation: The Fossil Record. Press I, (ed.);. Available at: `https://ieeexplore.ieee.org/book/5263042`.

114. Holland JH. Adaptation in Natural and Artificial Systems. Cambridge, MA, USA: MIT Press; 1975.

115. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation. 2002;6(2):182-97.

116. Kung HT, Luccio F, Preparata FP. On finding the maxima of a set of vectors. Journal of the ACM (JACM). 1975;22(4):469-76.

117. Deb K, Jain H. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. IEEE Transactions on Evolutionary Computation. 2014;18(4):577-601.

118. Seada H, Deb K. A Unified Evolutionary Optimization Procedure for Single, Multiple, and Many Objectives. IEEE Transactions on Evolutionary Computation. 2016;20(3):358-69.

119. Ashley H. On making things the best-aeronautical uses of optimization. Journal of Aircraft. 1982;19(1):5-28.

120. Schmit LA. Structural synthesis-its genesis and development. AIAA Journal. 1981;19(10):1249-63.

121. Pironneau O. On optimum design in fluid mechanics. Journal of fluid mechanics. 1974;64(1):97-110.

122. Hicks RM, Henne PA. Wing design by numerical optimization. Journal of Aircraft. 1978;15(7):407-12.

123. Schmit LA, Thornton WA. NASA CR-144. Synthesis of an airfoil at supersonic Mach number. National Aeronautics and Space Administration; 1965.

124. Haftka RT. NASA TN-D-7264. Automated procedure for design of wing structures to satisfy strength and flutter requirements. National Aeronautics and Space Administration; 1973.

125. Haftka RT. Optimization of flexible wing structures subject to strength and induced drag constraints. AIAA Journal. 1977;15(8):1101-6.

126. Livne E. Integrated Aeroservoelastic Optimization: Status and Direction. Journal of Aircraft. 1999;36(1):122-45.

127. Kroo I, Altus S, Braun R, Gage P, Sobieski I. Multidisciplinary optimization methods for aircraft preliminary design; 1994. .

128. Antoine NE, Kroo IM. Framework for Aircraft Conceptual Design and Environmental Performance Studies. AIAA Journal. 2005;43(10):2100-9.

129. Henderson RP, Martins JRRA, Perez RE. Aircraft conceptual design for optimal environmental performance. The Aeronautical Journal. 2012;116(1175):1–22.

130. Martins JRRA, Lambe AB. Multidisciplinary Design Optimization: A Survey of Architectures. AIAA Journal. 2013;51(9):2049-75.

131. Balling RJ, Sobieszczanski-Sobieski J. Optimization of coupled systems - A critical overview of approaches. AIAA Journal. 1996;34(1):6-17.

132. Chittick IR, Martins JR. An asymmetric suboptimization approach to aerostructural optimization. Optimization and Engineering. 2009;10:133-52.

133. Sobieski IP, Kroo IM. Collaborative Optimization Using Response Surface Estimation. AIAA Journal. 2000;38(10):1931-8.

134. Kim HM, Michelena NF, Papalambros PY, Jiang T. Target Cascading in Optimal System Design . Journal of Mechanical Design. 2003 09;125(3):474-80.

135. Lambe AB, Martins JRRA. Extensions to the Design Structure Matrix for the Description of Multidisciplinary Design, Analysis, and Optimization Processes. Structural and Multidisciplinary Optimization. 2012 Aug;46(2):273-84.

136. Gray JS, Hwang JT, Martins JR, Moore KT, Naylor BA. OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization. Structural and Multidisciplinary Optimization. 2019;59:1075-104.

137. Cramer EJ, Dennis JE Jr, Frank PD, Lewis RM, Shubin GR. Problem Formulation for Multidisciplinary Optimization. SIAM Journal on Optimization. 1994;4(4):754-76.

138. Di Bianchi DH, Secco NR, Silvestre FJ. A framework for enhanced decision-making in aircraft conceptual design optimisation under uncertainty. The Aeronautical Journal. 2021;125(1287):777-806.

139. Pagliuca G, Kipouros T, Savill M. Surrogate modelling for wing planform multidisciplinary optimisation using model-based engineering. International Journal of Aerospace Engineering. 2019;2019.

140. McKay MD, Beckman RJ, Conover WJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics. 2000;42(1):55-61.

141. Kocis L, Whiten WJ. Computational investigations of low-discrepancy sequences. ACM Transactions on Mathematical Software (TOMS). 1997;23(2):266-94.

142. Sobol I, Levitan YL. A pseudo-random number generator for personal computers. Computers & Mathematics with Applications. 1999;37(4-5):33-40.

143. Demange J, Savill AM, Kipouros T. Multifidelity optimization for high-lift airfoils. 54th AIAA Aerospace Sciences Meeting; 2016. p. 0557.

144. Broomhead DS, Lowe D. Radial basis functions, multi-variable functional interpolation and adaptive networks. Royal Signals and Radar Establishment Malvern (United Kingdom); 1988.

145. Jones DR. A taxonomy of global optimization methods based on response surfaces. Journal of global optimization. 2001;21:345-83.

146. Jones DR, Schonlau M, Welch WJ. Efficient global optimization of expensive black-box functions. Journal of Global optimization. 1998;13(4):455.

147. Forrester AI, Keane AJ. Recent advances in surrogate-based optimization. Progress in aerospace sciences. 2009;45(1-3):50-79.

148. Krige DG. A statistical approach to some basic mine valuation problems on the Witwatersrand. Journal of the Southern African Institute of Mining and Metallurgy. 1951;52(6):119-39.

149. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. Statistical science. 1989;4(4):409-23.

150. Schmidhuber J. Deep learning in neural networks: An overview. Neural networks. 2015;61:85-117.

151. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. nature. 1986;323(6088):533-6.

152. Goodfellow I, Bengio Y, Courville A. Deep Learning. Cambridge, MA: MIT Press;.

153. Stone M. Cross-validatory choice and assessment of statistical predictions. Journal of the royal statistical society: Series B (Methodological). 1974;36(2):111-33.

154. Kohavi R, et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. Ijcai. vol. 14. Montreal, Canada; 1995. p. 1137-45.

155. Chen M, Ebert D, Hagen H, Laramee RS, van Liere R, Ma KL, et al. Data, Information, and Knowledge in Visualization. IEEE Computer Graphics and Applications. 2009;29(1):12-9.

156. Inselberg A, Dimsdale B. Parallel coordinates: a tool for visualizing multi-dimensional geometry. Proceedings of the first IEEE conference on visualization: visualization90. IEEE; 1990. p. 361-78.

157. Multidimensional detective. Proceedings of VIZ'97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium. IEEE; 1997. p. 100-7.

158. Inselberg A, Dimsdale B. Parallel coordinates. Human-Machine Interactive Systems. 2009:199-233.

159. Kipouros T, Inselberg A, Parks G, Savill AM. Parallel coordinates in computational engineering design. 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference; 2013. p. 1750.

160. Piotrowski W, Kipouros T, Clarkson PJ. Enhanced Interactive Parallel Coordinates using Machine Learning and Uncertainty Propagation for Engineering Design. 2019 15th International Conference on eScience (eScience); 2019. p. 339-48.

161. Agency IE. World Energy Statistics 2016. OECD; 2016. Available at: `https://www.oecd-ilibrary.org/content/publication/9789264263079-en`.

162. Airbus. Global Market Forecast 2022-2041; 2022. Available at: `https://www.airbus.com/en/products-services/commercial-aircraft/market/global-market-forecast`.

163. Boeing. Commercial Market Outlook 2022–2041; 2022. Available at: `https://www.boeing.com/commercial/market/commercial-market-outlook/index.page`.

164. Commission E, for Mobility DG, Transport, for Research DG, Innovation. Flightpath 2050 : Europe's vision for aviation : maintaining global leadership and serving society's needs. European Commission; 2011.

165. Zaporozhets O, Isaienko V, Synylo K. Trends on current and forecasted aircraft hybrid electric architectures and their impact on environment. Energy. 2020;211:118814.

166. Bravo-Mosquera PD, Catalano FM, Zingg DW. Unconventional aircraft for civil aviation: A review of concepts and design methodologies. Progress in Aerospace Sciences. 2022;131:100813.

167. Epstein AH, O'Flarity SM. Considerations for Reducing Aviation's CO2 with Aircraft Electric Propulsion. Journal of Propulsion and Power. 2019;35(3):572-82.

168. Kim HD, Perry AT, Ansell PJ. A review of distributed electric propulsion concepts for air vehicle technology. 2018 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS). IEEE; 2018. p. 1-21.

169. Gray JS, Martins JRRA. Coupled aeropropulsive design optimisation of a boundary-layer ingestion propulsor. The Aeronautical Journal. 2019;123(1259):121–137.

170. Sahoo S, Zhao X, Kyprianidis KG, Kalfas A. Performance Assessment of an Integrated Parallel Hybrid-Electric Propulsion System Aircraft. vol. Volume 3: Coal,

Biomass, Hydrogen, and Alternative Fuels; Cycle Innovations; Electric Power; Industrial and Cogeneration; Organic Rankine Cycle Power Systems of Turbo Expo: Power for Land, Sea, and Air; 2019. .

171. Bradley MK, Droney CK. Subsonic Ultra Green Aircraft Research: Phase II, Volume II. Hybrid Electric design exploration. National Aeronautics and Space Administration; 2015.

172. Rheaume JM, Macdonald M, Lents CE. Commercial Hybrid Electric Aircraft Thermal Management System Design, Simulation, and Operation Improvements. 2019 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS); 2019. p. 1-23.

173. Trawick D, Perullo C, Armstrong M, Snyder D, Tai JC, Mavris DN. Development and application of GT-HEAT for the electrically variable engine (TM) design. 55th AIAA Aerospace Sciences Meeting; 2017. p. 1922.

174. Antcliff KR, Capristan FM. Conceptual Design of the Parallel Electric-Gas Architecture with Synergistic Utilization Scheme (PEGASUS) Concept. 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference; 2017. .

175. Gesell H, Wolters F, Plohr M. System analysis of turbo-electric and hybrid-electric propulsion systems on a regional aircraft. The Aeronautical Journal. 2019;123(1268):1602–1617.

176. Benegas Jayme D. Evaluation of the Hybrid-Electric Aircraft Project Airbus E-Fan X. MSc Thesis. HAW Hamburg; 2019.

177. Aero Z. Zunum Aero Aircraft; 2020. Available at: `http://zunum.aero/aircraft`.

178. Swannet K. Optimal Control And Energy Management For Hybrid Aircraft. MSc Thesis. TU Delft; 2022.

179. Sahoo S, Zhao X, Kyprianidis K. A review of concepts, benefits, and challenges for future electrical propulsion-based aircraft. Aerospace. 2020;7(4):44.

180. Gkoutzamanis VG, Tsentis SE, Valsamis Mylonas OS, Kalfas AI, Kyprianidis KG, Tsirikoglou P, et al. Thermal Management System Considerations for a Hybrid-Electric Commuter Aircraft. Journal of Thermophysics and Heat Transfer. 2022;36(3):650-66.

181. Pornet C, Gologan C, Vratny PC, Seitz A, Schmitz O, Isikveren AT, et al. Methodology for Sizing and Performance Assessment of Hybrid Energy Aircraft. Journal of Aircraft. 2015;52(1):341-52.

182. Riboldi CED, Gualdoni F. An integrated approach to the preliminary weight sizing of small electric aircraft. Aerospace Science and Technology. 2016;58:134-49.

183. Cinar G, Emeneth M, Mavris DN. A Methodology for Sizing and Analysis of Electric Propulsion Subsystems for Unmanned Aerial Vehicles. 54th AIAA Aerospace Sciences Meeting; 2016. .

184. de Vries R, Brown M, Vos R. Preliminary Sizing Method for Hybrid-Electric Distributed-Propulsion Aircraft. Journal of Aircraft. 2019;56(6):2172-88.

185. Pornet C, Gologan C, Vratny PC, Seitz A, Schmitz O, Isikveren AT, et al. Methodology for Sizing and Performance Assessment of Hybrid Energy Aircraft. Journal of Aircraft. 2015;52(1):341-52.

186. Zamboni J, Vos R, Emeneth M, Schneegans A. A Method for the Conceptual Design of Hybrid Electric Aircraft. AIAA Scitech 2019 Forum; 2019. .

187. Trawick DR. A Methodology for the determination of optimal operational schedules of hybrid electric architectures. PhD Thesis. Georgia Institute of Technology; 2018.

188. Zhang J, Roumeliotis I, Zolotas A. Model-based fully coupled propulsion-aerodynamics optimization for hybrid electric aircraft energy management strategy. Energy. 2022;245:123239.

189. Voskuijl M, Van Bogaert J, Rao AG. Analysis and design of hybrid electric regional turboprop aircraft. CEAS Aeronautical Journal. 2018;9:15-25.

190. Trawick D, Milios K, Gladin JC, Mavris DN. A Method for Determining Optimal Power Management Schedules for Hybrid Electric Airplanes. 2019 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS); 2019. p. 1-21.

191. Perullo C, Mavris D. A review of hybrid-electric energy management and its inclusion in vehicle sizing. Aircraft Engineering and Aerospace Technology: An International Journal. 2014.

192. Brelje BJ, Martins JR. Electric, hybrid, and turboelectric fixed-wing aircraft: A review of concepts, models, and design approaches. Progress in Aerospace Sciences. 2019;104:1-19.

193. Smith N. Interview Iain Gray. Engineering & Technology. 2017;12(11):66-9.

194. Eisenhut D, Moebs N, Windels E, Bergmann D, Geiß I, Reis R, et al. Aircraft requirements for sustainable regional aviation. Aerospace. 2021;8(3):61.

195. Williams CK, Rasmussen CE. Gaussian processes for machine learning. vol. 2. Cambridge, MA: MIT press; 2006.

196. Spinelli A. PDOPT - Probabilistic Design and OPTimisation; 2022. GitHub. Available at: `https://github.com/spinjet/pdopt-code`.

197. Joe S, Kuo FY. Constructing Sobol sequences with better two-dimensional projections. SIAM Journal on Scientific Computing. 2008;30(5):2635-54.

198. McKay M, Beckman R, Conover W. Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. Technometrics. 1979;21(2):239-45.

199. Joblib: running Python functions as pipeline jobs;. Available at: `https://joblib.readthedocs.io/`.

200. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825-30.

201. Blank J, Deb K. pymoo: Multi-Objective Optimization in Python. IEEE Access. 2020;8:89497-509.

202. Spinelli A, Anderson L, Enalou HB, Zaghari B, Kipouros T, Laskaridis P. Application of Probabilistic principles to Set-Based Design for the optimisation of a hybrid-electric propulsion system. IOP Conference Series: Materials Science and Engineering. vol. 1226. IOP Publishing; 2022. p. 012064.

203. Spinelli A, Enalou HB, Zaghari B, Kipouros T, Laskaridis P. Application of Probabilistic Set-Based Design Exploration on the Energy Management of a Hybrid-Electric Aircraft. Aerospace. 2022;9(3):147.

204. MacMillan WL. Development of a modular-type computer program for the calculation of gas turbine off-design performance. PhD Thesis. Cranfield University; 1974.

205. DuBois D, Paynter GC. Fuel Flow Method2 for Estimating Aircraft Emissions. SAE Transactions: Journal of Aerospace. 2006;115:1-14.

206. Filippone A, Bojdo N. Statistical model for gas turbine engines exhaust emissions. Transportation Research Part D: Transport and Environment. 2018;59:451-63.

207. ATR. ATR-42-600 Factsheet; 2020. Available at: `https://www.atr-aircraft.com/wp-content/uploads/2020/07/Factsheets_-_ATR_42-600.pdf`.

208. Edstrom K, Dominko R, Fichtner M, Otuszewski T, Perraud S, Punckt C, et al.. BATTERY 2030 Roadmap; 2020. Available at: `https://battery2030.eu/wp-content/uploads/2021/08/c_860904-l_1-k_roadmap-27-march.pdf`.

209. Ojeda-Rodriguez A, Gonzalez-Vizuete P, Bernal-Mendez J, Martin-Prats MA. A survey on bidirectional DC/DC power converter topologies for the future hybrid and all electric aircrafts. Energies. 2020;13(18):4883.

210. Sirimanna S, Thanatheepan B, Lee D, Agrawal S, Yu Y, Wang Y, et al. Comparison of electrified aircraft propulsion drive systems with different electric motor topologies. Journal of Propulsion and Power. 2021;37(5):733-47.

211. Spinelli A, Krupa GP, Kipouros T, Berseneff B, Fiette S. Investigation of the operational flexibility of a regional hybrid-electric aircraft. 12th EASN International Conference on "Innovation in Aviation & Space for opening New Horizons". vol. 2526 of Journal of Physics: Conference Series. IOP Publishing; 2023. p. 012021.

212. Ceraolo M, Lutzemberger G, Huria T. Experimentally-Determined Models for High-Power Lithium Batteries. SAE 2011 World Congress & Exhibition. SAE International; 2011. .

213. Chin JC, Look K, McNichols EO, Hall DL, Gray JS, Schnulo SL. Battery Cell-to-Pack Scaling Trends for Electric Aircraft. 2021 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS); 2021. p. 1-15.

214. Ecker M, Nieto N, Käbitz S, Schmalstieg J, Blanke H, Warnecke A, et al. Calendar and cycle life study of Li(NiMnCo)O2-based 18650 lithium-ion batteries. Journal of Power Sources. 2014;248:839-51.

215. Spinelli A, Krupa GP, Kipouros T. Set-Based Design Space Exploration to Investigate the Effect of Energy Storage Durability on the Energy Management Strategy of a Hybrid-Electric Aircraft. AIAA SCITECH 2023 Forum; 2023. p. 0837.

216. Spinelli A, Krupa GP, Kipouros T. Battery pack technological considerations for hybrid-electric regional aircraft feasibility. AIAA Journal. 2023, In preparation.

217. ATR. ATR-72-600 Factsheet; 2022. Available at: `https://www.atr-aircraft.com/wp-content/uploads/2020/07/Factsheets_-_ATR_72-600.pdf`.

218. Schmalstieg J, Käbitz S, Ecker M, Sauer DU. A holistic aging model for Li (NiMnCo) O2 based 18650 lithium-ion batteries. Journal of Power Sources. 2014;257:325-34.

219. Montgomery DC, Runger GC. Applied Statistics and Probability for Engineers. Hoboken, NJ: John Wiley & Sons; 2010.

220. Kipouros T, Spinelli A. D2.2: Design Methodology Report. Project FutPrint50;.

221. Brenner F, Eisenhut D, Enalou HB, Spinelli A, Goyal J, Mangold J, et al. D4.2: Design Methodology Report. Project FutPrint50;.

222. Glymour C, Zhang K, Spirtes P. Review of causal discovery methods based on graphical models. Frontiers in Genetics. 2019;10.

223. Blank J, Deb K. Pymoo: Multi-Objective Optimization in Python Documentation; 2020. Available at: `https://pymoo.org/index.html`.

224. Blank J, Deb K. A Running Performance Metric and Termination Criterion for Evaluating Evolutionary Multi- and Many-objective Optimization Algorithms. 2020 IEEE Congress on Evolutionary Computation (CEC); 2020. p. 1-8.

Appendices

Appendix A

Example PDOPT Setup

This appendix presents the step-by-step process for performing a PDOPT analysis. For this tutorial, the specific details of the simulation function will be overlooked, focusing more on developing an entire runnable test case. Generally, it is composed of an understanding and definition of the problem in terms of input parameters and output quantities and the layout of the run file to perform this analysis. The example presented here can be found in the GitHub example folder and is based on experiment CCL presented in Chapter 6.

A.1 Definition of the problem into PDOPT components

The problem is a design space exploration of hybrid-electric aircraft energy management strategies, where the goal is to minimise fuel consumption and NO_x emissions under a maximum take-off mass constraint. After understanding the problem to analyse, the user should identify the input parameters of the model and the responses he desires to optimise or impose constraints on. For this example, the following input parameters have been identified:

Name	Type	Range	Description
climb_h0	continuous	[0,1]	The DOH at the beginning of the climb segment
climb_h1	continuous	[0,1]	The DOH at the end of the climb segment
cruise_h0	continuous	[0,1]	The DOH at the beginning of the cruise segment
cruise_h1	continuous	[0,1]	The DOH at the end of the cruise segment

Table A.1: Identified input parameters for the problem under analysis.

The user should decide if these parameters are continuous or discrete, i.e. describe a range of possible values or a set of discrete choices. In the first case, the user should also fix the boundaries of this range. This information will be used in defining the `input.csv` file and the arguments of the evaluation function.

Then the user should list the responses they wish to analyse and define them as objectives or constraints. Special care should be given to objectives, considered ”soft constraints” for the exploration phase. This is useful if the user wishes to understand which areas of the design space are guaranteed to provide at most (or least) a certain value. For

instance, the user wishes to see which areas of the design space are more likely to have a minimum value of NO_x emissions under a certain figure. For the example problem, the following requirements are identified:

Name	Type	Operator	Value	Description
Mf	objective	min (minimise)	nan	Fuel Burnt during the mission
M_NOx	objective	min (minimise)	nan	NO_x Emission during the mission
TOM	constraint	lt (less than)	20000	Take-off mass, constrainted to 20000 kg

Table A.2: Identified response parameters for the problem under analysis.

These response parameters will then define the output of the evaluation function and the `responses.csv` file.

A.2 Definition of the .csv Files

With the problem broken down into its PDOPT components, it is possible to define the input file with the format described in Sec. 4.6.1. In this case, the input file is as follows:

```
name,type,lb,ub,levels,uq_dist,uq_var_l,uq_var_u
climb_h0,continous,0,1,4,nan,nan,nan
climb_h1,continous,0,1,4,nan,nan,nan
cruise_h0,continous,0,1,4,nan,nan,nan
cruise_h1,continous,0,1,4,nan,nan,nan
```

Listing A.1: `input.csv` contents.

The number of levels for each parameter should be selected with caution. Few levels make a faster analysis, especially during the search phase; however, the exploration phase would likely average the probability, leading to a potential loss of resolution of the decision boundary.

On the other hand, a high number of levels leads to an unnecessary number of surviving sets, which would considerably slow down the search process. It is advised to use 3 or 4 levels when starting and adjusting it by performing a few exploration steps.

Along the input file, the responses csv file must be defined as described in Sec. 4.6.2. Using the information from the problem breakdown, the `response.csv` file is as follows:

```
name,type,op,val,pSat
Mf,objective,min,nan,nan
M_NOx,objective,min,nan,nan
TOM,constraint,lt,20000,0.5
```

Listing A.2: `response.csv` contents.

As discussed previously, it is possible to include ”soft constraints” on the objectives for the exploration phase. It is suggested not to do so at the first run of the problem unless the user is already aware of the range of response values. Furthermore, soft constraints present a phenomenon of ”leakage”, whereas some sub-optimal sets are kept as they might have a few points satisfying the constraint.

A.3 Definition of the Evaluation Function

In this case, the evaluation function was developed by extending the `ExtendableModel()` class. For two reasons, this was preferred over defining just a Python function (as described in Sec. 4.6.3). First, it is possible to load in the `ExtendableModel()` object data regarding what is being modelled (for instance, model parameters that are fixed during each run of PDOPT). Second, it allows to package with the model data other routines, such as a post-processing function. This is the case for this example problem. It must contain the `run()` method, which must output the data required from the `response.csv` file as a Python dictionary.

The model class is, therefore, defined as:

```
class Experiment(ExtendableModel):

def __init__(self, input_parameters, architecture, mission_file):

## This constructor method allows to store in the Experiment
## object some information regarding the aircraft and mission
self.inp        = list(pd.read_csv(input_parameters)['name'])
self.arch       = architecture
self.arch0      = architecture
self.mission    = mission_file

def run(self, *args, **kwargs):

## Omitted code for performing the analysis

output = {
 'TOM'      : analysis.iloc[-1].mass,
 'Mf'       : analysis.iloc[-1].m_fl,
 'M_NOx'    : analysis.iloc[-1].m_NOx,
}

return output

def postprocess_analysis(self, *args, **kwargs):

## Omitted post-processing code, runs the same analysis
## but returns more information

return results
```

Listing A.3: Model class

A.4 Definition of the Runfile

Now that all the ingredients required to perform the analysis are present, it is possible to construct the Python script to run it.

First a `DesignSpace()` object has to be created, which will store all the information regarding the problem. It contains a list of all the sets (which are instances of the `DesignSet()` object) and the Pandas DataFrames containing the exploration and search phases results. The directory of the `'input.csv'` and `'response.csv'` are required as

arguments. To avoid loss of data, the `pickle` library is used to store each PDOPT object after each step is performed.

```
folder = 'run'
architecture = json.load(open('data/architecture.json', 'r'))
mission = 'data/mission_original.csv'
experiment = Experiment(folder + '/input.csv',
architecture, mission)

# Check if a design space is already present otherwise create it
if exists(folder + '/design_space.pk') and restart:
design_space = pk.load(open(folder + '/design_space.pk','rb'))
else:
design_space = DesignSpace(folder + '/input.csv',
folder + '/response.csv')
pk.dump(design_space, open(folder + '/design_space.pk','wb'))
```

Listing A.4: Design space setup.

The next step is defining the exploration phase. The `ProbabilisticExploration()` object is used. It requires as arguments the design space object and the model object. Optional arguments include the definition of a surrogate data file, useful to expedite multiple analyses and the number of samples to use. By default, it will use 100 sampled points in the entire design space (using a Latin Hypercube scheme) and 30 points to validate its regression. These points are sampled from the provided model. Once the exploration object is trained, it is run with the `.run()` method, by passing two arguments: the number of samples for evaluating each set and the minimum satisfaction probability (as discussed in Sec. 4.3). The results from the exploration phase are stored in the `DesignSpace()` object; they can be retrieved with the `.get_exploration_results()` method.

```
n_exp_train = 100

# Check if there is already a trained exploration object
if exists(folder + '/exploration.pk') and restart:
exploration = pk.load(open(folder + '/exploration.pk','rb'))
else:
exploration = ProbabilisticExploration(design_space,
experiment,
surrogate_training_data_file=folder + '/samples.csv',
n_train_points=n_exp_train)

for k in exploration.surrogates:
s = exploration.surrogates[k]
#Print the R score of each surrogate, to check
#their training. For smooth problems it should be
#above 0.8.
print(f'Surrogate {s.name} with r = {s.score:.4f}')

pk.dump(exploration, open(folder + '/exploration.pk','wb'))

n_exp_samples = 100
P_exploration = 0.5

# Check if exploration has been done already
if exists(folder + '/exp_results.csv') and restart:
pass
```

```
else:
exploration.run(n_exp_samples, P_exploration)
design_space.get_exploration_results().to_csv(folder \
+ '/exp_results.csv', index=False)

#Update the saved design object
pk.dump(design_space, open(folder + '/design_space.pk','wb'))
```

Listing A.5: Exploration phase.

It is possible to perform different exploration analyses by using copies of the `DesignSpace()` object and changing the satisfaction probability. This allows one to study how restrictive the requirements are over the design space. Once complete, the exploration step marks some areas of the design space as "Discarded" by setting the `.isDiscarded` boolean of each set as true. The following phase is the search step. A multi-objective optimisation algorithm is introduced in the surviving sets to identify the local Pareto front. The `Optimisation()` object is the standard deterministic multi-objective optimiser. The arguments are the design space, model, and options regarding the stopping criteria or the genetic algorithm population size. For the PDOPT analysis, it is recommended to use a fixed number of evaluation functions (`n_max_evals`) or generations (`n_max_gen`). This would allow a fair search in every set, as it would ignore the topology of the evaluation function. Instead, using criteria based on convergence would introduce distortions, as some areas of the design space might be shallow (slow convergence) or have multiple local minima/maxima.

Once set up is complete, optimisation is started with the `Optimisation.run()` method. Once complete, results can be retrieved from the `.get_optimum_results()` method of the design space object. It returns a Pandas DataFrame containing each set's optimal points and the constrained quantities' return value. The optimisation step code is:

```
optimisation = Optimisation(design_space, experiment, n_max_evals
  =2000)


# Check if optimisation has been done already
if exists(folder + '/opt_results_raw.csv') and restart:
pass
else:
optimisation.run()
df_opt = design_space.get_optimum_results()
df_opt.to_csv(folder + '/opt_results_raw.csv', index=False)

#Update the saved design object
pk.dump(design_space, open(folder + '/design_space.pk','wb'))


## Post-processing code to recover other output quantities from
  the model
if exists(folder + '/opt_results.csv') and restart:
pass
else:
inp_pars = design_space.par_names

df_opt['M_batt'] = 0
df_opt['eff']    = 0
```

```
df_opt['M_CO']    = 0
df_opt['M_CO2']   = 0
df_opt['eff_GT_cl'] = 0
df_opt['eff_GT_cr'] = 0
df_opt['eff_cl'] = 0
df_opt['eff_cr'] = 0

for index in tqdm(range(len(df_opt))):
t_out = experiment.postprocess_analysis(*df_opt.loc[index,
  inp_pars].tolist())

df_opt.loc[index, 'M_batt'] = t_out.iloc[-1].m_bat
df_opt.loc[index, 'eff'] = t_out.iloc[-1].eff
df_opt.loc[index, 'M_CO'] = t_out.iloc[-1].m_CO
df_opt.loc[index, 'M_CO2'] = t_out.iloc[-1].m_fl * 3
df_opt.loc[index, 'eff_GT_cl'] = t_out.loc[t_out['tag'] == 'climb'
  ].P_GT_out.sum() / t_out.loc[t_out['tag'] == 'climb'].P_GT_in.
  sum()
df_opt.loc[index, 'eff_GT_cr'] = t_out.loc[t_out['tag'] == 'cruise
  '].P_GT_out.sum() / t_out.loc[t_out['tag'] == 'cruise'].P_GT_in.
  sum()
df_opt.loc[index, 'eff_cl'] = t_out.loc[t_out['tag'] == 'climb'].
  P_req.sum() / t_out.loc[t_out['tag'] == 'climb'].P_tot.sum()
df_opt.loc[index, 'eff_cr'] = t_out.loc[t_out['tag'] == 'cruise'].
  P_req.sum() / t_out.loc[t_out['tag'] == 'cruise'].P_tot.sum()

t_out.to_csv(folder + f'/missions/{index}_mission_output.csv')

df_opt.to_csv(folder + '/opt_results.csv', index=False)

#Runtime Report
generate_run_report(folder + '/report.txt',
design_space, optimisation, exploration)
```

Listing A.6: Search phase.

By design, the evaluation function returns only the quantities of interest specified in `response.csv`. In order to recover other information after the optimisation run, a post-processing evaluation function can be used by inputting the optimal points found. Finally, the `generate_run_report()` function is used to produce a text output containing information on the total run time and number of discarded sets.

A.5 Running the Case and Post-processing the results

The file is run now that the full script is ready with the required input files. While the code is running, it will output to the console the current progress (Listing A.7). In particular, after training the Gaussian processes, it will output the R^2 score (Coefficient of Determination) of each trained surrogate model. This gives a good indication of the reliability of the exploration process. Usually, if the model's response is smooth, it is expected to be at least above 0.8. Increasing the number of training points may be necessary for non-smooth models.

The output during the search phase is the underlying U-LSGA-III output from the `pymoo` library (201). It informs the user about the progress of each generation (`n_gen`)

with the number of function evaluations made so far (n_eval), the minimum and average constraint violations (cv (min) and cv (avg)), the number of non-dominated solutions found (n_nds) and the change of the Pareto convergence indicator (columns epsindicator). The pymoo documentation (223) provides more detail about this output in the Display section and points to an article by Blank and Deb on the multiobjective convergence indicator (224).

```
Generating Training Data:  100%|**********************|100/100 [01:23<00:00,
  1.20it/s]
Generating Testing Data: 100%|**************************| 30/30 [00:28<00:00,
  1.06it/s]
Training Surrogate Responses: 100%|************************| 3/3 [00:00<00:00,
  36.59it/s]
Surrogate M_NOx with r = 0.9931
Surrogate TOM with r = 1.0000
Surrogate Mf with r = 0.9999
Exploring the Design Space: 100%|**********************| 256/256 [00:00<00:00,
  639.79it/s]
Searching in the Design Space:   0%|                              | 0/25 [00:00<?,
  ?it/s]

  ==========================================================================

n_gen |  n_eval |   cv (min)   |   cv (avg)   |  n_nds  |     eps      |
  indicator

  ==========================================================================

1 |     101 |  0.00000E+00 |  0.00000E+00 |      14 |            - |            -
2 |     202 |  0.00000E+00 |  0.00000E+00 |      17 |  0.012666136 |        ideal
3 |     303 |  0.00000E+00 |  0.00000E+00 |      19 |  0.009879243 |        ideal
4 |     404 |  0.00000E+00 |  0.00000E+00 |      19 |  0.017813004 |        ideal
5 |     505 |  0.00000E+00 |  0.00000E+00 |      21 |  0.046526161 |        ideal
6 |     606 |  0.00000E+00 |  0.00000E+00 |      21 |  0.017590293 |        ideal
7 |     707 |  0.00000E+00 |  0.00000E+00 |      20 |  0.058322279 |        ideal
8 |     808 |  0.00000E+00 |  0.00000E+00 |      20 |  0.007842796 |            f
9 |     909 |  0.00000E+00 |  0.00000E+00 |      20 |  0.001397298 |            f
10 |    1010 |  0.00000E+00 |  0.00000E+00 |      20 |  0.003076413 |
  f
11 |    1111 |  0.00000E+00 |  0.00000E+00 |      20 |  0.005812082 |
  f
12 |    1212 |  0.00000E+00 |  0.00000E+00 |      20 |  0.046733100 |
  ideal
13 |    1313 |  0.00000E+00 |  0.00000E+00 |      21 |  0.007279993 |
  f
14 |    1414 |  0.00000E+00 |  0.00000E+00 |      21 |  0.003993163 |
  f
15 |    1515 |  0.00000E+00 |  0.00000E+00 |      21 |  0.005201088 |
  f
16 |    1616 |  0.00000E+00 |  0.00000E+00 |      21 |  0.001729105 |
  f
17 |    1717 |  0.00000E+00 |  0.00000E+00 |      21 |  0.000351185 |
  f
18 |    1818 |  0.00000E+00 |  0.00000E+00 |      21 |  0.002476596 |
  f
19 |    1919 |  0.00000E+00 |  0.00000E+00 |      21 |  0.005252863 |
  f
20 |    2020 |  0.00000E+00 |  0.00000E+00 |      21 |  0.003703162 |
  f
Searching in the Design Space:   4%|*                    | 1/25 [20:00<8:00:08,
  1200.34s/it]

  ==========================================================================

n_gen |  n_eval |   cv (min)   |   cv (avg)   |  n_nds  |     eps      |
  indicator

  ==========================================================================
```

```
1 |      101 |  0.00000E+00 |  0.00000E+00 |      13 |            - |            -
2 |      202 |  0.00000E+00 |  0.00000E+00 |      18 |  0.066779342 |        ideal
3 |      303 |  0.00000E+00 |  0.00000E+00 |      19 |  0.013040164 |            f
4 |      404 |  0.00000E+00 |  0.00000E+00 |      19 |  0.003240776 |        ideal
5 |      505 |  0.00000E+00 |  0.00000E+00 |      19 |  0.006436898 |            f
6 |      606 |  0.00000E+00 |  0.00000E+00 |      18 |  0.013058896 |        ideal
7 |      707 |  0.00000E+00 |  0.00000E+00 |      18 |  0.008253434 |            f
8 |      808 |  0.00000E+00 |  0.00000E+00 |      17 |  0.023029914 |        ideal
9 |      909 |  0.00000E+00 |  0.00000E+00 |      17 |  0.002872064 |            f
10 |    1010 |  0.00000E+00 |  0.00000E+00 |      18 |  0.008945702 |
  f
11 |    1111 |  0.00000E+00 |  0.00000E+00 |      18 |  0.002731946 |
  f
12 |    1212 |  0.00000E+00 |  0.00000E+00 |      18 |  0.006083088 |
  nadir
13 |    1313 |  0.00000E+00 |  0.00000E+00 |      18 |  0.006646305 |
  f
14 |    1414 |  0.00000E+00 |  0.00000E+00 |      18 |  0.004031693 |
  f
15 |    1515 |  0.00000E+00 |  0.00000E+00 |      18 |  0.002041885 |
  f
16 |    1616 |  0.00000E+00 |  0.00000E+00 |      19 |  0.004039912 |
  f
17 |    1717 |  0.00000E+00 |  0.00000E+00 |      19 |  0.001856166 |
  f
18 |    1818 |  0.00000E+00 |  0.00000E+00 |      19 |  0.004233351 |
  f
19 |    1919 |  0.00000E+00 |  0.00000E+00 |      19 |  0.004437145 |
  f
20 |    2020 |  0.00000E+00 |  0.00000E+00 |      19 |  0.028033354 |
  nadir
Searching in the Design Space:   8%|**                | 2/25 [39:41<7:35:48,
  1189.07s/it]

  ==========================================================================================

n_gen |  n_eval |   cv (min)   |   cv (avg)   |  n_nds  |     eps      |
  indicator

  ==========================================================================================

1 |      101 |  0.00000E+00 |  0.148803098 |      15 |            - |            -
2 |      202 |  0.00000E+00 |  0.00000E+00 |      17 |  0.012496447 |            f
3 |      303 |  0.00000E+00 |  0.00000E+00 |      18 |  0.017854883 |            f
4 |      404 |  0.00000E+00 |  0.00000E+00 |      18 |  0.037275050 |        ideal
```

Listing A.7: Console Output

The analysis outputs are saved in the `.csv` files, as specified in the script, alongside the pickled Python object for inspection and debugging. First, inspect the report file, which provides information about the computational time and the number of discarded sets (Listing A.8). This information is useful for tuning the number of levels for each set and the number of samples for training and set evaluation.

Listing A.8: Report Output

```
Total Number of Sets          : 256
Number of Surviving Sets      : 26

Total Surrogate Train Time    :       0.021 s
Total Exploration Time        :       0.861 s
Total Search Time             :    5358.473 s
Number of Cores Used          :          16
```

```
Train time and score of each Surrogate:
TOM     0.0211(s)       0.9998

Search time and f_evals of each Set:
0   205.0636(s)
1   206.2956(s)
4   204.2401(s)
8   203.9787(s)
16   203.4907(s)
17   206.1708(s)
20   203.5766(s)
32   204.3334(s)
33   206.6882(s)
36   207.6279(s)
48   206.3984(s)
64   203.8703(s)
65   206.5425(s)
68   206.0475(s)
80   205.4995(s)
81   208.0759(s)
84   209.0785(s)
96   206.8746(s)
112   206.2059(s)
128   206.7444(s)
129   207.0894(s)
132   207.0260(s)
144   207.7437(s)
160   205.8206(s)
192   206.0405(s)
208   207.9498(s)
```

A printout of the Pandas data frame of the exploration results is shown in Listing A.9. This data is stored in a `.csv` file.

Listing A.9: Exploration Results Dataframe

```
      set_id  is_discarded  climb_h0  climb_h1  cruise_h0  cruise_h1     P  P_TOM
      0         0             0         0         0          0          0  1.00
1.00
      1         1             0         0         0          0          1  1.00
1.00
      2         2             1         0         0          0          2  0.44
0.44
      3         3             1         0         0          0          3  0.00
0.00
      4         4             0         0         0          1          0  0.98
0.98
      ..      ...           ...       ...       ...        ...        ...   ...
...
      251     251             1         3         3          2          3  0.00
0.00
      252     252             1         3         3          3          0  0.00
0.00
      253     253             1         3         3          3          1  0.00
0.00
```

```
        254     254             1         3         3         3          2  0.00
0.00
        255     255             1         3         3         3          3  0.00
0.00

        [256 rows x 8 columns]
```

A printout of the Pandas data frame of the search results is shown in Listing A.10. This information is also stored in a `.csv` file for easy interchange with other data analysis software. Unlike the exploration data frame, each row represents a single design point here: filtering by `set_id` is necessary for recovering the Pareto front of each surviving set.

Listing A.10: Search Results Dataframe

```
        set_id  climb_h0  climb_h1  cruise_h0  cruise_h1           TOM           Mf
M_NOx
        0          0.0  0.019019   0.042392   0.176987   0.136356  18488.123494  1053.114840
6.390926
        1          0.0  0.248721   0.247232   0.248985   0.246306  19570.910964  1019.602066
6.012993
        2          0.0  0.247531   0.247817   0.228041   0.242325  19497.310632  1021.829023
6.047204
        3          0.0  0.061144   0.103122   0.119441   0.056398  18250.701614  1060.478028
6.395150
        4          0.0  0.242335   0.001371   0.010946   0.010773  17969.991130  1070.045548
6.416366
        ..         ...       ...        ...        ...        ...           ...
...       ...
        590      208.0  0.771168   0.282262   0.089347   0.093017  19435.377825  1025.854261
5.904732
        591      208.0  0.843885   0.322598   0.078514   0.163723  19780.236214  1015.011526
5.812730
        592      208.0  0.975473   0.255174   0.126832   0.076693  19750.269531  1015.474685
5.853289
        593      208.0  0.773123   0.411460   0.108772   0.104280  19701.794994  1017.944584
5.820675
        594      208.0  0.752590   0.252407   0.005897   0.001813  18873.928896  1044.221176
6.062478

        [595 rows x 8 columns]
```

The `.csv` format allows for easy interchange between the PDOPT results and other visualisation programs. On the other hand, the `pandas` and `matplotlib` Python libraries can be used for visualising the output.

Appendix B

Ethical Approval Letter

27 November 2020

Dear Mr Spinelli,

Reference: CURES/12461/2020
Title: Robust design-space exploration for complex novel systems: a set-based optimization framework for hybridelectric aircraft design.

Thank you for your application to the Cranfield University Research Ethics System (CURES).

We are pleased to inform you your CURES application, reference CURES/12461/2020 has been reviewed. You may now proceed with the research activities you have sought approval for.

If you have any queries, please contact CURES Support.

We wish you every success with your project.

Regards,

CURES Team

Lo duca e io per quel cammino ascoso
intrammo a ritornar nel chiaro mondo;
e sanza cura aver d'alcun riposo,

salimmo sù, el primo e io secondo,
tanto ch'i' vidi de le cose belle
che porta 'l ciel, per un pertugio tondo.

E quindi uscimmo a riveder le stelle.

Dante Alighieri
Inferno, Canto XXXIV