CRANFIELD UNIVERSITY

Myrtle Julia Balasani

INTERACTIVE OPTIMISATION FOR HIGH-LIFT DESIGN

Ph. D, Doctor of Philosophy

Aerospace

October 2018

Doctor of Philosophy, Aerospace Division
School of Aerospace, Transport & Manufacturing
Cranfield University, Cranfield, Bedfordshire MK43 0AL, United Kingdom

BLANK

Interactive Optimisation For High-Lift Design

Myrtle Julia Balasani

Registered Academic Term:
02 March, 2015 – 30 June, 2017 (FT) | 31 October, 2018 (PT)

Final Examination: 19 March, 2019

Primary Supervisor
Prof. Anthony Mark Savill

Secondary Supervisor
Dr. Timoleon Kipouros

Independent Academic Representative/ Chairperson
Dr. Al Savvaris

Evaluation Board

Internal Examiner: Dr. Joao Teixeira
External Examiner: Dr. Jon Summers
Chair of Viva: Prof. Keith Mason

This thesis is submitted in fulfilment of the requirements for the degree of Doctor of Philosophy, Ph. D

BLANK

Thanks

For everything comes from him and exists by his power and is intended for his glory. All glory and thanks to him forever! Amen.

To my mother, who endured and my father, for giving me wings; for the things they gave and gave up for us. They encouraged adventure; though unwilling at times and often unintentional, ended up raising brave girls. My grand-parents who have been the backbone, my sisters who give so much love and distress, my extended family, friends & foes, well-wishers, both old and those I've met along this journey, thank you.

I was back to full-time studies after working for a few years in between and the experience was a mixture of delight and discomfort. It has been an unusual, out-of-the-ordinary life journey of many firsts, alternating physically, mentally and emotionally between countries and cultures. I've met several people, experienced new places, learning and unlearning several things along the way. The best education I received as part of this research period was a two day technical workshop undertaken by the then 83 year old David Cooke. Returning to job with about eight months left to finish research lead to mismanaged events.

This document is the result of a project which started with a lot of ambiguity and blurred direction, a challenge of sorts, as the combination of subject matter, context and approach was uncharted to some extent. However, there was a gradual growth in the industry's interest by the time this research concluded, paving way for new understanding and projects. Each new day brings with it new thoughts on editing and improving this work and report, but I have to wind it up here.

Thanks to supervisors who made this research opportunity possible. I also acknowledge the assistance offered by extended supervisory panel, examination board members, my fellow researchers, post-docs, IT & administration teams at Cranfield university and Airbus UK. Support, suggestions, assistance offered and not, comments, critiquing, tensions, both good and bad have shaped this phase of time and work.

And to the girls who take the roads less travelled and persevere, either by choice or none, here's to you!

To faith, hope, love, family, friends, truth, laughter, health, wealth, joy, peace, life, and happiness!

Cranfield | Bristol

Wings & Wisdom:

‘‘He gives power to the weak and strength to the weary; those who trust in God will renew their strength; they will soar on wings like eagles.’’

- 40.31, Book of Isaiah

“Men know how to mine silver and refine gold, to dig iron from the earth and melt copper from stone. Men know how to put light into darkness so that a mine shaft can be sunk into the earth, and the earth searched and its deep secrets explored. Into the black rock, shadowed by death, men descend on ropes, swinging back and forth.

“Men know how to obtain food from the surface of the earth, while underneath there is fire.

“They know how to find sapphires and gold dust— treasures that no bird of prey can see, no eagle’s eye observe— for they are deep within the mines. No wild animal has ever walked upon those treasures; no lion has set his paw there. Men know how to tear apart flinty rocks and how to overturn the roots of mountains. They drill tunnels in the rocks and lay bare precious stones. They dam up streams of water and pan the gold.

“But though men can do all these things, they don’t know where to find wisdom and understanding. They not only don’t know how to get it, but, in fact, it is not to be found among the living.

“‘It’s not here,’ the oceans say; and the seas reply, ‘nor is it here.’

“It cannot be bought for gold or silver, nor for all the gold of Ophir or precious onyx stones or sapphires. Wisdom is far more valuable than gold and glass. It cannot be bought for jewels mounted in fine gold. Coral or crystal is worthless in trying to get it; its price is far above rubies. Topaz from Ethiopia cannot purchase it, nor even the purest gold.

“Then where can we get it? Where can it be found? For it is hid from the eyes of all mankind; even the sharp-eyed birds in the sky cannot discover it.

“But destruction and death speak of knowing something about it! And God surely knows where it is to be found, for he looks throughout the whole earth, under all the heavens. He makes the winds blow and sets the boundaries of the oceans. He makes the laws of the rain and a path for the lightning. He knows where wisdom is and declares it to all who will listen. He established it and examined it thoroughly. And this is what he says to all mankind: “Look, to fear the Lord is true wisdom; to forsake evil is real understanding.”

- Chapter 28, Book of Job, ~6 BC

- dedicated to daddy -
I owe it to my father, earthly & heavenly

BLANK

Abstract

Interactivity always involves two entities; one of them by default is a human user. The specialised subject of human factors is introduced in the context of computational aerodynamics and optimisation, specifically a high-lift aerofoil. The trial and error nature of a design process hinges on designer's knowledge, skill and intuition. A basic, important assumption of a man-machine system is that in solving a problem, there are some steps in which the computer has an advantageous edge while in other steps a human has dominance. Computational technologies are now an indispensable part of aerospace technology; algorithms involving significant user interaction, either during the process of generating solutions or as a component of post-optimisation evaluation where human decision making is involved are increasingly becoming popular, multi-objective particle swarm is one such optimiser.

Several design optimisation problems in engineering are by nature multi-objective; the interest of a designer lies in simultaneous optimisation against two or more objectives which are usually in conflict. Interactive optimisation allows the designer to understand trade-offs between various objectives, and is generally used as a tool for decision making. The solution to a multi-objective problem, one where betterment in one objective occurs over the deterioration of at least one other objective is called a Pareto set. There are multiple solutions to a problem and multiple betterment ideas to an already existing design. The final responsibility of identifying an optimal solution or idea rests on the design engineers and decision making is done based on quantitative metrics, displayed as numbers or graphs. However, visualisation, ergonomics and human factors influence and impact this decision making process.

A visual, graphical depiction of the Pareto front is oftentimes used as a design aid tool for purposes of decision making with chances of errors and fallacies fundamentally existing in engineering design. An effective visualisation tool benefits complex engineering analyses by providing the decision-maker with a good imagery of the most important information. Two high-lift aerofoil data-sets have been used as test-case examples; a multi-element solver, an optimiser based on swarm intelligence technique, and visual techniques which include parallel co-ordinates, heat map, scatter plot, self-organising map and radial coordinate visualisation comprise the module. Factors that affect optima and various evaluation criteria have been studied in light of the human user.

This research enquires into interactive optimisation by adapting three interactive approaches: information trade-off, reference point and classification, and investigates selected visualisation techniques which act as chief aids in the context of high-lift design trade studies. Human-in-the-loop engineering, man-machine interaction & interface along with influencing factors, reliability, validation and verification in the presence of design uncertainty are considered. The research structure, choice of

optimiser and visual aids adapted in this work are influenced by and streamlined to fit with the parallel on-going development work on Airbus' Python based tool.

Results, analysis, together with literature survey are presented in this report. The words human, user, engineer, aerodynamicist, designer, analyst and decision-maker/ DM are synonymous, and are used interchangeably in this research.

In a virtual engineering setting, for an efficient interactive optimisation task, a suitable visualisation tool is a crucial prerequisite. Various optimisation design tools & methods are most useful when combined with a human engineer's insight is the underlying premise of this work; questions such as why, what, how might help aid aeronautical technical innovation.

NOTE:

This report is written in British English. Wherever a possibility of variation exists in style and terminology, the British version has been used and the system of measurement applied is metric.

KEYWORDS:

Interactive Optimisation; Multi-Objective; High-lift Design; Human-in-the-loop; Human Factors; Decision-Maker; Digital Human; Particle Swarm; Modelling & Simulation; Transonic Wing Design; Visual Analytics; Interactive Visualisation; Virtual Product Engineering

This work was carried out as part of Enhanced Fidelity Transonic (EFT) Wing Research Project, funded by Innovate UK, EPSRC council and ATI. In cooperation with participating universities and organisations, it is led by Airbus as chief participator in the United Kingdom.

Contents

Chapter 4: VISUALISATION OF SOLUTIONS

Chapter 5: OPTIMISATION TRADE STUDY

Chapter 6: ASSESSMENT & SCOPE

Figures, Tables

Abbreviations

AGARD	Advisory Group for Aerospace Research and Development
AOA	Angle of Attack
EFT	Enhanced Fidelity Transonic
ATI	Aerospace Technology Institute
CFD	Computational Fluid Dynamics
DM	Decision-Maker
DSS	Decision Support System
EFT	Enhanced Fidelity Transonic
EPSRC	Engineering and Physical Research Council
FT	Full-Time
GDF	Geoffrion-Dyer-Feinberg
GRIST	GRadient based Interactive Step Trade-off
HCI	Human-Computer Interface/ Interaction
HITLS	Human-In-The-Loop-System
HLS	High-Lift System
HTML	HyperText Markup Language
I-MOPSO	Interactive-MOPSO
ISWT	Interactive Surrogate Worth Trade-off
MCDM	Multiple Criteria Decision Making
MCOP	Multi-Objective Combitorial Optimisation Problem
MDMV	Multi-Dimensional Multi-Variate Visualisation
MDO	Multi-Dimensional Optimisation
MIT	Massachusetts Institute of Technology
MOO	Multi-Objective Optimisation

MOPSO	Multi-Objective Particle Swarm Optimisation (Optimiser)
NASA	National Aeronautics and Space Administration
OS	Operating System
POD	Proper Orthogonal Decomposition
PSO	Particle Swarm Optimisation (Optimiser)
PT	Part-Time
RadViz	Radial Coordinate Visualisation
RAE	Royal Academy of Engineering
SFC	Specific Fuel Consumption
SHEL	Software- Hardware- Environment- Liveware
SOM	Self-Organising Map
SoS	System of Systems
SPOT	Sequential Proxy Optimization Technique
STEM	STEp Method
STOM	Satisfying Trade Off Method
TAM	Technology Acceptance Model
UK	United Kingdom
USA	United States of America
WISDOM©	Wide Ranging Industrial Shape Design and Optimisation Module

----

Symbols

	Greek	**S**	Parameter Space / Standard Deviation
α	Angle of Attack	**T**	Temperature
∞	Infinity	**U**	Total Velocity
δ	Displacement Thickness	**W**	Inertia Weight
η	Transverse Shear Layer Coordinate	**X**	Horizontal Translation
μ	Dynamic Viscosity	**Y**	Vertical Translation
ρ	Density		
Θ	Rotation Angle (deg)	**c**	Chord of the aerofoil
	Latin	f	Objective Space/ function
A	Area	max_{par}	Maximum Parameters
C_1, C_2	Personal, Global Weights	min_{par}	Minimum Parameters
C_D	Drag Coefficient	**n**	Normal vector
C_L	Lift Coefficient	**p**	Pressure
C_p	Specific heat at constant pressure	par_{new}	New Parameters
D	Drag	par_{old}	Normalised/Old Parameters
L	Lift	**q**	Velocity
M	Mach Number	**t**	Time
N	Total Number of	**u**	X-component of velocity
O_x	Objective x	**v**	Y-component of velocity
P_x	Parameter x	ν	Kinematic Viscosity
Re	Reynolds Number		

BLANK

1

Introduction

1.1 RESEARCH OVERVIEW

Wings of an aeroplane are vital components of a fixed wing aircraft. Analysis and design of aeroplane wings is one of the important applications of the science of aerodynamics. A wing is the primary source of lift and also a dominant contributor to drag and hence any endeavour to increase aerodynamic performance is of significant interest. A good wing design provides lift in an efficient way as far as possible and high-lift devices are usually used. A wing's high-lift aerodynamics is actively influenced by the requirements and repercussions of other domains, often encompassing multiple subject areas and design philosophies A final aerofoil or wing solution is therefore a best achievable compromise, balancing multiple competing criteria by identifying a most preferred alternative leading to an iterative design process.

Optimisation is a necessary aerofoil design activity and modern design methods draw on extensive computing facilities and innovative computational design strategies. A computer is usually used to work out thousands of repetitive calculations involved, the outcome is an extensive list of numbers, not equations, which in turn drive an animated simulation or visual representation.

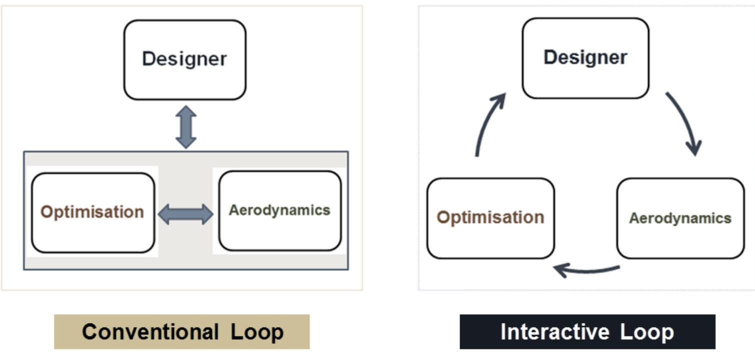


Figure 1.1: Aerodynamic Design Optimisation: Conventional Vs Interactive Loop

Interactive optimisation and visualisation aim at turning efficient optimisation methods into effective decision tools. This involves improving procedure efficiency, refining the model or input values that have been chosen for a particular optimisation problem, analysing, interrogating and navigating through datasets for betterment of results. An interactive optimisation technique is handicapped without an assisting interactive visual technique.

In interactive optimisation, the user of the optimisation system is actively involved in the optimisation process and can change or influence the results or performance (Figure 1.1). The human user is the decision-maker, playing the most valuable role in exploiting optimisation system by interaction.

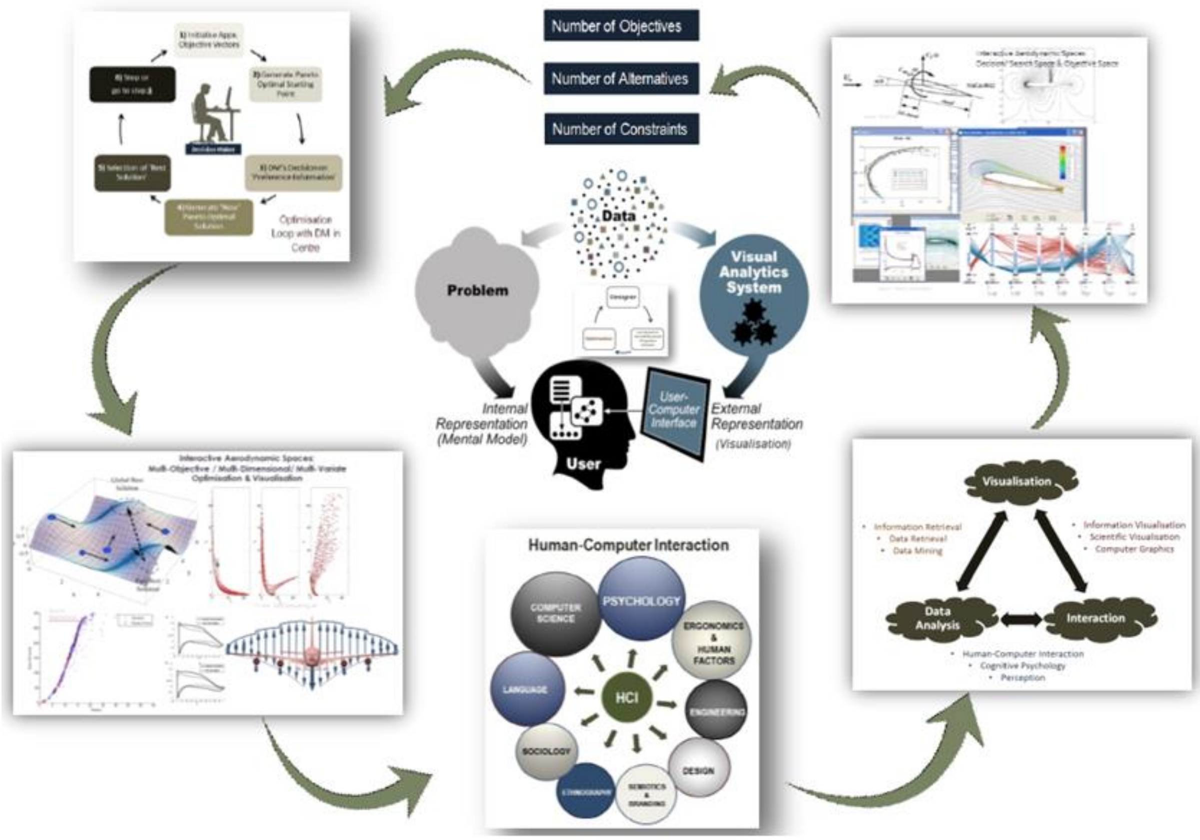


Figure 1.2: Research summarised in a picture showing interactive optimisation and the role of decision-maker in the design analysis cycle. The picture was part of various poster presentations. Sections of this comprehensive figure have been explained separately in ensuing chapters.

Algorithms where a user significantly interacts, either during the process of solution generation or as part of post-optimisation analysis where human decision making is involved are increasingly becoming popular. A vital fundamental assumption of such a man-machine system is that there are certain phases of problem solving in which the computer asserts an advantage and other phases where a human has leverage (Fisher, 1986). It is not a dispute anymore to acknowledge that man–machine interaction can be beneficial for solving complicated optimisation tasks. However, (Barthelemy et al., 2002) note that it is startling that a relatively little consideration has been given to the study of interactivity in the optimisation domain.

1.2 MOTIVATION

There is a constant demand for more innovation and use of more advanced technologies in all fields of engineering and aircraft design is no exception. Computer technologies are helping manufacturers to speed up time-to-market while minimising costs, utilising resources at an optimum level, at the same time also ensuring environmental efficiency, creating advanced, complex engineering products and systems to be competitive in terms of design, quality, performance and life-cycle value. Various software are now absolute vital parts of aircraft components and systems from design conception through to flight operations.

Aerodynamics is a major contributor for generating and creating sustainable aviation products and services, meeting the needs of global citizens and society. Computational aerodynamics is an important part of the product design and development; efficient use and working with computational aerodynamics is in a way an art, it is most times not possible to get to grips with this art without running a well validated code.

In the midst of this technology innovation is the innovator, the engineer, the decision-maker who plays a vital role either directly or indirectly. Design decisions taken by individuals, teams, organisations are key in driving the above mentioned factors, also influencing next-generation air and space vehicles. Thinking digital is no more an option but an industrial necessity. Much attention has been given to the human operator, user in an aircraft operational setting but there is a lack of consideration in a design environment where several technologies, products are engineered with a significant rise in utilisation of software.

1.3 PROBLEM

Optimisation remains a demanding task in computational aerodynamics. Ability to simulate the physics of flight has tremendously increased and is poised to continue in future, Figure 1.3 is a projection of future capabilities aspiring to reach real time simulation alongside the estimated growth in computational capacities, achieving better solutions and solving large, complex calculations. High-lift systems are a necessary design in modern fixed wing aeroplanes and these generally take into account certain factors such as run-way length, velocity at take-off, lift-over-drag ratio, and other such influencing factors while all the time trying to maintain the weight of the system as low as possible.

The major goal of the work carried out by the larger team involved in this project has been to test various optimisation strategies for aeroplane design utilising high-lift configurations with the intent to demonstrate the capability against various solvers, test-cases and visual techniques and also be knowing to extend any knowledge thus gained to three dimensional high-lift design.

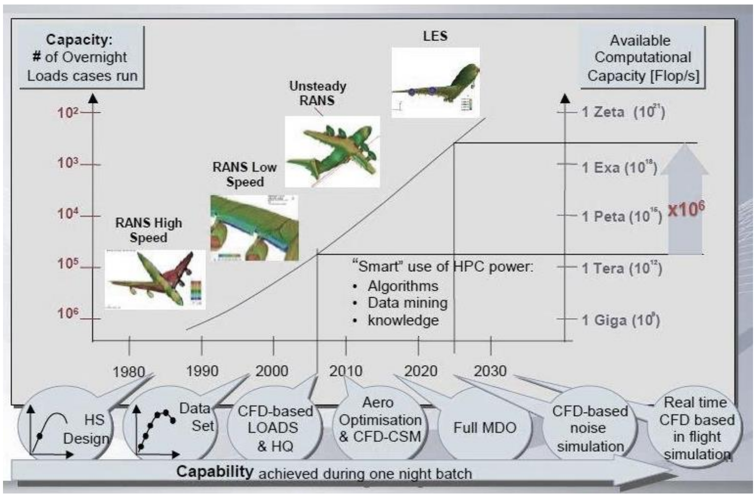


Figure 1.3: Progress in Flight Physics Simulation Capability (Source: Airbus)

Various optimisation strategies for numerical problems are very well explored and several hundred algorithms exist, which makes it difficult to test all of them so as to decide on the most appropriate strategy for a high-lift design analysis. Multi-objective particle swarm optimisation technique is used in this work against various visual techniques, it assists the user in finding a near optimal compromise among the proposed set of solutions; designer-in-the-loop engineering perspective seeks to examine human-computer interface and interaction.

Key questions are:
- How can the interactive optimisation strategy be explored to gain maximum process advantage?
- What does the emerging field of multi-objective visualisation offer high-lift design trade studies?
- Decisions, preferences, choices of the designer, amongst others are vital during an interactive optimisation trade study cycle. What is their significance?

Information visualisation analysis before, during and after optimisation makes use of computer assisted visual processing to gain an understanding. This has become a subject of serious improvement and research in the recent past, specifically for multi-objective problem tasks. Practical application of several information visual aid tools involves selecting, representing and transforming complex data in a manner that aids human interaction for exploration and comprehension. A study of such tools to help in conveying, understanding and create new knowledge between individuals and groups is a continuing and evolving effort; selected visualisation techniques are considered in this work.

In general, interactive optimisation involves human decision-maker and the computer, which is a machine. There is a lack of available knowledge in aeronautics on human interaction with computers on various virtual engineering tasks. Although high-lift design, multi-objective and interactive optimisation subjects are not entirely new areas of research, their combination and application to high-lift design trade study option and aeronautical applications remains relatively novice.

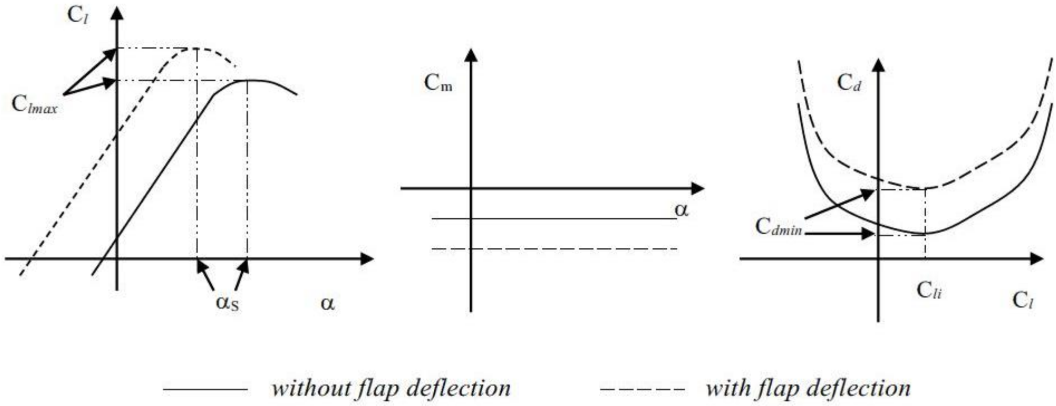


Figure 1.4: Typical Effects of High-Lift on Wing (Sadraey, 2012)

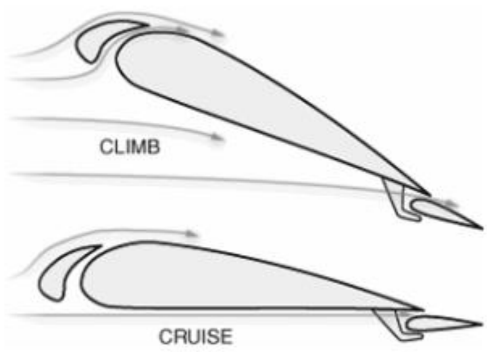


Figure 1.5: Effect of Leading Edge Wing Slat for High-Lift (Source: Zenithair)

It is now a norm to design aeroplanes for transonic air speeds which generates wave drag leading to instability. A transonic wing is designed to delay the onset of flow separation and shock waves by using a flattened upper aerofoil surface, thus allowing the supersonic flow to terminate in a weak shock. Figures 1.4, 1.5 show typical effect of using high-lift systems on wings; deflection of flaps provides more lift but also increases drag. High-lift devices are a design trade-off, a compromise between different flight requirements and presents as a suitable interactive optimisation problem task.

Some of the generic problems facing high-lift designers are summarised as follows:

- Limited & hard to manage shape optimisation
- Selecting and maintain proper mesh quality, turbulence models and optimisers
- Lack of tools for monitoring the optimisation
- Inadequate visualisation techniques
- Underusing computational tools below their maximum capabilities
- Need for design exploration skills
- Insufficient quality of currently available visual aids
- Manage large scale calculations
- Improper decision support systems to aid the designer
- Lack of knowledge on the impact of designers in design analysis and decision making

1.4 AIM & OBJECTIVE

This work is part of a wider research project called Enhanced Fidelity Transonic (EFT) Wing and the comprehensive objective of this parent project significantly aims to strengthen the performance assessment fidelity of transonic wings, reducing risk and ambiguity in the aeroplane design process, thus enabling aeroplanes and their design to be driven to higher performance standards.

Top-level challenges of EFT project deal with the following:

- Determining and improving wing's maximum lift
- Determining transonic drag rise characteristics
- Knowledge of wing shape in all performance conditions
- Accuracy improvement in wing's aero-elasticity assessment
- Knowledge and use of better tools and techniques for design optimisation

This research project aims at studying human factors in aerodynamics by implementing interactive optimisation with visualisation as a chief aid in the context of high-lift design and resulting analysis. It supports the work on advancing various tools and methods. Human-in-the-loop engineering, man-machine interaction & interface along with influencing factors, reliability, validation and verification in the presence of design uncertainty are considered. This work assumes a generic approach to present civil aeroplane and its wing design although many elements of this research could equally be applied and valid for other aircraft designs and optimisations.

An abstract of secondary level aims and objectives pertaining to this research work are as follows:

- Improve on the existing methods and tools used in aerodynamic design analysis and optimisation
- Minimise tool errors and improve working efficiency, ease-of-use and robustness
- Study of designer interfaces, tools and machine analysis
- Interactive analysis of tasks

- Improve design feedback module of which visualisation plays a major role
- Recognise and discern various visualisation techniques and their working in order to better steer the overall design process
- Understand the role of decision-maker in the process
- Drive continuous optimisation

1.5 METHODOLOGY

Computer science has been using particle swarm optimisation and implementing it for several industrial purposes but it has only recently been introduced for aerofoil parameterisation analysis. Aeronautical industry has been traditionally using several off-the-shelf simulation and analysis tools which are now being upgraded to take advantage of advanced computation including graphical user interfaces to cater to ambitions of individual manufacturers.

Improving, building on the existing software tool, exploring man-machine dynamics, specifically in a high-lift aerofoil design analysis setting is the research's framework. A specific trade study context is considered for interactive optimisation but the approach philosophy is presented in the generic context of aeronautical innovation.

Multi-element aerofoil tests cases, Garteur and SC2-0610 have been used. They are three element aerofoils comprising of slat, main element and a flap. MSES solver was run together with Python based I-MOPSO as optimiser. This research builds on previous work which covered tests and analysis using a combination of I-MOPSO and parallel coordinate visual technique. New visual techniques to aid decision-maker are supplemented to the existing tool which was not previously available; heat map, self-organising map, radial coordinate visualisation and a combined view are added as an extension in this research work. Test runs focussed on the optimisation of two objectives: lift coefficient and drag coefficient along with the influence of various slat, flap parameters.

The decision-maker plays a vital role by controlling, expressing preferences, deciding, using visual aids as support, learning and exploring. Interactivity enables control of some of the aspects of visual representation of information be available to the human designer; changes thus made by designer are incorporated into the visual tool in a timely manner. The designer interaction with optimisation task in this work chiefly reflects a single decision-maker, and could be extended to emulate a group of decision-makers agreeing together on a set of preferences.

Several decision making rules, support systems and approaches are available to design engineers in the industry as of today, however there still lacks a simple approach. Asking why outlook in a flight physics environment could aid innovation and is investigated in various forms. A literature survey reveals the existence of several interactive techniques; three interactive optimisation approaches, namely information trade-off, reference point and classification are adapted here to understand human decision-maker and the relevant environment.

1.6 THESIS STRUCTURE

The thesis report is made up of six chapters. Chapter 1 serves as an introduction to the research work and chapter 2 introduces briefly elements of generic aerodynamics, high-lift design and aspects of simulation, making it easier even for a reader from non-core aeronautics background to understand. Chapter 3 covers interactive optimisation, its methods, interaction and interface along with the role of decision-maker and decision making. Chapter 4 delves into visualisation which is a key part of the interactive optimisation module. Design and decision spaces, various plots, their qualities and influencing factors are discussed. Chapter 5 looks into the three interactive trade study approaches with test runs and their analysis; it explains the module which includes solver and optimiser with various visual outputs. An assessment of this research work is summarised in chapter 6 along with pros, cons and further scope; this chapter also presents an assessment of the digital human in a man-machine setting and contemplates future of flight.

The study attitude adapted tries to reflect the current stage of aeronautical industry with an attempt to contribute to knowledge towards the industry's present-day technological and digital transformation. Effort has been put such as to read the individual chapters both as separate sections in themselves and also as part of the larger entity.

1.7 ACKNOWLEDGEMENT

Aerofoil data was provided by Airbus and A.S.Rao which also find their use in several other research projects. Thanks to T.Tusar for answering visualisation queries and K. Jake for assisting with coding. The early version of MOPSO code used and reflected in this report was the work of J.Hettenhausen and his team (Griffith University, Australia). It was later modified by G.Tilocca (M.Sc, Cranfield University) and subsequently updated for the work carried out in this research. MSES solver, developed under Prof.Drela was provided by MIT, USA for academic research work. Codes or libraries used as part of this work are public material holding permissions for commercial, modification, distribution, patent and private use. Figures used in this report, apart from the author's own have been extracted or adapted from the referred literature, publicly available material on the internet and Airbus documents©; effort has been made to mention the authors and sources as far as possible.

2

High-Lift Design & Simulation

2.1 INTRODUCTION

Before the availability of computers to execute several lengthy calculations in relatively short time periods, two primal methods of investigation were used by aeronautical engineers to visualise flow fields around a flight vehicle combined with engineering experience and intuition:

- Wind tunnel testing
- Flight testing

To many engineers in the industry, prediction of high-lift flows is generally a challenge of practical interest. Multiple element configurations like slats on leading edge and flaps on trailing edges can be particularly challenging for computational codes and turbulence models. Numerical optimisation is now playing a strategic role in future aeroplane design and computational tools are extensively used to predict and provide calculated estimates of aerodynamic performances of a wing, usually in cruise flight conditions (Figure 2.1).

On most aeroplane configurations, high-lift systems have a considerable impact on size, costs and safety. The complex combination of flow physics, structures and systems has caused a lengthy and demanding experimental development process (Van Dam, 2002). However, because of advancement in computer hardware and software in recent times, engineering design has evolved significantly, and with that, the design of multi-element high-lift systems has only gained more importance.

Aerofoil design faces a burgeoning challenge to enhance reliability of aerodynamic predictions. Combinations of small, continuous refinements drive the high performance constancy. In seeking out step changes, the success and realisation of a design also depends on mitigating complicated aerodynamic risks. A wing's maximum lift (Cl_{max}) is a basic characteristic in determining an aeroplane's performance in terms of speed and efficiency. A betterment of the Cl_{max} uncertainty estimation to ± 5 calls for a revision in structural understanding and modelling approaches. The transonic drag rise characteristics of present-day wings also demands extremely correct predictions and are also vital to rightly capturing basic, underlying design trade-offs.

Figure 2.1: Computational Analysis (Source: Airbus)

Wing configurations are moving towards more efficient and flexible structures. An accurate knowledge of aerodynamics under all conditions is a necessity for the success of enhanced designs. There is also an increase in the utilisation of theoretical methods for predicting aerodynamic loads throughout an aeroplane's design envelope, thus enabling greater levels of optimisation.

2.2 AERODYNAMIC SIMULATION

There has been an enormous increase in simulation power over the past two decades in both software and hardware. Computed aerodynamic simulations either reveal flow fields around an element or its various characteristic calculations. Figure 2.2 is one such computer generated image showing the flow field analysis around a three element aerofoil section explaining the interaction between pressure and velocity fields. Along with the typical boundary layer regions near the walls, recirculation, mixing of boundary layers, wakes and secondary flows through slots are shown. Wind tunnel and simulation results are directly analogous; they represent data sets for specified flow configurations at different Mach numbers, Reynolds numbers etc. A computer programme has the advantage of size and mobility say via internet & numerous data storage devices unlike a wind tunnel, which is usually heavy and inconvenient. A source programme in a given computer's memory can be remotely accessed by people spread across the globe via various workable terminals. A software code is a readily transportable means, a 'mobile wind tunnel' (Anderson Jr. 1995) which facilitates carrying out numerical experiments.

Advantages of Simulation:

- Investigate what cannot be measured
- Reduce the need for testing
- Design Optimisation: narrow the design space
- Proactive instead of reactionary design
- Simplified geometry
- High-Speed CFD based scaling
- High fidelity aerodynamics simulation
- Reduce standard wind-tunnel testing
- Flexible & better aircraft and individual component/section representation
- Enable new solutions to aerodynamic problems
- Allows exploring areas of flight regime without risk of human or material loss
- Conditions can be analysed for which physical simulation is either very expensive or not possible

The aim of current aerodynamic simulation technology is to arrive at full scale simulation with multiple parallel analysing capabilities. However, trust and reliance on a simulation is influenced by the system's reliability and its user's awareness. Simulation codes are constructed based on various numerical algorithms that handle fluid flow problems. (Versteeg & Malalasekera 2007) state that all codes consist of three main parts:

- Pre-processing: formulation of problem and mesh construction
- Solving: Solution of discrete governing equations
- Post-processing: Analysis and visualisation of results

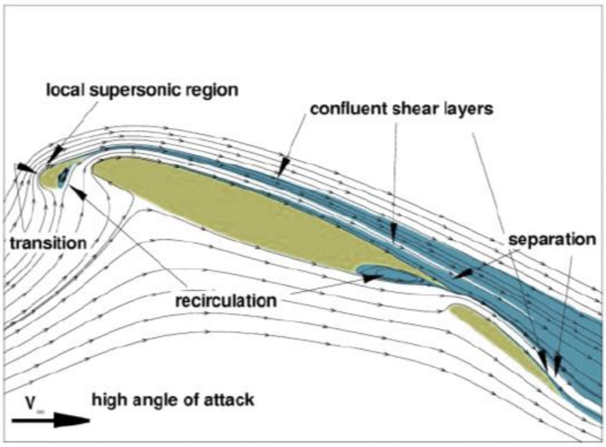


Figure 2.2: Flow field around a three-element aerofoil (Parthasarathy et al., 2015)

2.3 HIGH-LIFT SYSTEMS

An aerofoil's shape, wing area and velocity determine the amount of lift generated by a wing. A *high-lift device* is an added device mechanism which bolsters an increase in lift above that attainable from an aeroplane's classical main components. The device mechanism could be either fixed or movable which is deployed on requirement. Typical high-lift devices belong to either one of the two classes:

- Flaps
- Slats

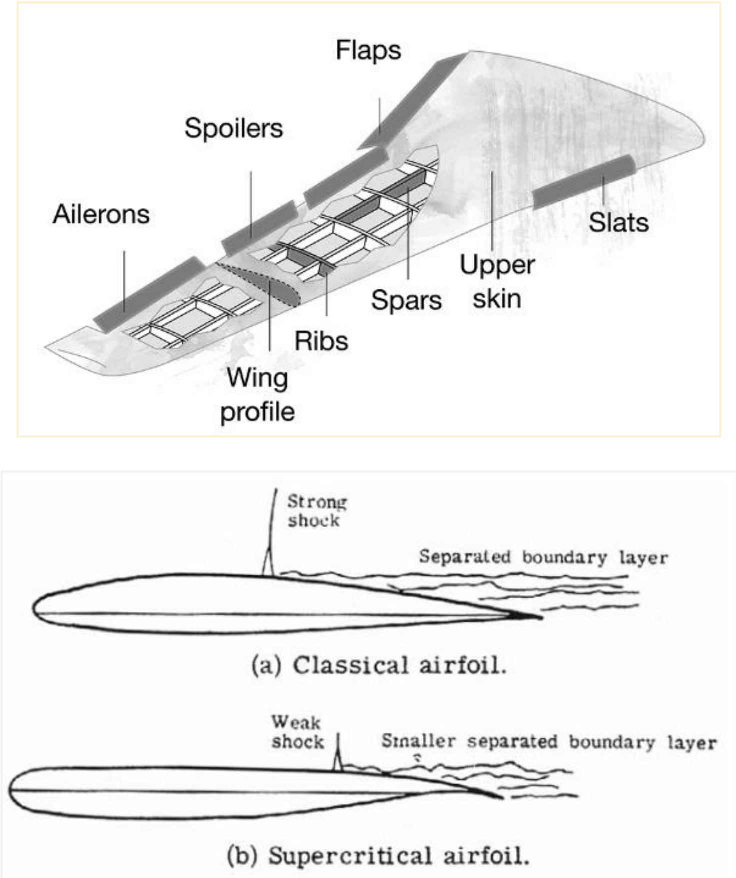


Figure 2.3: Top: Basic parts of a monoplane fixed wing design | Bottom: Difference between classical and supercritical aerofoil designs (Source: NASA)

During take-off and landing, an aeroplane's velocity is comparably low. In order to maintain lift high, aeroplane designers attempt to increase the wing's area and facilitate alteration of aerofoil shape by using movable device mechanisms on the wings' leading and trailing edges. Slat is the device on the leading edge, while that on the trailing edge is called a flap. Flaps and slats move along actuation mechanisms built inside the wings. The wing area is increased when the flaps are moved toward the tail (aft) and the slats moved forward. When the slat's leading edge and flap's trailing edge are pivoted downwards, the aerofoil's effective camber is increased, in turn increasing the lift. Also, the aft projected large area of the flap gives rise to an increase in the aircraft's drag, thus causing the aircraft to slow down for landing. A leading edge component like a slat increases the stall angle of attack and a trailing edge component such as a flap creates an upward shift in the lift curve.

There are different types of flaps in use and a specific choice is made depending on the speed, size and complexity of the aeroplane on which they will be utilised; the age and time period during which a particular aircraft was designed will also be considered. Plain flaps, slotted flaps and fowler flaps are commonly used; Krueger flaps are arranged on the wing leading edges of several jet aeroplanes.

In order to increase Cl_{max} , slats used on the wing's leading-edge to increase the camber of an aerofoil. Unlike a trailing edge flap, a negative pitching moment is not produced by the leading-edge slat. However, it may create a slight positive nose-up pitching moment, depending on its efficiency.

A slat is a slot that can be opened and closed. At high angles of attack, the slat moves forward and or downward, increasing the camber and area. The slot through the wing exerts high-pressure air from the underside of the wing over the top surface, delaying stall when the wing is at a high Angle of Attack (AOA). Drag does not increase much as the slot is not exposed to the airstream.

The choice of a wing's composition, such as size, lift capability is a bargain between contrasting requirements. A larger wing tends to provide more lift, reduces takeoff and landing distance, but will also increase drag during cruise flight, leading to a reduction in performance when in flight. High-lift devices aid in annulling some of these differences, allowing the use of efficient wings in flight, whilst adding lift on takeoff and drag on landing. A Supercritical aerofoil refers to an aerofoil design that has been designed in principal to delay the beginning of wave drag in transonic speed range. It is flatter on the upper side, curved on the bottom; upper trailing edge is accented with a downward curve to recover lift lost by the flat upper surface.

2.4 MODELLING A TRANSONIC WING AND ITS DESIGN PROBLEM

The high-lift wing is designed for achieving various performance requirements usually based on the cruise wing geometry. Viscous effects play a dominant role. The maximum lift of a well-designed high-lift profile is often times limited by the onset of flow separating on main wing or leading edge devices. Controlling the expansion of supersonic flow speed and its subsequent recompression is the key to transonic aerofoil design and it remains a difficult task.

Key factors of transonic aerofoils are:

- A large leading edge radius helps in expanding flow at the upper surface of leading edge, thus generating more lift.
- To maintain supersonic flow along a constant pressure area, or to slightly slow down the shock onset, the upper surface tends to be flatter than most typical aerofoil designs. By delaying the flow going into shock, a comparatively weak shock to the amount of lift generated is used to induce the flow decelerating to subsonic speed.
- Using an aft camber is another means of achieving lift without strong shocks at transonic speed. A large zero lift pitching moment is a potential drawback in using aft camber.
- Upper and lower surfaces at the trailing edge are almost parallel in order to avoid flow separation, a set thickness resulting at the trailing edge. The base drag is small at transonic speeds in comparison to profile drag reduction.

The above mentioned are necessary elements of a supercritical design, aerofoils designed for transonic speed range. Aerodynamic designers of today choose the best aspects of these elements to suit their specific applications (Mason, 2006).

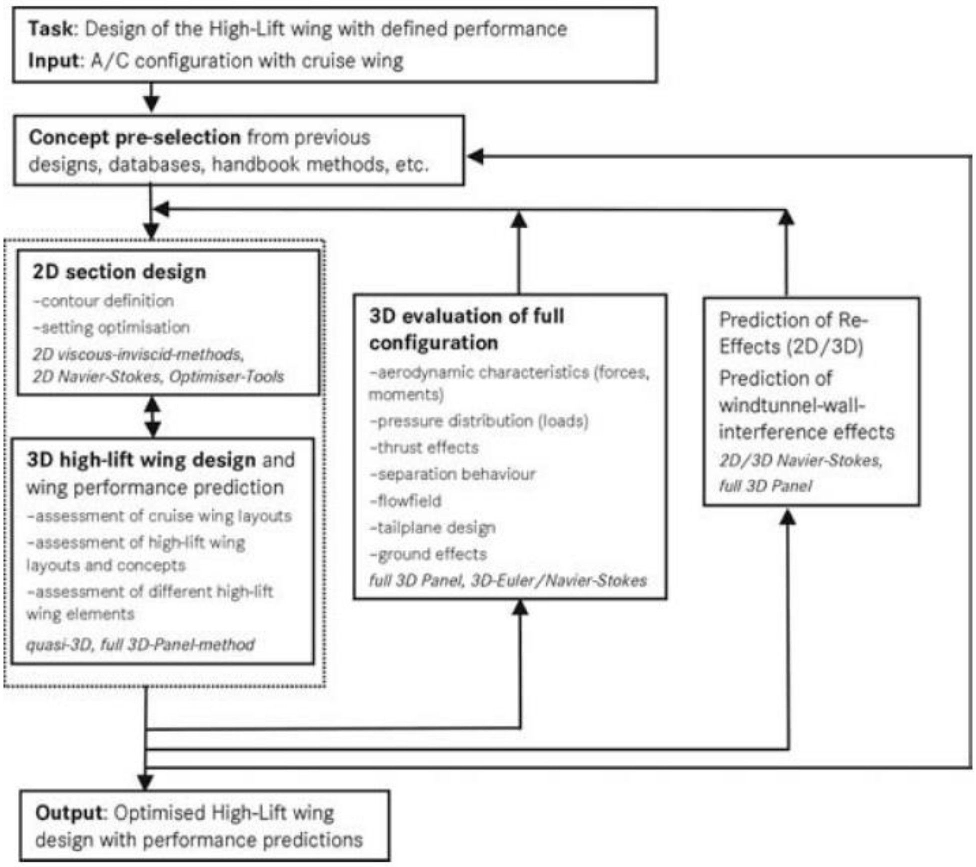


Figure 2.4: Sequence of CFD based High-Lift Design Process (Reckzeh, 2004)

Design methods are two-dimensional or three-dimensional. This work employs two-dimensional techniques. Experience shows that optimising a two-dimensional shape at intentionally chosen span-wise sections of a finite wing is adequate for three-dimensional shape design of high-lift devices, nevertheless the chosen optimisation must take into account the three-dimensional flow (Flaig & Hilbig 1991). Figure 2.4 explains the general sequence of various CFD steps in a high-lift design process and Figure 2.13 shows the high-lift design process through pre-development, development and pre-flight phases.

A blended subsonic and supersonic local flow in the same flow field, generally with free-stream Mach numbers from $M = 0.6$ or 0.7 to 1.2 gives rise to transonic flow . Generally the supersonic region of the flow gets terminated by shock waves, permitting the flow to weaken down to subsonic speeds, thus creating a complex case for both computations and wind tunnel testing. Also, there is limited analytic theory available for guidance in designing for transonic flow conditions. Importantly, the outer inviscid portion of the flow is not only governed by nonlinear flow equations, but the nonlinear flow features usually require that the viscous effects be included immediately in flow-field analysis for accurate design and testing.

$$\mathrm{Re} = \frac{\text{inertial forces}}{\text{viscous forces}} = \frac{(\text{mass})(\text{acceleration})}{(\text{dynamic viscosity})(\text{velocity}/\text{distance})(\text{area})} = \frac{(\rho L^3)(v^2/L)}{\mu(v/L)L^2} = \frac{\rho v L}{\mu} = \frac{vL}{\nu}$$

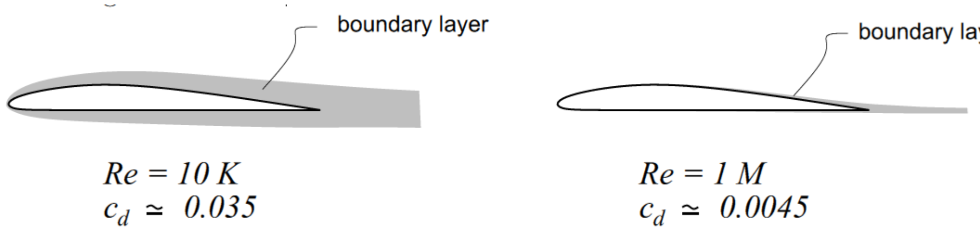


Figure 2.5: Reynolds number is an important dimensionless quantity steering the design's transition point from laminar to turbulent flow (source: unknown)

Reynolds number (Figure 2.5) is an important dimensionless quantity which helps to predict flow patterns, transitions from laminar to turbulent flows. For a constant Reynolds number with increasing Mach number, the drag remains steady until an intense increase of pressure drag occurs at Mach number close to 0.8. The critical Mach number directly depends on the thickness and lift of an aerofoil. Thickness, angle of attack, twist and camber increase the velocity on the upper surface of an aerofoil.

The important dimensionless parameters which determine the character of a given aerodynamic flow condition are the Reynolds number and Mach number; their values in any given flow condition will decide on the type of flow. Reynolds number is a measure of pressure forces relative to viscous shear forces, it helps in determining fluid behaviour patterns. Therefore, if Reynolds number increases, a flow's viscous effects become increasingly less.

2.5 HIGH-LIFT TRADE STUDY

Trade studies aid selection of the best or most balanced solution (Figure 2.6). It is a multi-disciplinary trade off activity, the viable solutions are judged by their fulfilment of a series of measures which describe the desirable attributes of a solution, including cost functions. Trade studies are vital since in practice, most information needed is uncertain, evolving, conflicting, opinionated, qualitative and quantitative at the same time.

The method by which one chooses various design variables leading to the 'best' design are many fold. All of these require that several analyses be carried out, often thousands of times. This necessitates that the model be simplified to the point that it is fast enough, but not to the point that it becomes ineffective and futile. The process is often simple when the design may be described by only a few parameters. A designer usually examines several test cases, and can typically see easily where the optimum occurs. With the increase in the number of design variables, the need for optimisation also increases. Obtaining results analytically and numerical optimisation are the two optimisation approaches commonly used.

When an analytical representation of the objective function can be made, it is sometimes possible to construct derivatives regarding design variables and produce a set of accompanying equations to be solved for the optimum. As the analysis of most aeroplane design problems involves iterating, complex computations limit the use of simple analytical approaches leading to the implementation of numerical optimisation techniques. Despite its obvious utility, (Holt, 1982) points that numerical optimisation seems to have been talked about a lot more that it has actually been used by aeronautical industry.

General objectives of a high-lift trade study are:

- Supporting decision needs of the overall high-lift system engineering
- Evaluate alternatives, such as requirements, functions and configurations
- A balanced integration of considerations like cost, performance, production, testability, compatibility, supportability etc.
- Develop and refine system concept
- Determining if any additional analysis, synthesis, or trade-off studies are required to make a design selection

In general, according to (Tabors & Steinberg, 2000), the following are to be noted in a trade study:

- Are the suggested solutions as good as possible
- What is the trade-off in order to obtain the most desirable objective
- How are bad, worse, good and better defined with regard to the design optimisation problem at hand
- Question/ challenge the assumptions; the forecast could be usually wrong
- Communication is key; any visualisation aids should make the analysis study more understandable, not more complicated
- Important variables have to be analysed, not the variables that are easy to analyse

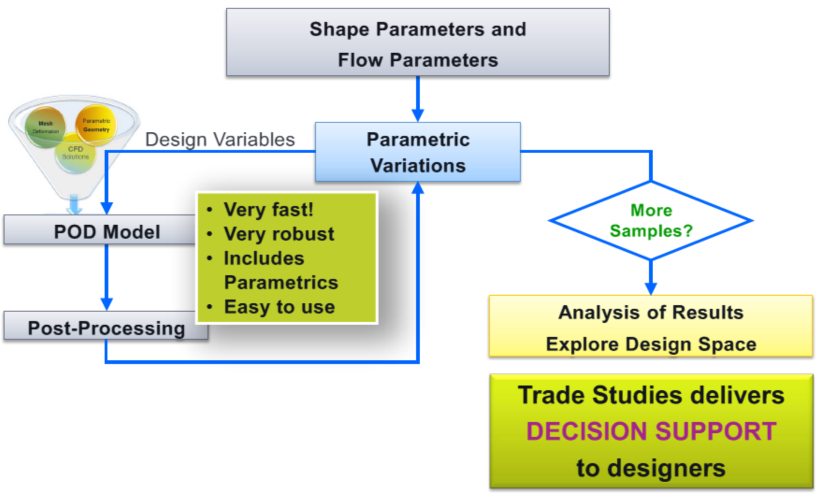


Figure 2.6: Flow chart explaining the requirement and benefit of a trade study in a design and analysis process cycle; Trade studies primarily offer decision support to designers. Mesh customisation, parametric geometry, CFD solutions are inputs to POD model. (AIRBUS, 2014)

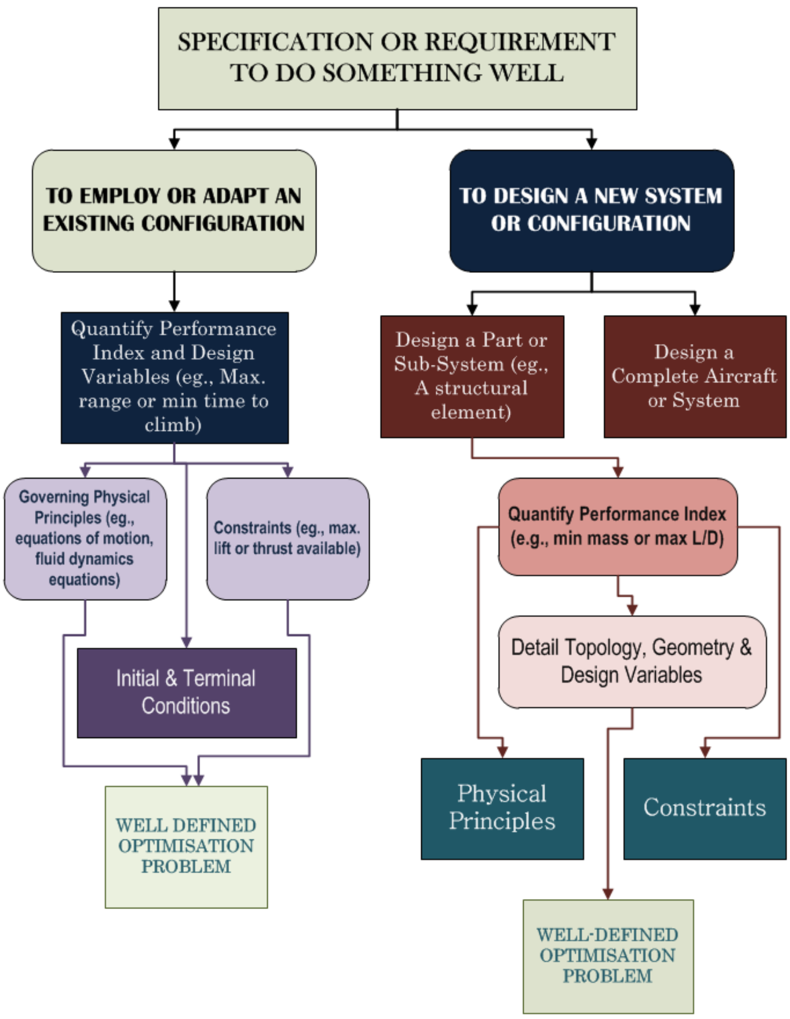


Figure 2.7: Morphology of optimisation in Aeronautics (Holt, 1982)

2.6 DESIGN OBJECTIVES & CONSTRAINTS

In aerofoil design, optimisation problem with continuous variables includes both constrained and multi-modal problems. The design process has, historically, ranged from trial, error, and natural selection, to sophisticated computer-aided design programmes and spans over various design phases (Figure 2.13).

The goal of design optimisation processes, regardless of the form taken, is to optimise design, in what is, in a sense, the best aerofoil or high-lift system design. This requires addressing of three basic questions:

- What is meant by best?
- How can a designer estimate the characteristics so that two designs or parameters can be compared in a quantitative way?
- How can one choose the design variables which yield an optimum?

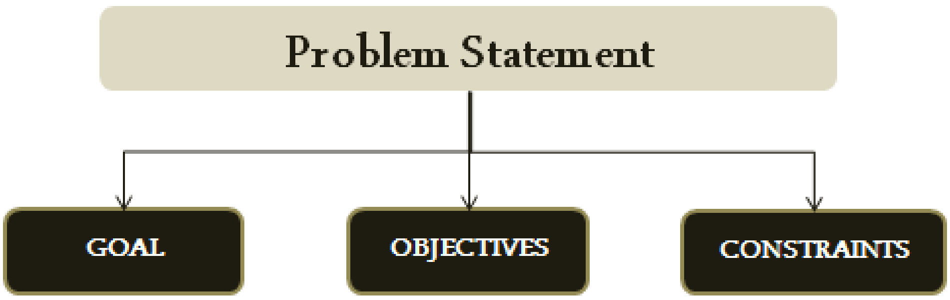


Figure 2.8: Three elements of a problem statement (Sadraey, 2012)

The first of these questions is perhaps the most important one. If a designer is not aware of what one is trying to achieve, or is set on the wrong goal, no matter how good the optimisation and analysis method might be, or how efficient the optimiser, the optimisation endeavour falls short of the original intent.

The best overall design is often times a compromise in some sense. Optimising an objective function with respect to some variables in the presence of constraints on those specific variables is constrained optimisation. The objective function is either maximising lift, minimising drag, minimising weight, or maximising direct operating costs (TKU, 1995).

The variables of a programme have much in common with numerical parameters. Parameter values are supplied by the modeller or computed from other values. When increasing the number of variables, number of evaluations increase as well. While goal functions with few variables are likely attainable, optimisations with about twenty or more variables are usually more difficult to achieve.

Classical optimisation techniques require several restart points and multiple runs anticipating that a different solution may be identified on every run, however, there is no surety. Swarm intelligence techniques provide a collection of possible solutions, which are computed iteratively. On termination of algorithm, it offers multiple good solutions instead of only the best solution. Finding and maintaining the set of these multiple solutions is where the challenge lies.

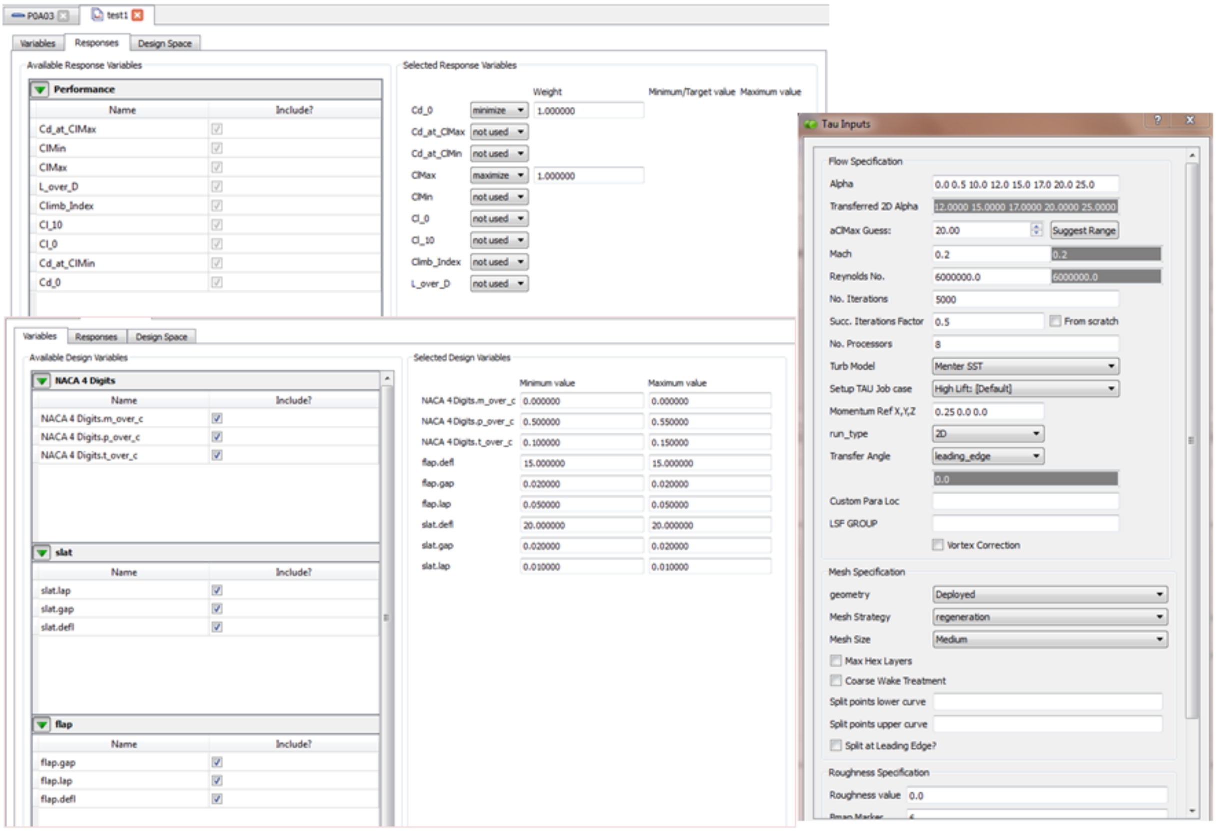


Figure 2.9: Various variables', objectives and solver settings tab in WISDOM© tool for a two-element aerofoil design analysis (source: Airbus)

When structural or cost constraints exist, awareness of multiple solutions to an optimisation task is notably beneficial. The best results may not always be realisable; therefore, if multiple solutions are known, another solution can be promptly implemented and still achieve the best possible system performance (Wong, et al., 2011). Weighing out various solutions also helps in analysing hidden properties or various relationships underlying an optimisation problem, also making them important sources for domain knowledge.

2.7 VISUALISING HIGH-LIFT TRADE STUDY

The aerodynamics of High-Lift System (HLS) are difficult to simulate with standard numerical codes and strongly depends on other values such as a flight's Reynolds number. High-lift system design is a function of several, often conflicting objectives with no one unique optimal solution but instead a set of potential solutions, called Pareto solutions, defined by the fact that one objective cannot be improved in one dimension without being worsened in another (Legriel, Le-Guernic, Cotton, & Maler, 2010). The Pareto front is the set of all Pareto solutions representing the problem trade-offs, the possibility to sample this set in a representative manner is a very beneficial assistance in decision making.

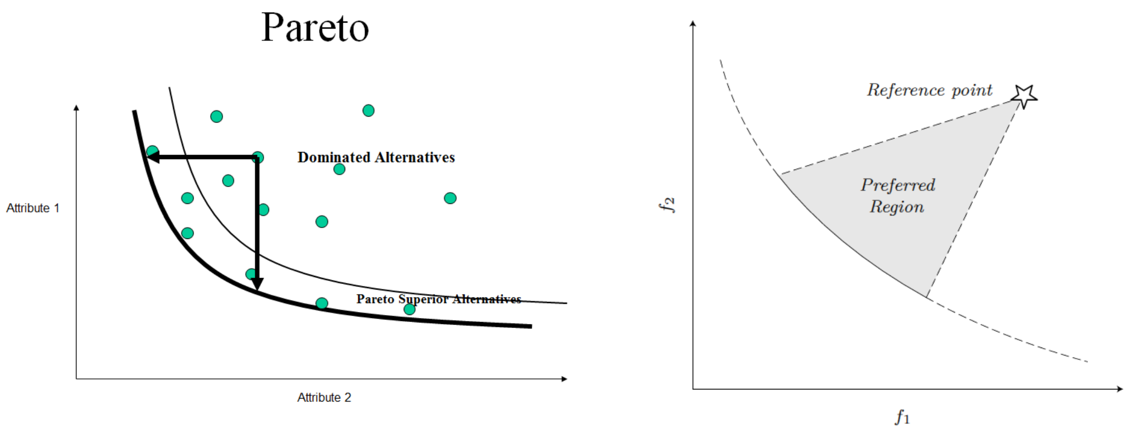


Figure 2.10: Pareto front between two attributes/ functions showing the dominated alternatives spread (left) and preferred selection region (right) (Carrese, 2012)

An appropriate visualisation tool must be able to display the location, range, shape, and distribution of calculated non-dominated solutions. Most existing, commonly used visualisation tools in many-objective optimisation fail in one or more features to show the shape of the Pareto front and allow the designer to further pursue design exploration.

A solution is treated as a Pareto optimal if no other possible solution is available, one which could better all objective functions at the same time (Figure 2.10). The set or group of all solutions satisfying this criteria is referred as the Pareto optimal set. It is made up of all non-dominating solutions; its portrayal on the domain space is called Pareto front.

Pareto fronts could be concave, convex, linear, mixed or disconnected; decision-makers can make use of a good visualisation tool to visually navigate large, multi-objective solution sets, observe the progress, visualise the relative location of a solution, evaluate trade-offs among objectives and select preferred solutions.

A suitable visualisation tool is an essential requirement for an effective interactive optimisation. Various visualisation techniques could be based on the grouping of resultant sets, such as visualising a single solution, a finite solution set, or an infinite set of solutions. Selection of a visualisation technique depends on the decision-maker's preferences.

2.7.1 Data & Information

Data is crude and unrefined, having no significance in and of itself. It can exist in any form useful or not. Data that has been given meaning by way of relational connection is called information. The meaning attributed may or may not be beneficial. The proper, applicable accumulation of information is called knowledge, such that its aim is to be practical and handy. Knowledge is deterministic, asserting a certainty in process. In aircraft design, most applications exercise some type of stored knowledge.

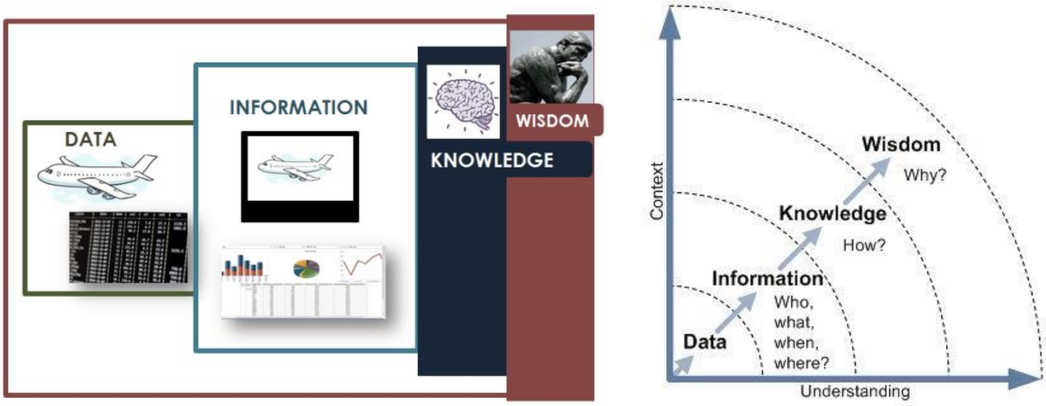


Figure 2.11: (left) Five categories of human mind: data, information, knowledge, wisdom by (Bellinger et al., 2004) (right) DIKW model for understanding (Murphy, 2016)

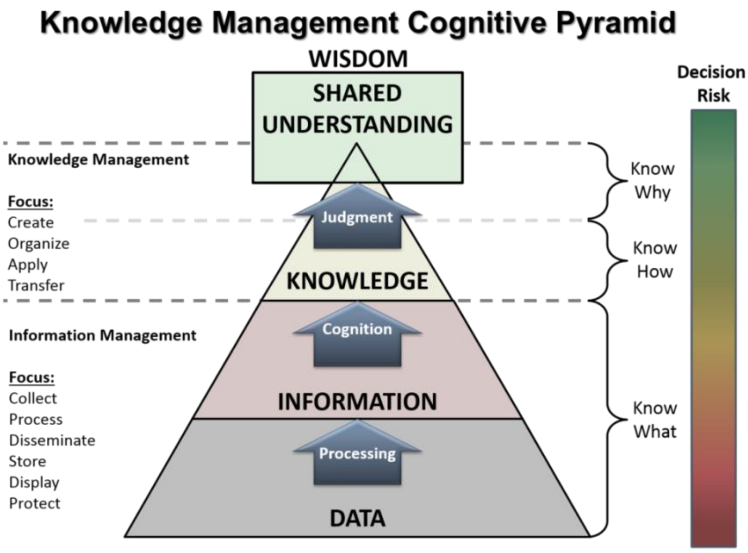


Figure 2.12: DIKW pyramid adaptation by (Viel, 2016)

Understanding is analytical and cognitive, a process by which a designer synthesises new knowledge from formerly held knowledge. Wisdom builds upon all the previous levels of consciousness and specifically upon an individual's characteristics such as morals, ethics etc. and goes beyond understanding itself. Wisdom is apparently a unique human state and not possessed by computers, a process by which one discerns, judges, chooses between right and wrong, good or bad (Bellinger, Castro, & et al., 2004).

Data stored electronically in files serves as input for an information system. Various programmes make up an information system to process or transform data, producing information as an output. The meaning of data is revealed by information when converted into visual images in the form of plots, graphs, figures and charts.

2.8 COMPLEXITY OF OPTIMISATION PROBLEMS

Optimisation has advanced by evolving towards the study and utilisation of algorithms to solve mathematical problems on computers. An optimisation task is the problem of finding a best solution from among all available possible solution options. The complexity or hardness of an optimisation problem could be described in terms of how worst the computing time grows as the problem size grows, in order to generate a result output on a given machine. Problem size is a complicated notion, but it can be roughly summarised as the number of constraints and variables in an optimisation problem, together with the cost of evaluating the objectives and constraints for a given choice of variables.

Not all optimisation problems are created equal. Some problems, such as finding a solution to a set of linear equalities or inequalities are easy and they can be solved in a feasible amount of time and memory on a computer. Others are inescapably hard and involve finding a path among a combinatorial number of choices.

Almost all practical engineering problems are non-linear with the impact of linear algebra algorithms on engineering being immense, and continues to be. Engineering modelling reveals a persistent reciprocity between what a designer would like to do (model systems as accurately as possible) and what a designer can do (analyse or design simple models) (Calafiore et al., 2014). Thus, while complex, non-linear models may be generally an attractive work challenge, when concerned with practical design analysis problems, engineers often rely on tested, simpler linear models to perform computations and find approximate solutions.

Optimisation models by nature try to fit the solution to the available data. There exists a possibility of the solution exhibiting an extreme sensitivity to changes in the problem data and this makes any effort at optimisation a risky suggestion. In aerofoil design, optimisation problem with continuous variables includes both constrained and multi-modal problems.

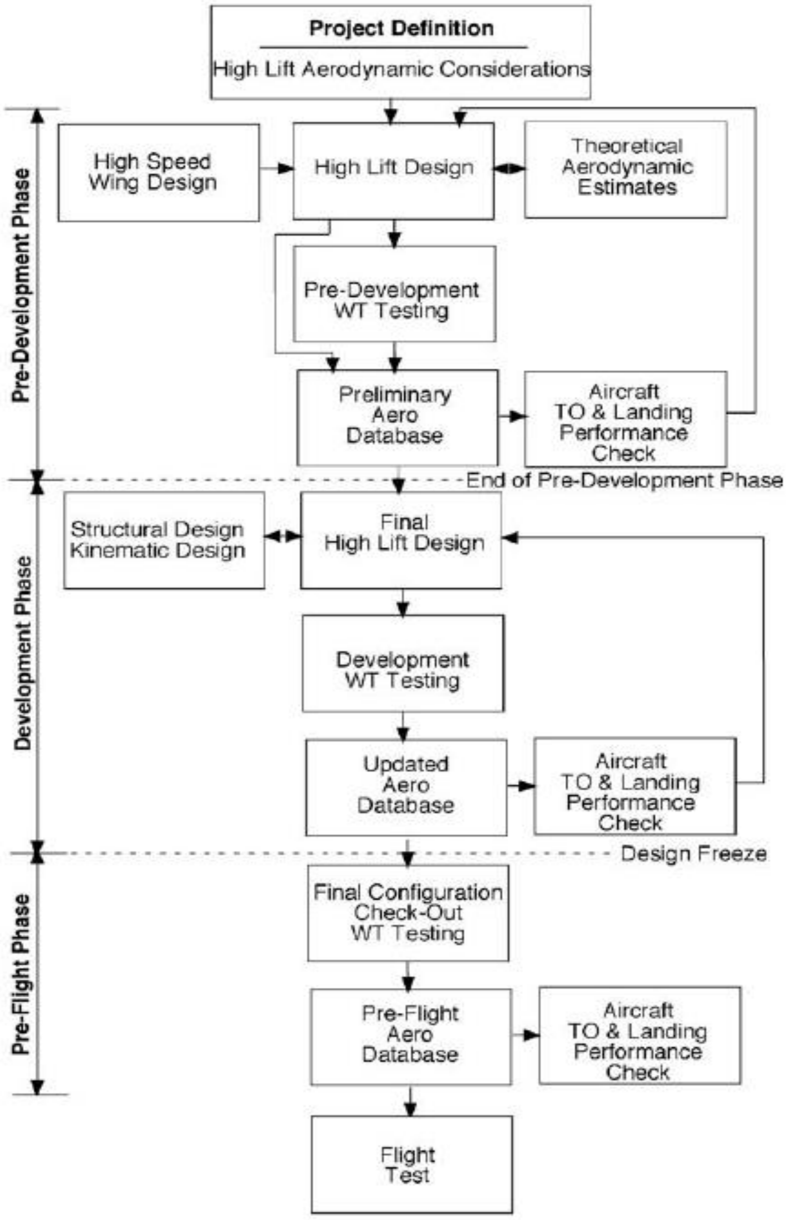


Figure 2.13: High-Lift Design Process (Van Dam, 2002)

2.9 THE HIGH-LIFT DESIGNER

Designers are an essential part of high-lift design along various development phases and are becoming increasingly digitally dependent; several designs are visualised and assessed on computers with the help of virtual tools, thus allowing concepts to be optimised at early stages of the process. The pre-development process is very repetitive and aids in designing, evaluating a broad range of configurations and to select a system that best fulfils the requirements. It is computationally intensive. The development phase refines the high-lift configuration which is the result of pre-development phase where an effective high-lift system is modified in terms of performance and costs that fits well within the final wing design strategy.

Aerodynamics is a multi-disciplinary field, dealing with two basic topics :

1. What various aerodynamic loads such as lift, drag, side force, moments are generated by a specific choice of body shape?
2. What type of aerodynamic shape will create an advantageous pressure distribution?

A lifting surface is a specific sort of structure of interest to an aerodynamicist. It is desired of a practicing engineer to be skilled in the design and analysis of such conceptions. Along with the use of a wide range of computational and physical tools, design engineers often employ numerical optimisation techniques to assist in the evaluation and comparison of new aerofoil configurations.

The most striking aspect of various processes in the industry is the fact that a high-lift designer is generally not offered much creative design space to devise and put forth an effective system (Van Dam, 2002). This limited design space, along with several design changes that continually spread along the various stages, and within a competing business market squeezes the time available for comprehensive design cycles, thus making a system design very challenging, leaving little room for creativity and exploration. Most times, the goal of the designer is to devise a high-lift system which minimises penalties while meeting the target performance requirements.

Apart from aerodynamics, high-lift system design involves application of specialised knowledge from diverse domain areas such as systems, structures, manufacturing and production, reliability and finance (AGARD-CP-515, 1993). The design and optimisation of such a system, while maintaining safety standards and minimising the direct operating costs of an aeroplane, remains a very complicated but essential task for the designer.

3

Interactive Optimisation

3.1 INTRODUCTION

‘Optimisation’ is applicable to any branch or functional area. In this research work, the use of term is directed towards high-lift aerodynamics of an aerofoil. Several industrial problem tasks containing targets, plans and decisions involve numerous objectives that are in conflict with one another; these ought to be simultaneously considered and they are broadly called Multiple Criteria Decision Making (MCDM) problems. Multi-element aerofoil design task is one such multiple criteria problem involving optimisation of conflicting objectives where the problem formulation is non-linear.

In search of a final, most preferred solution, multi-objective optimisation problems typically involve a human Decision-Maker (DM). This most preferred solution is called a Pareto optimal solution of which the DM is confident that it is the best possible option. In arriving at the best solution, a DM’s participation is necessary; she/he is likely expected to have an understanding and insight into the problem under consideration. The decision-maker should be capable of specifying related preference instructions of various objectives under consideration and steer different, alternative solutions.

A solution pattern is generated when making use of interactive methods through iterative solution algorithms; a repetition of the different steps involved occurs and preference information is progressively specified by DM during the solution process. This information is used to construct an approximate model reflecting the local preferences of DM. New solutions are generated based on this model which likely fit the DM’s preferences better. In this way, the solution process is directed by DM and sometimes only a part of the Pareto optimal solutions could be generated and assessed; selections and preferences of the DM can be corrected and defined during the solution process.

3.1.1 Non-Interactive Approach

There are three classes of non-interactive methods based on whether DM takes no part in the process leading to a solution or whether DM indicates a choice option either before or after the solution process.

- No-Preference
- A Posteriori (Search and then Decision Making)
- A Priori (Decision Making and then Search)

In solution processes where there is no DM involved are called no-preference methods. Without the availability of any additional preference information, the task is to find a solution of neutral compromise. Certain reasonable inferences are made to generate a compromise solution instead of asking the DM for their preference input. In all other categories, the involvement of DM in the solution process is presumed.

In a posteriori (Latin, literally 'from later') method, one solution is chosen by the decision-maker from among the set of possible solutions generated by the solver. To support decision-maker in choosing, DM is enabled to delve into the entire solution according to specified preferences, DM is thus better able to understand the various adjustments between the criteria.

In a priori (Latin, literally 'from earlier') method, the decision-maker is presumed to assess first-hand the importance of each objective and their influence in the overall design. The result is the transformation of a multi-objective combinatorial optimisation problem (MCOP) into a single-objective problem, which can be solved by traditional optimisation methods (Basseur et al. 2006). This sort of approach is simple and direct but the difficulty is that the DM is not necessarily aware of the several limitations or possibilities of the problem in advance and may either be too expectant or cynical (Branke et al. 2008).

3.1.2 Interactive Approach

In interactive approaches, a iterative solution pattern is generated and repeated, usually many times. After the specified iteration interval, DM is presented with some information and is required to indicate preference information, usually in a way that the chosen technique in use can utilise, such as by answering certain displayed questions or refreshing the search pattern, a mutual effort between the code and user (Figure 3.1).

This interactive way of transferring DM's preference methods is very important. The DM is able to specify and adjust choices between iteration intervals and at the same time also learn of the individual problem interdependencies, about one's own inclination and thought processing as well.

Different methods from different interactive approaches contain their inherent advantages and deficiencies, and so, availability of various approaches is required. During optimisation, the interest of an engineer lies in deciding optimal settings for various factors and the degree to which a given factor can impact the outcome in a process. Design and decision-making processes make up an interactive problem task (Figure 3.2); however, user interaction takes precedence during optimisation in expressing preferences, setting values, validating and deciding. A graphical user interface is the link between decision-maker and the optimisation system (Figure 3.3); user’s information preferences generates a preferred model, while at the same time making room for an opportunity to learn of user’s choices.

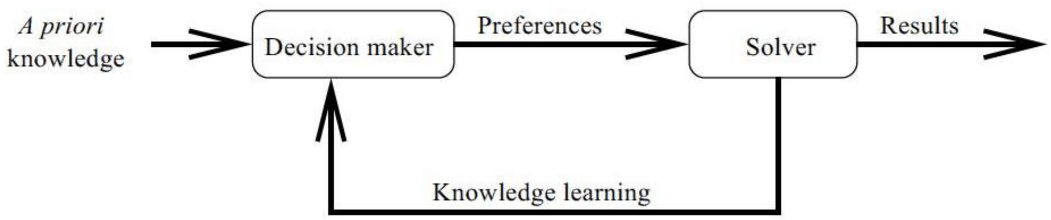


Figure 3.1: Interactive Approach: Progressive mutual effort between the solver and the decision-maker (Basseur et al. 2006)

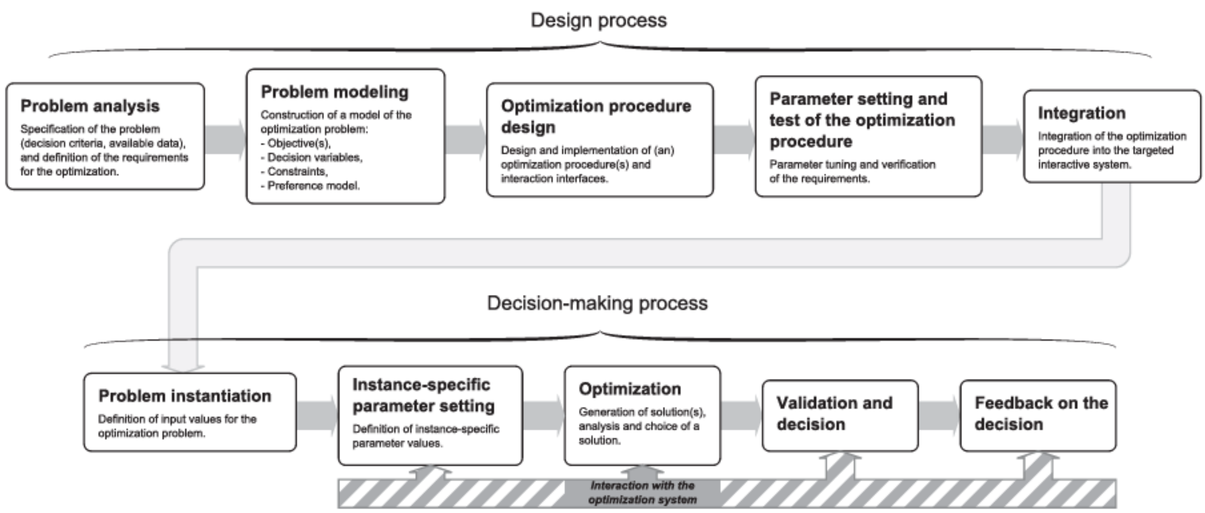


Figure 3.2: Design and decision-making processes in relation to an interactive optimisation system (Meignan et al., 2015)

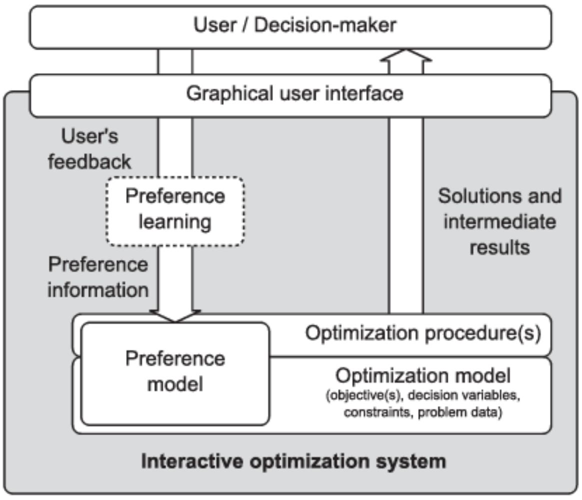


Figure 3.3: Components of an Interactive Optimisation System (Meignan et al., 2015)

3.2 OPTIMISATION MODEL

The difference between real-time optimisation problems cases and their computation models plays a vital role in interactive optimisation. A problem challenge or decision -maker's dilemma refers to the real setting of an optimisation problem for which a decision must be taken. The depiction of this context in the optimisation system is called an optimisation model. Similarly, a criterion points to the methods used to examine various alternatives of an actual problem; while the objective specifies a mathematical function for examining solutions of an optimisation model.

Generally, in order to represent a particular problem instance, both the objective function and the set of constraints are depicted with parameters that ought to be set. The problem data, generally known when a certain problem must be solved, allows the defining of such parameters. Generally, the real problem for which a decision must be made is only partially captured by the optimisation model. The differences between an optimisation model and real problem may arise from instances such as simplifying the problem case so as to make the problem computationally manageable or in situations where the modelling process poses built-in limits requiring approximations and generalisations to be made (Meignan et al., 2015).

In an optimisation problem P, a set X representing the solution space and an objective function $f : X \rightarrow R$. The objective is to identify a best solution in the set X with respect to the objective function f. Specifically, for minimisation problems, a solution $x^* \in X$ with $f(x^*) \leq f(x)$ for all $x \in X$ is to be found; for maximisation problems, a solution $x^* \in X$ with $c(x^*) \geq c(x)$ for all x is to be found.

Any solution x of P is stated by a set of values allocated to the variables influencing decisions. The likely values are restricted by a set of constraints, generally presented in the form of algebra of inequalities or mathematical logical expressions. In essence, a mathematical model of an optimisation problem, or simply optimisation model is made up of a set of decision variables, constraints and an objective function.

3.2.1 Key Components

In any optimisation system, the optimisation model and its related procedures are the basic parts that generate potential solutions to given problem cases. In interactive optimisation, both the model and its procedures are part of the interaction loops along with the user. The optimisation holds within it the description of decision variables, objectives, and constraints pertaining to the problem to be solved. When an optimisation problem is presented, the problem related data determines values of parameters of the optimisation model.

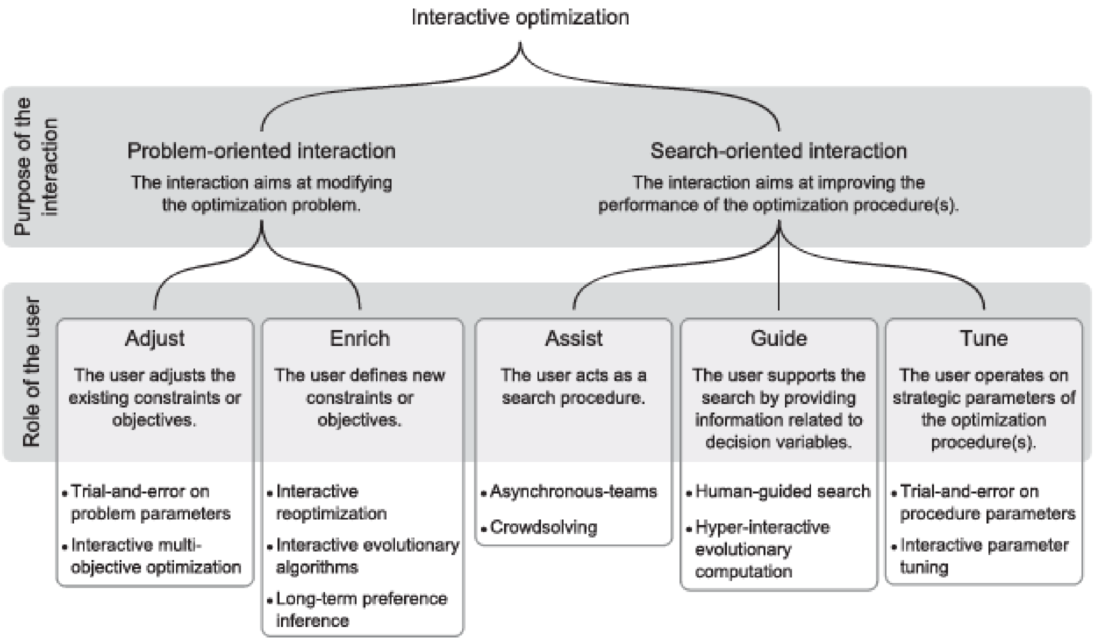


Figure 3.4: Classification of Interactive Optimisation methods according to the purpose of the interaction and the role of the user in the optimisation process (Meignan et al., 2015)

Interactive methods are classified into two depending on whether the preference model modifies the optimisation problem, or whether the preference model primarily impacts the optimisation procedures (Figure 3.4). A method is called a problem-oriented interaction when feedback loops try to alter the optimisation problem. When feedback loops try to alter the optimisation procedure, it is called a search-oriented interaction such as adjusting parameter settings of an optimisation procedure.

Although sometimes not implemented systematically, a preference-learning procedure tries to generalise user feedback so as to create a model of the user's preferences. Interactive algorithms vary in this regard; there is a learning process that explicitly generalises the user's feedback, or the user feedback is made part of the preference model like assigning a value given directly by the user to be directly assigned to a parameter of the optimisation model.

3.2.2 Choice of Interactive Technique

Several types of interactive techniques have been proposed and promoted till date; however, none of them can be considered as being better or superior to others. Certain methods are better suited to particular DMs and problems than the others. The availability and willingness of the DM to actively be part of the solution process by steering it is an essential characteristic of interactive methods.

Various styles of interaction and arrangement of technical elements differentiate various interactive methods from one another. An interaction style refers to the form in which information is presented to the DM and the fashion in which preference information is specified by DM. Arrangement of technical elements involves the kind of end solution generated, it can be weak, proper or Pareto optimal or none at all. The type of optimisation problem managed, any numerical assumptions made, the method's mathematical convergence if any and the type of a scalarising functions used also influence the solution.

DM's perception and satisfaction is often times very significant, that the DM identifies a specific technique worthwhile, acceptable and execute it correctly. While using a specific computational tool, adopting dynamic icons, animations and various sounds usually aids communication of a state of operation, thus setting up an impression of interaction and response. Interface features like fonts, color palettes, and graphical layouts also affect the perceived effectiveness of an interface; (Sharp, et.al., 2007) points to studies have shown that paying attention to visual and aesthetic features will impact the perception of a user in terms of acceptance and usability.

3.3 INTERACTIVE METHODS

A human cannot match a computer's repetitive tasks and consistency. On the other hand, a computer needs algorithmic analogy of a 'human brain' to guide solution steps. Although software engineering plays a major role in an engineering design loop; interactive design optimisation is adjusted more for satisfying the wants and wishes of most users for a chosen method or tool used in a project. The flow of the communication is primarily concentrated on the computer side of the interaction. Various Interactive methods differ from one another based on:

Style of interaction:

The manner in which information is presented to the DM, the form and type of preference information specified.

Technical elements:

The type of final solution obtained; the kind of problems handled whether based on a mathematical assumption or set on the problem, a method's mathematical convergence and the type of a scalarising function used.

Three types of interactive approaches or methods are broadly identified based on the different types of preference information. Various other interactive methods also exist but will not be covered in detail here.

- Information Trade-off Approach
- Reference Point Approach
- Classification-Based Approach

The above methods are well suited for a high-lift design trade study and are applied in their fundamental form. Choice of an interactive approach depends on the nature of optimisation problem task. Some techniques are highly specific in their search pattern and could be suitable for other types of analysis and trade studies.

3.3.1 Information Trade-Off

A trade-off is an exchange, a loss in one aspect of the problem, in order to gain additional benefit in another aspect. In multi-objective optimisation, a trade-off represents giving up in one of the objectives, which allows the improvement of another objective. Among the various trade-off based interactive techniques found in literature, two most commonly used are: *Objective & Subjective trade-off*. Both concepts are used within interactive optimisation in order to move from a Pareto optimal solution to another.

In good decision-support a system, distinction is made between the subjective part of the knowledge, concerning to user preferences referred as *preferential model* of the decision situation. The relating objective part, representing certain selected understanding regarding relevant features of decision situation which are not completely objectively selected, but devised with an objective goal is referred as the decision situation's *substantive model*. Objective trade-offs belong to substantive model, and subjective trade-offs are part of the preferential model (Branke et al. 2008).

Considering two feasible solutions x^1 and x^2, and the corresponding objective vectors $f(x^1)$ and $f(x^2)$, the ratio of change between f_i and f_j is denoted by $T_{ij}(x^1, x^2)$, where $T_{ij}(x^1, x^2) = f_i(x^1) - f_i(x^2) / f_j(x^1) - f_j(x^2)$. $T_{ij}(x^1, x^2)$ is considered a partial trade-off involving f_i and f_j between x^1 and x^2 if $f_l(x^1) = f_l(x^2)$ for all $l = 1, \ldots, k, l \neq i, j$. If there exists an index $l \in \{1, \ldots, k\} \setminus \{i, j\}$ such that $f_l(x^1) \neq f_l(x^2)$, then $T_{ij}(x^1, x^2)$ is the total trade-off including f_i and f_j between x^1 and x^2.

3.3.1.1 Objective Trade-Off

An objective trade-off scales changes in one objective with respect to changes in another one, taking into consideration the structure of the problem when alternating from one feasible solution to another.

In the method by (Zionts & Wallenius 1976), several objective trade-offs at each iteration are presented to the DM who is in return expected to respond with a like, dislike or an indifference with respect to each trade-off. In interactive surrogate worth trade-off (ISWT) method (Chankong & Haimes 2008), an elaborate input is needed from the DM for many objective trade-offs at every iteration; DM then ranks every one according to −10 to 10 scale, according to its perceived benefit (or −2 to 2 scale, as put forth by (Tarvainen 1984).

Among the objective trade-offs, the following various concepts exist.

- Total trade-off
- Partial Trade-off
- Indifference trade-off
- Indifference rated trade-off or Marginal rate of substitution

3.3.1.2 Subjective Trade-Off

When a trade-off calculates how much the DM considers desirable to sacrifice the total value of a certain objective function so that there is an improvement in another objective to a certain level, it called a *subjective trade-off*.

Three important methods are part of subjective trade-off. The GDF method by (Geoffrion et al. 1972) makes use of Frank–Wolfe algorithm to carry out a line search utilising the information of subjective trade-off specified by DM to steer the direction of search. In SPOT method (Sakawa 1982) also, DM's subjective trade-offs are used to determine the search direction, however a proxy function is instead applied to run an optimum step length. Thirdly, the GRIST method (Yang & Li 2002) utilises normal vector to steer subjective trade-offs onto a tangent plane to generate Pareto front.

3.3.2 Reference Point Approach

Using a Decision Support System (DSS) in a reference point method usually involves the following:

- Aspiration and range levels (reference points) are specified by the decision-maker for all objective functions. To assist the user in starting the process, DSS usually computes a neutral solution first, responding to reference levels of average objective function ranges.

- The DSS generates response by maximising the achievement function, an easy, nonlinear aggregate of objective functions and feasible approximation of DM's value functions. The information is present in range estimates of objective functions and in aspiration, reservation level settings.
- The DM has the freedom to alter reference points as she/he wills, using this opportunity to gain knowledge about the problem situation and to study interesting sections of Pareto optimal set.
- Different techniques could be utilised to support DM in this study exploration as long as they do not limit DM's freedom.

Human preferences have essentially a nonlinear character, including a preference for balanced solutions. Any linear approximation of preferences e.g., by a weighted sum distorts them, favouring unbalanced solutions. An aggregation of linear weighted sum is easy but could be very simplistic in depicting typical human preferences which are usually nonlinear. Employing a simple approach may generate unfavourable and unforeseen extra effects.

Another assumption of reference point approaches is that a decision is made by DM by making a comprehensive evaluation of the decision situation. In order to support DM in such evaluations, it is expected of a DSS to compute and inform the DM of relevant range values of objective functions.

A reference (aspiration, range) level or fixed points are not considered as fixed preference wishes but as a tool for adaptive and holistic learning about the decision situation. (Wierzbicki 1999) points that even if the convergence of a reference point's solution most preferred by the DM could be proved, this feature is not emphasised. Other characteristics of the approaches are considered more important. Even if DM's reference points could be concluded in some objective steps, independent of DM's preferences, the diversity of such objective determinations is vital to note as it makes comparing results optimal solutions possible.

DM can pick a Pareto optimal solution by altering reference points and maximising the achievement function as they have full controllability feature providing freedom to the DM. The basic aim is to enhance the power of intuition of DM (Wierzbicki 1997) by enabling holistic learning about the decision situation as modelled by the substantive model. The same applies when using reference point approaches for supporting negotiations and group decision making.

To help decision-maker in an overall evaluation, a DSS computes and informs the DM regarding ranges relevant to objective function values. The ranges could be characterised in different ways; the two basic ones are:

- The full range of objective functions includes defining lower bound z^{lo}_j and upper bound z^{up}_j , with respect to feasible decisions $x \in S$ $(j = 1, \ldots, k)$
- Optimal ranges of objectives are considered over Pareto optimal solutions. The lower bound is an ideal objective vector z^*_j and is generally set as z^{lo}_j. The upper bound is a nadir objective vector z^{nad}_j (a construction from the worst Pareto-optimal objective values in a multi-objective optimisation problem)

Generally, $z^{nad}_j \leq z^{up}_j$, it is simpler to decide on a nadir objective vector in case of two-objective problems. Independent of objective ranges that are utilised, it is usually beneficial to accept that all objective functions or quality indicators with their values $f_j(x)$ for decision vectors $x \in S$ are lowered down to a comparitive scale by transformation: $z^{rel}_j = f^{rel}_j(x) = (f_j(x) - z^{lo}_j) / (z^{up}_j - z^{lo}_j) \times 100\%$.

3.3.3 Classification-Based Approach

Alternating between one Pareto optimal solution to another suggests a trade-off approach. To put it differently, it is a decision to move to another Pareto optimal solution to improve the value of an objective function by worsening the value of some other objective function. This approach is partly borrowed into classification-based methods. In classification-based approaches, DM's preferences are indicated by classifying objective functions who then decides on which objective functions to improve and which to be allowed to diminish from their current values. Pareto optimal solution is presented to the DM and asked of any changes in the objective function values which might generate a more favoured solution. It was shown by (Larichev 1992) that such classification approaches allow expressing of DM's preference information in a cognitively valid way.

Several classification-based interactive multi-objective optimisation methods exist. They differ from one other based on the number of class availability, the preference information asked from DM and how this information is used to produce new optimal solutions.

When the DM classifies objective functions (say O) for a current solution, DM categorises them into a certain class. The number of classes available varies in different sub-methods. Below are some generic classifications:

- O < whose values to be improved (decreased) from current level
- O ≤ whose values should improve until a certain desired aspiration level
- O = whose values are acceptable for present solution
- O ≥ whose values could be impaired (increased) until a certain upper bound
- O* whose values are momentarily permitted to freely change

The various classification related preference levels and upper bounds are obtained from DM if needed; DM is usually expected to classify all objective functions.

3.3.3.1 Step Method

Step method (SteM) is one of the earliest introduced interactive methods for multi-objective optimisation, originally developed multi-objective linear programming problems. SteM uses the concept of moving from one weak Pareto optimal solution to another. A solution optimum could be assured and the idea of classification seems easy for DM. However, it could be complicated to estimate the amount a certain function should be relaxed so as to generate preferred improvements in another.

SteM is an iterative procedure for exploration where the best compromise is achieved after a certain number of cycles. Each cycle consists of a computation phase, a decision-making phase which includes communication of DM’s analysis. During the decision making phase, DM studies the results of computed calculations and will be able to give new information about the objectives after evaluation (Benayoun at al., 1971).

3.3.3.2 Satisfying Trade-off Method

The satisfying trade-off method (STOM) (Nakayama & Sawaragi 1984) is built on concepts similar to reference point approaches. As its name suggests, the approach focusses on arriving at a satisfying solution.

DM is supposed to categorise objective functions into three classes: those that could be improved, those that could be relaxed and those values that are satisfactory; the DM should specify preferred target levels for these functions.

The DM is only required to state desirable function levels; the upper bound for functions is calculated from trade-off rate information so that the DM is not loaded with the responsibility to specify more information. Functions are assumed to be twice continuously differentiable. The solution process goes on until the DM decides not to improve or worsen any objective function value. The DM however has the choice to alter the calculated values if they are not satisfying.

STOM approach can also be implemented even if the assumptions allowing automated trade-off are not valid. In such instances, the aspiration levels and upper bounds have to be defined by DM.

3.4 DESIGNER-IN-THE-LOOP ENGINEERING

Human-in-the-loop Systems (HITLS) include a biological system: the human being that cannot be engineered. It centres on interactions between people and devices (computers). Several computational aerodynamic software tools used by engineers for design and analysis are very good at repetitive tasks. However, when any out of the ordinary situations occur and requires actions to be taken, the devised system, for the most part cannot react accordingly. The essential aspect in interaction systems is the Human-Computer Interface (HCI).

To know the human-machine system, it’s vital to understand the manner in which human beings:

- Perceive information from system devices
- Interpret information and make decisions about their next actions
- Interact with the device, its related components, and/or its controls

It's also essential to comprehend the ways in which computer devices:

- React to user inputs
- Provide feedback to the user regarding the effects of their inputs/ actions

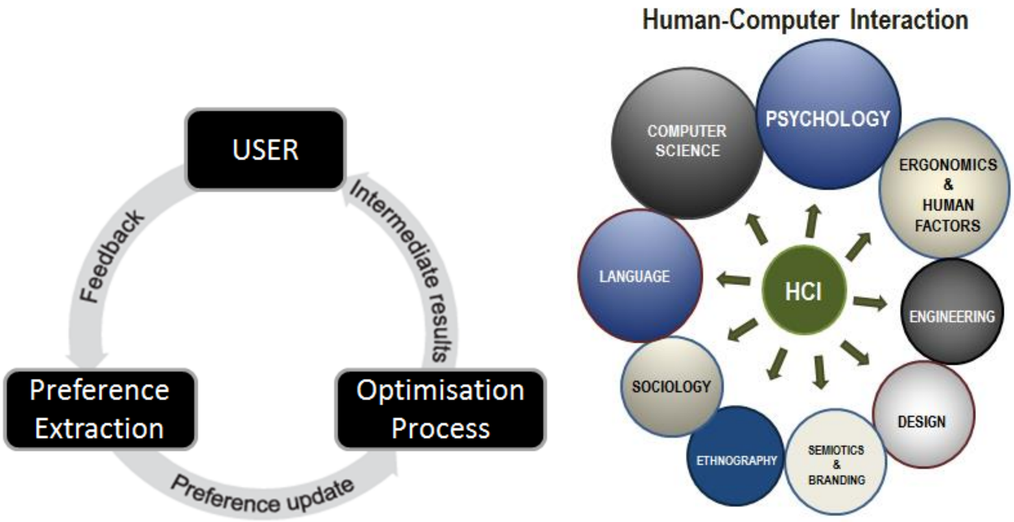


Figure 3.5: (left) Human-in-the-loop approach for optimisation (Meignan et al., 2015) (right) The field of HCI (adapted picture; source: unknown)

Human designers are one of the significant origin points of errors in any complicated system; most of a system's failures and errors could be tracked to a human connection. Many design & analysis errors are associated to badly developed human-computer interface; HCI encompasses various spheres of influences relating to both humans and computers (Figure 3.5, right). However, the human in the loop is often expected to be failure-proof in a chiefly automated system environment. Even the most professionally trained and skilled designers are subject to disinterest and weariness. Negative factors such as fatigue and stress, linked with human cognition, can impact a designer's performance severely, potentially risking the capacity to carry out tasks.

The user involved in an optimisation process is able to influence the end result or performance of optimisation (Figure 3.5, left); user's expertise of application domain is valuable in such a system. The computer must provide suitable response to the designer to support her/him in making well informed decisions depending on the most recent information. Aeronautical engineers must ensure that the aeronautical methods & tools are easy and intuitive for human users, but not so uncomplicated that it pacifies the designer into a state of complacency and lowers their alertness to the demand of being creative and innovative in their approach towards optimised engineering.

Following are some of the important HCI requirements that computational design and analysis tools & methods should take into consideration (Thakur, 2015):

- Human Computer Interaction must be robust and allow recoverability
- A good interface design should encourage the designer to carry out tasks correctly and discourage them from making errors
- Interface must be relatively uncluttered and easy to use without eliminating innovation, exploration of design development & optimisation.
- However good the underlying system functionality might be, end-user designers will always judge the system by its interface
- Elements like display colours, menu layout, label text, ease of navigation etc. are some key points which must be thought through well

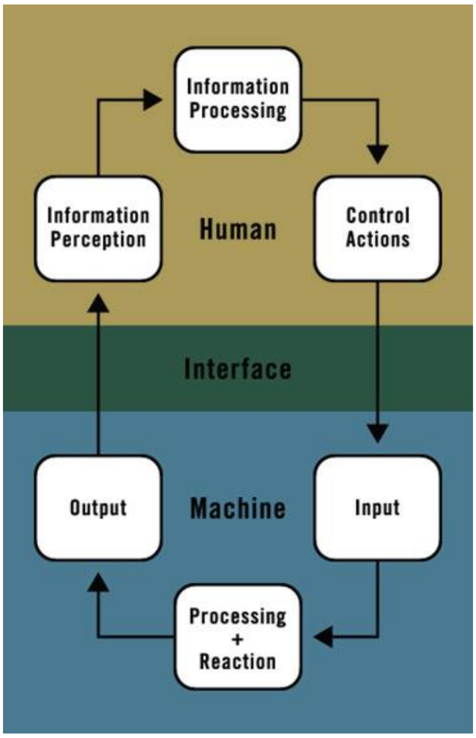


Figure 3.6: Human-Machine Interface Flowchart **(**Source: Redmill & Rajan 1997)

3.4.1 Humans in HCI

Design is a process of defining and exploring a vast space of possibilities that requires the building up of knowledge and familiarity with the constraints and trade-offs involved. The amount of data required to successfully pursue unconventional aircraft designs can rapidly become overwhelming. Humans are superior to machines at managing new occurrences, but bad at performing repetition tasks for a long time. All computational tool designers are humans, and usually, they are also the end-users.

The analyst or decision-maker, when faced with such a data overload problem, is limited in her/his ability to conduct any kind of trade-offs, test hypotheses, explore the design space, and detect unexpected trends, detail or relations, as the data sets cannot be visualised. Consequently, she/he cannot fully comprehend the problem to be solved, or understand the behaviour of the system under consideration. While unprocessed data does not hold any intrinsic value, it can result in missed opportunities for critical actions, which may, in turn, result in poor designs and significant loss of time and money.

To alleviate this problem, it is necessary to move away from static representations and visualisations and develop ways that support interaction between information, and the human cognitive and perceptual systems, while simultaneously allowing users to integrate their background, expertise, and cognitive capabilities into the analytical process. The need to address these aspects has given rise to a multidisciplinary perspective named Visual Analytics (Mavris et al. 2010).

A human's capability and limits have been categorised in various styles, and one of them is the SHEL model (Figure 3.7), specifically designed for aviation and has remained popular in human factor studies. This conceptual model describes the following components:

- **S**oftware
- **H**ardware
- **E**nvironment
- **L**ive-ware

The model centres on the human-being, or live-ware.

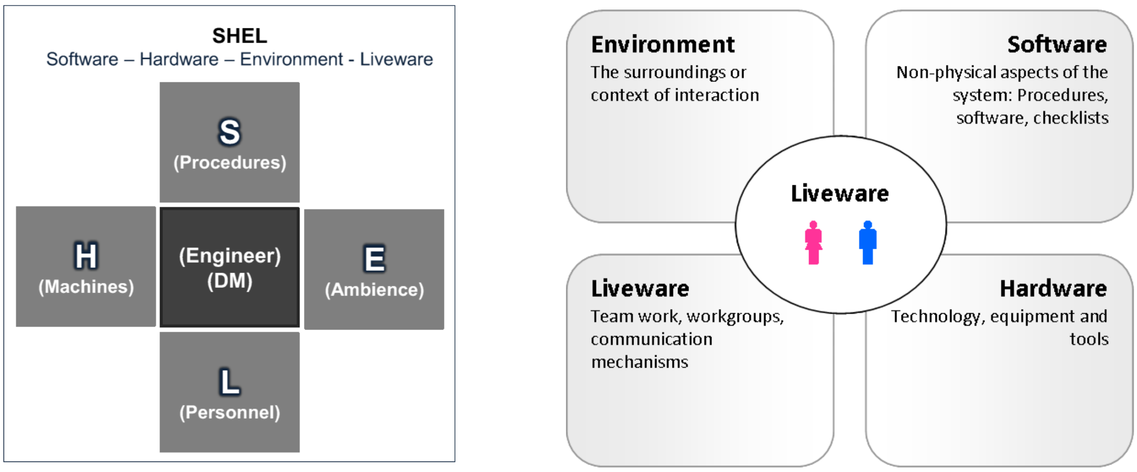


Figure 3.7: A conceptual model of human factors, SHEL

(Source: Aviation Online Magazine)

If the designer must be routinely involved in a design and analysis, she/he is prone to making errors and adapts to a usual mode of operation. Also, if the designer has a constant mental model of the system in its normal mode of operation, she/he will tend to avoid data pointing to an error unless it is displayed very prominently to catch attention. The HCI must provide sufficient novelty to maintain user alertness and keep her/ him interested in the task, but not so extremely complex that the designer will find it hard to use the computational methods and tools.

Description	Error Probability
General rate for errors involving high stress levels	0.3
Operator fails to act correctly in the first 30 minutes of an emergency situation	0.1
Operator fails to act correctly after the first few hours in a high stress situation	0.03
Error in a routine operation where care is required	0.01
Error in simple routine operation	0.001
Selection of the wrong switch (dissimilar in shape)	0.001
Human-performance limit: single operator	0.0001
Human-performance limit: team of operators performing a well designed task	0.00001

Figure 3.8: Generic human-error probability data in various operating conditions (Kirwan, 1994)

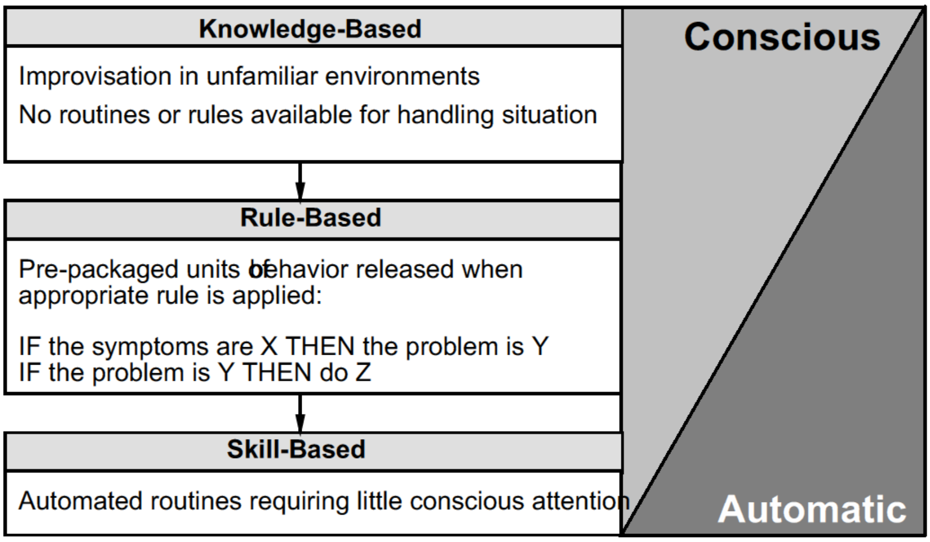


Figure 3.9: The Continuum between Conscious and Automatic Behaviour

(Reason, 1990)

Human error could be quantified for the likelihood of errors involved to determine the overall effect of human error on a system's functionality and reliability (Figure 3.8). However, this approach works when what constitutes an error is clearly identified and defined so that errors or associated risks could be reduced to an acceptable level. Human behaviour and error could be understood by examining the use of rules learned as a result of interaction or by already acquired experience (Figure 3.9); the level of conscious behaviour is intermediate between knowledge and skill based modes.

The present technology offers tremendous capacity for data transfer. The availability of bandwidth increased processing levels and data, enabling easy inter-operations between various systems. One of its chief consequences is that general users and professionals alike are being immersed in an overwhelming data and information while frequently, most times lacking real knowledge and adequate understanding (Gersh, et.al., 2005).

3.4.2 Computers in HCI

Making computers widely available to be utilised by people has always been closely related to the subject of interface. The interaction of humans with computational technology has evolved through the years and continues progress and mature with new and emerging technologies, systems rolling out almost every day.

Human-Computer Interaction/Interface (HCI) is by default linked to the evolution of computational technology, the machine. No matter how advanced a machine might be, it is often of little value unless it can be used properly by humans. A machine's functionality and usability are the basic criteria to be contemplated in the design of any HCI.

What a designed system can accomplish defines why a system is actually designed; the various functions of a system help achieve its purpose. A system's functionality is characterised by the set of actions or outputs that it provides to its users. However, its value is only appreciated when it can be efficiently utilised. A system's usability with specific functional features can be defined as the range or degree of the system's efficient and adequate use to accomplish certain goals, sometimes for certain users. A system can be called effective when its functionality and usability are well balanced (Karray et al. 2008).

Interaction with hardware using input and output (I/O) devices happens through software interfaces such as Graphical User Interface (GUI) generating displays (Figure 3.10). Software and hardware must coordinate with one another, so that the processing of user inputs is fast enough and computed outputs do not disrupt the workflow. Understanding both computers and humans is necessary to understand their interaction. At a fundamental level, when designers interact with computers, they are either transferring information to the computer, or receiving information from it. Usually, the information received is as a response to the information conveyed to the computer. Therefore, interaction is transfer of information, a two way process.

Computers are best used as logic machines; their precision, reliability, and long endurance give them an edge over human beings. Much of present day industrial work has been rationalised and broken down into precise, specific tasks and actions. Human beings tend to spend several hours carrying out repetitive or pre-determined steps, yet they will not be able to achieve precision levels of a computer which is good at following rules. However, human workers exhibit valuable traits such as intuition, judgement, deducing meanings and flexibility, reaching conclusions based on evidences and reasoning. These qualities have been difficult to be injected into computer machines (Dreyfus, Dreyfus, & Athanasiou, 2000).

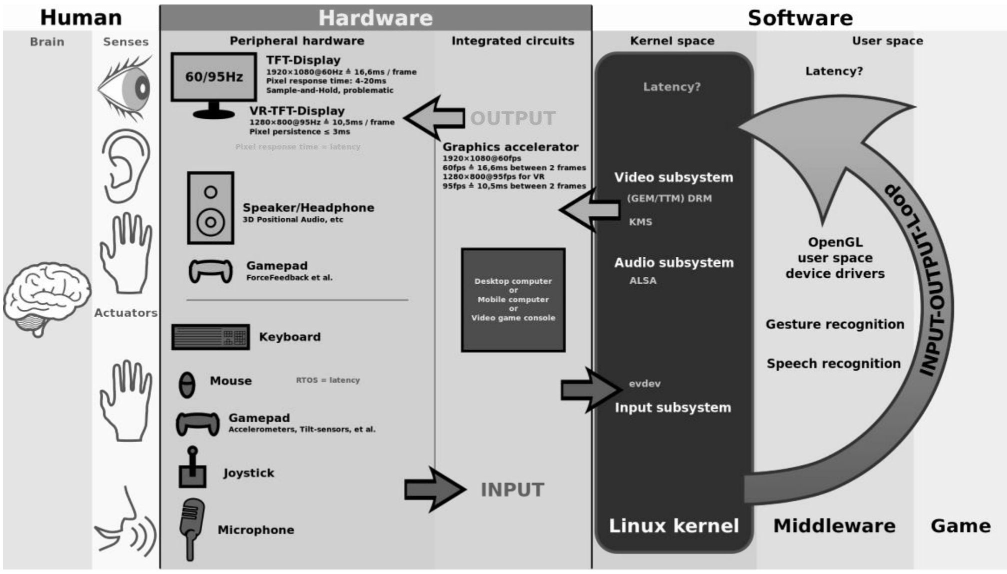


Figure 3.10: Human-Hardware-Software Interaction **(**Source: ScotXW, 2014)

Interactive computing involves input/output communication with the user during computation and interactive programming is a procedure of writing parts of a programme while it is already active. Interactive techniques are useful in aerofoil design, especially when no clear specification of the problem to be solved can be given in advance. Using software's interaction model, a designer can create interactivities in a short period of time, suited to the problem at hand.

3.4.2.1 Programming

Programming an interface is a challenging and time-consuming affair. As a result, the success of any computed solutions becomes precious to the programmer and gives rise to a risk of these results being defended and minimise any further changes or improvements. The coding process starts with an original formulation of a certain computing problem through to an executable program involving analysis, developing, understanding, generating algorithms, verification of requirements, checking correctness and implementation.

Providing the right computing tool kit helps a designer's problem solving ability. The advantage of good programming toolkits is that smaller, easily-understood components can be created in certain ways in order to create larger tool modules. When these larger tools are accepted and understood, they can be merged with other tools. The process can be continued which provides an opportunity for those with good computing skills to enhance productivity (Dix et al., 2004).

The purpose of programming for aeronautical applications is to work on a series of instructions that will automate performing problem specific tasks. The various process steps often demand a programmer to have knowledge of various subjects. An important part of solving the problem is to clearly specify the goal.

3.5 HUMAN FACTORS ENGINEERING

The motivation for human factors engineering is that the current and future aviation systems are expected to continue to depend on humans in the process loops for effective, efficient, and safe design and operations. There are several evidences showing that a failure to sufficiently include humans in the design and operation lead to inefficient systems and may be dangerous. This concern grows with the continuing growth in technology. Nevertheless, (Abbott 2001) points out that several technological advances have recognised past mistakes and have provided improvements in designs, operations and safety levels, and will continue into the future.

A new technology, a method or tool is often developed and introduced to undertake problems already known or to grant certain operational benefits. While such new introductions may solve some problems, they also usually give rise to other problems; it may bring out issues that may not have been anticipated and which may not be ignored in the larger work or industrial context. Such issues are to be dealt with specifically with respect to the human operator.

3.5.1 Interfaces

The humans have a variety of channels to gather information about the world around them. Colours, lights, sounds, movements, patterns are different types of inputs perceived by humans. This incoming information requires some attention to effectively set up and design communications between computer systems and the human operator.

Another vital design aspect is in understanding how a human user processes the information received. Inadequately designed human-machine interfaces or systems which fail to consider a human’s capabilities and limitations in terms of information processing can strongly influence system effectiveness. (Abbott 2001) states that several human errors can be traced to short- and long-term memory limitations, cognitive processing and decision-making processes.

Humans, similar to that of equipment, are designed to function effectively under certain environmental conditions and limitations. Variations in factors such as temperature, pressure, noise, humidity, time of day, light, darkness etc. could be reflected in their performance. A boring or stressful working environment will also impact their work.

3.5.2 Usability

A system's usability is very connected to its acceptability by the end users; therefore, it is an important factor for a design's success, (Nielsen 1994) defines usability according to the following multiple components:

Learnability of the system; it should be easy to learn

Memorability of the system; be easy to remember

Efficiency of the system; be efficient to use

Errors of the system; it should be designed in such a way that the users make less errors while using, and can easily recover from those that are made

Satisfaction of the system; should be user friendly so users are emotionally satisfied when using it.

3.5.3 Workload

Workload is a multi-aspect element expressed by the duties, amount of work, or a person's number of tasks to manage or achieve; tasks bounded by a certain time interval or to be carried out in a specific assignment context.

Workload could be physical or psychological. Over-loading (high workload, potentially leading to skipping actions or incorrect, incomplete execution) and under-loading (low workload, potentially causing lack of attention and complacency) should be addressed when scrutinising the impacts of work on human-machine performance (Abbott 2001).

3.5.4 Situation Awareness

Situational awareness is defined by (Palmer et al. 1995) as a state of alertness and insight on the part of a worker having relevant information about the task at hand and the external environment. It is an understanding of various effects with respect to a specific assignment immediately and in the near future, emphasising on the values of work.

Situation awareness is as an issue and considered as an important characteristic to workload. As part of the work process, a worker's information necessities must be identified, and properly made known to ensure adequate situation awareness.

Sometimes, although the required information needed by a user is available, it may not be in a form that could be easily used, and so will be of little value. (Reason 1997) points that another area that gaining recognition quickly is the subject of various organisational processes, policies and practices at all levels of an establishment; It has become obvious that these aspects have a significant influence and if left dormant will lead to potential susceptibility in design and operations.

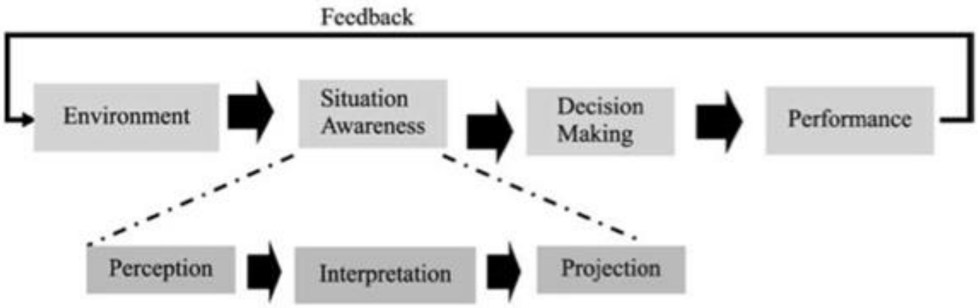
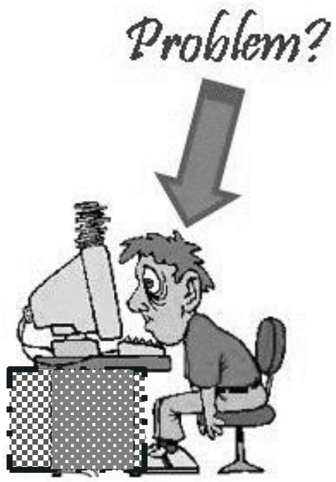


Figure 3.11: Situational Awareness Feedback Loop (Endsley 2000)

3.6 DECISION-MAKER (DM)



"Codes do not produce results, people produce results using codes"- Dave Whitfield, a long time CFD code developer and user.

The availability of Decision-Maker (DM) and the willingness to be part of the solution process, directing it in accordance to her/ his preferences is the most vital, fundamental belief to the successful use and application of interactive methods.

The perception of the DM to find the chosen method acceptable and advantageous to use is always important. The manner in which preference information of DM is specified, the overall understanding of a method's way of working must be interesting and easy.

In general, two phases are identified in a solution process: a learning phase and a decision phase. The DM first learns regarding the problem and various possible solutions and then decides on the most preferred solution which is found during analysis in the first phase. These two phases are presented iteratively in an optimisation problem.

It is a constructive process to solve a multi-objective optimisation problem interactively because through learning, the DM is builds on a confident belief of the types of solutions available, of what is possible. This gained knowledge is then faced with her/his preferences which may also evolve over time. Therefore, apart from a mathematical solution convergence, a psychological convergence usually occurs when making use of interactive methods.

3.6.1 DM and Trade-Off Information

When solving a multi-objective optimisation problem using interaction techniques, it is vital and advantageous for the DM to be aware of objective trade-offs when shifting between Pareto optimal solutions. This awareness allows the DM in deciding if the search should be continued for other optimal preferred solutions looking for specific aspects or not.

The preference information needed from DM varies in the level of difficulty depending on the type of trade-off information method used for a particular problem. However, coherence in DM's feedback is important in all the methods for an absolute convergence of the method. Even though most methods allow revisiting solutions and go back in the process, irregular feedbacks generally not preferred.

3.6.2 DM and Reference Point Approach

Reference point approaches enhance DM's intuition power by allowing an overall, comprehensive learning of the situation and decisions as presented by the problem model. The DM has the opportunity to learn about the situation and is free to modify the reference points to delve into interesting sections of the Pareto optimal set; various approaches can be adapted in such explorations.

The goal functions generally possess full controllability feature. DM can select any Pareto optimal solution by altering reference points and maximising the goal or achievement variable, thus paving way for a complete dominance by DM (Branke et al. 2008).

3.6.3 DM and Classification Based Methods

Classification approach relies on DM's intuition to steer the solution process in finding the solution most desirable because no imitation ideas are used. The DM tends to deal with objective function which deem meaningful, important and understandable for her/him.

An expectation about improved solutions can be expressed by the DM by directly comparing how well the desired expectation fares when the next solution is generated. Additionally, DM's desire for the extent of solution improvement or permitted levels of deterioration may be required as input information.

With respect to stopping criteria, classification-based methods and reference point approach methods share similar philosophy that a DM's sense of satisfaction is the most vital stopping factor which means that the process of searching will continue as long as the DM wants it to. Converging mathematically is not very important (unlike in trade-off based approach) but instead a psychological convergence is given priority. This supports the reality scenario where a DM naturally desires to be in control and does not usually prefer a chosen method to advise them regarding the generation of their most preferred solution.

3.7 DECISION MAKING

The designer when working with interactive techniques should be aware of various modes of unclear, inadequate communication, misunderstandings and glitches that could take place in order to proactively manage them by minimising any disruptions. The various steps during the design process involving decision-making activities are most times complicated (Figure 3.12), and the decisions taken will have critical impact on the design solution and the process.

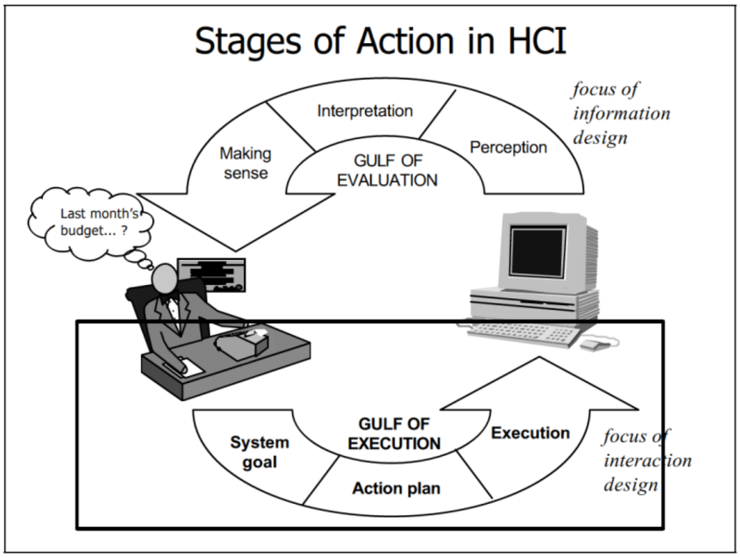


Figure 3.12: Planning and Course of Action (Norman, 2002)

An empirical study of engineers working in an industrial practice was carried out by (Ahmed, 2001) . It was observed that:

- All of the design strategies which were in place to aid various decision-making were observed during various instances. It was noted that the engineering designers when involved in decision-making activities did not make use of the available decision methods to structure their choices or conclusions.

- (Pahl & Beitz, 1996) stated that when evaluating between alternatives, all alternatives should be developed to similar levels and presented for decision making. Two designers were observed in this regard. It was noted that the designers did not present their design alternatives unless their first evaluation had been deemed successful; if not they generated another design solution alternative. Therefore, the evaluation was done between alternatives, it was carried out in a sequential manner, a 'synthesis and evaluation' style.

- When decisions were to be taken, the designers did not seem to present a set of relevant related criteria in the decision instance; rather they seem to focus on the known issues of the activity, specifying these issues appeared as an automatic sub-activity in making decisions. However, mention of activity issues was done many times even without activity task. This suggests that considering activity issues was an approach that helped the designer to specify decision criterion.

The study stated indicates that rather than making use of an available decision method, an engineering designer is likely to follow a work strategy which relies on her/his understanding of the ongoing design process situation. An extraordinary gap exists between the study and model proposals of decision-making in design methodology literature and the practical findings from experimental studies of engineers in practical industrial settings. (Hansen & Anderasen, 2004) suggests that there is a challenging need to combine results from theory and practice in order to boost the designer’s acceptance and use of research results.

3.7.1 Identifying Novel Views & Objects

A general human tendency is to remain with something that one is convicted about. The original source of conviction is emotion (Jastrow, 1917) and emotions influence decisions. Personality, culture, gender, power, social conventions, roles, fear are some major influencing factors of human emotion. A designer may perhaps recognise that a better solution exists, however it is convenient to believe in conventional, widely accepted solutions since the designer ‘knows’ they work. Settling for known solutions hinders the very objective of optimisation. Generally, an individual’s aptitude and creativity impact idea generation

It is a fact that some people are capable of producing excellent, inspired designs while some others try very hard to present any ideas at all. A lot of things in the world are not completely new but adaptations of things that already were. In general, innovations are birthed through crossing over of various ideas from different applications, the evolution of a product already existing by betterment notions generated through its use and observation, or direct duplication of other, identical products (Sharp et al., 2007). A study report by (IEGD, 2006) argues that diversity generates innovation in the form of new products, processes and systems and generally has a positive impact on business.

Aircrafts and the study of aeronautics mimic the natural world; moveable wing surfaces, winglets, silent-flight concept, light-weight flight structures, ‘groovy skin’ concept, formation flying, to name a few, though might come across as new technology are in fact inspired by observation of nature. Deliberately looking out for new ideas and inspiration is an important and beneficial step in any design process.

3.8 DISCUSSION

This chapter discusses various types of interactive methods and the importance of human decision-maker in the engineering loop along with the consideration of several elements of human factors.

A large collection of various types of interactive methods are available in literature, the task of choosing the most suitable technique for specific decision situations remains difficult. A 'decision situation' could be perceived as a DM, with a given set of choices (due to the several possible solution permutations and combinations) facing wholly or part of a decision problem.

To facilitate the availability of various techniques in a single decision system, some literature suggests proposals to develop open architectures or combined systems (Luque et al. 2007). Some relationships among various types of information that interactive techniques usually ask of DM like weights, trade-offs, reference points etc., are studied by (Luque et al. 2007). Combining the advantages of various techniques in order to reduce their flaws is one way of developing new, improved methods. Another algorithm proposition is to generate an approximation of the Pareto optimal set by combining reference points (Klamroth & Miettinen 2008); only those parts of the Pareto optimal set in which the DM is interested are approximated and sections are controlled by using a reference point for evaluation.

4

Visualisation of Solutions

4.1 INTRODUCTION

An aeronautical designer needs visual support and computational tools help in making the right things visible; various calculations over bodies such as aerofoil sections are presented to the user in an attempt to optimise the design solution. Such a project starts with several unknowns. Higher the complexity of a design system, higher the chance of a vaguely defined design parameters and requirements. A design process typically involves many incremental learning experiences and interactive visualisation attempts to allow a decision-maker to visualise computational results graphically, with greater utility to solve practical engineering problems. An efficient visualisation system is a relation between input data, the application in the background and the images generated (Figure 4.1).

By 1960s, the digital computers had advanced to the point of making it possible to attempt calculations of aerodynamic characteristics of aeroplane components by solving appropriate mathematical models. The use of visualisation to represent underlying information is not a new occurrence and the field of computer graphics has undergone some of the most important advancements in information visualisation paving way for other computer tools in exploring large amounts of data.

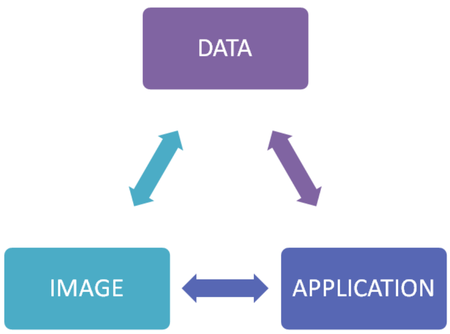


Figure 4.1: Visualisation in Software Applications

A computational problem task often involves two actions: (a) Solving problem using mathematical calculations and (b) choosing a preferred solution from several available alternatives. The first action includes design and analysis of various aeroplane components such as an aerofoil while the second is primarily a decision-making process (McCormick, et.al., 1987). The principles of visualisation have been developed and advanced within various fields of study and professional sectors such as photography, gaming, animation, typography, medicine and engineering simulations. (Blackwell, 2010) notes that an artistic mind-set, together with skills and understanding are required to build on the current available visual tools to improve them. Those working on such tools should be able, when required, to develop novel and improved visual representations.

Visualisation is an interdisciplinary branch of computing. It transforms data into multi-variant datasets and geometries, enabling engineers to observe computed simulations. It provides an arrangement to see the large or unseen data, enriching the analysis process and fosters thorough, sometimes unexpected insights. The field of visualisation is rapidly evolving and revolutionising the way engineers do work (McCormick et al., 1987) and its goals are:

- Analysis and Insight
  - Extract the content's information
  - Make things/ relations visible that are not obvious
  - Analyse data by means of various visuals
- Communication
  - Allow non-experts to understand
  - Show specific information in ways that all involved people can follow
  - Aid the skilled users in correct choices
- Steering
  - Interactively control and drive application
  - Use visual representations to grasp task situations and avoid delays

4.2 VISUAL PHASES

Visual representations are a means to assist thinking. They support externalising knowledge about a complicated task or domain, and trigger reasoning through interaction. Visualisation is not a separate entity but linked to its user; in order to try to understand it, it is also important to understand how its user thinks. Such a grasp is vital and it is complicated to process because there are several types of users and not a single set individual (Ziemkiewicz, et.al., 2012). In a specific work context, understanding visualisation requires making sense of how users vary and why. Visualisation plays an important role in computational simulation and following are the major phases with multiple steps occurring within each phase (Edwards, 1989):

- Modelling
- Simulation
- Analysis

Developing computational model is the first step for a specific problem of interest such as an aerofoil. Modelling phase, the pre-processing phase, is the transformation of a physical problem into a mathematical problem and this may carry over certain modelling errors (Figure 4.2). Generally, this phase does not use computation and visualisation is not an important part. However it is one of the important phase in the simulation process as the designer decides on which physics sections, boundary conditions to be included and which to be ignored, chooses a mathematical model of the physical model such as time-dependent Navier-Stokes equations, and any extra modelling assumptions to collect system information such as turbulence, transition.

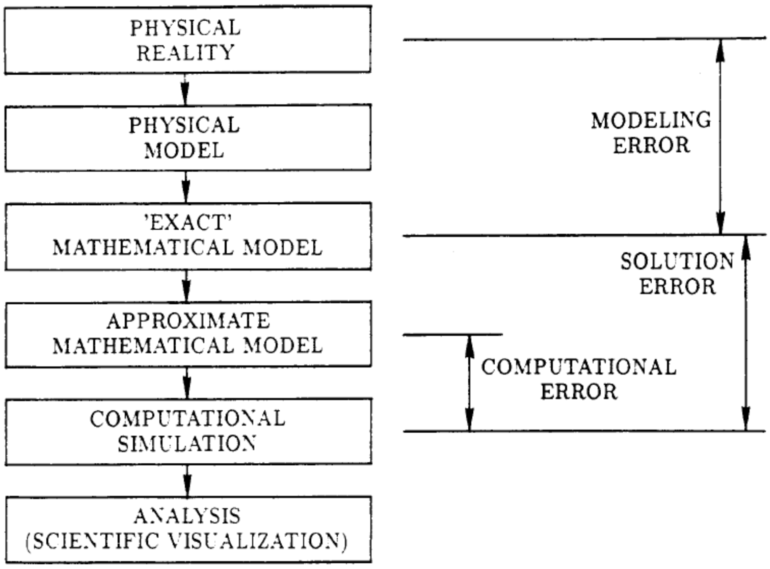


Figure 4.2: Stages of a Computational Simulation (Edwards, 1989)

The simulation phase tries to compute a near accurate and efficient solution to the mathematical problem. The choice of computational techniques plays a vital role in the simulation process and visualisation techniques are useful for interaction, controlling and monitoring of iterations. There are associated errors with various types of approximations introduced into the mathematical model (Figure 4.2), these have to be identified and controlled. Such critical observation and examination depends on the user's knowledge, awareness of errors and interest taking corrective actions.

Visualisation is advantageous in this phase to error detection and controlling. Specifically, the application of interaction graphics will offer the designer an efficient way to observe and steer simulations. This allows unstable or divergent solutions to be recognised and cancelled, saving time and resources. At other times, the user could adjust various computational parameters to interactively direct the simulation to a stable solution. A data set is the result of simulation phase checked in the analysis phase.

Gathering of information happens in the analysis phase; solution data set from the simulation phase is interpreted and evaluated. The results are examined to check for any errors due to assumptions made in the modelling phase. It also provides an opportunity to examine physics of the data set enabling any improvements required in either the physical or mathematical model. Any negligence in correcting errors could lead to analysis of results which may be perceived as being correct and good, while the results themselves could be a culmination of several unchecked intermediate errors (Figure 4.2). When significant changes are made, the overall simulation process is repeated along with any appropriate model modifications. This phase of computational simulation gained importance due to the tremendous advancement in large scale computing.

Much time is spent in the development of computer programmes to simulate physical model and processes; and a larger proportion of time is used in understanding and interpreting simulation results. Visual analysis is vital in computational simulation, it can be used to examine data sets for any numerical errors and the computational simulation could be repeated with different parameters if necessary. It also aids in viewing transformations, positions and number of data sources.

4.3 INTERACTIVE SPACES

“The purpose of computing is insight, not numbers” – R.W Hamming, 1961

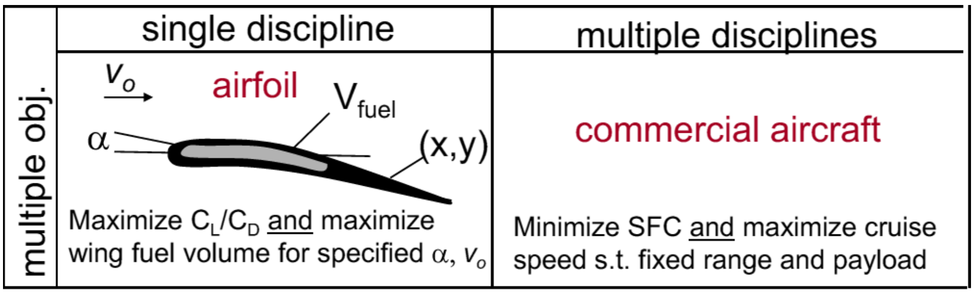


Figure 4.3: Multi-disciplinary Vs Multi-objective spaces (Alfaris, 2010)

Interactive spaces allow exploring designs, utilising human insight and understanding along with numerical methods. Figure 4.3 explains the terms multi-disciplinary and multi-objective; this research examines single discipline and multiple objectives. When data set sizes increase, so does the need for scientific visualisation. Evaluating performance of a single parameter and its optimisation could take several long hours of computer processing. In such situations, there is a limit to the number of parameter configurations that could be evaluated, and it seems beneficial to adapt approaches that achieve good results on a small number of evaluations.

In multi-objective aerodynamic problems, there is no unique optimum because the objectives are usually in conflict; apart from the relationships between objectives, the relationship between variables and objectives is also of interest to the analyst. Figure 4.4 and 4.6 explain the mapping of particle vectors between spaces and Figure 4.5 explains interactive features offered by the visual module.

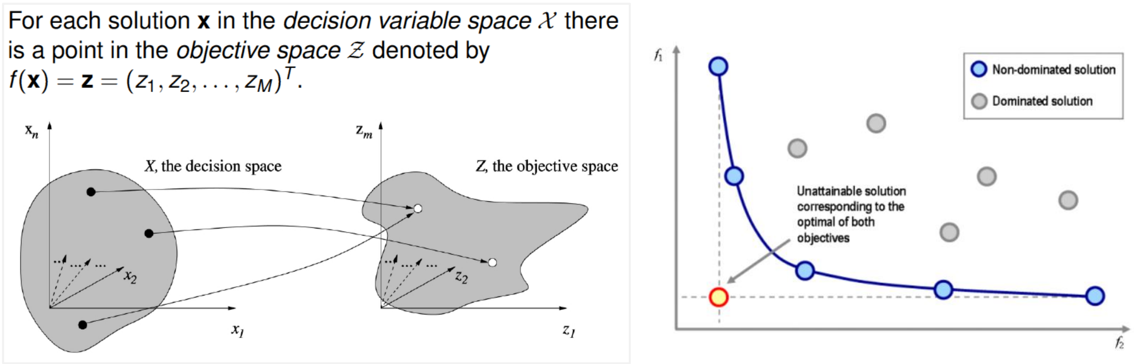


Figure 4.4: Mapping between decision space and objective space (Narzisi, 2008) Decision vectors are called Pareto-optimal if they are globally non-dominated. The set of all non-dominated decision vectors is the Pareto-optimal set.

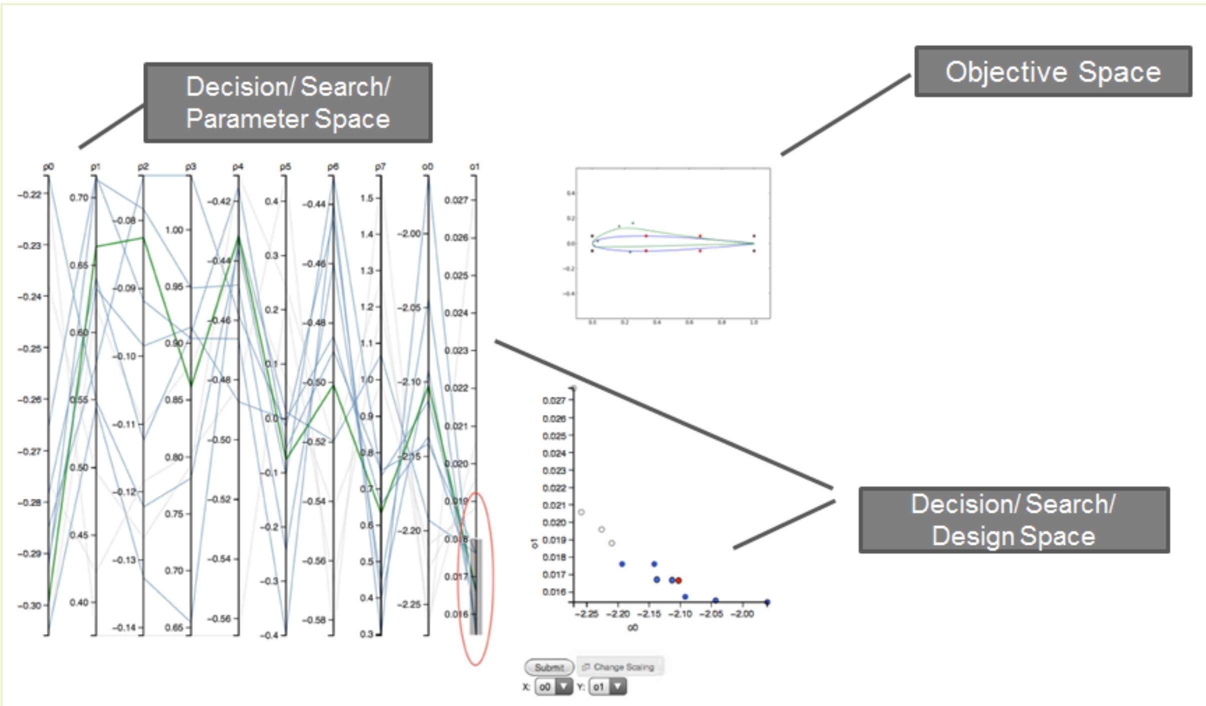


Figure 4.5: Screen-shot of the test module illustrating interactive spaces. A range in objective space is selected correlating to parameter ranges with feasible solutions (highlighted). The scatterplot in the lower right corner shows O_1 Vs O_2; decision-maker can further select any of the available solutions, either in scatter plot or parallel

coordinates plot. The corresponding point is then highlighted in both plots and also displayed in the upper right hand corner (Hettenhausen, et al., 2013). The feasible space is commonly denoted as the decision space or parameter space, and the image of the decision space, subject to $\mathbf{f}(\mathbf{x})$, is referred as the objective space.

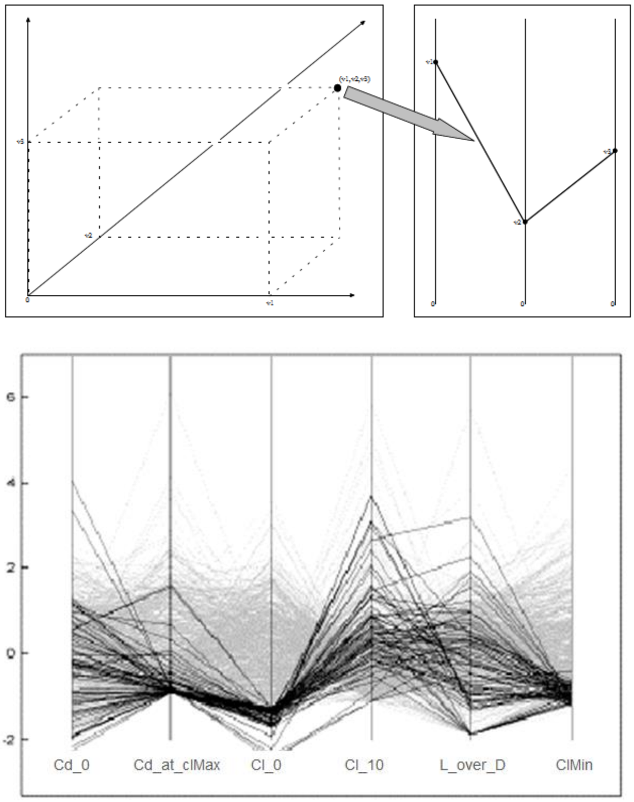


Figure 4.6: Parallel Coordinates and the representation of a point from n-dimensional space to n-1 line segments in 2D space (Kriwaczek & Rustem, 2000)

4.3.1 Search/ Decision Space

The search for a ‘feasible’ design is the search for a design which considers all design constraints and this is a difficult task. The complexity is linked to the clustering of decision or search space, geometric complexity of the layout, solver, search algorithm and their relations, leading to the use of heuristic techniques for solving multi-variate, multi-objective problems.

Multi-objective optimisation involves a solution specified by the decision-maker and a series of acceptable, but not necessarily feasible solutions iteratively in response to generated solutions. These solutions are points in the decision space. The decision-maker begins to discern any acceptable solutions and accordingly adjusts specifications of what one finds feasible.

To successfully use this approach, the decision-maker (DM) must:

- perceive clearly how close the wishes are with respect to what can be achieved
- specify correctly on what compromises the DM is willing to undertake

The above require support from an interface that:

- can display several points in n-dimensional space, showing clearly where and by how much the various points differ from one other
- allows the user to adjust positions of generated points to express wishes and steer the system in searching for next solutions

I-MOPSO module framework was set up to accommodate such a possible kind of domain space by implementing the following:

- Scale and Translate
- New range

Scaling and translating or interpreting is simple and predictable. The module set-up continues to generate particles according to the specified interval. A scaling and translating factor are utilised after particle creation to present those particles in the correct interval,. The decision-maker chooses on the search space interval and defines the value via input file; scale and translate factors are computed and sent to the model-runner where coordinate transformation happens prior to the particles move to the model.

Scale and translate factors are calculated as follows:

$$scaling(j) = \frac{\max_{\mathrm{par}_j} - \min_{par_j}}{2}$$

$$translation(j) = \frac{\max_{\mathrm{par}_j} + \min_{par_j}}{2}$$

Limits for each parameter domain are given by $\max_{\mathrm{par}}$ and $\min_{\mathrm{par}}$. The factors are calculated for each parameter j; the normalised parameters ($\mathrm{par}_{\mathrm{old}}$) are transformed to new parameters ($\mathrm{par}_{\mathrm{new}}$) according to:

$$\mathrm{Par(j)}_{\mathrm{new}} = \mathrm{Par(j)}_{\mathrm{old}} * \mathrm{scaling(j)} + \mathrm{Translation(j)}$$

This approach of generating new range allows for a better adaptability; the parameter's upper and lower bounds are changed within code runs. Diversity is promoted through updating of positions for creating new solutions or by mutation, turbulence operator.

Study of I-MOPSO module with different parameter ranges is an area to be further investigated. The two factors were however intended to add flexibility to the framework.

The decision-maker is provided with a chance to choose various swarm distributions and intervals for initialisation; values of $c1$, $c2$, w, the turbulence intensity and archive size can be modified via input file.

Based on the notion of user intent, (Soo Yi, et.al., 2007) set up the following seven categories of interaction:

- Select: mark something as interesting
- Explore: search something else
- Reconfigure: show a different arrangement
- Encode: show a different representation
- Abstract/Elaborate: show either more or less details
- Filter: show something conditionally
- Connect: show relation between items

Selection offers a designer with the capability to mark interesting single or multiple data items by keeping track. By making items of interest visually distinct, keeping track becomes easy. Exploration is to analyse a different subset of data. Both computational and human cognitive limitations affect data and information processing. Reconfiguration changes spatial arrangement of representations and aids different point of views of the data set, especially to reveal hidden aspects of data and their relationships. Encoding alters visual representation of the data fundamentally impacting its visual appearance by modifying colours, sizes and shapes of data elements.

Abstract feature techniques support a designer with the ability to adjust levels of data representation abstraction, aiding in the alterations, overviews or detailed views of individual data sets, often with several intermediate levels. Filtering enables the change of data sets shown according to certain specific settings, so that only parts satisfying those criteria are displayed. Connect technique refers to highlighting links, relationships between various data items represented, also to present data that is relevant to a specific selection.

4.3.2 Objective/ Design Space

The aspects of a design space have been defined by (Schulz, et.al., 2013) as '5 W's': Why, What, Where, Who, and When and sometimes How.

These aspects are very often used to describe a problem or task situation from all appropriate angles. They can be useful in communication and documenting technical data. They support visualisation, in analysing tasks and for understanding user intentions. These aspects are defined as follows:

- **Why** is a task being taken up? What is the task's goal?
- **How** is a task being executed? What are the methods?
- **What** is expected of the task? What are the resultant characteristics?
- **Where** does the task fit into overall work? What is the task's direction?

- **When** is a task carried out? What is the order and timing?
- **Who** is doing the task? Who is the user or decision-maker?

The last two aspects (when, who) are not separate considerations but form part of a larger, overall task context. They depend on the context's previous and succeeding tasks in a given sequence, and also involves capabilities, responsibilities to carry out assignments in a cooperative setting.

A generic multi-objective optimisation problem is presented in the form:

Minimise $\{ f_1(x), f_2(x), \dots, f_k(x) \}$ subject to $x \in S$

Objective Space is a vector space including all objective functions. Objective vectors are images of decision vectors and consist of objective function values.

$Z = f(x) = (f_1(x), f_2(x), \dots, f_k(x))^T$; $S \subset R^n$; $f_i : R^n \rightarrow R$; $k(>= 2)$

The projection of the feasible region in the objective space is referred as feasible objective region, $Z = f(S)$.

In multi-objective optimisation, objective vectors are considered optimal if none of their components can be made better without worsening at least one of the other components. A decision vector $x' \in S$ is called Pareto optimal if there does not exist another $x \in S$ such that $f_i(x) <= f_i(x')$ for all $i = 1, \dots, k$ and $f_j(x) < f_j(x')$ for at least one index j.

The set of Pareto optimal objective vectors can be denoted by $P(Z)$ and the set of Pareto optimal decision vectors can be denoted by $P(S)$. An objective vector is Pareto optimal if the related decision vector is Pareto optimal. The set of Pareto optimal solutions is a subset of weak Pareto optimal solutions set. A decision vector $x' \in S$ is a weak Pareto optimal another $x \in S$ does not exist such that $f_i(x) < f_i(x')$ for all $i = 1, \dots, k$ (Branke, et.al., 2008).

If the objective functions are bounded over the feasible objective region, the range of Pareto optimal solutions in this region provides valuable information regarding the problem under consideration. Lower bounds of a Pareto optimal set are available in an ideal objective vector, obtained by minimising each of the objective functions individually subject to a feasible region. The upper bounds of a Pareto optimal are generally difficult to acquire and there is no effective way to get an exact objective vector for non-linear problems. Estimation could be obtained but may be unreliable, hence raising the need for involvement of a decision-maker and analyst.

4.3.3 Search Space Vs Objective Space

In Pareto optimisation, two spaces are considered - the decision space or search space '**S**' and the objective space '**Y**'. The vector valued objective function $f : S \rightarrow Y$ provides the mapping from the decision space to the objective space. The set of feasible solutions X can be considered as a subset of the decision space, i.e. $X \subseteq S$. Given a set X of feasible solutions, Y is defined as the image of X under f.

The sets S and Y are usually not arbitrary sets. To define an optimisation task, it is necessary that an order structure is defined on Y. The space S is usually set up with a neighbourhood structure. This neighbourhood structure is not required for defining global optima, but it is exploited by an optimisation algorithm such as particle swarm optimiser that gradually approaches optima. During the formulation of local optimality conditions, the choice of system's neighbourhood could significantly influence the difficulty of an optimisation problem. The defining of neighbourhood gives rise to many characteristics of functions, such as local optimality and limitations; neighbourhood structures needs to be especially mentioned in discrete spaces because the continuous optimisation locality usually then refers to Euclidean metric.

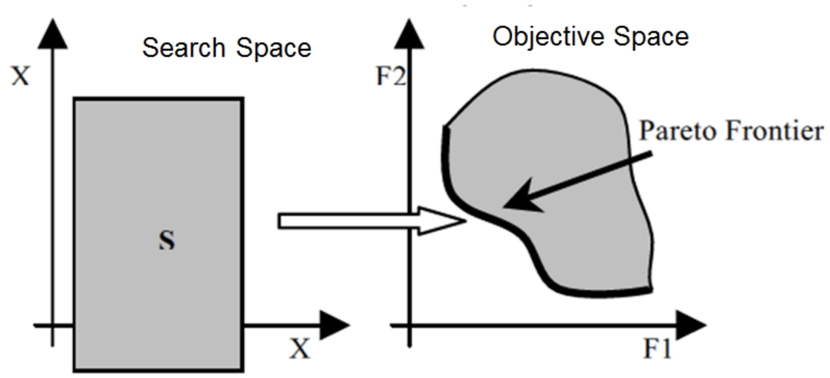


Figure 4.7: Mapping from Search Space to Objective Space (Agarwal et al., 2004)

Euclidean metric or distance is the straight-line distance between two points, satisfying certain specified relationships (distance, angles, translation, rotation) in an Euclidean space. With this distance, Euclidean space becomes a metric space.

In defining a landscape, it is useful to distinguish the general concept of a function from the idea of a function with a neighbourhood determined on the search space and a partial order determined on the objective space. In general, all local minima are also global minima.

In order to define a landscape in finite spaces, two important structures are needed: a graph in search space where edges connect nearest neighbours and a design in the objective space. For several optimisation related definitions, there is no need to specify a height function, it is sufficient to specify an order on the search space (Emmerich & Deutz, 2006).

4.4 MULTI-OBJECTIVE DATA VISUALISATION

Validating solutions for multi-objective problems continues to be a challenging matter till date. The most practical way to validate solution to a multi-objective problem is to make use of visualisations to check the computed results; such visualisation is a progressing area of research for multi-objective problem applications (Agarwal et al., 2004).

A quick and meaningful examination of the problem space or data set could be achieved by integrating the capability to interact in real-time with generated displays. The intention of multi-dimensional multi-variate visualisation (MDMV) is to interpret large amounts of data into meaningful, intuitive visual representations . To achieve this goal, several methods and applications have been developed, but they are also restricted by hardware and software limitations. Almost all visualisation techniques try to transform a multiple dimensions or variates of a problem or dataset to be mapped to a 2D or 3D visual space.

Cognitive elements such as spatial and verbal abilities, and working memory competencies vary between individuals, and affect logic and interpretation in various ways. Particularly, spatial and perception capabilities have an impact on how well users are able to perform several different tasks in a visualisation system (Ziemkiewicz et al., 2012).

(Keim, 1997) divided visual data exploration techniques for multi-objective data into six categories: geometric, icon-based, pixel-oriented, hierarchical, graph-based and hybrid techniques. Based on this, (Wing-Yi Chan, 2006) classified them into four broad classes depending on the overall approaches adapted to generate resulting visualisations (de Oliveria & Levkowitz, 2003):

- Geometric Projections
- Pixel-Oriented Techniques
- Hierarchical Displays
- Iconography

A plot is a graphical technique for representing a data set, a visual representation of relationships between two or more variables, playing a vital role in analysing data. Plotting methods are both quantitative as well as graphical, providing insight into elements of a data set for testing assumptions, selecting models, validating, estimation, identifying relationships, determining related factors, providing a comprehensive view of the underlying structure of the data to the inquisitive designer, including any unusual observations.

Three aspects or objects define a plot: data, layout and figure. Data in more than two dimensions are difficult to represent. In programming, data is actually a list objects, containing all the traces that a designer wishes to plot. A trace is a collection of data and the specifications of which one wants that data plotted. Multi-objective optimisation is not linear. For a two-variable plot, unit changes in the x-variable will not always bring about the same change in the y-variable, resulting in a curve instead of a straight line.

Geometric projection techniques were already used even before the emergence of information visualisation. Cartesian space is familiar and understanding representations is not difficult. However, when the dimensionality of data increases, it requires some effort to understand.

Individual data items are encoded into pixels in pixel-oriented techniques; several data points mapped to corresponding pixels appear at the same position in each respective window. With appropriate rearrangements, designers can notice the various attribute relationships along with trends and patterns of the hidden data.

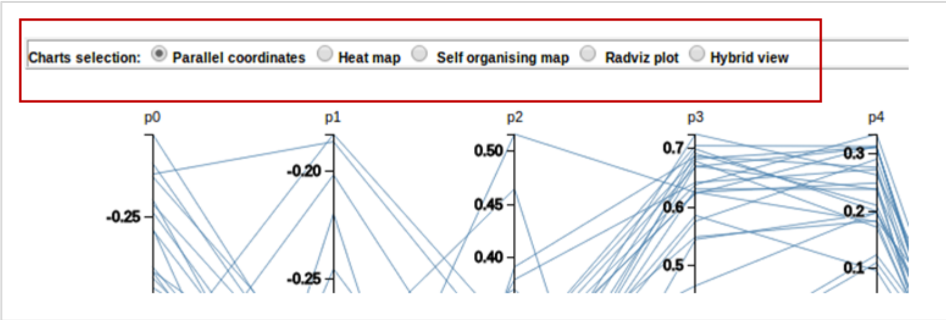


Figure 4.8: An extract of I-MOPSO module showing the various visualisation techniques

Hierarchical displays are derived from the concept of hierarchical trees and are effective in visualising hierarchical data; it is also their limitation. The generated plots of pixel-oriented techniques and hierarchical displays are not as easy as those of geometric projections; knowledge and skill on the designer's part is required to understand these hierarchical visuals.

Iconography makes use of a multi-dimensional icon, or glyph, as the unit of visualisation. Data attributes map to a glyph that has several graphical properties. When the glyphs or data items are densely packed together, they produce certain texture patterns. This aids in studying the overall features and data relationships. Colour has been extensively used to add extra dimensions; these could also be replaced by textures to generate graphical attributes for data visualisation.

4.5 PLOTTING

Data analysis is usually done quantitatively or graphically. Quantitative techniques are classical; these procedures yield numbers or tables. Graphical techniques or plotting is a large collection of statistical tools and engineering heavily relies on these tools as they are the quickest path to gain insight into a data set in terms of assumptions, choices, estimations, relationships and outlier detection. Some of those graphical approaches are addressed here.

A wide ranging industrial shape design and optimisation module, abbreviated as WISDOM© is an Airbus flight physics tool currently under development. Choice of plotting techniques adapted in this research is influenced by the parallel, ongoing work on this tool. Some aspects, features of the development tool are mentioned in this report and trade studies were carried out using I-MOPSO module.

4.5.1 Scatter Plot

Scatterplot, in general, is used for bi-variation discrete data in which two attributes are projected, along the x and y axes of the Cartesian coordinates. Scatterplot matrix is an extension for multi-dimensional data where a collection of scatterplots is organised in a matrix simultaneously to generate correlated information among the attributes.

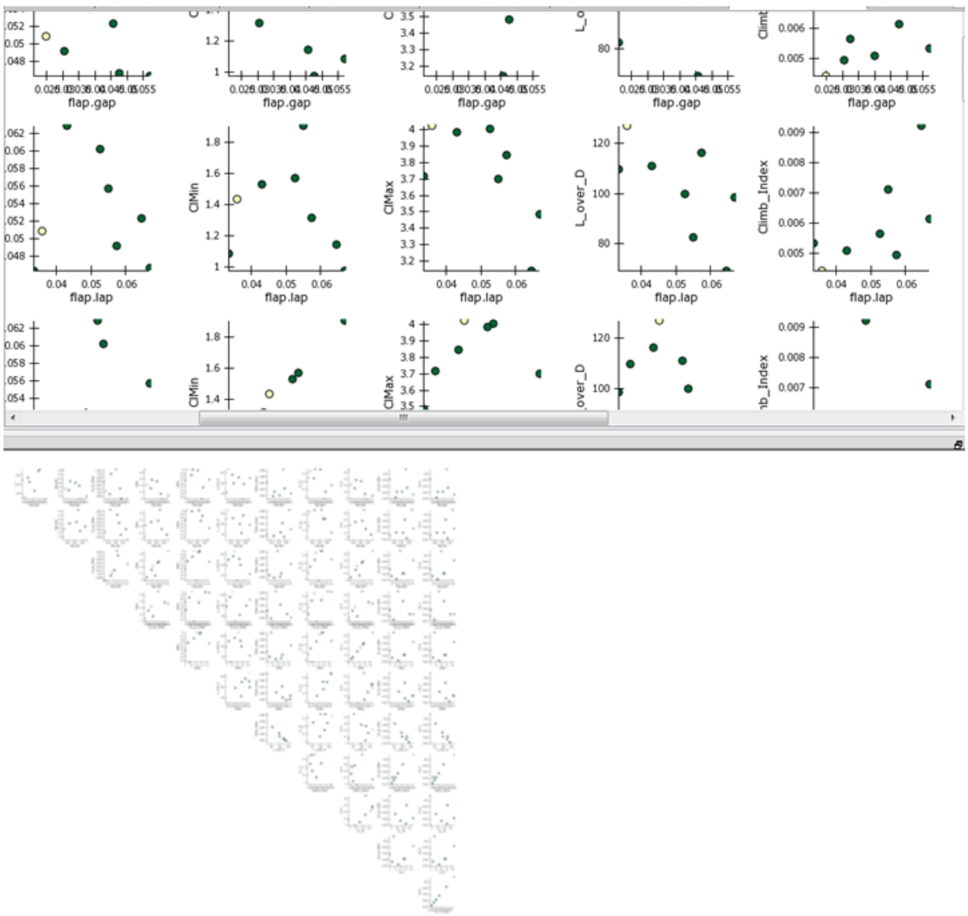


Figure 4.9: Example of two-dimensional scatter plot matrix generated over a low-speed, three element, deployed aerofoil section using HiLi solver in WISDOM© tool. The top view is an expanded view showing individual graphs clearly; the lower is a compact view displaying all possible parameter/ objective combinations on one page which is unclear and not helpful to the human decision-maker.

Scatterplot is an easy visualisation method projecting all vectors to a low-dimensional space by disregarding all the dimensions of the vector that are beyond those that can be visualised. When done for all possible combinations of lower-dimensional spaces, a scatter plot matrix is obtained.The scatter plot matrix is a very fast, simple, and robust visualisation technique that displays information on approximation sets, also the different distribution of vectors. Patterns in the relationships can easily be observed by the designer between pairs of attributes from the matrix; higher dimensions may aid in revealing important patterns which may otherwise be difficult recognise. When the number of points or data items becomes too large, visualising becomes confusing and is a limitation (Figure 4.9).

The technique of brushing could be used to deal with the above problem. Brushing tries to interpret by highlighting a specific *n*-dimensional subspace in the visualisation; the corresponding points of interest are highlighted or coloured in each scatterplot of the matrix.

4.5.2 Parallel Co-ordinates Plot

Parallel coordinates are able to reveal relationships between multiple variables, specifically useful when a designer wants to identify which choices correlate highly to a particular outcome. Parallel coordinates plot represents multi-dimensional data using lines. Each dimension or attribute is represented by a vertical line. The maximum and minimum values of the dimension are generally scaled to the upper and lower bounds on these vertical lines. An n-dimensional point is represented by n-1 lines connected to each vertical line at the appropriate dimensional value.

Lines are chiefly used to encode time-series data. Changes through times from one value to another are indicated by the up and down slopes. The lines in parallel coordinate displays do not indicate a change. A single line in a parallel coordinates graph connects a series of values, each linked to a different variable which calculates multiple aspects of an attribute, such as lift and drag coefficients. The problem of obstruction occurs in a parallel coordinates' display. This could be reduced by dividing the overall data into separate series of parallel coordinates graphs, separating each range of influencing parameters into an individual graph.

Each axis in parallel coordinates plot can have at most two neighboring axes, one on the left, and one on the right (Figure 4.11). For a d-dimensional data set, at most d-1 relationships can be shown at a time. In time-series visualisation, there exists a natural predecessor and successor and in such special case, a preferred arrangement exists. However when the axes do not have a unique order, finding a good axis arrangement requires the use of heuristics and experimentation. In order to explore more complex relationships, axes must be reordered. In a 3D space, axes can be re-arranged, however, the visualisation becomes difficult to interpret and interact with in comparison to that of a linear order.

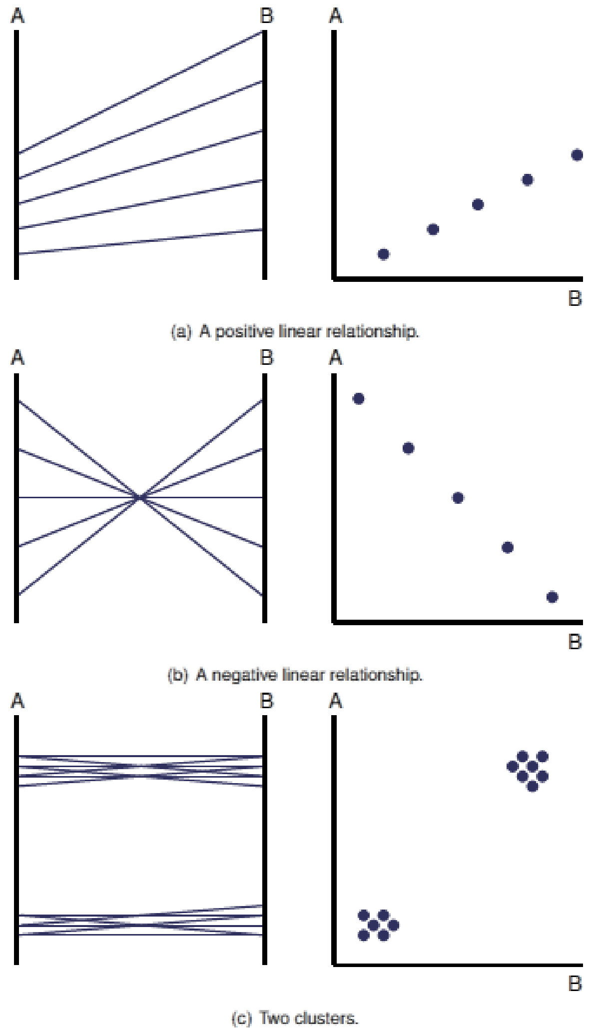


Figure 4.10: Parallel coordinates (left) and scatter plots (right) showing common features in data. The two-dimensional points in Cartesian coordinates map to lines in parallel coordinates (Johansson, 2008).

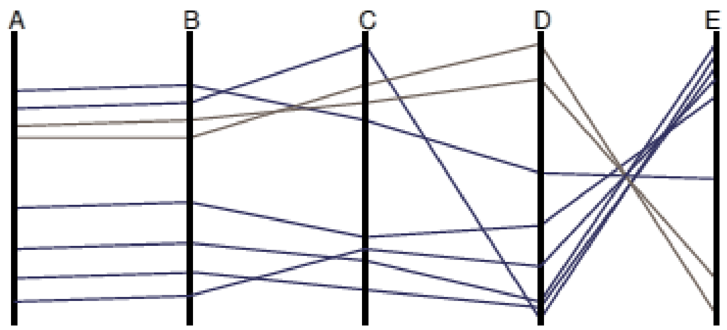


Figure 4.11: A parallel coordinates representation of a data set with five variables. Such a representation supports a number of analysis tasks. The example shows the identification of negative relationship between variables D and E and the similar shape over all variables seen for the two selected items highlighted in brown (Johansson, 2008)

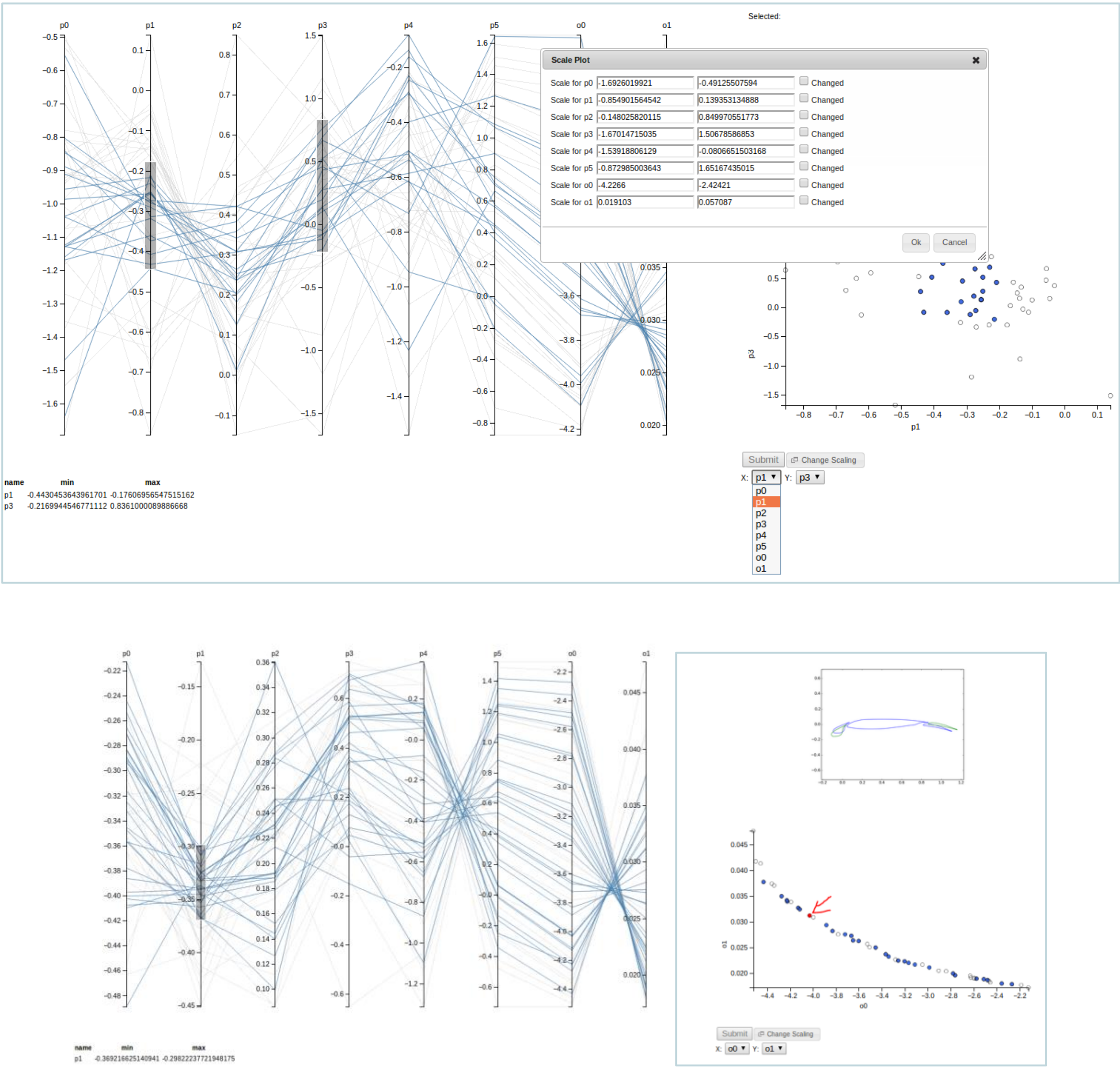


Figure 4.12: I-MOPSO and Graphical User Interface. Shown here is the parameter selection of a particular range on P2 vertical coordinate (below) and its corresponding dependent results impacting other parameters and objectives are highlighted. Selecting a particle of the interval is shown by green lines in parallel-coordinates' plot and by red dot in the relative scatter plot.

4.5.3 Heat Map

A heat map uses colours to represent values and is a two-dimensional representation of data. In general, it refers to any computed visual that utilises colour to depict quantitative data. A simple heat map provides an immediate visual summary of information while more detailed heat maps allow the designers to comprehend complex data sets.

There are several ways to display heat maps, however all approaches make use of colour as a means of information exchange to show links between data values which might otherwise be difficult to understand and examine if represented numerically in a spreadsheet. When colour is used, it is necessary to understand their correlated values and representation such as high, low or intermediate. Various grid structures can be adapted into a heat map display; Figure 4.13 shows a hexagonal grid. I-MOPSO module uses a rectangular grid.

Multi-variate heat map matrices are often used by designers to analyse data, they tend to use heat maps combined with a additional display such as a dendogram; a dendogram is a tree structure, organising entities hierarchically, based on similar multi-variate profiles. Heat maps don’t have the problem of obstruction where objects hide behind or are restricted by other objects, where in comparison, parallel coordinates seem cluttered with several lines displayed in the close proximity. Each computation in a heat map is limited to its own cell within the matrix and eliminates obstruction (Few, 2006).

Multi-variate displays exhibit a series of colours are sometimes not easy to perceive and remember by a human user when compared to a lines’ pattern formed in parallel coordinates display. Grid shape and size affects a user’s view and perception of results.

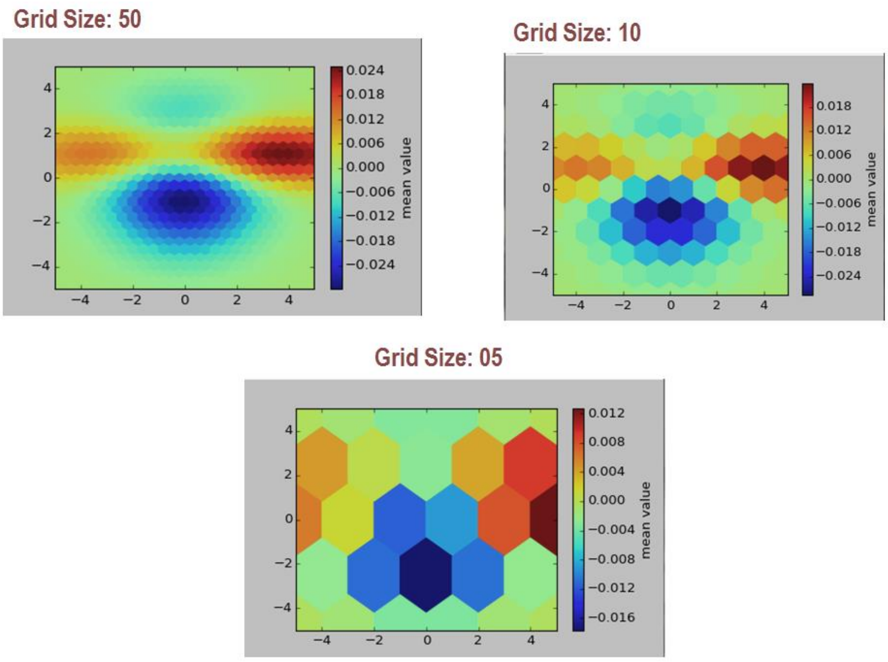


Figure 4.13: Hexagonal grid heat map generated in basic Python visual window showing the difference in visuals depending on the grid size. Colours are much clearer to identify in a smaller grid size compared to a larger grid size where the view is hazy and boundaries are not clearly defined.

4.5.4 Self-Organising Map

Self-organising maps (SOM) make use of neural network algorithms which project multi-variable data onto a two-dimensional array of nodes without supervision. Each node is correlated to a randomly initialised n-dimensional weight vector, where n is the dimension of the data to be visualised. The algorithm automatically sorts out various values and adapts itself so that identical data are displayed closer. Each value component can be visualised by adapting grid colours depending on component selection values. Thus, many dimensions could be visualised at the same time by displaying several mapping components side by side.

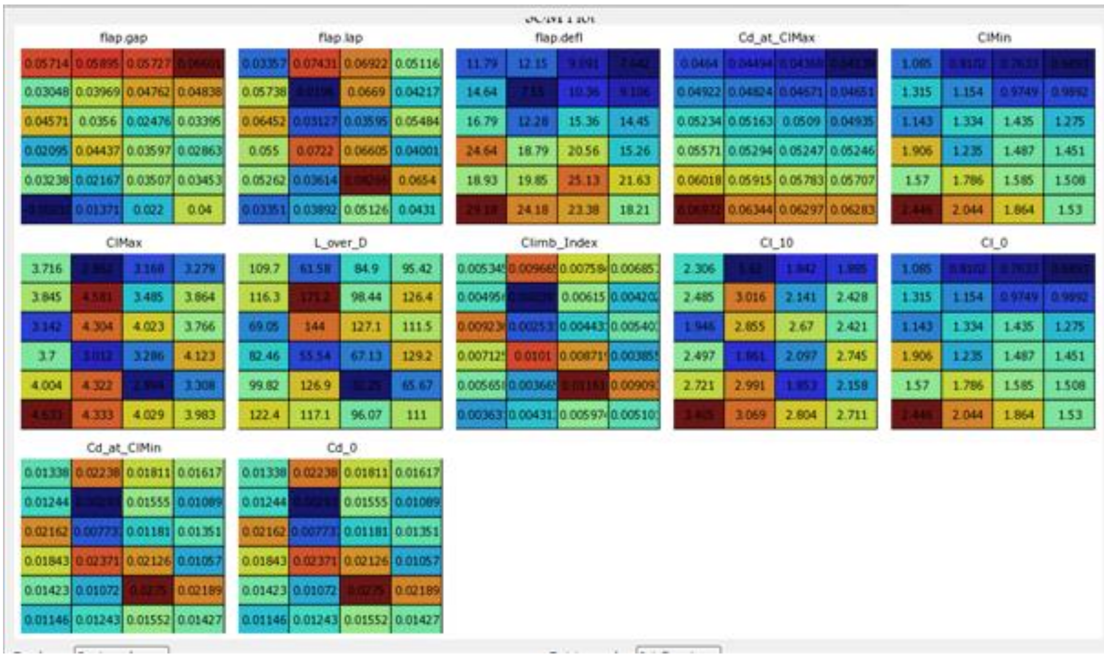


Figure 4.14: Example of self-organising map using rectangle blocks generated over a low-speed, three element, deployed aerofoil section in WISDOM© tool. The calculated values were also shown inside each block, along with its associated colour which was an available feature in this particular tool.

By adjusting algorithms, SOMs can be visualised in various ways. Depending on the chosen colour palette, SOM will map those colours onto a hexagonal or rectangular grid area. One of the popular approaches is U-matrix (unified distance matrix) where the distance between adjacent neurons is displayed with different colours. Clusters of similar or lower value nodes are represented by light-colour parts and dark areas indicate cluster boundaries or higher values. A linear SOM arranges all nodes in a single cluster and is easy to interpret in comparison to a spherical SOM where the exact number of clusters is difficult to establish.

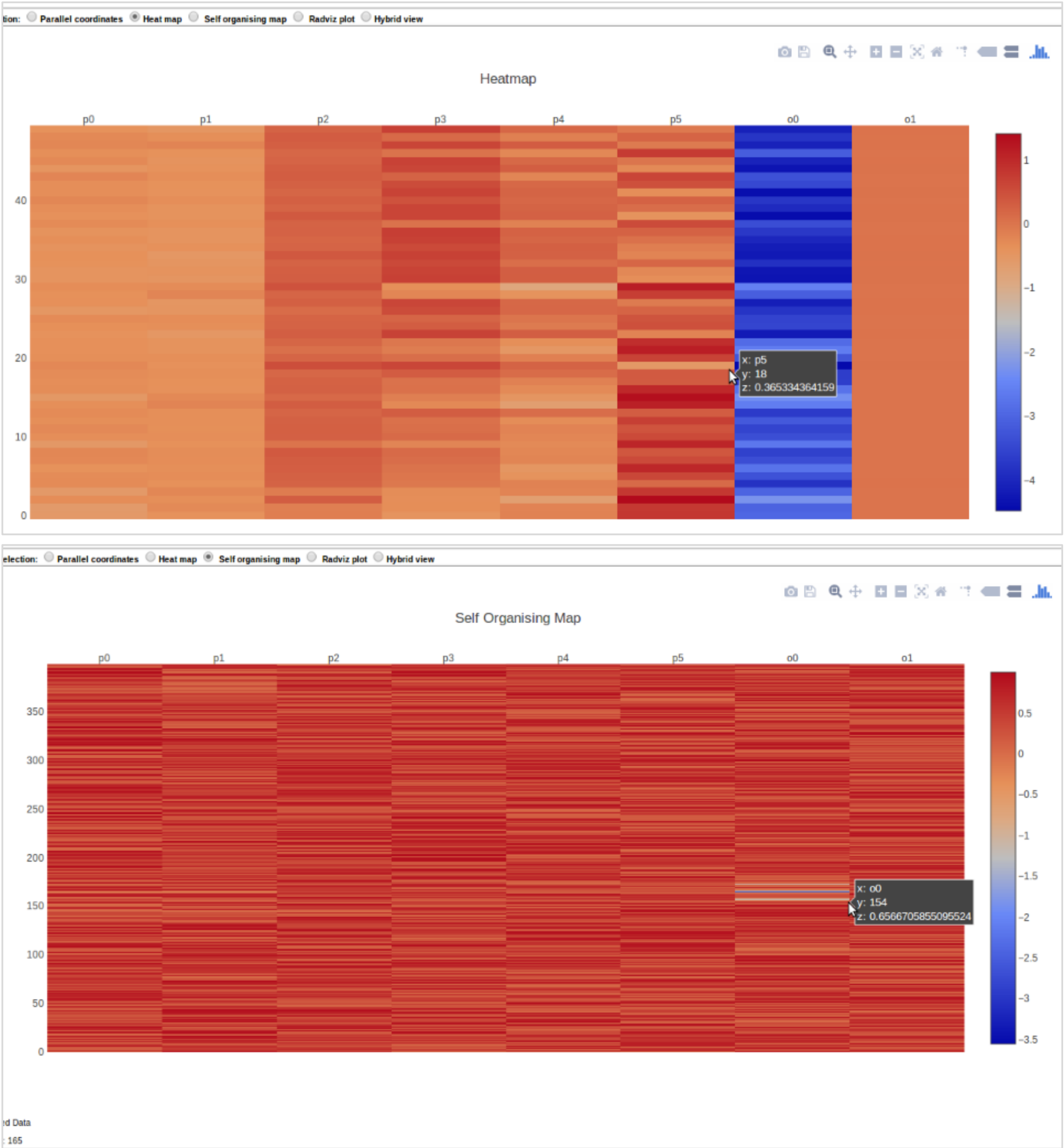


Figure 4.15: An extract of I-MOPSO module showing optimisation results in a heat-map (above) and self-organising map (below). The test run was carried out on Garteur aerofoil section.

4.5.5 Radial Co-ordinate Visualisation (RadViz)

The Radviz technique (Hoffman, 1999) represents each n- dimensional data item as a point in a two-dimensional space. The points are positioned inside a circle whose perimeter is divided into *n* equal arcs. The points are equally spaced along the circle's perimeter, referred to as anchor-points or dimensional anchors; each data dimension is linked with one anchor-point. For n-dimensional data, each data point will be associated with n anchor-points through n different springs. These data points are then shown at the position that generates a spring force sum of zero.

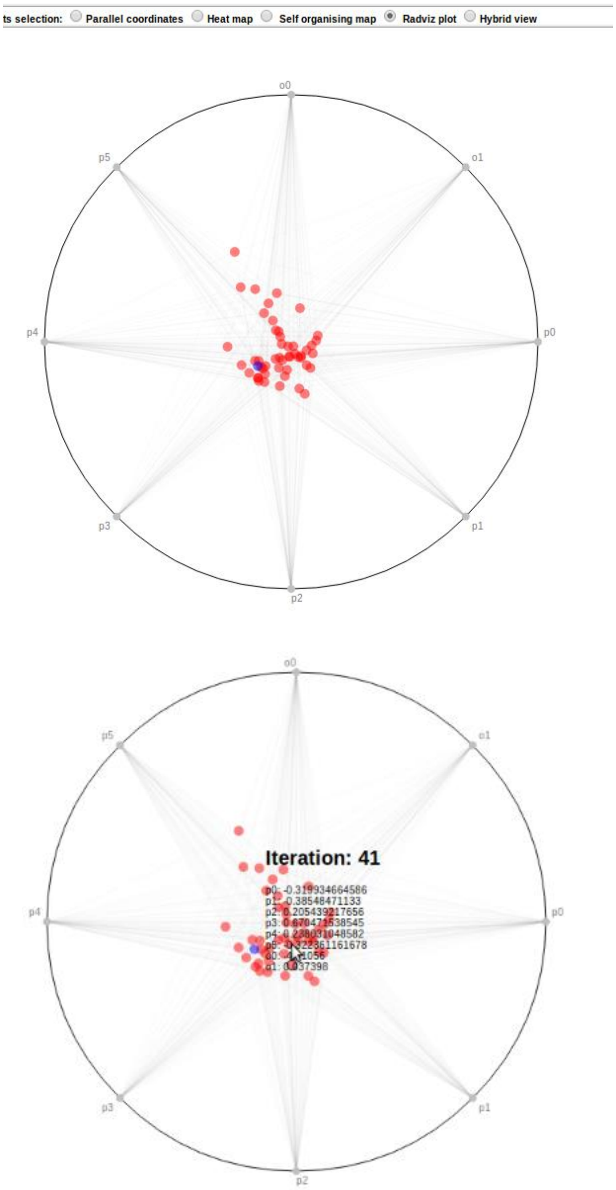


Figure 4.16: The Interactive RadViz projection of I-MOPSO data set

Each data dimension's values are normalised to the range [0, 1]. In the original, undisturbed range, those variables with higher values than others will influence the display on the spring. The data points will position exactly in the centre of the circle if all n coordinates have the same value irrespective of whether they are low or high. A unit vector point lies exactly at the fixed point on the circle's edge, where that dimension's spring is fixed.

4.5.6 Other Multi-Criteria Plotting Methods

There exist numerous multi-Objective, multi-dimensional, multi-variable visualisation methods. As this work concentrates on visualising approximation sets, the techniques are rounded for this purpose. Visual techniques are both general and specific. The general methods can also find their use also outside the realm of multi-objective optimisation.

General MDO/MOO Visualisation Methods	Methods Specific for Visualising Approximation Sets
✓ SCATTER PLOT	• DISTANCE & DISTRIBUTION CHART
• BUBBLE CHART	• INTERACTIVE DECISION MAP
✓ RADIAL COORDINATE VISUALISATION	• HYPER SPACE DIAGONAL COUNTING
✓ PARALLEL COORDINATES	• TWO-STAGE MAPPING
✓ HEAT MAP	• LEVEL DIAGRAMS
• SAMMON MAPPING	• HYPER RADIAL VISUALISATION
• NEUROSCALE	• PARETO SHELLS
✓ SELF-ORGANISING MAP	• SERIATED HEATMAP
• PRINCIPAL COMPONENT ANALYSIS	• MULTI-DIMENSIONAL SCALING
• ISOMAP	

Table 4.17: Table showing various visualisation techniques

General multi-objective optimisation visual methods were used in this work, methods highlighted in orange in Table 4.17 indicate the methods adapted. A combined view has also been set up in I-MOPSO module. Various other techniques are also mentioned and the list is not exhaustive; some techniques are generic while some are more suitable for specific problem tasks.

4.5.7 Combined Visualisation

Various visual techniques differ from one another by the number of dimensions a particular approach is capable of projecting. The ability to detect trends in Pareto front is another differing characteristic, i.e., how various variables, objectives, and constraints interact in various regions. In visualising information, a problem's dimensionality refers to the number of variables or attributes that are present in the data to be visualised. Traditional tools like tables and scatterplots can be effectively used for visualising one-dimensional data or univariate data, consisting of one or two attributes.

By plotting various visualisation techniques simultaneously, the user has an advantage of making use of essential features offered by different representations. Cross-plotting enables better understanding of trade-offs among the computed representations. A selected design attributes can be checked against the compromises made, in comparison to the best design according to design objectives. Trade-off plots help in visualising continuity or discontinuity allowing the designer to experiment between lower and upper bounds.

If a designer is interested in the continuity of the Pareto Optimal front, a chosen visual technique should be able to recognise and display the gaps in the Pareto front in order to choose a single design set or a few possible designs. To have an overall view of an optimisation problem, there is an advantage in combining all available methods and taking advantage of each method's capabilities as some visual approaches are more attractive to the user than others, also more easy to use.

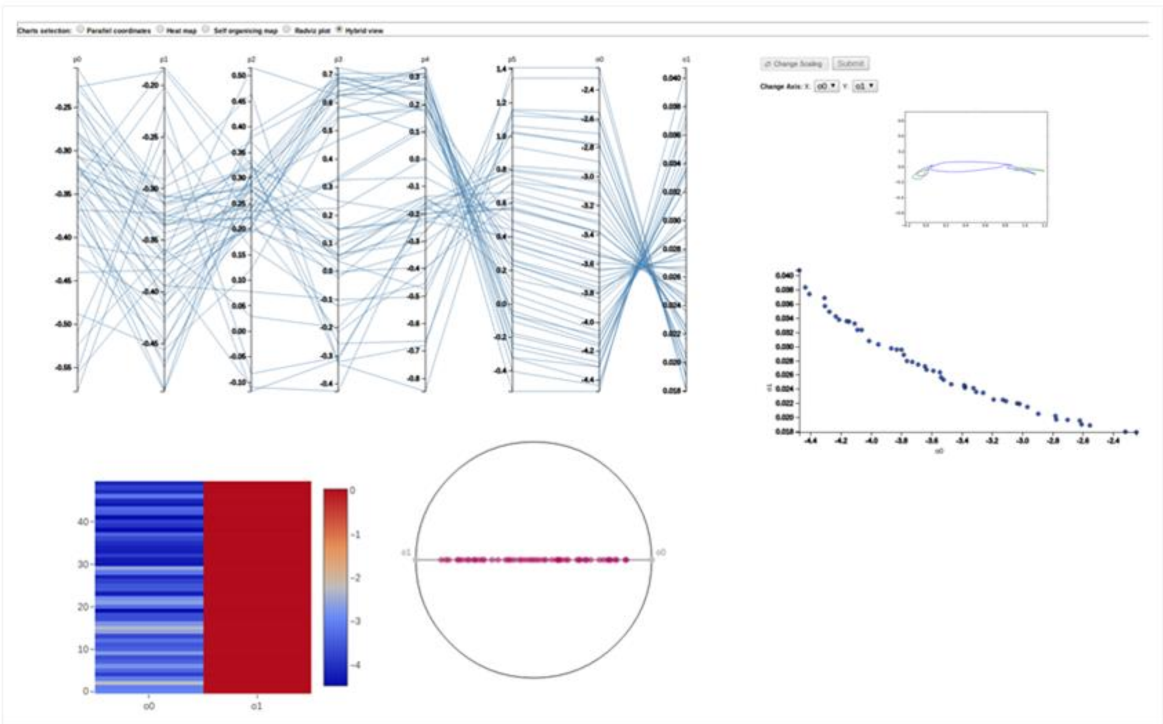


Figure 4.18: An extract of I-MOPSO module with the 'combined view' tab showing various plots on a single page; named as 'hybrid view' in the module

Radial visualisations consist of computed visual features along a circle, ellipse, or spiral. Several radial techniques could be considered as visual projections from a cartesian coordinate system onto a polar coordinate system. There is no specific known advantage of RadViz plots over other techniques. Unless there is a clear reason to favour a RadViz plot in terms of evaluating certain features, other techniques are better at showing convergence, patterns and perform better with respect to perception accuracy.

Scatter plots, parallel coordinates and heat maps are simple, easy to understand and compute; they do not require complex mappings of vectors and are therefore fast. Except for scatter plots, the others can easily be scaled in several dimensions. Parallel coordinates and heat maps allow visualisation of the decision space along with the objective space.

Self-organising maps perform sophisticated dimension reduction mapping to the 2D space. They are also scalable to many dimensions, relatively easy to understand and implement. However, they are computationally more expensive than other methods and every SOM generated might be different. They are not very robust as the mapping used for visualisation depends on the objective vector values in approximation sets.

4.6 FACTORS AFFECTING OPTIMA

Visualisation usually begins with ‘raw data’ generated as a result of initial from the data population process, and these data are often not suitable for direct visualisation in their generated form (Post & Van Walsum, 1993). One of the general factors affecting the intensive development of aerofoil design and analysis is the availability of high-performance computing and parallel algorithms. Choice of geometric shape, choice of flow control, meshing technique and settings, choice of optimiser determine characteristics of the problem under consideration.

It is not easy to develop various interaction technique categories that are clear and inclusive. However, (Bondarev & Galaktionov, 2014) point that digital revolution has had a tremendous impact on experimental dynamics causing a revolution in experiments due to transfer of digital technologies for image registration and visualisation of experimental results. This allowed for a direct comparison of experimental and computational results in aerodynamics.

4.6.1 Optimiser search pattern

The topology of the swarm defines the subset of particles where each particle exchanges information with neighbouring particles, also defined as swarm communication structure. Multi-objective Particle Swarm Optimiser (MOPSO) used in this work searches the objective space by dividing it into hyper-cubes which have even sizes. Each hypercube has a score which is inversely proportional to the number of non-dominated particles inside its boundaries. Roulette wheel selection is used to choose a non-empty cube; the global guide is then randomly selected from the chosen cube, which try to promote cubes with lesser non-dominated points and an even coverage of the Pareto front.

Swarm technique represents a type of randomised local search heuristics which maintains a population of solutions that evolve through a series of generations. At every step, the algorithm generates a new population of maximum size N (where N is a pre-specified constant) from the current state by using a set of randomly applied operators which allow erasing old solutions and create new ones.

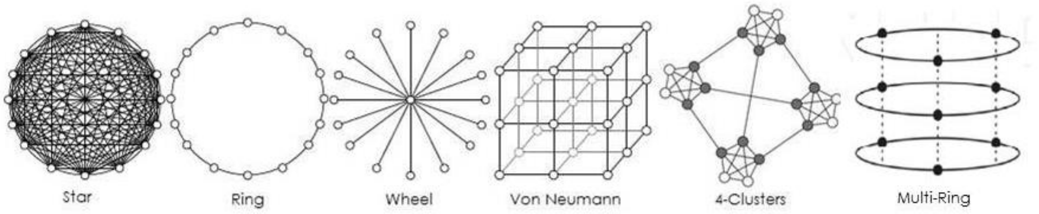


Figure 4.19: Different MOPSO topologies influence how particles share information with one-another

Many different solutions can be tested and modified at the same time; therefore such procedures offer an inherent parallelism. A suitably defined function defines and evaluates the quality or fitness of each solution in the current population. In several cases, the fitness of a solution corresponds to its measure.

Current swarm solutions are probabilistically selected to exist in the new swarm with respect to their relative fitness. New individuals are randomly generated as long as the swarm has size less than N (Cossio et al., 2003).

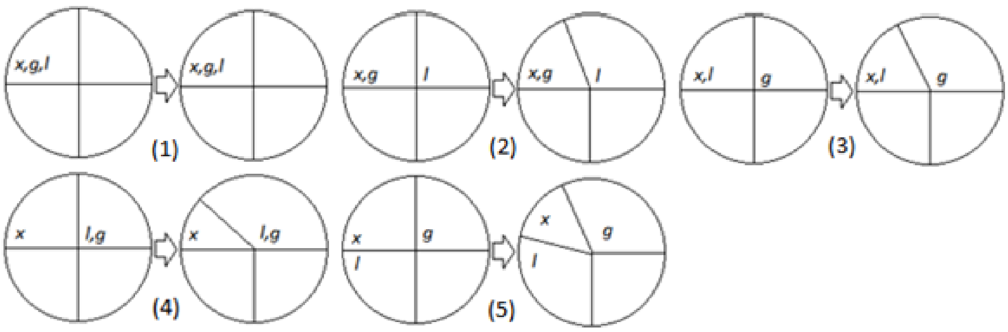


Figure 4.20: The ideal roulette probability updates (Smythe, 2012)

Tests were carried by (Smythe, 2012) to analyse the roulette's average probability update behaviour of the PSO. A check was done before and after probability updates, looking for set requirement violations. Five possible configurations were identified of the relative locations with respect to each other in a random space:

- The global best, local best, and current locations are the same
- The global best and current locations are the same, but the local best location is different
- The local best and current locations are the same, but the global best location is different
- The local best and global best locations are the same, but the current location is different
- The local best, global best, and current locations are all different

4.6.2 User friendliness of tool

Simple installation, easy updating, intuitive, efficient, pleasant with easy-to-navigate graphical-user interfaces, easy removing, less or no need of third-party software, easy troubleshooting, adhering to required standards and effective error handling are desirable characteristics of a software tool. User friendliness of a tool is user empowerment. Designer tool interaction control is a matter of choice and degree. It is possible to develop variable interfaces which can manage a variety of users and design situations.

Engineers are generally eager to invest time and effort to learn work and master tasks that are necessary for routine activities. Users don't seem to have much interest in learning to carry out infrequent tasks efficiently or effectively for long, because of the likelihood that the knowledge will be obsolete, replaced or forgotten. Usability, sustainability and maintainability are important characteristics of a user-friendly tool.

Better than employing automated techniques for solving a problem, designers could instead benefit from better visualisation tools which can aid in better design decisions. Over time, as designers build on a collection of information in the form of evaluations for certain data points, a visualisation system that will support in better managing the available information is advantageous (Shaffer, Knill, & Watson, 1998).

4.6.3 Decision-Maker knowledge and skills

Engineers are both agents of decisions and decision-makers all through their professional lives. Environments where engineers are active and situations requiring decision making quite often involve multiple criteria, imprecise and incomplete data, multiple actors and pressure groups.

An engineer is likely to be a better at professional activities if she/ he possess good knowledge, skills, combined with good decision making abilities and several engineering sciences, specialisations aim at improving these qualities and productivity. Psychological characteristics such as the ability to quickly comprehend a problem case through a broad perspective and to generate good solutions with limited resources impact an engineer's overall performance. Nevertheless, knowledge of the tools and their way-of-working can contribute significantly at improving an individual's or group's decision making abilities (Autran & Gomes, 2011).

A seven step decision making model has been complied by (Faraclas & Koehler, 2005) based on the heurist models found in literature: 1) Problem formulation 2) Defining Objectives 3) Solving the problem 4) Assess actions 5) Analysis and decision making 6) Analysis validation 7) Result communication. While literature proposes several models, none of these have obtained unanimous approval in practice; and most are replaced with the arrival of new approaches.

It is traditionally assumed that engineering only includes mathematics and sciences as a required qualification for the discipline. However, engineering now spans a wide variety of topics and knowledge of other areas of study is now a necessity to a successful engineer background and attributes. Together with mathematics and science skills, technical computer, social sciences, business and leadership skills also play an important role. It is equally vital to recognise early on how other disciplines have an integral part and influence the overall decision making process.

4.6.4 Personality Traits of Decision-Maker

Individual personality differences reflect one's outlook and behaviour patterns, and have a significant effect on performance; however, various study inferences are not straightforward. Personality effects are strongest when tasks are complex, under cognitively demanding situations, providing a window on how users differ in using a visualisation to support higher-level reasoning.

Five-factor model is a common personality psychology model with five dimensions: extraversion, neuroticism, openness to experience, conscientiousness and agreeableness. An individual can be categorised under these personality traits, and according to (Roberts & Del Vecchio, 2000), these traits remain consistent throughout an individual's adulthood.

Personality factors significantly correlate a designer's preference for visual interfaces and complex task performances. For example, an introversion trait is consistently positively correlated with both programming abilities and efficiency with computer-assisted instruction tasks. Individuals with more openness and spatial abilities may have an easier time switching between different design imageries, such as those found in a multi-view system (Ziemkiewicz et al., 2012).

An understanding of complex relationships among personality, visualisation and performance aids in the design of visual interaction interfaces, supporting user thinking.

4.7 COMMON VISUALISATION ERRORS & PROBLEMS

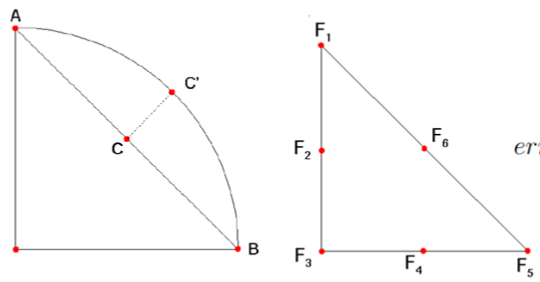
To be literate in graphics is the ability to understand graphically presented information which includes a general knowledge on how to read, extract information and make inferences from the various formats. Education and continuous skill development are required to interpret graphical information; both perceptual and cognitive processes play an important role. A major obstacle to graphical literacy is the widespread availability of high end software which most often fails to communicate clear and accurate information in a manner that is easily perceived by its user.

4.7.1 Particle Clustering

When doing multi-objective analysis of the big picture, a designer usually starts observing for meanings to be deduced from the clutter. A display with large data such as parallel coordinates is not advantageous in exploring the details, but it is very useful in examining dominant patterns and exceptions. Separating clusters of similar data into separate graphs is often useful to easily focus on specific sections independent of others and to compare multi-variate profiles.

Searching for entities with a specific multi-objective, multi-variable profile is another useful task when exploring multi-objective data, either one that is displayed by a certain entity, such a specific aerofoil design, or one that a designer perceives to be interesting. Effectiveness of visual techniques with large data output is reduced by visual cluttering, thus preventing effective revealing of underlying patterns in large datasets.

The primary cause of visual clutter arises from too many polylines (such as in parallel coordinate plots); (Zhou, Yuan, Qu, Cui, & Chen, 2008) points that most of the existing efforts to reduce clutter are data centric. Data are clustered before being plotted and the computed patterns are limited by the choice of algorithm used. Also, multi-dimensional data clustering itself is a difficult problem. The underlying algorithms, whether data or visual based, effect the geometric clustering while being plotted and define whether designers are able to control the levels of visual cluster with respect to preferences and further explore clustering results by enhancing colour or opacity. Interactions in visualisation significantly improve completion times of problem comprehension and altering model configurations.



$$err = max\begin{pmatrix} |F_2 - \frac{F_1+F_3}{2}| \\ |F_4 - \frac{F_3+F_5}{2}| \\ |F_6 - \frac{F_5+F_1}{2}| \end{pmatrix}$$

Figure 4.21: Error estimation: distance C-C’ is the geometric error for the deformed cell and field error is given by the shown formula (Brodlie, Osorio, & Lopes, 2012)

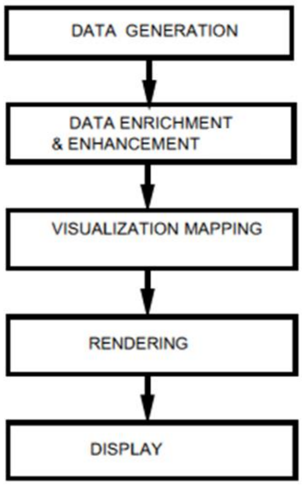


Figure 4.22: A Pipeline model of Visualisation Process (Post & Van Walsum, 1993)

4.7.2 Searching, Filtering & Extracting

When searching through data, the first step usually is filtering or reconstructing through interpolation or approximation, generating a model of the problem section’s underlying data. A visualisation algorithm produces geometry which is interpreted as an image. Uncertainty appears at all stages; visualisation of uncertainty focusses on the data stage, while the uncertainty of visualisation starts at the filtering phase and passes through to the rendering phase.

The method for reaching a target is determined by the way in which a visualisation task or action is carried out. Navigation includes all methods that alter the range or granularity of represented data such as by data browsing or searching; for granularity, this is realised by data elaboration or summary. It does not reorganise the data itself (Schulz et al., 2013).

Organisation and re-organisation include all methods that actually adjust the data to be displayed either by diminishing or enhancing it. Extraction methods like filtering or sampling and abstraction such as aggregation or generalisation are usual ways of data reduction.

The commonly used enrichment means are to gather added data from external sources to derive further metadata, data describing other data. A relation includes all ways that position certain data into context such as by looking for similarities via comparison, for differences by examining variations or discrepancies, or by seeking relations (Brodlie et al., 2012).

4.7.3 Colour Contouring

The numerical analysis of computational results convey a designer’s interested engineering parameters, but the graphical representation of those results explains why.

Colour is used in computational simulations in two key ways: (a) for geometry visualisation allowing engineers to be confident of the generated construction model being a good depiction of the problem situation and (b) colour is used in post-processing the data from simulations to illustrate the complex relationships for analysis and investigation (Kinnear, Atherton, Collins, Dokhan, & Karayiannis, 2006).

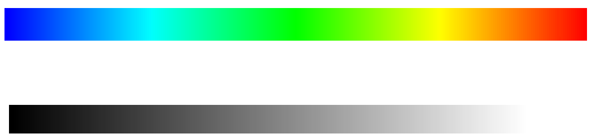


Figure 4.23: The rainbow and grayscale colour maps

Plots such as heat maps make use of many different colour schemes with perceptual advantages and disadvantages. A rainbow colour map which is commonly applied in data visualisations is based on the order of colours in the spectrum of visible light, expecting blue to mean low and red to mean high. However, this could confuse the user as there is no natural perceptual order of the spectral colours. Apart from causing visual confusion, this drawback of natural order could delay tasks because the user might refer to the colour key often to help interpret visualisations.

The human visual system has low spatial resolution in responding to colour variation when compared to variation in brightness (mathworks, 2017). Hue, saturation and brightness are the three basic characteristics of a colour. The green and cyan parts of the rainbow colour palate are not distinct to perceive, which creates the illusion of displaying data in the corresponding ranges appear to be uniform.

A same light stimulus could generate different colour appearances depending on internal and external aspects; colour is not a simple function of external stimulants. The health of the user's eyes and their gender affects colour vision raising a deficiency. A user with colour blindness will be unable to distinguish reddish and greenish hues since their eyes cannot construct red-green opponent mechanism. Confusion of greens, reds and yellows is usually hereditary and is a common type of colour blindness although other types also exist.

It is difficult for a person with normal colour vision to comprehend the visual world of someone with colour deficiency. Men are more colour vision deficient compared to women and genetics is a common cause. According to studies by British colour blindness awareness group, colour blindness affects approximately 1 in 12 men and 1 in 200 women around the world.

4.7.4 Brushing & Rendering

Multiple views are not of much use if the user is unable to interact with them. Brushing is an added technique with which a designer can select points to be examined. Brushing is generally carried out directly on the visible areas by using interfaces such as a mouse for interaction which could open a rectangular region of interest. It can also be carried out independent of displays such as by entering values, using sliders to select areas of interest, or by certain complicated means like selecting a cluster where the points inside are brushed allowing brushing possibilities in dimensions that may not otherwise be displayed clearly (Kosara, Hauser, & Gresh, 2003).

Composite brushes can be formed by combining brushes allowing the user to select complicated shapes and define the points of interest precisely. Without linked views, brushing by itself is of limited advantage. Linked views allow information exchange regarding points that are brushed, so the user can visualise easily the same points of data brushed in various views. It usually works from any view to all other views.

Rendering is a way of generating images from 2D or 3D models by means of computer programmes, transforming the software data into a picture; the results of such models are also referred as rendering. It involves a large number of complex calculations, keeping the computer occupied for a long durations, gathering data from sub-systems and interpreting data appropriate to features such as mapping textures, shading and lighting. Rendering involves user's choices which impacts quality and speed of data or images rendered.

Progressive rendering is the name given to techniques used to render content for display as quickly as possible. Reducing computing time continues to be a challenge in order to solve visualistaion problems at the rates of interactivity. Progressive rendering is a method where a close enough visual image is generated by discretely choosing a sample set. New samples may be selected to generate new images if the desired quality image is not reached; the process can be repeated till a desired quality is attained. The user has a burden of deciding the casting of new image samples; generally, the choices made depend on empirical criteria based on statistics.

4.7.5 Clutter & Reordering

A good visualisation reveals clearly the structures inside data, helping the designer to recognise patterns and detect trends. Clutter obsctructs the structure in visual displays which is characterised by congested and disordered visual objects. Clutter is not a desirable characteristic since it prevents a user's understanding of the displayed matter. However, it is inevitable for users to confront clutter when the dimensions or number of data items increase, irrespective of the visual method used.

The arrangement of data and dimensions has a considerable effect on visual representations, in revealing a data's various aspects and impacts a display's perceived clutter and structure. It is possible for entirely different conclusions to be drawn depending on the available displays.

Apart from dimensional effects on clutter, there could be other aspects of displays that impact a display's clutter or structure for various visual techniques; knowledge of them can aid users in interpreting visuals.

4.7.6 Discontinuities

Discontinuity is a one of the basic message in the simulation process, any functional value discrepancies provides details on solution convergence. When high accuracies are required by computational simulations for analysis, it is sometimes inevitable to deal with some geometrically deformed cells.

Common visualisation usually do not offer sufficient features to handle discontinuities, especially when discontinuities are combined with unstructured topology. Standard visualisation tools and approaches are optimised for studies when variables exhibit linear behaviour. In order to optimise interactive performance, linear approximation of functions for rendering purposes is one of the primary difficulties to be addressed (Brodlie et al., 2012). A clear, proper representation of discontinuities is an important problem yet to be resolved for displaying discontinuous fields.

4.7.7 Uncertain Information

When plots are used as visualisation tools, exploring potential relations takes precedence over presentation of facts. Uncertainty is a critically vital issue in data visualisation because most people tend to perceive both data and computers as being less fallible than the humans who make decisions for them. Most visualisation methods have been developed on the assumption that the represented data is free of uncertainty and that the data being displayed is exact. Yet, this is rarely the case. In most generic visual representations of uncertainty, much priority is given to data of most uncertainty.

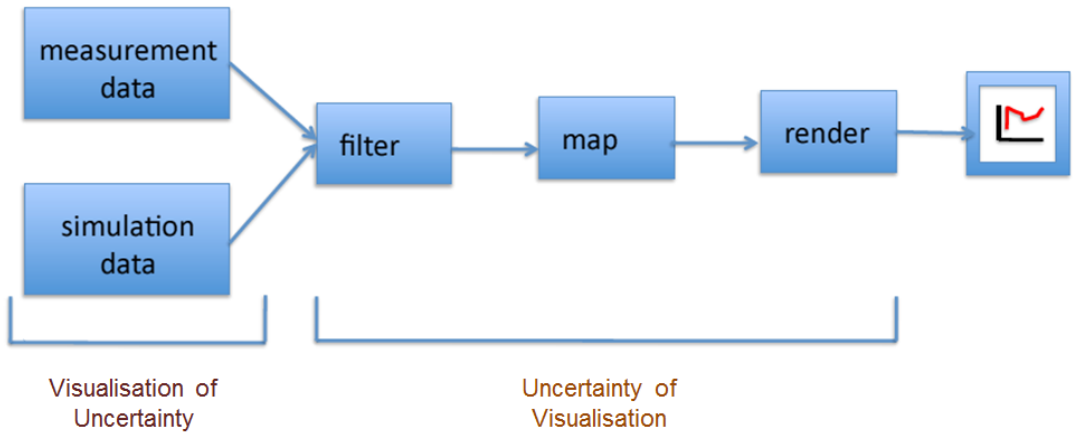


Figure 4.24: Haber and McNabb model: visualisation of uncertainty and uncertainty of visualisation (Brodlie et al., 2012)

Spatial aggregation leads to certain visual variations; in addition, aggregating attribute values also adds to this variability, and in turn increases uncertainty. All data are categorised and individual measurements are retained in the database; accuracy of these measurements are impacted by their mathematical precisions to a large extent (MacEahren, 1992). Components of computer graphics, hardware and software inevitably introduce error. This has been the norm since early graphical plotters to present day graphics processing units; they allow approximating curves and surfaces by straight lines and triangles (Brodlie et al., 2012).

The uncertainty level of mapping to locations depends on the quality of values, variance of the mean values which usually represent units, and a unit's spatial variability. When approximations are made over the test case body, their relative representation as lines and points remains weak.

Lines and colour blocks can alsobe visualised as a series of shorter segments. This might usually generate good results in 2D, but difficult in 3D without further depth cues. Also, only a limited number of representations can be displayed without any user confusion (Post & Van Walsum, 1993).

4.7.8 Image Discontinuity

Identifying uncertainties in visuals often depends on the user's visual capabilities. Ability to recognise an image's discontinuities and to interpret them as sections with specific data features is up to the decision-maker. Visualisation techniques that communicate discontinuities rely on blurring, texture, translation, scaling, rotation, warping, and distortion of images that represent computed data.

Making use of animation features could highlight regions of interest, distortion, blur or enhance differences in visualisation parameters. A few basic animations have been applied to the module used in this work such as highlighting, colours and linked user interactions.

4.7.9 Visual Selection

The goal of a visualisation task according to (Schulz et al., 2013) is the motive with which the task is carried out and the three high-level goals are: exploration, confirmation and presentation.

Exploration analysis is an undirected search and draws assumptions from an unknown dataset. Confirmation analysis aims to examine identified or assumed hypotheses regarding a dataset. Used analogously, sometimes an undirected search is also a directed search. Presentation is a description and exhibition of confirmed analysis results. The goals point to the motive of a task's actions and not the action itself, such as searching, extraction, filtering or sampling data and they are independent. A similar motive can drive diverse actions and a same action can be carried out for diverse motives.

Data + Task = Visualisation

This combination asks which visualisation is best suited to pursue a given task on given input data. It caters directly to the **visualisation design**

Data + Visualisation = Task

This combination asks which tasks can be pursued and how well on a given visualisation for a given dataset. It caters directly to the **visualisation evaluation**

Figure 4.25: Relationship between tasks, input data and visualisation for design and evaluation (Schulz et al., 2013)

Various tasks are affected by input data relations and the visual representations used. Different aspects of visualisation can be selected or parameters modified when the user has knowledge of the task. This could be a generic steering of the overall visualisation design or specific approaches for individual aspects, such as selected mappings, colour scales or searching for certain data-task combinations. A machine recommendation of visualisation techniques to the user is still an area of further research.

As most design dimensions and variables are customisable, visual selection options offer fundamental opportunities for working with tasks. The design space as a whole is usually fixed but individual preferences could be added or further subdivided at various levels as desired to attain a good level of granularity for specific applications or tasks at hand. The design space could capture entire tasks, whether abstract, specific, or even unusual.

4.7.10 Reading Graphs & Data Interpretation

Graphs and charts are used to illustrate information and it is vital to be able to interpret them correctly. The primary purpose of scales is to permit one to read values from the graph at any chosen point.

Before conducting an investigation into a problem case, it is important for the user to be aware of how to organise collected information or generated data. By organising data, one can conveniently interpret what has been observed. A data table organises data into rows and columns, while graphs are generated from data tables. Ideally, they should be self-explanatory and a designer should be able to understand them without detailed references to other information to aid support but this is not the case most times.

It is desirable to keep numerical values shown in a table or graph to be as simple as possible, while also have adequate useful details and information to support. It is vital for summaries generated by tables or graphs to display clearly what was calculated to avoid uncertainty in interpretations and units. It is also helpful for the user to have information regarding where the data was gathered so that the extent of the coverage is clear, explicit time periods and if any data was borrowed from elsewhere (SSC, 2000).

Graphical representations allow visual images of a decision-maker's observations, simplifying interpretations and aid in reaching conclusions. Drawing conclusions is a final step of a design investigation, its accuracy depends on well organised and clear interpretation of data involving tables and graphs (CSEF, 2016).

4.7.11 Graphs Vs Tables

In choosing between a favourable choice between tables and graphs for data visualisation, the answer usually depends on the audience or the person and how that data is meant to be used. Individuals interact very differently with these two types of visuals.

Tables, with their rows and columns of data, interact primarily with one’s verbal system. Tables are read, scanned across rows, columns to compare various values. In terms of communicating structured numeric information, tables are better than graphs while graphs are good at displaying trends, show relationships and assist in making comparisons. Tables are not effective for multi-disciplinary visualisation, while graphs, on the other hand, show how different variables relate to each other. Graphs are a high bandwidth information flow from what one’s eyes see to the comprehension in one’s brain, which can be very useful when done well.

4.8 VISUAL INTELLIGENCE

There are still several issues regarding visualisation techniques such as distributed processing, managing data, standardising of data formats, user interface and interaction that call for more study and exploration. Several combinations of style and mode of interaction are possible, depending on the problem case, simulation process or extent of visualisation control (Hearn & Baker, 1991).

The interaction style offered by general purpose visualisation codes enables the user to create data flow networks, allowing alteration of visualisation process, they do not impact objects representing the data. (Post & Van Walsum, 1993) state that the concept of direct manipulation of data requires more design clarification, including interactive probing and interrogation.

A human mind is a creative genius, and although it may come across as effortless, it far outstrips the most valiant efforts of supercomputers. The mind only has to see. In a fraction of a second, a human’s visual intelligence can construct the objects, patterns, images, colours, or any of countless other scenes of such subtlety and complexity, far surpassing any advanced computational technology in recollection, repertoire and speed (Hoffman, 1998); intuition also works in a similar way. When dealing with unsure and ambiguous areas, a human has an advantage over algorithms; they are able to draw parallels quickly to understand and develop insight on new situations.

Any design framework requires a cumulative discussion at various levels where the designers are able to discuss the framework together in terms of:

- Benefits
- Drawbacks
- Applications

The design space of visual representations is not yet exhausted; however it can be noticed that it is increasingly becoming difficult to develop entirely new visual techniques and representations which will significantly extend and alter the already existing ways of such representations in a field. It is also obvious that there is, in general, no visual technique that is clearly superior for a given problem case or dataset; all visual approaches have their advantages and disadvantages (Javed & Elmqvist, 2012). In order to balance these strengths and weaknesses, efforts have been made in combining different visualisations. This also gives rise to novel visual representations which can be generated by combination of the existing ones.

4.9 DISCUSSION

This chapter primarily dealt with visualisation in the context of multi-objective optimisation. It recognises the importance of visual support system to the aeronautical engineer and the several advancements made till date and those that are projected in future.

Various phases and spaces of interactive visualisation have been presented. Scaling and translation of particles in I-MOPSO module has been introduced; a further explanation of the module is covered in the next chapter. Selected visual techniques that were primarily used in this research are introduced and their characteristics explained. In searching for an optimum solution, various factors that affect this search have been discussed which are both machine and human related.

Common errors and problems faced, both cognitive and perceptual have been presented. Usefulness and drawbacks of various tools in view of decision-maker and visualisation of data, tasks have been presented. A practical analysis of these techniques is presented in the following chapter as part of trade studies.

5

Optimisation Trade Study

5.1 INTRODUCTION

While optimisation is a helpful technology, it is only one of the available methods in the toolbox of design exploration. Design optimisation algorithms have two sections: codify problem formulate and solution convergence, assuming that a problem is formulated before optimization search and aims at a convergence. A design exploration strategy establishes that a problem formulation evolves during the search process and converges (Figure 5.1), ultimately leading to an informed optimal solution. Optimisation thus is both divergent and convergent.

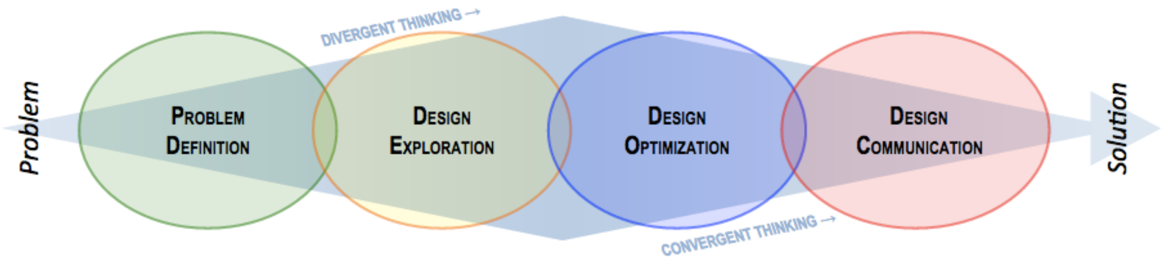


Figure 5.1: Four Phases of Engineering Design Process Showing Divergent & Convergent Methods (Johnson, et al., 2015)

Optimisation depends on a good optimisation problem set up or formulation, it generally involves the following, and are usually multidisciplinary in nature:

- an objective function
- constraints, variables
- the expression of user preferences

The formulation of an optimisation problem predefines the optimum solution, allowing the optimistaion algorithm's mathematical logic to generate optimum solution through an iterative search. On the other hand, design exploration, assumes that the optimum design is initially unknown and not characterised. The process of design exploration eventually discovers design conditions through various forms of experiments and defines an optimal design. After this step, a decision on final solution can be found through a convergent design optimisation algorithm (Jenkins, 2014).

A problem formulation and good design according to (Will & Perng, 2011) are as follows:

- Identify key design parameters
- Identify the variation of design performance with respect to design parameter variations
- Make the right decisions based on the right information with the appropriate tools

5.2 SWARM INTELLIGENCE

Swarm intelligence is a commonly shared behaviour emerging from social insects working under very few rules. Self-organisation is the main theme with limited restrictions from interactions among agents (Figure 5.2). A swarm should be capable of responding to altering factors and able to easily compute related to its surrounding environment. Resources are generally not concentrated in specific regions but distributed and swarms should be able to adapt to fluctuations.

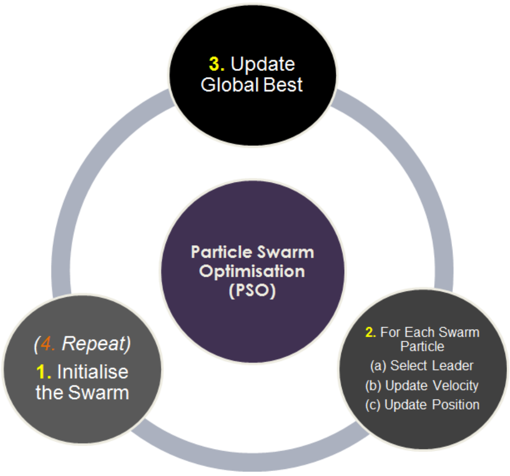


Figure 5.2: Basic Particle Swarm Optimisation Cycle

Algorithms based on swarm intelligence techniques find optimal values follow the pattern of animal groups who have no leader. Particle swarm optimisation consists of a swarm of particles, where a particle represents a potential solution, a better condition. Particles move through a multi-dimensional search space to find the best position in that space, maximum or minimum values (Figure 5.3).

Particle Swarm Optimisation (PSO) is a population based heuristic based on the social behaviour of birds within a flock. In a PSO, algorithm of each potential solution to the problem is called particle and the population of solutions is called swarm. The way in which PSO updates the particle x_i at the generation t is through the formula:

$x_i(t) = x_i(t - 1) + v_i(t)$
where velocity factor $v_i(t)$ is given by:
$v_i(t) = w .v_i(t - 1) + C_1 . r_1 . (x_p best_i - x_i) + C_2 . r_2 . (x_g best_i - x_i)$

In this formula, $x_p best_i$ is the best solution stored by x_i; $x_g best_i$ is the best particle, also known as the leader that the entire swarm has scanned, w is the inertia weight of the particle and controls the trade-off between global and local experience, r_1 and r_2 are two uniformly distributed random numbers in the range $[0, 1]$, and C_1 and C_2 are specific parameters which control the effect of the personal and global best particles (Durillo et al., n.d.).

Particles move through the solution space, and are evaluated with respect to certain fitness criterion after each time-step. Particles are accelerated over time towards those particles within their communication grouping which have better fitness values. The advantage of such an approach over other global minimisation strategies is that the large number of members that make up the particle swarm make the technique impressively resilient to the problem of local minima.

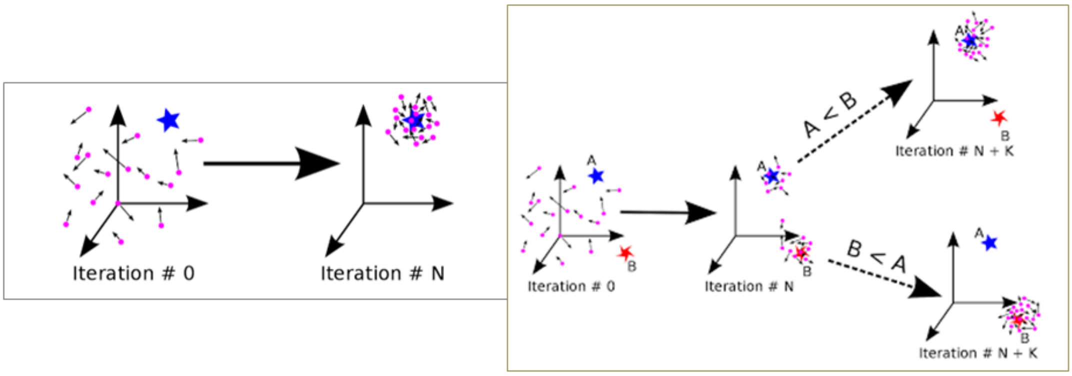


Figure 5.3: PSO is a minimisation optimiser. Swarm moves towards point A or B depending on the decision-maker's selection of target search point

Although there are several multi-objective evolutionary algorithms (MOEA's) available in literature, popularly used are genetic algorithm based NSGA-II and particle swarm intelligence based multi-objective particle swarm optimisers (MOPSO's). Particle Swarm Optimisation (PSO) is now a very well established optimisation technique used in a variety of fields and contexts. It is similar in some respects to other evolutionary algorithms, except that the potential solutions/ particles move rather than evolve through the search spaces.

PSO consists of several candidate solutions called particles each of which has a position and velocity, and experiences linear attraction towards two factors:

- Best position attained so far by that particle (particle attraction or personal best)
- Best particle attractors in a specific neighbourhood (neighbourhood attractor or global best)

Multi-objective PSO (MOPSO) has become popular due to its easy implementation, population based approach, successful handling of continuous search spaces and concepts of individual position and velocity.

5.3 INTERACTIVE MOPSO

Interactive MOPSO (I-MOPSO) uses interactive algorithmic extensions allowing the decision-maker to interact and present preference information at different times during the optimisation process; specifying any unacceptable values or goal levels and restricting the solution space to be searched. It has been test run using the computational solvers, X-foil for single-element and MSES for multi-element aerofoils.

Preferences are captured in the form of preferred ranges in the parameter space, depending on the decision-maker's ability to find desirable solutions and guide the MOPSO algorithm accordingly. Figure 5.4 and 5.5 present the optimisation flow chart. The user interface is mainly by visuals, using parallel coordinates, scatter plot, heat-map, self-organising map, radviz and a combined view: they display parameters and objectives of known non-dominated solutions. When using parallel coordinates, the decision-maker can select any two axes to visualise in a scatterplot. A visual inspection of presented candidate solutions could be examined by clicking on a solution in either plot, which also highlights the point in the other visualisation modes. Such inspection can also be carried out in other plot displays. Preferences act as guidance information; the underlying algorithm also makes use of virtual guide particles to search selected regions with few or no solutions.

The user interface has evolved as part of Interactive MOPSO module and provides a tool for the decision-maker to use linked visualisations to study the known, non-dominated solutions and analyse preferences using various approaches in the parameter space. While visualising results, the set of computed solutions displayed in plots is limited to the present known non-dominated set by default. Most of these features can also be found in several post analysis tools, however their target functionality has not been directed towards aerodynamics analysis and also do not involve decision-maker in interactive optimisation loops. The interface tries to facilitate problem discovery and decision making in a easy and flexible manner. One of the benefits of such visual presentations is high dimensional spaces represented in an easy and intuitive manner. As is the case with many software products, there is also much room for the tool's development to add and enhance interface features.

Parallel coordinates' plot are capable of displaying the entire space in one graph, and so is the user's primary interface to interact and exploit the data. Scatterplots, heat-maps, self-organising maps and radviz plots are available as additional representations of the data. Scatterplots allow any number of dimensions to be plotted against each other per plot, this feature is also available in combined view for self-organising map and radviz. Hybrid or combination view is also made available for the user to view computed output on a single page.

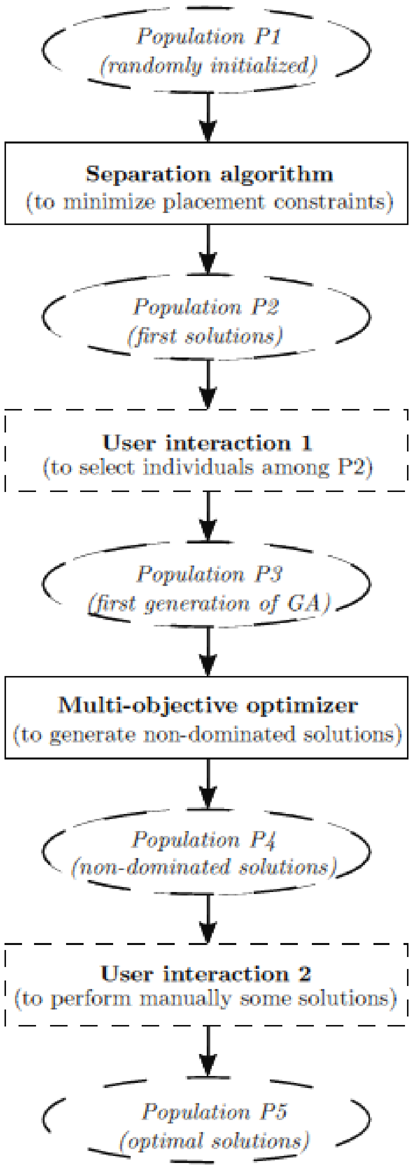


Figure 5.4: Schematic representation of the interactive optimisation strategy (Benabes, et al., 2010)

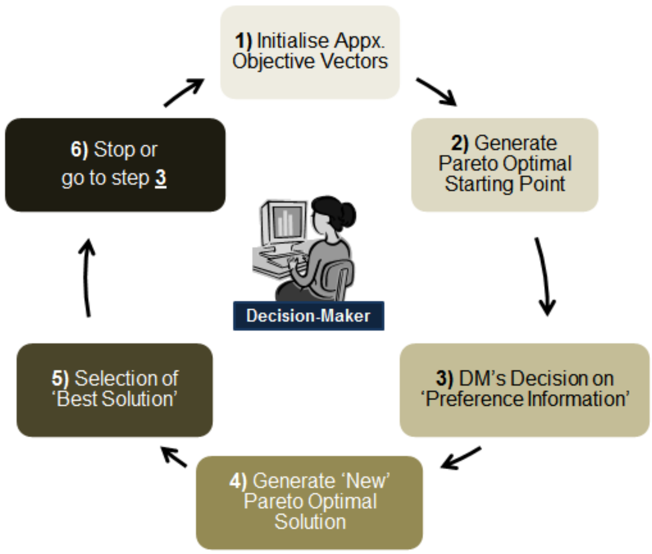


Figure 5.5: Generalised interactive optimisation flowchart for I-MOPSO

5.3.1 I-MOPSO Algorithm

A particle is a collection of all potentially possible solutions and a swarm is a populace of all solutions. Any one perfect solution or a unique global, personal maximum does not exist for multi-objective problems. At first, the algorithm initialises the swarm generating a set of non-dominated particles which are generally saved in an external file; a set of leaders is selected from the set of non-dominated particles. One or more leader particles are selected to guide the swarm in accordance with its respective topology. The algorithm adapts ‘turbulence’ which is a mutation operator to the particles. Later, every particle is assessed so as to refresh its related *p-best* and *leader*.

The process is repeated till the given maximum number of iterations is executed; a quality measure of the set of leaders is also assessed repeatedly. The algorithm maximises the number of Pareto optimal set elements and their spread and minimises the assumed distance between generated Pareto front and the real particle (assuming the location is known). Selection of leader, storing non-dominated solutions and diversity of swarm influence a MOPSO algorithm and there are several ways available depending on the user’s preference and choice of implementation.

I-MOPSO algorithm is run according to the following steps:

1. Initialise swarm population and velocity
2. Fitness evaluation and Pareto dominance for ranking particles (solutions)
3. Store memory: Personal Best (P_{best})= Swarm Population (and their ranks)
4. Store non-dominated solutions in external archive
5. Particles select ‘global leaders’ from external archive
6. Compute PSO equation
7. Fitness evaluation and Pareto dominance for ranking particles (solutions)
8. Apply mutation and perturbation
9. Update Personal best
10. Maintain external archive
11. Go to step 5 if stopping criteria is not met
12. Report solutions from external archive (pareto front)

Particle Swarm Optimiser (Pseudo Code)

```
Initialise Swarm ( )
Initialise Swarm Population/ Leaders Archive ( )
Determine Leaders Quality ( )
Set w, C1, C2, Max No. of Iterations (Gmax)
Start
While G < Gmax
  For Each Particle do
      Select Global Leaders from Archive ( )
```

```
        Select Personal Best ( )
        Update Velocity and Position (Flight) ( )
        Mutation/Perturbation ( )
        Evaluation ( )
        Update Pbest ( )
    End For
  Update Leaders Archive ( )
  Determine Leaders Quality ( )
  Generation ++
  End While
  Update External Archive ( )
```

5.3.2 Structure & Visual Interface

I-MOPSO structure provides the user with knowledge of search space structure and available solutions, thus lessens the inconvenience of expressing preferences without any previous knowledge. Computational costs are further reduced by using guidance to the search and by avoiding search regions that are of little interest. The understanding of decision-maker regarding design space is enhanced as optimisation becomes an assisted design tool.

The framework for I-MOPSO was originally developed by (Hettenhausen et al., 2013). A swarm of N particles is generated by the algorithm and stays consistent throughout the evaluation runs. The velocity equation governs swarm behaviour of the particles; it depends on the previous weighted velocity and two added factors, P-Best and PG-Best (Personal-Best and Personal-Global-Best). These factors present the knowledge of good solutions concerning the particles and the entire swarm, otherwise designated as the cognitive and social components. The particles are represented by non-dominated solution's archive in the Algorithm. PBest is particle specific and contains only non-dominated solutions found by the particle, containing either one or more points depending on the implementation. PG-Best holds non-dominated particles detected by the entire swarm. The position of the particle gets updated based on the velocity.

The algorithm operates based on choices articulated in the parameter space instead of the commonly used objective space. The approach is hinged on a visual representation of decision and objective spaces to effectively make use of human decision-maker's reasoning and knowledge of domain. The user interface is vital in interaction. The decision-maker reviews the progress of the test runs at regular intervals through the user interface options, having choices to specify a set value, delete or adjust boundary constraints in the decision space.

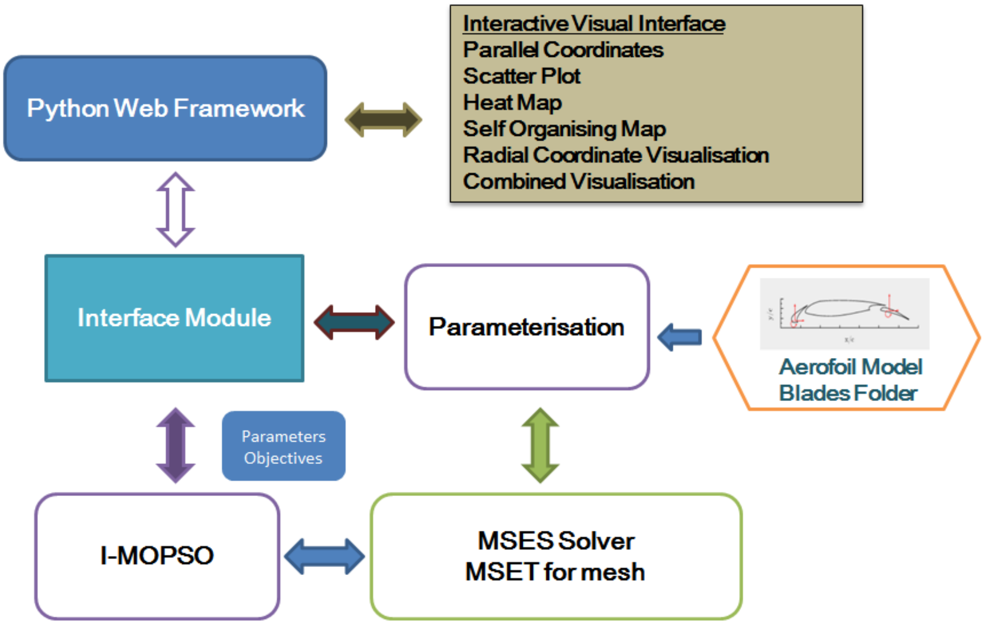


Figure 5.6: Structure of interactive MOPSO and visualisation module

The user interface comprises of the following visualisations:

- Parallel coordinates plot
- Scatter plot
- Heat Map
- Self-Organising Map
- Radviz
- Combined visual page

The first form of visualisation used in this module was the parallel coordinates plot (Hettenhausen et al., 2013b). The aim of other plots was to serve as an aid in the user's perception, exploration and decision making. Parallel coordinates offer a very good visualisation solution to see and comprehend high dimensional datasets. When used on multi-objective optimisation problems, the vertical axes represent each dimension of the parameter space and objective space with the ranges of each axis being independent of one another.

Every generated solution is presented by a line which connects and crosses each axis at the respective, appropriate value. For every axis, the user is able to select a range, depicted by a grey bar. On moving this grey bar, solutions are either highlighted or greyed out depending on the movement and solution values. This is an attractive way of visually analysing correlations between parameter space and objective space along with various influencing factors.

For example, if a certain combination of lift and drag are of interest to the decision-maker, an appropriate range for those two objectives can be selected. This then highlights the related solutions and parameters whose objectives are within the selected ranges in objective space. Such a comparison between highlighted and not-highlighted areas often presents an insight to the decision-maker into parameters, showing those that strongly match with desirable solutions and those that have a lesser influence. Based on preferences of the decision-maker, the interactive plot also suggests the directions in which an algorithm could be steered to further improve the approximate Pareto front. Apart from range selection in objective space, parameter space ranges can also be selected.

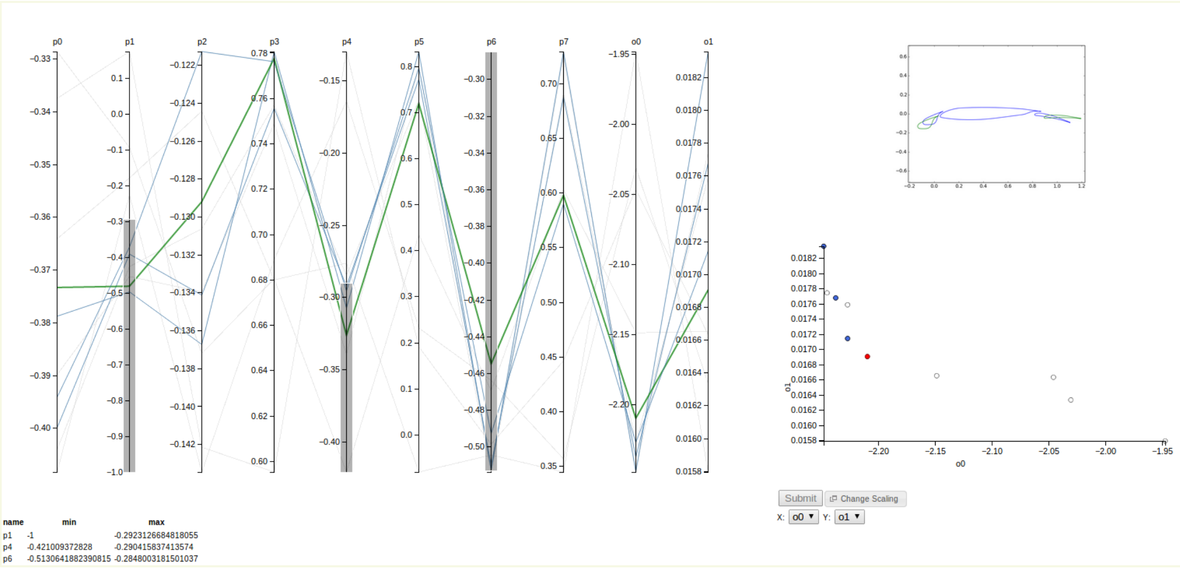


Figure 5.7: Typical user interface of I-MOPSO module with interactive optimisation framework. Optimisation problem minimises real function by choosing input values from within an allowed set, displaying o0 value as negative C_L

A Range is selected in the in objective space by the decision-maker considered advantageous. Values with lower or higher values depending on the objective functions are chosen; in some cases, the selection may be restricted by choosing a certain range for the first selected objective; highlighted and greyed-out solutions are compared. For example, the higher values of parameter six (P6) are connected to the solutions in the desired range of objective. After P6 range selection, and getting off O1 range, it can be visually noted that this selection provides insight information for next iterations. Such visual interpretations could be somewhat ambiguous and possibly one or more parameters need to be aligned to generate desired selections. If the decision-maker believes that exploring certain parameter or objective ranges in certain directions could lead to improvements, those ranges can be selected and the swarm will correspondingly respond (Hettenhausen et al., 2013).

By individually selecting specific, choice result points or solutions, the decision-maker can improve their understanding of selections and also enhance exploration abilities. The above visual view allows points to be selected either in the parallel coordinates plot or in the scatter plot. The corresponding points are then highlighted in both representations. When incorporating a scatter plot with parallel coordinates plot, heat map and self-organising map, any two dimensions can be chosen for the scatter plot. Additionally, the aerofoil shape, a visualisation of the actual design relating to selected points is also shown as a representation. This representation may not necessarily offer assistance for all optimisation problems; it aids the decision-maker to study suitable sections in a more discernible way. This may also generate design results that may be impractical as final, workable solutions but gives an opportunity for the decision-maker to explore and evaluate simulations.

5.4 COMPUTATIONAL SET-UP

Fundamental software input settings of I-MOPSO module are similar to those used for preceding research (Hettenhausen, et al., 2013)**,** (Tilocca, 2016). Although several new features were added to the module, most calculation code settings are unaltered in order to maintain configuration and facilitate ease of use, building on existing tool and integrating new processes with as little change as possible. A detailed description of the set-up is presented in the following sections.

5.4.1 Input Programme

AIRSET is an interactive programme that carries out geometry editing and manipulation; it reads blade.xxx file and allows geometry customising. It is able to manipulate the number of points defining a profile, modify contour, and other plotting functions. Separation-point definition is a valuable feature which allows manual selection of point where air separation is likely to occur. MSET identifies the setting and mesh is rearranged for separation bubbles.

Blade.xxx file contains input information corresponding to aerofoil geometry and grid; it can be generated through AIRSET programme for manipulating aerofoil geometry, or by using MSET initialisation programme. The first line of the blade file includes geometry name, such as ‘Garteur’. The second line indicates domain definition, the inlet and outlet coordinates. Four values are considered: Xinlet, Xoutlet are horizontal coordinates and Ytop and Ybottom are vertical coordinates as is common to several computational simulation problems. As aerofoils are formulated based on polynomial expressions, it is easy to calculate aerofoil surface coordinates. In the lines that follow, the co-ordinate points of the multi-element aerofoil geometry are listed; the geometry sections or elements of the aerofoil are separated by [999; 999] co-ordinates.

Gridpar.xxx file stores the parameters for mesh generation. Gird parameters can be stored in this file when a mesh is generated with MSES and similar configuration used for other test runs. This is a useful feature either when running several evaluations on same geometry or with various geometry sections to maintain coherence.

Mdat.xxx file is created by MSET representing the test case and holds data regarding each cell. While Gridpar file contains grid generation parameters, the Mdat file consists of the mesh itself. It serves as a file for MSES solver for input and output. MSES updates Mdat file after each iteration with the most recent dell data. This also serves as input information for post-processing stage, for MPlot programme.

5.4.2 Input File

The module allows the user to plug different aerofoil models via a text input (.txt) file. Before the start of the optimisation, data from the input file is gathered by the code into the framework's various scripts. The input file used in this module has three parts:

- General
- Initialisation
- MOPSO

The user specifies input test geometry, number of parameters (6) and objectives (2), and the geometry section's directory (with input file's name). Transformation factors are calculated automatically; the choice parameter interval can be mentioned directly. The scaling factor for the objectives is beneficial to present a normalisation or for reversing the sign of one or more objectives. As the MOPSO code executes only minimisation solutions, this factor aids maximisation of objectives.

Choice can be made between Gaussian and Uniform distribution through the initialisation section. Initialisation parameters such as mean, standard deviation, interval can be modified. Parameters of MOPSO algorithm can be modified via Input file, they are:

- personal and global weights (C_1,C_2)
- inertia weight (W)
- Archive size
- Standard deviation of turbulence mutation factor

5.4.3 Script Runner

The script runner acts as the link between geometry model and simulation framework; it permits information exchange with various models.

Each set of parameters (particle) is imported by the function run_model which transforms parameters to take on assigned intervals from the normalised interval. The geometry is then evaluated generating the standard-output and standard-error files. The user can carry out any additional operations with data.

```
------------------- -GENERAL - -----------------------------------------------------------------
-(1) PARAMETERS:    "6"
=>  number of parameters -> EX: "6"

-(2) OBJECTIVES:    "2"
=>  number of objectives -> EX: "2"

-(3) DIRECTORY: "/server/par-mses-evaluator"
=>  directory of the model/solver

-(4) NAME:  "par-mses"
=>  name of the model/solver executable -> EX: "model.exe"

-(5) MAX PAR:   "0.1 0.1 20 0.1 0.1 20"
=>  maximum values ("+1" by default - if not desired leave '0' as default) -> EX: "4 0.5 -2 3 1 4"
    (please separate the numbers using a space)

-(6) MIN PAR:   "-0.1 -0.1 -20 -0.1 -0.1 -20"
=>  minimum values ("-1" - if not desired leave '0' as default) -> EX: "0 200 200 0 -200 -200"
    (please separate the numbers using a space)

-(7) SCALE OBJ: "-1 1"
=>  objective scaling factor (optional - if not desired leave '0' as default) -> EX: "-1 1"
    (please separate the numbers using a space)
```

```
-------------- - INITIALISATION - ----------------------------------------------------------------
    the following parameters affect the initialisation in the NORMALISED interval [-1,1]
-(8) INITIALISATION: "g"
=>  choose the initialisation method: 'g' for Gaussian, 'u' for Uniform

-(9) UPPER BOUND uniform: "0.02"
=>  upper bound for uniform initialisation - you can insert a different distribution for each parameter
   for fixed UPP_BOUND (all par.) type: "value" - EX: "-1" - in alternative specify each par. (like in MAX-PAR)

-(10) LOWER BOUND uniform: "-0.2"
=>  lower bound for uniform initialisation - you can insert a different distribution for each parameter
   for fixed LOW_BOUND (all par.) type: "value" - EX: "+1" - in alternative specify each par. (like in MIN-PAR)

-(11) MEAN gauss: "0"
=>  gauss mean for Gaussian initialisation - you can insert a different distribution for each parameter
   for fixed mean (all par.) type: "value" - EX: "0" - otherwise specify "value1 value2 ... valueN"

-(12) SIGMA gauss: "0.2"
=>  gauss sigma for Gaussian initialissation - you can insert a different distribution for each parameter
   for fixed sigma (all par.) type: "value" - EX: "0.2" - otherwise specify "value1 value2 ... valueN"

-------------- - MOPSO - -------------------------------------------------------------------------
-(13) C1:   "2"
=>  personal weight

-(14) C2:   "2"
=>  leader's weight

-(15) w:    "0.4"
=>  inertia weight -> EX: "0.4"

-(16) SIGMA-TURBULENCE: "0.02"
=>  stardard variation for the Turbulence correction -> EX: "0.05"

-(17) Archive size: "50"
=>  Size of the archive of non-dominated particles
```

Figure 5.8: Shown here is the I-MOPSO input file containing the data relative to the aerofoil trade study

Under default settings, the code only verifies whether the standard-error file is blank. If no errors have been gathered during model execution, the standard output file presents the results. By default, the final results are expected in [flag o0, o1, o2, o3, …oN] format. The code searches standard-output, the line having the string 'final results', and reads a list of values containing a flag followed by the number of objectives, 'N'. The flag has two values denoted by 0 or 1:

- 0: Invalid result
- 1: Valid result

The results are invalid if the code reads the flag = 0; a default value of 100 is assigned to all the objectives so that the invalid solutions are kept away from valid points N of the objective space. This value could be modified depending on the decision-maker. If the flag reads 1, the code carries out further filtering to check composition of real numbers and if the objective-list size corresponds to the specified number of objectives.

For I-MOPSO's particular two objectives' problem used in this work, the code verifies for the objective ratio not to be infinite. After scaling the objectives, if any error is detected, that particular particle is treated as invalid. The decision-maker could overwrite errors; this does not affect the modularity but aids user convenience.

5.4.4 MSES Solver

MSES, developed by MIT faculty, is an assortment of various programmes for analysis and design of single and multiple element aerofoils; the package used here, version 2.9 is a composition of several programmes, each with its specific functionality. The input and output programmes rely primarily on the data files; a user interface is available. MSES is a low-fidelity estimator, employs an Euler solver for the outer flow and an integral boundary layer, similar to that of XFOIL, for the viscous layer.

As an Euler method, it is able to predict the aerodynamic characteristics of aerofoils in the transonic region. MSES can predict transition using a full e^N method, in which a Newton iteration method is used to find the critical Tollmien-Schlichting frequency, or by means of the approximate envelope e^N method. The increased capabilities of MSES result in it not being as easy to use or as robust. In particular, the programme requires considerably more run time and convergence can be problematic.

Two input files are taken into consideration, mses.xxx and mdat.xxx; the latter is constantly updated with latest available solution for every iteration. The maximum number of iterations are specified by user. The simulation comes to a stop if the maximum number of iterations is reached or when convergence is achieved.

MSET is a built-in programme which initialises the grid, flow-field, and a variety of other variables. Blade.xxx is the input file. The user can manually specify inlet flow angle, which initialises the streamline, so that the expected mesh will be adjusted with the flow. The grid can be initialised depending on the specified spacing criteria and grid parameters.

MPLOT is the solution plotter displaying the results, convergence or non-convergence of the obtained solution. It reads Mdat.xxx file and offers various interactive plotting options to the user.

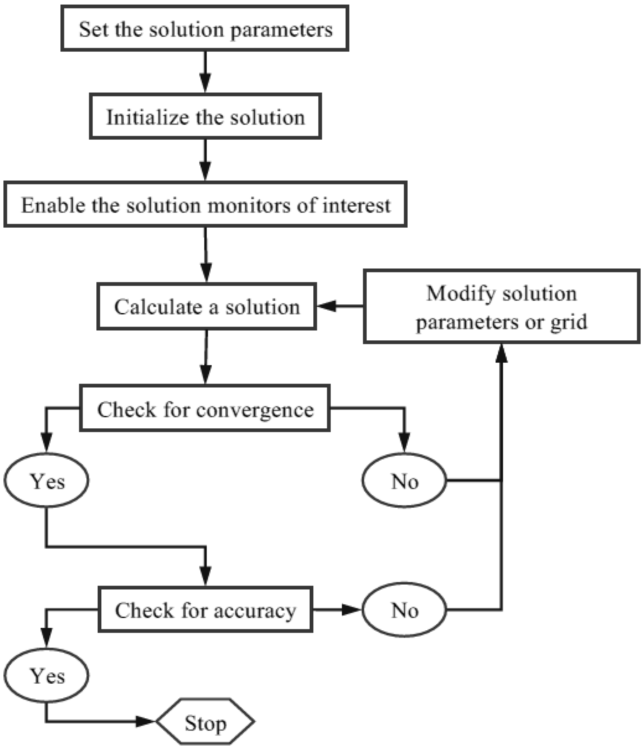


Figure 5.9: Setting Solver Parameters Overview (Ansys)

5.4.5 Visual Modules

The various visual modules used in this work are specific to the individual programming languages and tools. In making use of I-MOPSO framework, the most plotting codes were achieved through Javascript's D3 library and Matplotlib. D3 library allows the generation of dynamic, interactive visuals allowing users to control user needs and results. Matplotlib is a python library for making 2D plots of arrays; it provides an object oriented application programming interface for embedding plots using generic graphical user interface.

The plotting library is used in many different contexts, for automatically generating post-script files, or to deploy Matplotlib on a web application server or for interactive usage from Python shell. The library is designed with the philosophy of being able to create plots with just a few commands; also the base code can be redesigned without affecting the user code. Javascript framework generates visuals by binding data and object model. Wrapper function has been used to adapt various interfaces.

Plotting scripts call upon a parameterisation programme code, which, when given a set of parameters of each particle, return the calculated results or aerofoil's shape through a list of points. A python based web framework is a collection of packages or modules which allows various web-based applications without much low-level management, Django is one such popular high level framework generating clear and pragmatic visualisations; the code, conforming to a set of conventions lets the user to 'plug in' to the framework.

5.5 INTERACTIVE TRADE STUDY

The module settings used in the trade study are: Inlet Mach number, M=0.22, Reynold number of Re=4.51e6, at an inlet flow angle of α =20.3°. The initial maximum iterations for MSES was set at 200; however, the convergence of particles could already be noticed between 30-40. Only fewer particles are converging during later intervals (between150-200), the maximum iterations was reduced (100-150). Reducing the number of iterations gives rise to an increase in non-valid, non-converging and spread out particles; these will not help the decision-maker in a proper evaluation irrespective of whether the individual particles themselves might generate either a good or bad result.

Nevertheless, an increase in the number of non-convergent, rejected particles benefits computational cost by reducing the number of evaluations. When the evaluation process begins to diverge instead of converging, the MSES code has been programmed to stop iterations avoiding the solver to waste a full iteration interval for non-valid particles. This bypassing of diverging computations leads to a reduction in computational time and costs by cutting down on the maximum number of iterations.

After code initialisation, the initial positions of personal leaders are updated for each particle and the first swarm's non-dominated particles are added to the archive. Leader selection takes place in a uniformly divided 10x10 hypercube grid space and a score, inversely proportional to the second power of the number of particles assigned to each hypercube gives way to a roulette-wheel selection and a leader is randomly picked up. For each iteration, the position of each particle is updated. The process repeats until the set maximum number of iterations are completed.

The decision-maker guides the personal and global best particles by selecting the parameter range in the desired optimisation direction with the help of the optimiser code. For each selected dimension, a list of values or results that satisfy the constraints set by the decision-maker are generated. The interactive visual technique aids the decision-maker in steering the optimisation to further explore any particular areas of interest for a closer investigation.

User-computer interface, the computer software and its understanding, along with the overall optimisation process play an important role in interactive trade studies. In using interactive parallel coordinate's plot, the user is able to select a range and particles that belong to the selected interval are shown, both in parallel co-ordinates and simultaneously in the scatter plot. In the example discussed, a 2D aerofoil is plotted with six parameters and two objectives. Each vertical line represents one dimension of the parameter space and the objective space; any combination of parameters and objectives can thus be visualised for an aerodynamic design exploration.

5.5.1 Optimisation

Two test runs were carried out, one each for Garteur and SC2-0610 aerofoil sections. The configurations compute two objectives, lift and drag coefficients.

Garteur is a two dimensional profile taken from 59% Airbus A310 aeroplane wing. It is a three element aerofoil made up of slat, main element and flap. Take-off configuration is considered as a starting point. NASA SC2-0610 is a slotted supercritical aerofoil whose profile is widely used to research aerodynamic characteristics of large civil aeroplanes for pressure distribution at speeds of Mach 0.85.

The main element remains fixed and the geometry is modified by translating and rotating only two of the elements. Spatial coordinates are defined in the two-dimensional plane, X (horizontal translation), Y (vertical translation) and Θ (theta – rotation angle [deg]), the vector parameters are expressed as:

$$\vec{p} = [X_1, Y_1, \theta_1, X_2, Y_2, \theta_2]$$

Slat is element 1 and flap is element 2. The experiments carried try to optimise two objectives [o0, o1]: the maximisation of lift coefficient and the minimisation of drag coefficient. MOPSO framework is designed for minimisation problems only, hence the usage of additive inverse of lift.

The two objectives functions can be written as:

$$f_1(\vec{p}) = -C_L$$
$$f_2(\vec{p}) = C_d$$

5.6 TEST RUNS

Initially, the swarm is populated using a Gaussian distribution with a mean of $\mu=0$ and a standard deviation of $S=0.2$. The particles inside MOPSO are computed in the interval $[-1, 1]$; the change takes place just before they are sent to the model. All through the simulations, the inertia weight, personal and global weights are designated set values of $W = 0.4$, $C_1 = 2$ and $C_2 = 2$. The invalid points are given a value of 100, so as to identify them easily apart from the region of interest. During the interactivity runs, DM aims to explore solutions with $C_D < 0.0280$, trying to maximise C_L.

An iteration of 100 with a swarm of 70 particles and another iteration of 75 with 50 swarm particles was carried out. 25 iterations respectively were used to initialise the domain; no interaction was required during this interval. During the interaction stage of the optimisation, DM was required to interact every 5- 10 iterations.

A small interaction interval allows the DM to make active decision and guide the optimisation. The iterations could be stopped if the DM perceives to have achieved a favourable optimisation result. Optimisation study was carried out based on three interactive approaches.

5.6.1 Trade-Off Information

Using this approach, the lift-to-drag ratio was adjusted to trade-off parameter values. It is beneficial to be aware of objective trade-offs when alternating between Pareto optimal solutions. The key issue was to obtain a partial trade off rate for a Pareto optimal solution. However to seek the values of objective functions separately was convenient for practical reasons. DM had a say on the preferences on the displayed trade-offs whether the current solution seems acceptable or not. Depending on this the information is actualised and a new solution found.

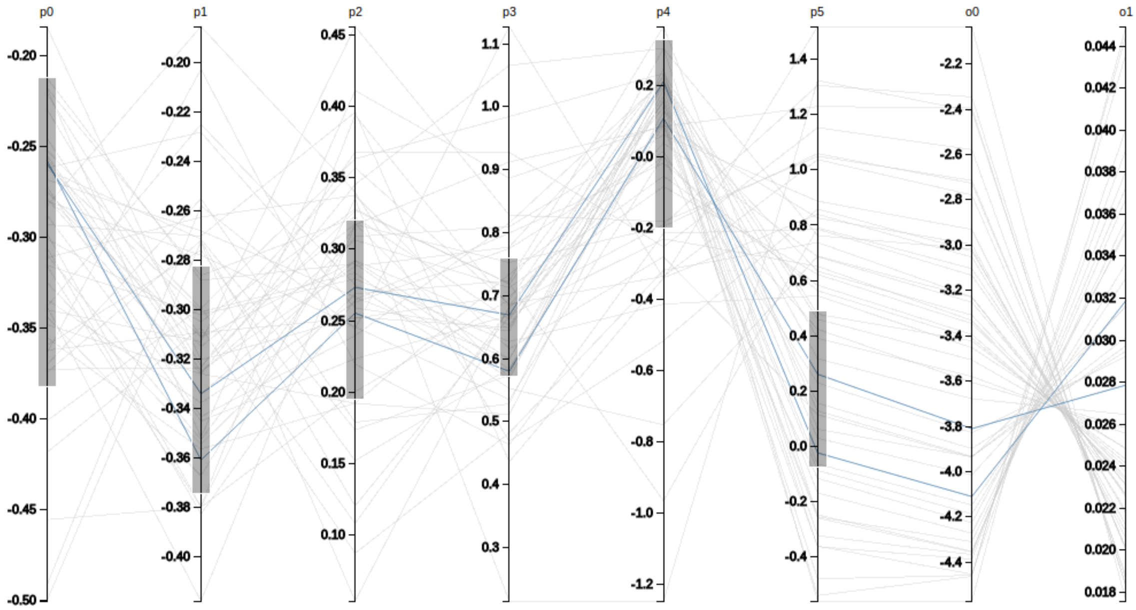


Figure 5.10: Iteration results for Garteur aerofoil section showing an average C_L 3.6 in parallel coordinates map

In the highlights shown in Figure 5.10, C_L at -3.6 seems to be the average obtained value. In order to be able to increase the lift to above -4.2, the drag value seems to increase above 0.036, while decreasing the values of P5 and P3, and increasing the values of P4 and P2 (Figure 5.11).

Parallel coordinate map provides the best interactivity capability among the various visual aids available. Scatter plot, heat-map, self-organising map and radviz plots offer a support to the DM in viewing overall results and in decision making. Individual plots viewed as separate entities may not offer much help and guidance to the user. The results generated by the heat-map was not sufficient for the user to make any useful conclusions of the iterations but combing those with the results of self-organising map offered a more clear view of the generated results, thus allowing the user to view high CL values. Radial coordinate plot seemed to offer the least help to the decision-maker while viewing and analysing the results. It is also necessary to have prior knowledge of the plots and the way the results are processed and shown to draw conclusions. There is a chance that the user will abide by the views that are more comfortable and easy to use rather than exploring every visual aid available for design exploration.

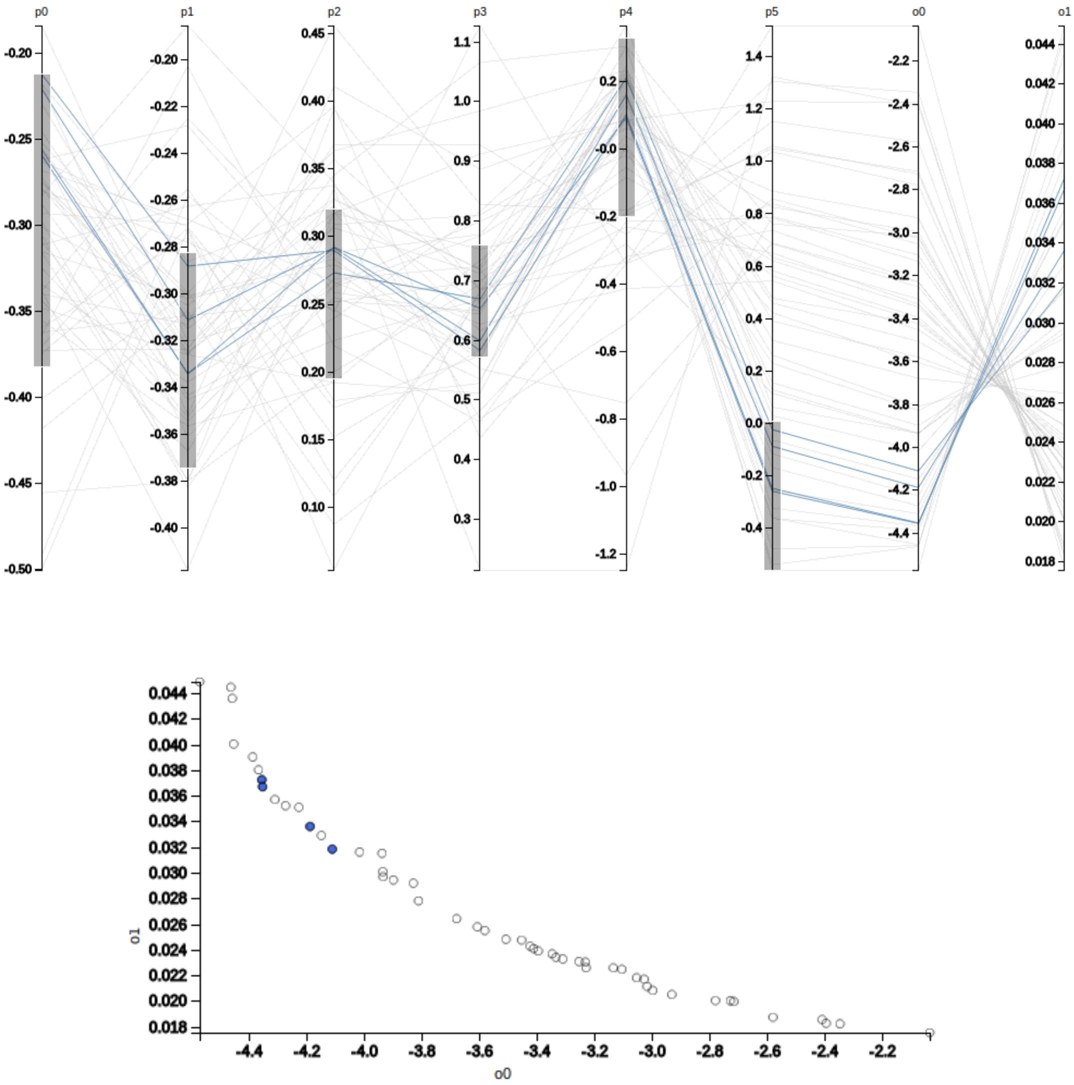


Figure 5.11 (1): Results for Trade off information approach at C_L 4.2 shown in various plots

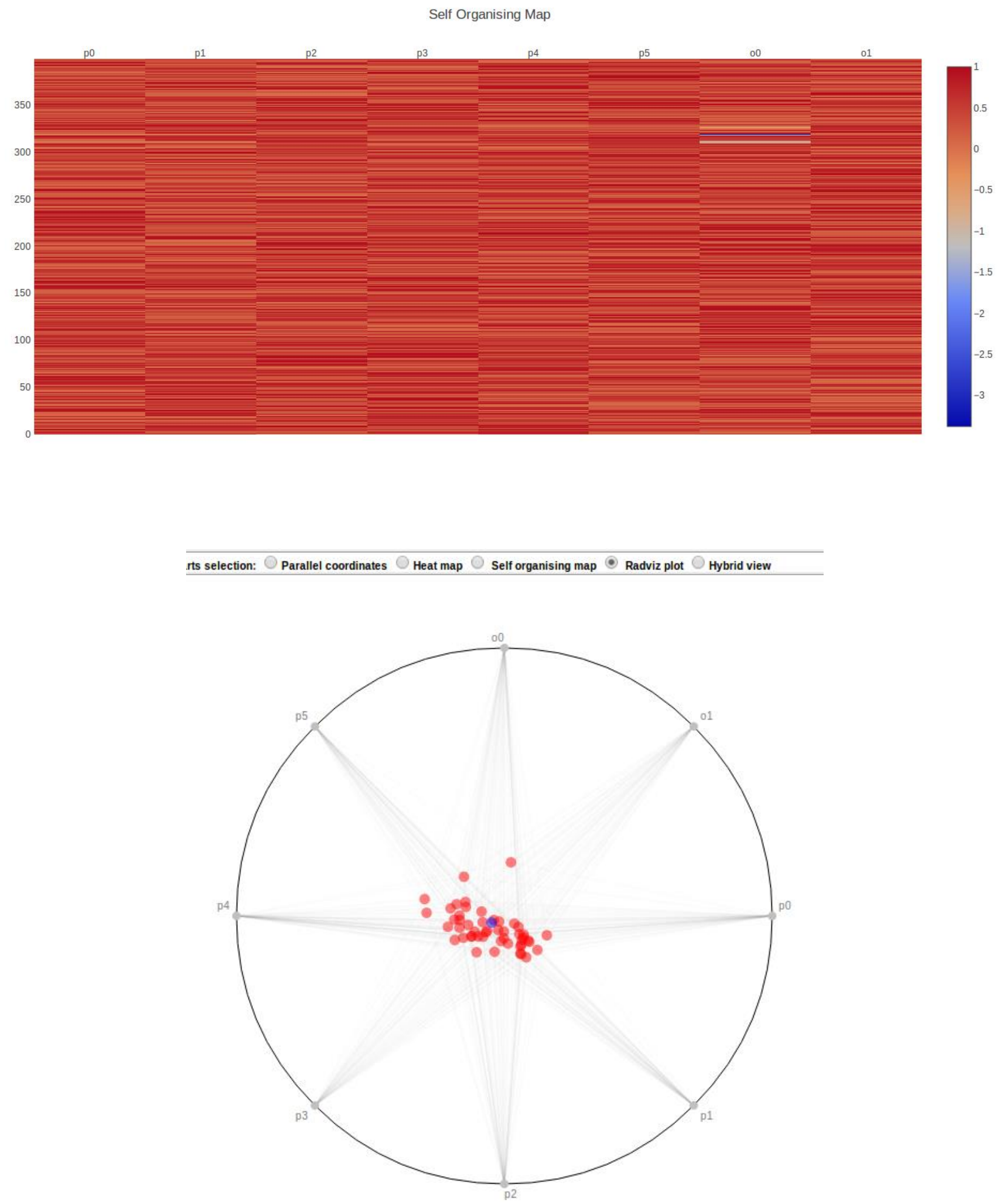


Figure 5.11 (2): Results for Trade off information approach at C_L 4.2 shown in various plots

The desired mathematical convergence was obtained at 67th iteration, proving that the method could be converged in limited iterations. The preference information required from the DM is not hard if the problem at hand and the variable information is clear. DM's consistent responses are vital for convergence and on bypassing a close-enough convergent iteration for another one, it could be that the DM has to wait a while for another good solution unless the tool features allow going back in the process. It is to be noted that a Pareto optimality of the final chosen solution is not guaranteed.

In another test run, an O_0 value between -3.1 and -3.3 and O_1 value between 0.021 and 0.023 was DM's desired setting and adjusting to trade off values of P1 and P2, trying to decrease the lift and drag values (Figure 5.12). An early result pattern could be perceived by DM as early as 30 iterations. The values of P1 seemed to decrease and those of P2 seemed to increase in order to obtain the DM's desired range. A quick view of the scatter plot between P1 and P2 offered some help to the DM in interacting with parallel coordinates plot. Half way through the iterations, a desired convergence of O_0/O_1 could already be observed.

Results shown in radviz plot were found as uncertain to the DM to most extent mainly because it was difficult to explore the relationship between parameters. However, viewing it together with other plots shed some light on the visualisation. Identification of trade-offs by searching and filtering for values using heat map or self-organising map alone was inconvenient to the user; parallel-coordinate plot played an important role in the decision making.

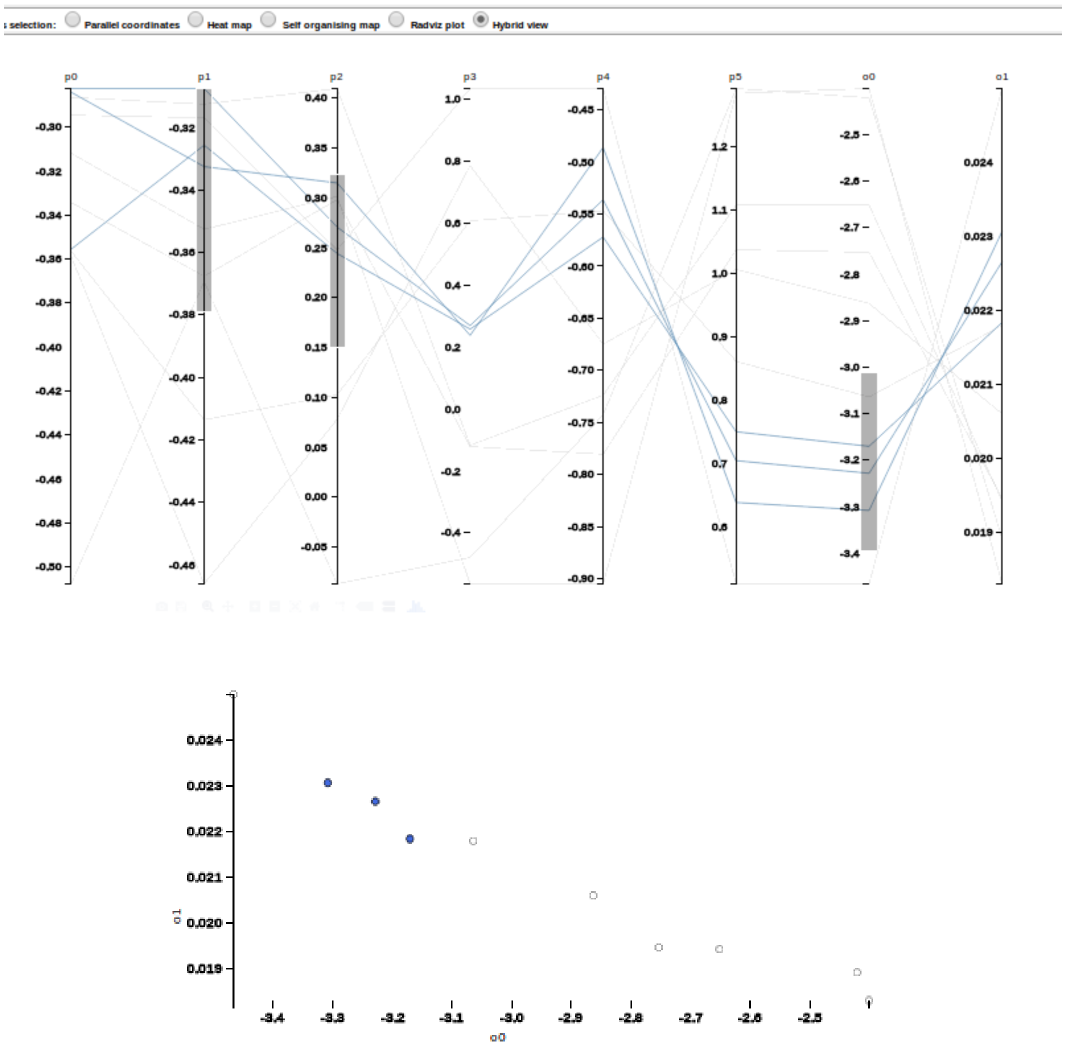


Figure 5.12: Results for Trade off information approach at C_L between -3.1 and -3.3 shown in parallel coordinates plot and scatter plot with 30 iterations

5.6.2 Reference Point Approach

In this approach, DM was looking for a preference O_0 /O_1 value which serves as the reference point. A prior knowledge in this approach regarding DM's preferences on the decision taken to arrive at a set goal played a vital role. This approach serves individual cases well having limited options for decision-making. In this approach, learning is a concurrent activity of the DM when interacting with the decision support system (DSS), which in this test scenario are the various visual aids. DM's preferences might change in the decision making process and has full right or even a necessity to be inconsistent during evaluations.

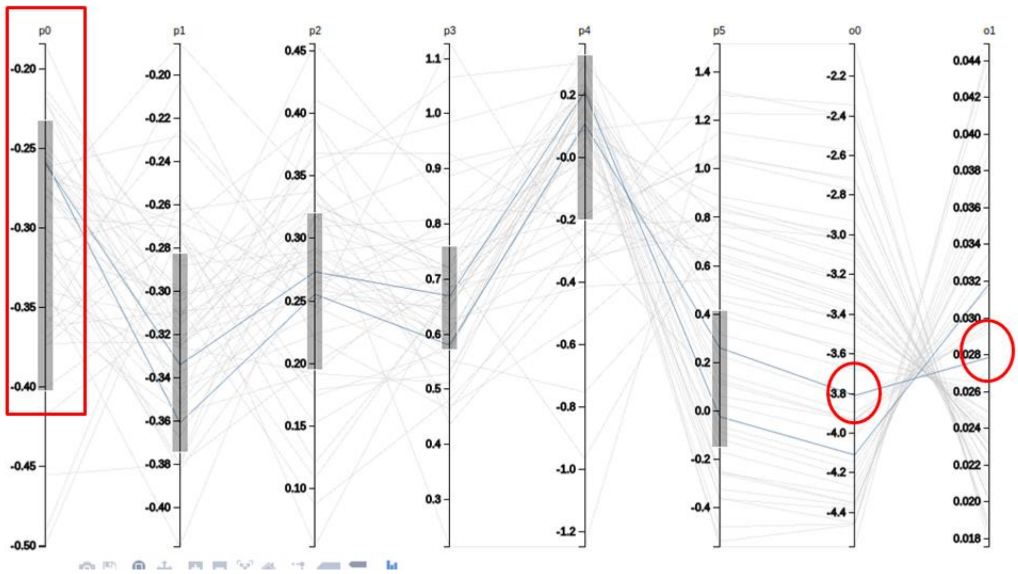


Figure 5.13: Results for Reference Point approach

In order to maintain a CL/ CD ration of 0.1, with CL at -3.8 and CD at 0.028, P0 should be maintained between -0.20 and -0.40 (Figure 5.13). This approach assumes from the start that all objective functions are subject to trade-off analysis. By improving or diminishing the value of one objective function, another function could be either improved or worsened.

This approach could promote decisions with unbalanced objectives as the natural human preference is for a balanced solution. DM makes decisions based on an overall evaluation of the decision situation. The various visualisation techniques help compute and communicate to the DM on suitable objective function value ranges, thus aiding the DM in an overall evaluation.

It is also to be noted that a reference point approach assumes that the reference (choices, restriction) levels and points are not considered as fixed representation of DM's preferences but are an adaptable tool which help in overall learning regarding the decision situation. Thus, even if the convergence of reference point approaches to a solution most preferred by the DM can be proved, this aspect is never stressed.

Determining the reference point by the DM decides the outcome of a Pareto optimal solution. Even if the reference points could be determined in an objective fashion, independent of DM's preferences, there remains a diversity of such objective determinations, thus making possible comparing of various resulting optimal solutions.

DM can select Pareto optimal solution by altering reference points and maximising the target function. This gives the DM full freedom which could be used to study the problem and decision situation, and to investigate various sections of the Pareto optimal set.

For a CL set at -3.3 and CD value of 0.023 which was DM's reference point, worsening of drag lead to improvement in lift, adjusting to trade off values of p1 and p2 (Figure 5.14). The values of P1 seemed to decrease and those of P2 seemed to increase in order to obtain the DM's desired reference point. This approach limits exploration as the reference is already set and offers less room for innovation, limiting identifying any novel views or patterns. In terms of visualisation, no advantage was noted over trade-off information approach; the pros and cons noted showed similarity.

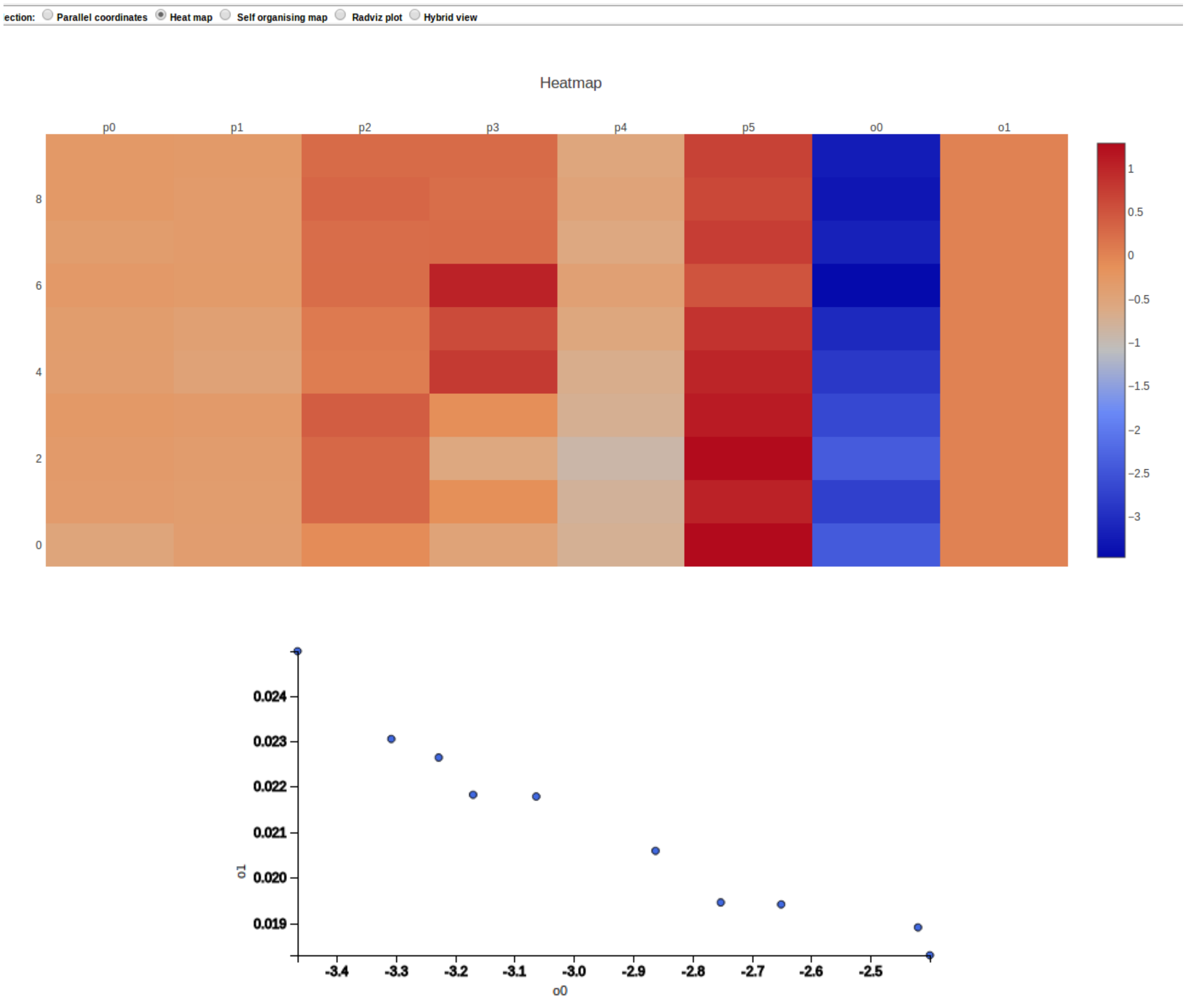


Figure 5.14: Results and evaluation for Reference Point approach CL set at -3.3 and CD value of 0.023

5.6.3 Classification Method

The classification based approach assumes the possibility of moving from one pareto optimal solution to another to improve one objective function value by permiting another objective value to get worse. DM indicates the preference by classifying objective functions. Such classification is a clear way of expressing the DM's choices.

The idea is to tell whether lift or drag functions should be improved or impaired from their present values. DM is shown a generated Pareto optimal solution who then defines what modifications to the objective function values might generate more satisfactory solutions. Classification is a conscious reasoning method for DM to steer the solution progress so as to identify solutions that are most preferred because no additional ideas are utilised; rather, the DM handles objective function values that are deemed essential and logical. The DM is able to define any possibilities regarding solution improvements to directly visualise and compare how good the anticipation could be achieved on generating the next solution.

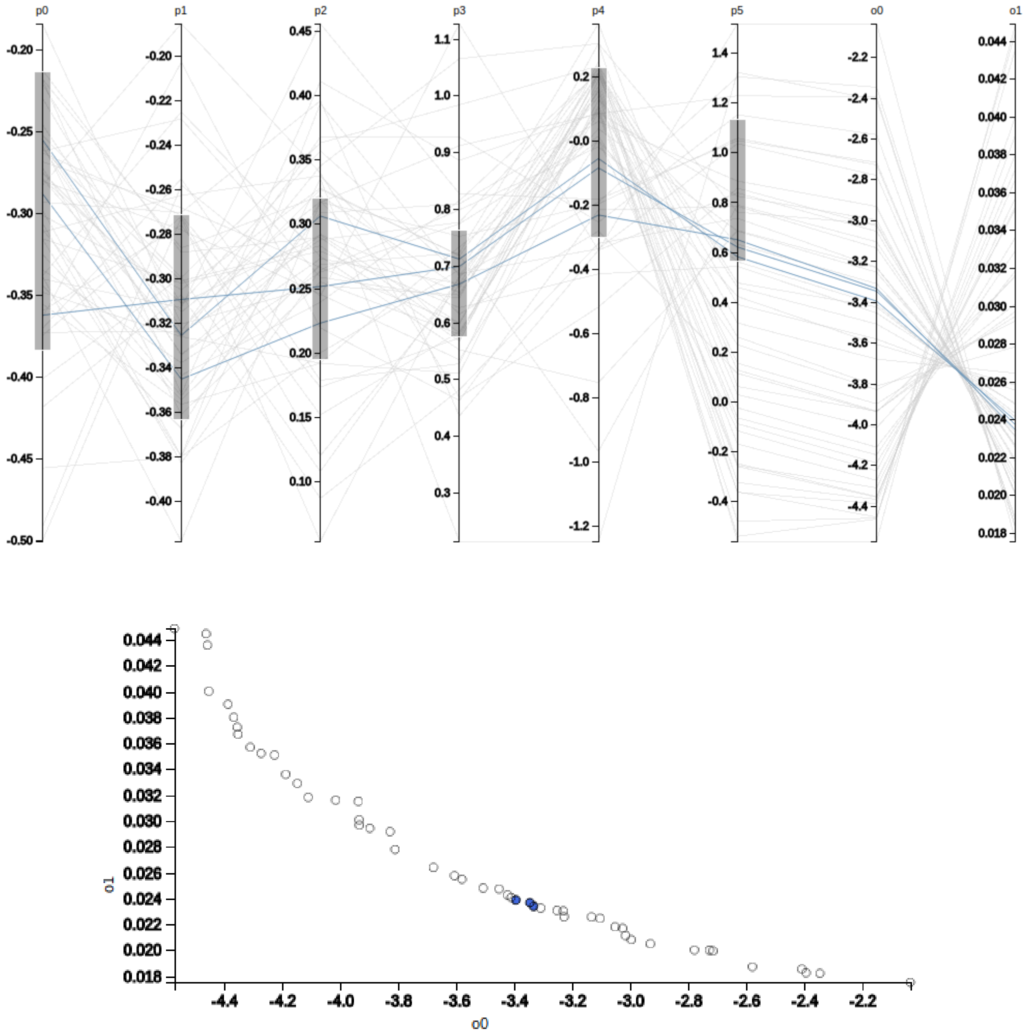


Figure 5.15: Evaluation of Classification method for CL at 3.4, drag has to be maintained around 0.024 and P4 between 0.2 and 1.2

To generate a particular set value for CL at 3.4, drag has to be maintained around 0.024 and P4 between 0.2 and 1.2 (Figure 5.15). In general, it is not feasible to make better all objective function values of a Pareto optimal solution, but the DM was able to express preferences without paying attention to this and to visualise the various solutions that are feasible. When using classification approach, there is a possibility of the DM to be more in control when selecting between objectives to be made better and the level of mitigation for others.

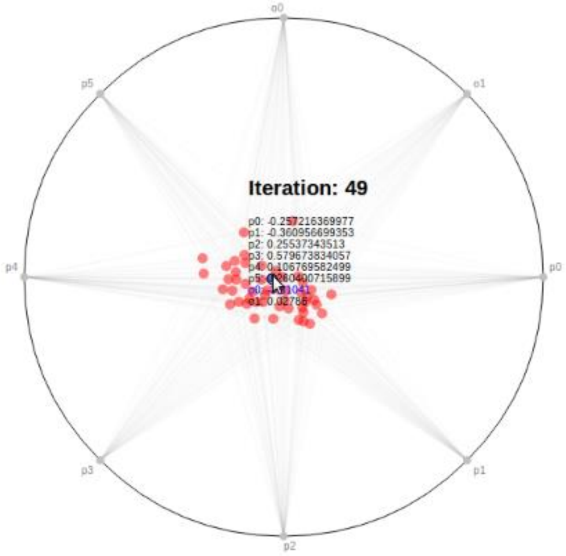


Figure 5.16: Results for classification method approach in Radviz plot

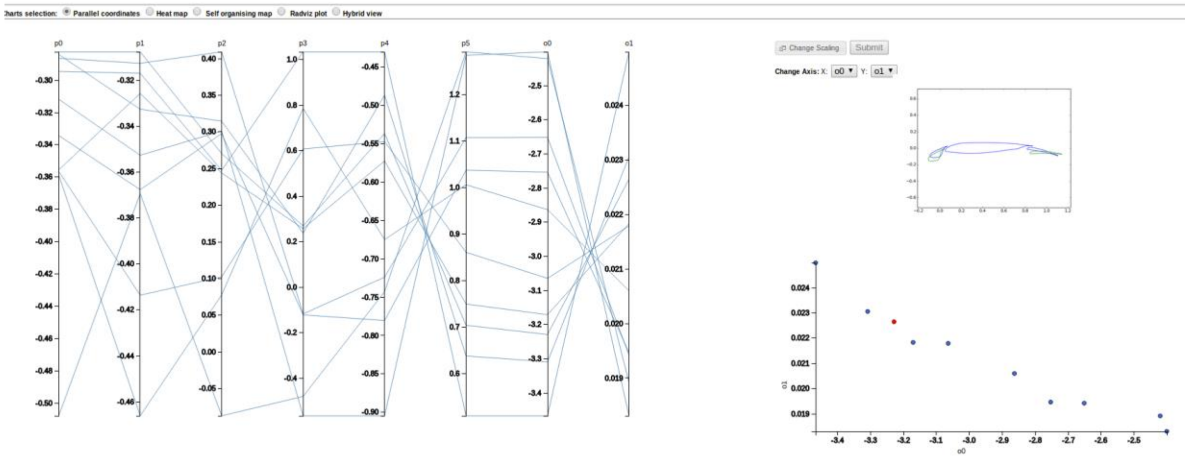


Figure 5.17: Evaluation for classification method approach for increasing lift from -3.1 to -3.4, which increased drag from 0.021 to 0.024

Another test looked at increasing Lift from -3.1 to -3.4, which increased drag from 0.021 to 0.024 (Figure 5.17). It was possible to stop the evaluation as the DM seemed to have achieved the desired evaluation result quickly.

As this approach chooses between objectives, O_0 and O_1, visualisation of solutions is uncomplicated as it eliminates the need for weighing out other variables, thus minimising clutter. Scatter plot and combined view offered good views to the DM. This approach also allows the DM to choose between two parameter variables if so desired. Parallel-coordinate plot offered the best view of solution convergence among the available visual aids.

Classification-based method shares the philosophy of reference point method in terms of stopping criteria; the DM's sense of contentment is the ultimate stopping factor. The search process could be continued as long as the DM wants to because it is not important for a mathematical convergence to occur as in trade-off based methods but instead the psychological convergence plays a vital role. This is supported by the fact that the DM typically wants to experience being in control and would not naturally want the method to dictate when the most preferred solutions are found, overriding their own preference.

5.7 DISCUSSION

Three interactive approaches using I-MOPSO module are presented in this chapter. User interaction through the designer-in-the-loop features offerd by algorithm and user interface through the various visual techniques have been analysed. The practical aspects presented in this chapter build on the study presented in previous chapters covering high-lift aerodynamics, interactive optimisation and visualisation.

The interactive utilisation of user preferences and choices is necessary for the reciprocal response generated and analysed through visualisation tools which is necessary to make the analysis of results easier, and to help in decision support. The choice of methods chosen is in accordance with the nature of optimisation problem task, a high-lift aerofoil design context; however, in an industrial setting, each discipline is then responsible for the choice of its preferred methods.

Several factors that affect design optima and evaluation criteria for visual techniques have been discussed in Chapter 4. A summary of these characteristics for visual tools used in the test runs is presented in Table 5.18; it rates various techniques against certain preferred visual tool features such as dimensionality, trend analysis, discontinuities, design selection, robustness and ease of use. The rating is done from 1 through 3, from weakest to strongest feature, along with its ability for dimensionality.

Parallel coordinates seem to offer a very good interaction support to the user with radviz plots offering least support, primarily to with the nature of data representation and ease of user interpretation. All other plots used in the trade studies served as support tools to interpret data and complimented one another. However, it was difficult for the user to rely on just any one visual tool in order to express preferences and analyse problem task.

FEATURE	SCATTER PLOTS	PARALLEL COORDINATES	HEAT MAPS	SELF-ORGANISING MAPS	RADIAL COORDINATE VISUALISATION	COMBINED VISUALISATION
DIMENSIONALITY	N_L	N	N	N	N	N
TREND ANALYSIS	2	3	2	3	1	3
DISCONTINUTIES	2	1	1	2	0	2
DESIGN SELECTION	2	3	2	3	1	3
ROBUSTNESS	3	3	2	2	1	3
EASE OF USE	3	3	3	1	1	3

0- No Detection **1**- Weak **2**- Moderate **3**- Strong **N**- Any Number **N_L**- Limited Number

Table 5.18: A Comparison of various visualisation methods against certain desirable features

	Trade-Off Information	Reference Point Approach	Classification Approach
Objective	To increase O_1/O_2, adjust to loose another value (P Values)	Searching for a specific O_1/ O_2 value	Choosing between O_1 and O_2
Analysis	• Able to shift between pareto optimal solutions • DM to decide if search should be continued looking for a more optimum solution • DM feedback is important for convergence	• DM intuitively searches for preference solution and has the freedom to modify reference points • DM has complete dominance in maximising the goal or achievement variable	• DM decides on an objective function that seems meaningful, important and understandable • Input information is required of DM's desire for the extent of solution improvement or permitted levels of deterioration • Psychological convergence is more important than mathematical convergence

Table 5.19: Evaluation of various interactive approaches

A combined view also provided the user with easy visualisation of preferences and results of choices by eliminating the need to move between various visual tabs; however, the size of the computer screen affected the clarity of the results displayed, a zoom or pan feature was then used for easy reading. The capacity of the hardware used to run calculations also impacetd the speed and clarity of visual dispalys.

Several factors pertaining to human factors engineering are discussed in sections 3.4 and 3.5. As the user was involved in coding the module's framework, it allowed for learning about the system and its functions, also influencing user's perception and interpretation of the projected results. There is always room for betterment of user interfaces and making the system user friendly. Encountering several programming errors, along with hardware system limitations lead to periods of disinterest; the effort, which started as a conscious, knowledge based attempt adjusted into a skill-based, automatic task (Figure 3.9). To avoid continuous errors and slipping into a usual mode of operation, it was necessary to opt for creative approaches in solving generated programming errors by also regulating psychological workload to minimise errors which might otherwise be a result of avoidable stressful situations.

It was necessary for the user to be aware of the nature of problem task, operation of solver, optimiser and their respective settings. Such previous knowledge allowed the user to identify errors during interaction. Graphical literacy also played a role in interpreting the displayed information. Table 5.19 presents a summary of interactive approaches and their evaluation; all approaches exhibited the importance of decision-maker in design trade-offs and the role played in presenting choices, preferred information of objectives and parameters.

BLANK

6

Assessment & Scope

6.1 INTRODUCTION

Along with the presentation of empirical findings (section 6.2) and overall conclusions (section 6.10), this chapter also delves into a detailed assessment of digital human in a man-machine setting and contemplates future of flight.

Several positive attributes and advantages have been presented throughout this exposition but this research also recognises that there are inherent limits to interactive optimisation (section 6.3); that analyses have boundaries in terms of uncertainties (section 6.3.2), visualisation hurdles (section 6.6.2), reliability and validation of software (section 6.3.3). However, in the midst of these constraints, wings of the future are designed now (section 6.9), future technologies are conceived today. This calls for continuous optimisation (section 6.6) to keep up design innovation (section 6.6.1). Automatic design will not drive novelty but interactivity could (section 6.8) and design prediction is part of it (section 6.5) which includes visual and data analytics (section 6.5.1). A thinking engineer is an indispensable part of realising innovation (section 6.7) where critical thought processing abilities play an important role (section 6.7.1). Often confined by several decision support systems and tools to aid at work, an engineer could best start by asking why (section 6.7.2).

6.2 EMPIRICAL FINDINGS

Interactions seem to show a positive impact on a designer's acceptance and assurance of analysis results. Interactions, and thus interactive optimisation, seems to aid the designer in learning about an optimisation problem, also helping the designer to grasp and better discern the optimisation approaches. When a working system is understood better, there is a greater likelihood of a designer's acceptance of it. Additionally, a fair amount of interaction loops may prevent circumstances in which a designer over-estimates the efficiency of an optimisation system. Reviewing an interactive optimisation method, such as particle swarm technique, also addresses the inherent limits of an optimisation model and the difficulty to obtain adequate performances. In circumstances where it is very costly to assess an individual configuration computationally, the interactive approach is especially suitable so that the comprehensive number of parameter arrangements calculated could be minimised accordingly.

Apart from the literature that justifies an interactive approach as a second argument, the fact that interactions help the designer or decision-maker learn of an optimisation problem and the related processes is a distinguishing trait common to several interactive optimisation methods. This work employs particle swarm optimiser for design optimisation; the primary benefit of this approach over other global minimisation techniques is that the problem of local minima is effectively resilient by the particles.

This, however, does not advocate that any particular optimiser is better or best. Choice of optimisers, similar to that of meshing techniques, solvers and visual aids, are best analysed in light of their use and analysis of results, depending on design objectives and the problem at hand. The work, nonetheless, emphasises interactivity, involving human in the engineering design loop, to take advantage of the super human brain over super computers. In an effort to digitally mimic human brain, according to (Heisler, 2016) scientists ran more than 82,000 processors on one of the fastest available supercomputers to emulate just 1 second of brain activity of an average human.

It is to be noted that the interactive nature is not limited to optimisers or visual aids alone but can be adapted to solvers, meshes or data handling, for just about every stage of aeronautical computational analysis. Selected visual techniques have been explored and their assessment presented; however, scope for improving visualisation, on both interactive and non-interactive fronts remains high.

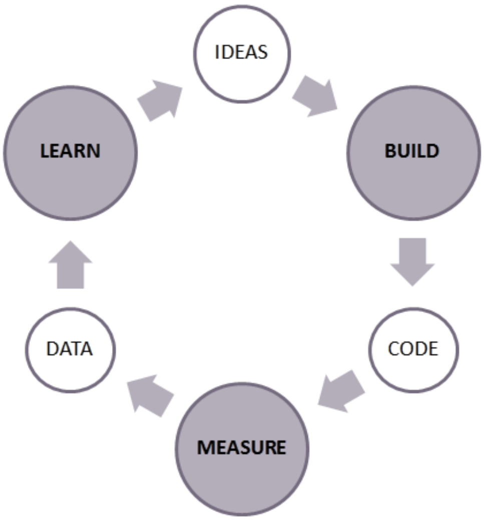


Figure 6.1: Principles of validated learning and innovation accounting (Ries & Hickman, 2012)

Several articles in literature research suggest and compare optimisation methods that are more effective than previous ones. For high-lift design too, proposals of advanced algorithms and enhanced results continue through the years; however, a gain in performance of optimisation methods is often acquired at the expense of simplicity and flexible adaptability of the methods used (Meignan et al., 2015). The overall assessment of numerical optimisation for aeronautical applications, including high-lift design can be summarised in the thoughts of (Holt, 1982) as follows:

- It is surely not correct that all novel and complex technologies require several years to be accepted by the working aerospace community.

- The literature bulges with theoretical knowledge on pros and cons of several aeronautical designs, yet design specifications do not make use of it; its methods are often unfamiliar and competitors in the industry do not seem to have embraced them. Figure 6.1 presents an order of lean principles to inculcate continuous innovation.

- Codes that are dependable are not easy to come by, and they tend to be much complicated to use in practice than even the largest and best analytical programmes.

- During development stages, optimisation has great ability as a stable way of choosing among alternative ideas. However, it may make things complicated later on in the development process. It is no good then in trying to optimise a bad idea.

- If one builds on a design idea that nobody wants, what does it matter if it is accomplished on time, on budget, with high quality and with a very good design? Achieving failure is successfully executing a bad plan. Behind every supposed technical problem is usually a human problem. (Ries & Hickman, 2012) mentions about fixing the cause, not just the symptom.

- Due to lack of experience when new technology is being tested for the first time, unexpected, startling and often very suitable discoveries can be generated when searching for extremes. Yet, there may also be counterproductive and even absurd results; certain undesirable consequences may result from omission or thoughtless execution of constraints.

- Even when there are several convincing factors to support the use of new technology, uninterested, negative attitude of individuals and teams may prevail and hinder.

A final optimisation decision is the result of a compromise among all alternatives including overall design requirement considerations. Subjects spanning mathematics, physical and engineering sciences are the norm of traditional engineering education; a lack of sufficient emphasis on design and creativity is a dilemma. (Sadraey, 2012) points that creative thinking and its attitudes are vital to a design success. The ability to be creative and win over strong hindrances is required in crafting novel, improvement designs.

The above discussion can be summarised in the following table, pros and cons of interactive optimisation:

PROS	CONS
Interactions have positive impact on both design analysis and the designer	Choice of optimisers and new algorithms continue to be proposed all the time
Interactive design analysis helps in gaining clarity of an uncertain optimisation problem and user choices	Choice of interactivity is a decision and an attitude which are highly dependent on the designer, either as an individual or as a team
Several software (eg: Python) are able to offer good interactivity features	I-MOPSO is not the better or best optimiser. In general, there is no guarantee of an optimal solution ever found
It does not take long time to adapt any novel & complicated technologies	Several programming languages remain to be explored and exploited
The problem of local minima by I-MOPSO algorithm is effectively resilient	More chances of designers relying on less-efficient approaches
Optimisation has the potential to help choose among alterative concepts	New technologies evolve much faster than their implementation
Tools can be developed simple enough for everyone to use and understand	There is room for extending interactivity to all phases of a computational design and analysis, not limited to optimisation
Parallel coordinate visualisation continues to be the most preferable MDO technique as of today	Several visual techniques are a disadvantage if the user/ designer lacks knowledge of available tools or their use
Interactive optimisation is suitable for computationally expensive configuration analyses	Many MDO visualisation techniques are available and several under development

Table 6.2: Table highlighting pros and cons of interactive optimisation

6.3 LIMITS OF INTERACTIVE OPTIMISATION

According to (Meignan et al., 2015), an optimisation model, including interactive optimisation methods are only a partial depiction of the real problem. An optimisation model contains inevitable errors due to the various limitations of modelling processes, which may be problematic for supporting a decision. Criterion such as in-flight activities, both machine and human generated risks, strenuous tasks, real-time boundary conditions, mesh functions, perceived duration of actual flight conditions are approximated out of necessity in an optimisation model and particularly difficult to model. This may result in impractical or unworkable computation solutions, or those that fail in capturing attributes related to the domain.

Most times, specifications of an optimisation problem are simplified in order to apply a computational optimisation approach, usually a single objective function combining several criteria, or by linear models that approximate relationships of variables. Apart from the inherent barriers to an optimisation problem, errors in optimisation model may also be related to difficulties in capturing an entire overview of the problem for a model design. Those designing the optimisation model may possess only a limited understanding of the actual circumstances of the optimisation problem, which is always an issue, especially with very large teams working across multi-system integration platforms.

The determination of the adequate choice of an optimisation technique remains a tough task at the design stage; it is a crucial point for the effectiveness of optimisation. The choices in design decisions are often the result of a trade-off between the validity of the chosen model and the likelihood of models being made easy so that they can be solved readily. Maintaining traceability of the problems and their validity always exists.

Technology Acceptance Model (TAM) is a user's technology acceptance mainly affirmed by the anticipated use of a system and perceived usefulness. A common limit of optimisation methods, such as interactive optimisation is the concern of a designer's agreement with the system and the assurance in the solutions generated by the chosen method. This assurance is especially critical for an optimisation system as it must solve complicated problems that will not be fully understood by the designer but the perceived usefulness, anticipated ease of use and faith in the system are necessary for an optimisation system to be received by its end users.

Having no trust in an efficient system or having confidence in an inefficient one are both factors of risk for the designer: solutions that may result in good choices may be rejected, the effectiveness of optimisation system may be overestimated, or solutions that could be made better by other techniques accepted. Whatever the situation, there is room for a designer to misunderstand the value of an optimisation system and its results. According to the result of a study conducted by (F. . Davis & Kottemann, 1994), users showed a lack of conviction in an efficient approach but had greater belief in a far less effective approach. This demonstrates the influence of end user perception and confidence as it ultimately decides on whether a system is accepted or not.

6.3.1 Analyses Limits

An aeroplane and its wings are designed taking for granted that the fundamental laws of physics governing our nature stay the same, which is a fundamental limit.

Every design of experiment consists of three basic parts while optimising:

- objectives under test
- variables or alternatives
- constraints which are fixed

Among these, a designer is able to influence only the variables when an objective is tested and optimised under the notion of fixed constraints. Sometimes, a degree of violation of constraints also serves as a key influencing variable. For every variable used, a lower and upper bound must be specified prior to any optimisation. A change in the value of the variables will influence (either improve or worsen) the measurable performance of the optimisation.

Trial and error methods are beset with certain fundamental difficulties, which must be clearly understood and appreciated to achieve maximum advantage even in the midst of performance limits.

6.3.2 Uncertainties

All decisions do not work out to be the correct ones, regardless of the amount of data collected. Occasionally, the effect of wrong decisions is small, other times having significant consequences. Whatever the result, decisions are made based on a perception of the system that is unlikely to be completely accurate, and this should be recognised in terms of design sensitivities and uncertainty (Figure 6.3).

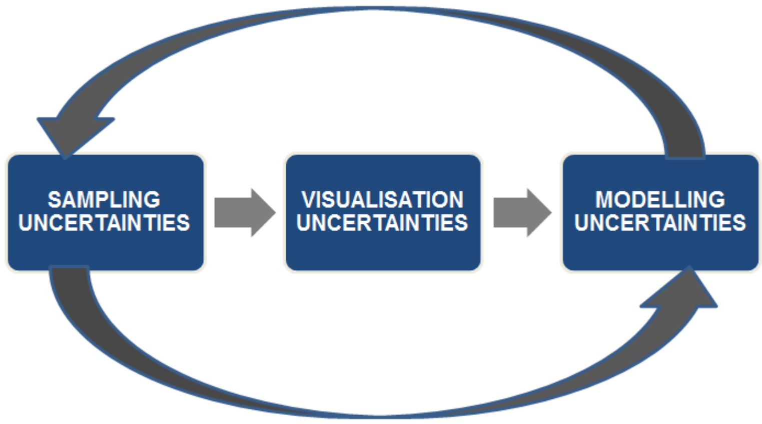


Figure 6.3: Sources of uncertainty; both sampling and modelling uncertainties affect each other and add to visualisation uncertainties (Bonneau et al., 2014)

Making choices under uncertainty is at the heart of decision theory (Pascal, 1670) and the actions chosen are usually the ones that give rise to the highest total expected value.

Evidence demonstrates that making decisions in an environment of uncertainty occurs in different parts of the brain in comparison to decision-making done under more certain conditions (Bonneau et al., 2014). The more complicated the task of making a decision under uncertainty, the more complex approaches are required and is likely to be influenced by one's experiences and events from the past. This theory is upheld by psychological evidence which demonstrates an increased engagement of brain areas vital to strategy formation and alteration, particularly the prefrontal and parietal cortex in uncertain situations (Paulus et al., 2001).

6.3.3 Reliability and Validation

Errors in computational modelling and solutions are inevitable. The reliable accuracy of any computational solution could be primarily measured in two ways:

- reference to analytical solutions
- largely accurate numerical solutions

For several aviation related problems, numerical solutions seem to be the answer rather than analytical ones simply because of the sheer size and complexity of the problems dealt by the industry. However, numerical solutions can be a good approximation under specific conditions or not so good under others.

The study of a solution's correctness or error is fundamentally empirical, albeit within set design limits. Rigorous reliability by demonstrating the study results for all possible applications of a CFD algorithm or code is practically impossible for complicated ones. Such a goal could however be achievable for examining specific calculations by using other validation codes- a code for a code.

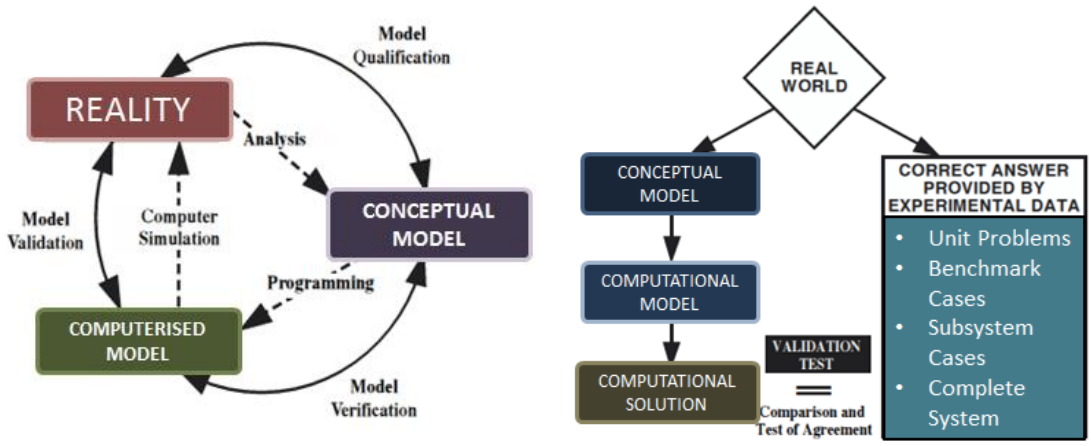


Figure 6.4: (left) Phases of Modelling & Simulation and the role of Credibility and Validation (right) Validation process of comparing computational results & simulation with experimental data from various sources (Oberkampf & Trucano, 2002)

Verifying a code's reliability requires meticulous proof that the conceptual model and its solution is represented correctly by computational implementation. This also requires evidence that the underlying partial-differential equations, along with its initial and boundary are correctly approximated by the algorithms implemented in the code (Figure 6.4).

Additionally, it must be proved that convergence of algorithms to the accurate solutions of the particular equations under all conditions in which the code will be utilised. It is very doubtful that such evidences, proofs will ever wholly be available for CFD codes. The shortcomings and inability to offer a reliable code verification proof is similar to the code validation problems. In an operational sense, verification is the insuffiency or absence of proof that the code is faulty, a continual consistency with experimental data and the outputs from other computational techniques (Oberkampf & Trucano, 2002).

Increasing confidence in detailed evidence supporting code verification is not easy. Evidence could be gathered from user communities on the efficiency of a CFD code to contribute to assessing verifications, although it is a multi-fold problem in itself for the aviation industry. Observation skills of a designer play an important role in identifying, eliminating any errors and also in gathering careful empirical assessment information to build a knowledge hub of iterative performances.

6.4 EVOLUTION OF TECHNOLOGY

Technology evolution can usually be broken down into stages. The introductory, proof-of-concept stage exhibits the basic capabilities and performance of a technique or technology; even trivial functional systems seem impressive in this stage and encourage imagination, but also sometimes raises doubts and misunderstanding.

In the second stage, or imitation, the technology starts mimicking existing technologies, with very few basic changes made to the operational interfaces; the technology begins to be steered less by novelty and usually starts attracting a larger interested audience in developing things more intensely by the way of science of understanding. This leads to the third stage, in which the technology achieves maturity; technology architects understand and exploit the intricacies of new knowledge and make way for unique experiences that offer capabilities that were not available until then (Tan & Nijholt, 2010).

It is vital to carry on with inventions and breakthroughs within a certain domain itself, but also equally important to build connections, leverage engineers and researchers, and exploit advantages from ongoing work in other fields. The domain of human-computer interaction continues to drive toward increasing the knowledge and understanding between humans and machines, more considerably to craft technologies that assimilate smoothly into day-to-day tasks.

6.4.1 Uniqueness & Limitations of Human Brain

Human brain is unique with remarkable cognitive capacity with the ability to recollect images, sounds, or objects from one's memory with intense accuracy. The brain has plasticity, which means, that one is not stuck in situations where nothing can't be done; subconsciously, a person is able to alter one's brain to accomplish something that they could have not imagined to achieve. It is capable of complex colour perception and depth perception.

On the other hand, in comparison to the large amounts of information that is generally available in the sensory input, a human's ability to form perceptual categories is very limited. Similarly, the capacity to generate guided actions by visuals is limited to one or few objects at a time. Visual recognition takes time, typically hundreds of milliseconds, as it depends on a series of physical processes in the brain. A perceiver's subjective bias for making certain kinds of categorisations and the given relative attention does influence a visual selection and analysis.

If workplaces permit people to do work they value and let them find meaning, they will be crafting a human character that values work. There is a possibility that people would rather interrupt sleeping and eating than give up practicing their arts when preoccupied by what they do (Aurelius, 2003). All people are creative individuals, born with creative streaks, children being the best example. It is regrettable, if often, several creative people are trained to be un-creative, thus limiting design exploration and optimisation. Human being continues to remain the most unrivalled machine.

6.5 PREDICTIVE ANALYSIS

Predictive analysis in design makes use of statistical techniques to study historical and current facts to generate predictions about design unknowns and aids in novel approaches. They help in exploiting patterns based on transactional or historical data to identify various opportunities and risks.

Relationships captured by design models among several other factors to support evaluation of negative potential or risks associated under a specific set of conditions, thus steering a designer in decision making.

6.5.1 Visual & Data Analytics

With the advent of computational applications in aeronautics, data is being generated at extraordinary rates. The scope of gathering and storing new data is growing very quickly, while at the same time, the capacity to analyse these large volumes of data is moving at much lower rates. This gap is leading to novel problems in the study and analysis processes because decision-makers as individuals and teams rely on concealed information in the data; the diverse patterns and connected relationships among various elements are of much interest. Gathering a large amount of diverse types of data very quickly does not generate appreciated value, an analytics strategy is required to uncover insights that will aid optimisation.

Technologies for visual representations and interaction offer a means for the designer to see and comprehend large volumes of information in a single instance. Human mind is capable of understanding complex information when delivered through appropriate visual means. This ability is utilised by visual analytics to facilitate scientific and rational thought processing. Visual representations and interaction techniques provide the designer with an understanding of evolving situations so that the user takes action (J. J Thomas & Cook, 2005)

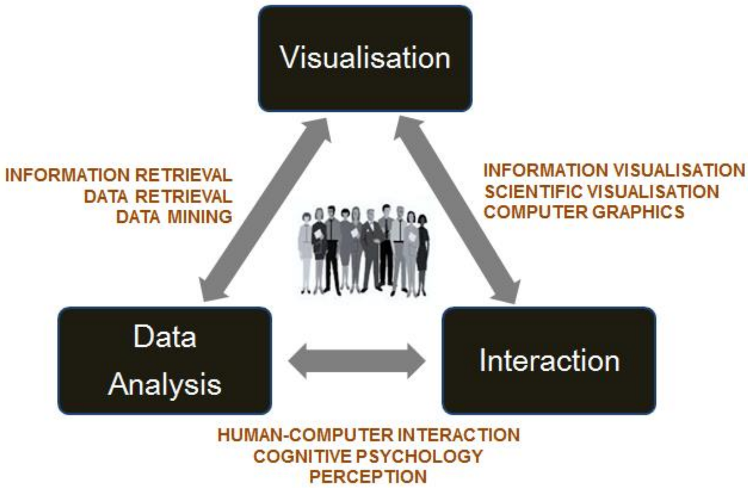


Figure 6.5: Visual analytics is shown as an integral approach combining visualisation, human factors and data analysis (James.J et al., 2005)

6.6 CONTINUOUS OPTIMISATION

No matter the many number of tests and empirical experiments one carries out, the capacity to monitor code efficieny is inadequate and limited. Designers might never be able to prove that a particular implementation of software has zero defects or is error free. Neither errors, nor the presence of limits can be completely eliminated; however, they can be minimised.

According to (AIAA-G-077-1998, 2002) the key phrase differentiating the definitions of uncertainty and error is ‘lack of knowledge’, primarily to do with lack of knowledge about the processes, possibilities and experimenting with probabilities that go into designing an aeroplane, a wing or an aerofoil.

The aviation industry relies heavily on a team culture rather than an individualistic culture to steer ideas, innovation, and the business itself chiefly because of its sheer size and nature. Individuals within aviation industry with brilliant ideas cannot easily take their work forward unless plugged into and helped by a larger industry team. In the same way, the industry without its smart and talented work force cannot make an impact the way it intends to or be so profitable in the long run. The ability to inspire one another should not be left to be practised by a chosen few, but by the majority.

The chief advantage of re-optimising existing projects is that the stock of already compiled programmes could be improved almost instantly with minimal effort, reducing computational resources, and also lowering costs. The disadvantage could be that new code releases would require optimiser maintenance to cater for possibly changed algorithms but this is trivial in comparison to the advantages gained in the long run.

It is also to be noted that, in general, as is with computational technologies, new updated codes frequently coincide with hardware upgrades; the faster and efficient hardware would usually more than compensate for various updated software programmes reverting to their pre-optimised versions.

6.6.1 Design Innovation

Most engineers follow a very similar process throughout a design & analysis loop. When an interactive optimisation is adapted, although similar methodologies were used previously, the final results may differ from one design engineer to the next because multiple solutions exist to the design challenges.

Some designs are already tried and tested, and hence are generally chosen for speed and simplicity of implementation. One of the keys to success in the aviation industry is innovation. The designers should always try to find newer and better solutions instead of reusing the old ones.

Biomimicry or biologically inspired engineering, the field of scientific study dealing with the natural world, involving examining what can be extracted, learned and duplicated from nature is one of the best sources of inspiration, ideas and innovation. There is no waste in nature, any leftovers from a plant or animal becomes food for another sort of species. Cyles in nature are efficient and any impotency gets dissolved quickly. Human engineers and designers can oftentimes consult nature for ideas and solutions to contemporary aviation problems (Gunther, 2016). Much knowledge remains to be tapped from bird flight, and fishes to some extent in making use of movable wing surfaces for aeroplanes.

6.6.2 Beyond Visuals

Data and graphical visualisation in aerofoil design have been traditionally viewed as tools for exploring data and forming hypothesis, a tool for examining and understanding. In current times, both the mainstream of computer graphics and the availability of data sources via internet seem to have a significant impact in aeronautics, giving rise to the various possibilities of generating visual representations of data on basic computer systems. Professional and amateur designers alike have taken interest and expanded the imaginary horizons of artistic space, making use of scientific techniques to create computational tools and designs that actively aid analytical reasoning, supporting new approaches and innovative designs.

According to (Phillips & McQuarrie, 2004), the resilience of various visual techniques to persuade advanced designs is the new typology of visual rhetoric.

6.7 THINKING ENGINEER

With the evident limitations of mathematics and scientific knowledge, engineers tend to use their judgement and experience to solve problems with a constant stimulation to eliminate risk or to limit it. Complexity is ever present and most successful creations make room for human fallibility (Engineering Council UK, 2010).

An education is the primary means by which a profession puts its code into practice. What engineers do is important and there is a difference between engineers, scientists and management in their information and knowledge.

Historical engineering standards lay down rules of conduct for engineers. In general, engineers develop these standards because all engineers doing things the same good way is better than each engineer choosing her/ his own way. The standards so defined are not, however, likely to freeze in the way pure conventions do. Most change from time to time as experience generates new options or shifts the weight of evidence favouring this or that old one. Engineering professionals are usually not always as well informed as they should be when compared to scientists or researchers. Although what a profession requires is crucial for determining whether a professional actually has professional autonomy, what the professional supposes her/his profession to require is crucial for determining whether the professional feels that they have such an autonomy (M. Davis, 1998).

Interactive design denotes a definitive representation, sensory, creative delivery and implementation of computer-supported technological perceptions (cognition and sensorics) and actions (visualisation) in diverse media. Digital information will begin to take on physical form sooner or later and design engineers will eventually become the engineers of experience (Buurman, 2001).

6.7.1 Tenets of Critical Thinking

Systems thinking, problem finding and creative solving, improving, visualising and adaptability are vital engineering habits of mind. Innovative, creative problem solving was considered an important trait in a research conducted by Royal Academy of Engineering (RAE, 2014). Although engineers could be using concepts that are not original and be ingenious, it could be that some may not see themselves as being creative.

However, it is to be noted that there are various kinds of creative thinking, that which calls upon disciplined thinking and that which explores the propagation of new ideas.

Why?	Why will it be done? Justification, Reason
What?	What will be done? Action steps, Description
Where?	Where will it be done? Location, Area
When?	When will it be done? Time, Dates, Deadlines
Who?	Who will do it? Who's responsible for it?
How?	How will it be done? Method, Process
How much?	What will it cost to make? Costs, Expenses involved

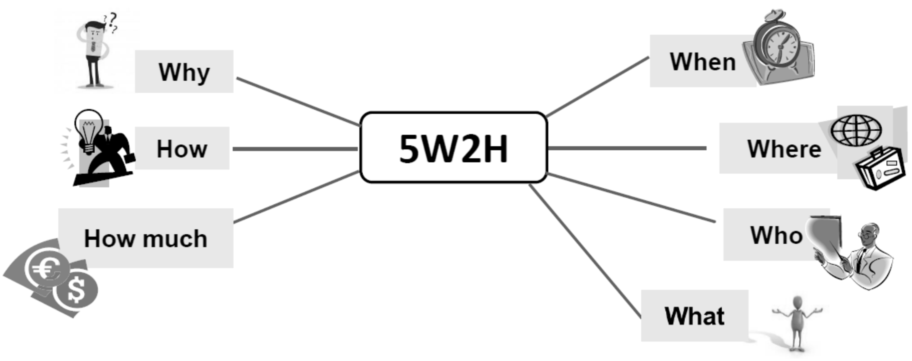


Figure 6.6: 5W2H method is a tool that helps in identifying a problem when trying to improve or optimise

Optimism, communication, collaboration and attention to principled considerations are also vital characteristics which aid an engineer to think and take action when faced with demanding problems. An engineer's work is most times about balancing series of tensions and the ability to move between two or more modes of reasoning and thought process.

The aim of education must be to influence the mind so that it may acquire good judgments on matters at hand. With regards to any subject proposed to be investigated, one must inquire not what others have thought, or what conjectures, but what could clearly be manifested and perceived by intuition or deduced with certainty.

5W2H is a method which asks questions whose answers are considered basic to information gathering on a subject for optimisation or problem solving (Figure 6.6). The questions are expected to have a factual answer and not a simple yes or no.

6.7.2 Starting With Why

The underlying factor throughout an optimisation process so far has been the importance of decision making. Decisions are to be taken effectively, quickly and more importantly, it is an absolute necessity to know what decision is to be made and why.

A critical link exists between the success of a design optimisation endeavour and the models or analysis used in solving any optimisation problem, its accuracy being only as accurate as the physical models used. Although a deemed successful optimisation may only require the model to predict the correct trends and not provide wholly accurate converged solutions throughout the design optimisation process, there is a decision step involved to check suggested optimal solutions with high level accuracy computational engineering methods at the end of the procedure.

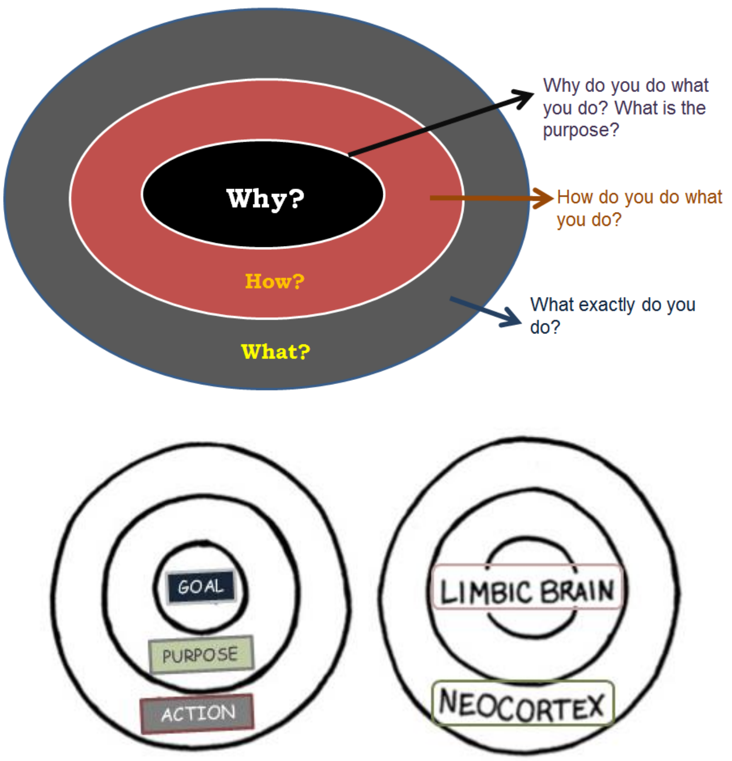


Figure 6.7: (top) golden circle showing why, how and what sections starting from inside out (below) parts of the brain connecting golden circle sections (Sinek, 2009)

Starting by asking 'why' (Sinek, 2009) explains that a purpose, cause or a belief that inspires an individual and ultimately an organisation or an industry, affects the product that it finally produces and sells.

The Neocortex of the human brain correlates to the 'what' level, responsible for analytical thinking, rationality and languages. The two intermediate sections comprise limbic brain, in charge for all human feelings like faith, trust, behaviour and decision making, but has no ability for languages (Figure 6.7).

When designers communicate from outside towards inwards, conveying 'what' is done first, it is relatively easy for others to discern a large amount of complicated information covering numbers, facts and figures but it does not drive behaviour. When communication is done inwards out, one communicates directly to the part of brain controlling decision making, and the brain's language section allows rationalising those decisions (Figure 6.7). Decision-making and capacity to analyse decisions exist in different parts of the brain, leading to the 'gut feeling'. Sometimes limbic brain drives behaviour that contradicts rational and analytical understanding of a situation.

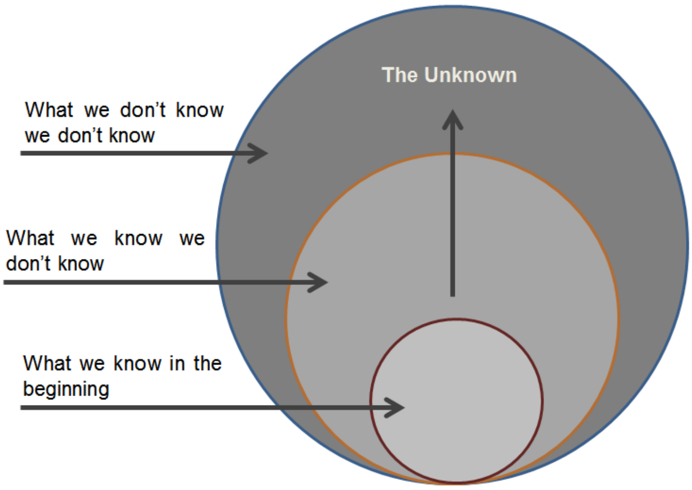


Figure 6.8: Agile engineering: 'Why' (purpose and requirements) is specific and tied to end value generation (source: crisp, 2016)

Starting from the inside out, an optimisation process could be trigged by starting with 'why', followed by 'how' and 'what'. When everyday work is defined just by 'what' a designer is doing instead of 'why', it can lead to stagnation of ideas, creativity and decision making. Knowing 'why' behind a task is a vital way to cultivate a lasting success, with a greater mix of flexibility and innovation.

With a clear destination, a human is capable of using one's own creativity, sense of innovation and problem solving skills to overcome hurdles in between, on the way, to get to the destination (Figure 6.8). Destination in the front is more vital than the route taken, which could be flexible and interactive optimisation offers this flexibility in design exploration.

6.8 INTERACTIVE Vs AUTONOMIC DESIGN

In using an interactive tool, the simulation is guided by the user in constructing a solution to the problem at hand by using a graphical user interface, while often on the back-end a time-and-memory consuming programme is being executed (Knezevic, Frisch, Mundani, & Rank, 2011). Interactivity always involves two entities; one of them by default being a human user. The act of making simple, working versions of what one is trying to build or modify forces the designer to uncover key problems, thus making room for the designer involved to find solutions, or to contemplate potential ideas to those problems much less expensively (Bricklin, 2017).

An autonomic computing tool will make decisions on its own, using high-level strategy codes, continually checking and optimising its status and automatically adapting itself to changing conditions. Such a tool would be most useful after a design freeze and during a production launch. The overarching goal of automated computing is to realise that a computer, software system or application can manage itself in accordance with high-level guidance from humans (Parashar & Hariri, 2005).

A human designer or decision-maker thus plays a crucial role whether a design tool or process adapted is either interactive or autonomous, albeit with varying levels of involvement in the design process. While high-end computational solvers, optimisation algorithms and visual techniques are making a human designer’s tasks less complicated, they do not make labour redundant, nor do they make a designer’s skills obsolete. Instead, jobs which are taken over by computers are constantly making room for new tasks because of a human’s inbuilt genius, creativity and insatiable desire which is constantly pushing engineering boundaries (Autor, 2016). Curiosity is the key. Today’s technologies were yesterday’s fantasies, and the same stands true for tomorrows.

6.9 WINGS OF THE FUTURE

The design tradition of using rudders, elevators, flaps, ailerons and spoilers to help an aeroplane move in a particular direction will change as designers harness tougher, stretching materials and wing-warping motors. Aerodynamics could be altered to fly efficiently at most altitudes if control surfaces can be designed to adaptively change an aeroplane’s wing camber geometry (Marks, 2016).

An increase in the demand for production and performance rates calls for profoundly new approaches to design and manufacture aeroplane wings. Next generation aeroplane wings may have a different shape or take on a new way of assembly, made of composites or advanced metallic materials. However, the industry is certain that with aeroplane production rates set to rise significantly, there will be a demand for faster wings, easier and cheaper to make and assemble.

The industry is also planning to study various wing sizes. Long, narrow wings tend to generate a high lift-to-drag ratio, improving fuel efficiency; however, several airport regulations limit the maximum length of the wings thus raising the need for experimenting with wing tips folded that could then be extended before flight and folded back when on ground. Such design research is underway and helps in assessing any gains in aerodynamic performance outbalancing the extra weight and costs (Airbus, 2017).

In the words of the Wing of the Future Programme at Airbus, “We need to have the courage to progress ideas. It’s not about what we know now; it’s about what we can achieve in the near future”.

6.10 CONCLUSIONS

Essential findings are discussed in section 6.2 along with observations that were drawn by experiments and theoretical study. Qualitative conclusions of interactive optimisation were drawn in Chapter 5 by way of three trade study approaches, sections 5.6 and 5.7. Limits of interactive optimisation and its countering means are analysed in sections 6.3 through 6.9. Scope for furthering this research is presented in section 6.12.

Key questions mentioned in section 1.3 are answered by the following:

- Interactive optimisation makes use of a chosen mathematical programming methodology which searches for a preferred pareto optimal solution in accordance with the decision-maker’s choices. A maximum process advantage can be achieved by allowing the computer to conduct logical operations and let the human decision-maker who is biological with an emotional intelligence characteristic to take advantage and steer the logical processes.

- A high-lift design trade study generates a lot of numbers; these data are an important source of information and knowledge. With the rise in multi-objective and multi-disciplinary algorithms, suitable visualisation support is a necessity. Visualisation techniques equipped with interactive aids that interpret and help understand these data are vital for a designer.

- In order to maintain a multi-disciplinary approach throughout a design phase by addressing various objectives, a decision-maker is tasked with vital choices that will influence a design decision and could impact an end product. A designer’s approach towards a design task, knowledge, skills, personality etc. will not only determine the outcome of the task but will reverberate through any disruptive advances in design and manufacture of new generation of aeronautical designs.

This work tackles the challenges of EFT project by introducing an optimisation approach for determining and improving an aerofoil's maximum lift in the context of multi-objective optimisation. It supports the knowledge and use of better tools and techniques for design optimisation by taking into account the human engineer of the present and future behind various virtual engineering tasks. Aims and objectives mentioned in section 1.4 have been thoroughly addressed in this research work and are summarised as follows:

- Several non-interactive optimisation techniques were already being experimented in an high-lift design analysis context while corresponding visual techniques were either in their inception stage or not yet introduced. An interactive method, multi-objective particle swarm optimisation technique has been introduced for high-lift design through preceding works which has been modified and improved through this research.

- Due to the nature of algorithms adapted in the module, several errors had to be dealt with and eliminated. New visualisation tool features along with a possibility for the user to experiment various multi-element aerofoil sets has been added. Ease of use and robustness are improved.

- Study of designer interface, tools and machine analysis has been discussed through graphical user interface and human-machine interaction. SHEL model explains human factors.

- Three broad interactive approaches were used to study task analysis on two multi-element aerofoil sections with six parameters and two objectives; conscious and automatic behaviors of decision-maker are explained along with the influences of optimser and visual techniques.

- Optimisation module has been improved for designer feedback. Any interaction in parallel coordinate plot reflects in scatter plot. Axes selection for scatter plot also reflects for heat map and radviz plots in combined view page.

- Several visualisation methods have been introduced, some of which find their use for generic applications while some are specific for viewing approximation sets. New visual techniques were added along with a tab displaying a combination view. Advantages and disadvantages of various techniques are mentioned along with commonly encountered errors and problems. A table comparing various methods against desirable visual features has also been presented.

- Complexity of optimisation problems and the vital role of decision-maker in design and analysis context have been addressed according to the purpose of interaction and the role of the user in the process. Designer-in-the-loop engineering explains information perceived, its interpretation and interaction.

- Limits of optimisation, reliability, validation and uncertainties have been addressed along with what a decision-maker could do to tailor optimised configurations for technological benefits. A good utilisation of an aerodynamic tool is only as good as its user or decision-maker and driving continuous optimisation requires leveraging the advantages of both the human and the machine in an interactive setting.

6.11 CONTRIBUTION TO KNOWLEDGE

The subject of interactive optimisation, where the prime components are a machine and a human is relatively new and has customarily been of interest to those involved in logical mathematics and computational techniques. Aviation industry has long been using computational tools for aerodynamics and its related tasks. Only recently has the industry started to exploit its own abilities on advancing software tools for aeronautical design activities by borrowing and implementing ideas and processes that are being used in non-aviation related applications, a trend triggered by the tremendous evolution in computational technology. The use of any such tools involves a human designer or decision-maker who interacts to bring forth new ideas and products aiding technological improvements and innovation.

Much study on human factors has been carried out with respect to flight crew and aircraft maintenance engineers in the aviation sector, however there is a breach of knowledge and information on how design engineers who are the brains behind several aviation products behave and interact with respect to their work environment, specifically modelling & simulation which involves the use of software tools.

This research work addresses that gap by way of an exemplar aerodynamics engineer in a high-lift design and analysis context. While several others have covered the use of various optimisation techniques for aerodynamics, this work tested the possibility of adapting particle swarm technique as one of the probable optimisers that could be used in a high-lift design tool module. The work demonstrates the advantages of an interactive approach over traditionally used non-interactive ways and analyses three approaches that a designer is likely to adapt in an optimisation context.

The research sheds light and draws attention to a designer's way of insight, preferences, analysis and decision-making, interactive communication and steering of a design process with its respective objectives and constraints. Several potential visualisation techniques are introduced in this work for a multi-objective optimisation framework, some of which are not yet being used in aerodynamics' tools while others are being exploited but not yet fully developed for aeronautical applications. Visualisation, with its phases and characteristics are only now being explored by the aviation community; this work contributes towards that graphical literacy awareness by explaining various errors, problems, search and display patterns, reading and interpretation.

Themes such as relationship between data, information, knowledge, understanding and wisdom, uniqueness and limitations of human brain, critical thinking have been brought together to elucidate an aeronautical design engineer's work setting, dilemmas and limitations. Several decision making tools and models are already employed in the aviation sector, most of which are constantly renewed by the arrival of new techniques and old ones dying out. A simple approach is introduced in this work to encourage designers in their thought process and decision making.

6.12 SCOPE FOR FUTURE WORK

Interactive optimisation is an emerging field, especially for aeronautical applications. There exists much scope for borrowing ideas from other fields and technologies that have already proven the advantages of interactive design exploration and optimisation. The test and application of several established optimisation techniques today are yet to be explored for aeronautical implementation.

Optimisation algorithms can be adjusted easily to the problem at hand. Almost any aspect of an algorithm could be modified and customised. On the other hand, although much research has been done on which algorithm is best suits a given problem; several practical questions continue to remain. While standard values generally provide a reasonably good performance, different configurations may yield more useful results and insight but will also take up time and resources.

There is scope for the current module to be migrated to Windows OS, however the current adapted version of MSES solver does not allow this possibility as it is coded for Linux environment. The visualisation framework could be migrated from Django to HTML. The complete tool module which is currently locally used could be extended to run on a local or restricted communications network (intranet) to control both front-end and back-end applications. The present module uses more than one language which generates several integration errors; using a single language and a customised single platform for all functionalities with better interactive features will be advantageous.

Other MOO/MDO visualisation techniques can be tested along with other interactive optimisation methods such as Nimbus, Surrogate, GDF, Guess,Tchebycheff+. The module could also be improved to execute more than one solver type for multi-element aerofoils.

Python, as of today, is one of the most widely used high-level, general-purpose, dynamic programming languages. It is able to run on a variety of systems, allowing the use of several third-party tools. One of its greatest strengths is its standard libraries, suited to many tasks. It offers several tools for an aeronautical designer to explore and this area of tool exploration could be traversed as interactive and innovate as possible as a designer's knowledge, skill, creativity and interest allows.

Apart from the few combinations that have already been tested, improvements on the existing I-MOPSO code and integrating it with various interface codes such as solvers, grid generators, and parameterisation techniques for a wider range of aerofoil designs can be further studied with 2D aerofoil models and further explore possibilities with 3D.

There is scope for studying practical aspects concerning the difficulty of a designer to submit preference information, in particular when numeric values have to be defined, and the potential prospect of reusing the same information for solving different problem cases. There is room for undertaking study experiments involving groups of design engineers or making use of several user data analytics already gathered by the industry.

Also, a general limit of interactive multi-objective optimisation is relying on the process of exploring Pareto front. There is room for expanding on the tendency of limited number of trade-offs explored and the effect that the systematic presence of a loss in some objective values generated during iterations. Designers work with several unconscious biases, like considering the losses in an objective value more important than equivalent gains (e.g. lift vs weight in an aerofoil design).

Others areas of further study include interactive decision making models for multi-objective optimisation, learning systems, advantages and disadvantages of offering ‘more choices for a designer’, critical thinking in engineering, brain and visual models, their interpretation, and design analysis with the help of established techniques such as six-sigma.

Apart from combined visualisation, hybrid and metaphor based optimisation algorithms leave much room for consideration.

Development and Integration of source code control with minimum effort to maximise usability and effective future developments secures innovative and competitive technological edge.

As part of work flow implementation and management, intelligent and interactive process controlling and monitoring, heterogeneous system platforms, minimising integration effort of any additional computational tools, making effective use of cloud computing and internet, improving reliability robustness to at least 99% and availability of multi-user operation capability are areas that could be improved.

BLANK

References

Abbott, K.H., 2001. Human Factors Engineering and Flight Deck Design. *The Avionics Handbook*, p.Chapter 9.

AGARD-CP-515. (1993). *High Lift System Aerodynamics*. Neuilly Sur Seine.

Agarwal, G., Lewis, K., Chugh, K., & Huang, C. (2004). Intuitive Visualization of Pareto Frontier for Multi-Objective Optimization in n-Dimensional Performance Space. *AIAA Symposium of Multidisciplinary Analysis and Optimization*.

Ahmed, S. (2001). *Understanding the Use and Reuse of Experience in Engineering Design,Ph.D Thesis*. Cambridge.

AIAA-G-077-1998. (2002). Guide for the verification and validation of computational fluid dynamics simulations. *AIAA*, *G-077-1998*.

Airbus. (2015). Meshing Philosophy, Internal published article.

Airbus. (2017). Wing of the future- No winging it here, Internal published article.

AIRBUS, E. (2014). *WISDOM Presentation: Wide Ranging Industrial Shape Design and Optimisation Module*, Internal document.

AirbusUK, & AltranUK. (2013). WISDOM User Guide 1.0, (1D1), Internal document.

Alba, E., & Martí, R. (2006). Metaheuristic Procedures for Training Neural Networks, 53–70.

Alfaris, A. (2010). *Multidisciplinary System Design Optimization (MSDO)*. USA.

Anderson, J.R. & Lebiere, C., 1998. *The Atomic Components of Thought*, Lawrence Erlbaum Associates.

Anderson Jr., J.D., 1995. Computational fluid dynamics- The basics with applications. *McGraw-Hill, Inc.*, 27(6), pp.1661–71.

Andreasen, M. ., & Hein, L. (2000). Considerations on a Design Typology. *3rd International Workshop IPD*.

Aurelius, M. (2003). *Meditations*. Modern Library.

Autor, D. (2016). *Will Automation Take Away All Our Jobs?* Cambridge: TED Talks.

Autran, L., & Gomes, M. (2011). *Teaching Decision Making Analytical Skills to Engineers: A New Paradigm*.

Balasani, M. J. (2015). *Interactive Multi-Objective Optimisation for High Lift Design, 1st*

Progress Review Report, Ph.D. Cranfield University.

Balasani, M. J. (2016). *Aircraft Design Optimisation with its Designer at Heart, 2016 Applied Aerodynamics Conference, Evolution & Innovation Continues, The Next 150 Years of Concepts, Designs and Operations, Royal Aeronautical Society*.

Balasani, M. J. (2016). *Interactive Optimisation for High-Lift Design, 2nd Progress Review-24 Months Report, Ph.D*. Cranfield University.

Balasani, M. J. (2017). *Interactive Optimisation for High-Lift Design, 3rd Progress Review-PPT Presentation, Ph.D*. Cranfield University.

Balicki, J., 2007. Tabu Programming for Multiobjective Optimization Problems. *IJCSNS International Journal of Computer Science and Network Security*, 7(10), pp.44–51.

Barthelemy, J.-P., Bisdorff, R., & Coppin, G. (2002). Human Centred Processes and Decision Support Systems. *European Journal of Operational Research*, *136*(2), 233–252.

Basseur, M. et al., 2006. Metaheuristics for Multiobjective Combinatorial Optimization Problems : Review and recent issues. *INRIA*, 5978, p.39.

Bellinger, T., Castro, D., & et al. (2004). *Data, Information, KNowledge and Wisdom*.

Benayoun, R., de Montgolfier, J., Tergny, J., & Laritchev, O. (1971). Linear Programming with Multiple Objective Functions: Step Method (STEM). *Mathematical Programming*, *1*(1), 366–375.

Blackwell, A. (2010). *Human-Computer Interaction, Lecture Notes*. Cambridge.

Blackwell, A. (2011). Visual Representation, The Encyclopedia of Human-Computer Interaction, 2nd Edition

Blum, C., & Roli, A. (2003). Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys*, *35*(3), 189–213.

Bondarev, A. ., & Galaktionov, V. . (2014). Current Visualization Trends in CFD Problems. *Applied Mathematical Sciences*, *8*(28).

Bonneau, G.-P., Hege, H.-C., Johnson, C. ., Oliveria, M. ., & Potter, K. (2014). Overview and State-of-the-Art of Uncertainty Visualization. In *Scientific Visualization* (pp. 3–27). London: Springer.

Boutilier, C., Regan, K., & Viappiani, P. (2009). Online feature elicitation in interactive optimization. *26th International Conference on Machine Learning*, *382*, 10.

Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Branke, J., Kalyanmoy, D., Miettinen, K., & Słowiński, R. (2008). *Multiobjective Optimization: Interactive and Evolutionary Approaches*. *Springer*.

Breitkopf, P.,Coelho, R., (2013). Multidisciplinary Design Optimization in Computational

Mechanics, Wiley Publication

Bricklin, D. (2017). *Meet the inventor of the electronic spreadsheet*. Beacon Street: TED Talks.

Brochu, E., Brochu, T., & Freitas, N. De. (2010). A Bayesian Interactive Optimization Approach to Procedural Animation Design. *Symposium on Computer Animation*, 103–12.

Brodlie, K., Osorio, R., & Lopes, A. (2012). A Review of Uncertainty in Data Visualization. In *Expanding the Frontiers of Visual Analytics and Visualization* (pp. 81–109). London: Springer.

Buurman, G. . . (2001). *Total Interaction: Theory and practice of a new paradigm for the design disciplines* (1st ed.). Brickhaeuser.

Calafiore, G.C, Ghaoui, L.E. (2014). Optimization Models, Cambridge University Press.

Cantwell, B. (2015). *Thin Airfoil Theory*. California.

Card, S., Mackinlay, J. & Shneiderman, B., 1999. *Readings in Information Visualization: Using Vision to Think*, Kaufmann.

Carrese, R. (2012). *Identifying Preferred Solutions for Multi-Objective Aerodynamic Design Optimization, Ph.D Thesis*.

Chankong, V. & Haimes, Y.Y., 2008. *Multiobjective Decision Making: Theory and Methodology*, Dover Publications.

Chen, S., Amid, D., Shir, O., & Schreck, T. (2013). Self-Organizing Maps for Multi-Objective Pareto Frontiers. *Visualization Symposium*.

Chernoff, H., 1973. Using Faces to Represent Points in K-Dimensional Space Graphically. *American Statistical Association*, 68, pp.361–368.

Chimani, M., Lesh, N., Mitzenmacher, M., Sidner, C., & Tanaka, H. (2005). A case study in large-scale interactive optimization. *Artificial Intelligence and Applications (AIA)*, *24-29*, 24–29.

Chong, E. K. ., & Zak, S. . (2001). *An Introduction to Optimization* (2nd ed.). John Wiley & Sons.

Christian, B. (2011). *The Most Human Human: What Artificial Intelligence Teaches Us About Being Alive*. Penguin.

Cockburn, A. & McKenzie, B., 2000. An Evaluation of Cone Trees. *Human-Computer Interaction*, XIV, pp.425–436.

Computing, P. (n.d.). Parallel Computing at a Glance. *Processing*, 1–13.

Connor, A.M. & Tilley, D.G., 1998. A Tabu Search Method for the Optimisation of Fluid Power Circuits. *IMechE Journal of Systems and Control*, (212), pp.373–381.

Cossio, M. L. T., Giesen, L. F., Araya, G., Pérez-Cotapos, M. L. S., VERGARA, R. L., Manca, M., … Héritier, F. (2003). *Complexity and Approximation*. *Springer*. Springer.

CSEF. (2016). Tables & Graphs, Resources Page, Colorado State Science Fair

Cummings, R.M. et al., 2015. *Applied Computational Aerodynamics- A Modern Engineering Approach*,

Davis, F. ., & Kottemann, J. . (1994). User Perceptions of Decision Support Effectiveness: Two Production Planning Experiments. *Decision Sciences*, *25*(1).

Davis, M. (1998). *Thinking Like an Engineer: Studies in the Ethics of a Profession (Practical and Professional Ethics)*. Oxford University Press.

De Oliveria, M., & Levkowitz, H. (2003). From Visual Data Exploration to Visual Data Mining: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, *9*(3).

Dickinson, S., Leonardis, A., Schiele, B., & Tarr, M. (1999). Object Categorization: Computer and Human Vision Perspectives.

Dix, A., Finlay, J., Abowd, G. ., & Beale, R. (2004). *Human-Computer Interaction*. Pearson Education Limited.

Drela, M. (1998). *Pros & Cons of Airfoil Optimization*. Massachusetts.

Dreyfus, Hu., Dreyfus, S., & Athanasiou, T. (2000). *Mind Over Machine: The power of human intuition and expertise in the era of the computer*. Simon and Schuster.

Dtom. (2007). Airfoil Lines.

Durillo, J. J., Garcia-Nieto, J., Nebro, A. ., Coello Coello, C. ., Luna, F., & Alba, E. (n.d.). *Multi-Objective Particle Swarm Optimizers: An Experimental Comparison*.

Edelman, S. (1999). *Representation and Recognition in Vision*. Cambridge: MIT Press.

Edison. (2013). The Pathway to Quieter, More Productive Wind Turbines.

Edwards, D. E. (1989). Three-Dimensional Visualization of Fluid Dynamic Problems. *AIAA'89*.

Emmerich, M., & Deutz, A. (2006). *Multicriteria Optimization and Decision Making*. Netherlands.

Erdos, P., & Posa, L. (1966). The representation of a graph by set intersections. *Canadain Journal of Mathematics*, *18*(1), 106–112.

Endsley, M.R., 2000. *Situation Awareness Analysis and Measurement*, Routledge.

Faraclas, E., & Koehler, C. (2005). The Engineering Decision Making Model: Its Importance As Applied Through The Context Of A World War II Simulation. *American Socirty for Engineering Education Annual Conference & Exposition*.

Few, S. (2006). *MultiVariate Analysis Using HeatMaps*.

Fass, D., 2006. Rationale for A Model of Human Systems Integration: The Need of A

Theoritical Framework. *Journal of Integrative Neurosciences*, 5(3), pp.333–354.

Fisher, M. . (1986). Interactive Optimization and Microcomputers. *Annals of Operations Research 5(1985/6)*, 541–556.

Flaig, A., & Hilbig, R. (1991). *High Lift Design for Large Civil Aircraft*. Airbus.

Flanker. (2007). Airfoil Geometry.

Flynn, M. . (2003). *Numeric Data Representation*.

Gemignani, Z., 2010. Better Know a Visualization: Parallel Coordinates. *JuiceAnalytics*.

Gennings, C. et al., 1990. Interpreting Plots of a Multidimensional Dose Response Surface in Parallel Coordinates. *Biometrics*, 46, pp.719–735.

Geoffrion, A.M., Dyer, J.S. & Feinberg, A., 1972. An Interactive Approach for Multi-Criterion Optimization, with an Application to the Operation of an Academic Department. *Managment Science*, (19), pp.357–368.

Gersh, J. ., McKneely, J. ., & Remington, R. . (2005). Cognitive Engineering: Understanding HUman Interaction with Complex Systems. *JOhns Hopkins APLTechnical Digest*, *26*(4).

Gianluca, L. (2007). *Simulation of Turbulent Flows*.

Giles, M. (2017). *CPUs and GPUs- Lecture*. Oxford.

Glover, F., 1990. Programming Integer Algorithms, Heuristics. *Interfaces*, 20(4), pp.79–94.

Glover, F. & Laguna, M., 1997. *Tabu Search*, Springer.

Golia, C., & Viviani, A. (2011). Airfoil Airflow Analysis of a High Lift System with Vortex Blob Method. *AIAA 2011-31010*.

Gunther, S. (2016). Eight Amazing Examples of Biomimicry.

Hansen, C. ., & Anderasen, M. . (2004). A Mapping of Design Decision-Making. *International Design Conference*.

Hansen, M.P., 1997. Tabu Search for Multiobjective Optimization : MOTS. *MCDM'97*.

Hearn, D. ., & Baker, P. (1991). Scientific Visualization. *Eurographics Technical Report Series*.

Heisler, Y. (2016). How Powerful is the Human Brain Compared to a Computer.

Hettenhausen, J., Lewis, A., & Kipouros, T. (2014). A Web-based System for Visualisation-driven Interactive Multi-objective Optimisation. *14th International Conference on Computational Science*, *29*(1), 1915–1925. http://doi.org/10.1016/j.procs.2014.05.176

Hettenhausen, J., Lewis, A., Randall, M., & Kipouros, T. (2013a). Interactive multi-objective particle swarm optimisation using decision space interaction. *IEEE Congress on Evolutionary Computation*, *1350*, 3411–3418. http://doi.org/10.1109/CEC.2013.6557988

Hettenhausen, J., Lewis, A., Randall, M., & Kipouros, T. (2013b). Interactive Multi-Objective Particle Swarm Optimisation Using Decision Space Interaction.

Hettenhausen, Jan, Lewis, A. & Mostaghim, S., 2010. Interactive Multi-Objective Optimization with Heat-Map Visualization Based User Interface. *Engineering Optimization*, 42(2), pp.119–139.

Hoffman, D. . (1998). *Visual Intelligence*. USA: W.W Norton & Company.

Holt, A. (1982). On Making Things the Best- Aeronautical Uses of Optimization. *AIAA*, *19*(AIAA 81-1738R).

Hutter, F., Bartz-Beielstein, T., Hoos, H. ., Leyton-Brown, K., & Murphy, K. . (2010). Sequential Model-Based Parameter Optimization: An Experimental Investigation of Automated and Interactive Approaches. In *Empirical Methods for the Analysis of Optimization Algorithms* (pp. 361–411). Springer.

Ibrahim, A., Rahnamayan, S., Martin, M.-V., & Deb, K. (2016). 3D-RadVis: Visualization of Pareto Front In Many-Objective Optimization. *WCCI*.

IEGD, European Institute for Managing Diversity., 2006, Diversity and Innovation: A business opportunity for all.

ILA-boy. (2008). NACA Profil, Internet page.

Inselberg, A., 2009. *Parallel Coordinates*, Springer.

Inselberg, A., 1985. *The Visual Computer: The Plane with Parallel Coordinates*,

Jaeggi, D. et al., 2005. A Multi-objective Tabu Search Algorithm for Constrained Optimisation Problems. *Lecture Notes in Computer Science*, 3410(October 2015), pp.490–504.

Jaeggi, D.M. et al., 2008. The development of a multi-objective Tabu Search algorithm for continuous optimisation problems. *European Journal of Operational Research*, 185(3), pp.1192–1212.

Jain, H., & Deb, K. (2014). An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, *18*(4), 602–622.

Järvinen, P., Puolamäki, K. & Siltanen, P., 2009. *Visual Analytics*, VTT Technical Research Centre of Finland.

Jastrow, J., (December, 1917), The Phychology of Conviction, Journal Article, The Scientific Monthly, Vol.5, No.6, P 523-544

Javed, W., & Elmqvist, N. (2012). *Exploring the Design Space of Composite Visualization*.

Jenkins, B. (2014). Design Exploration Vs. Design Optimization.

Jin, R., Chen, W., & Simpson, T. W. (n.d.). Comparitive Studies of Metamodeling Techniques Under Multiple Modeling Criteria.

Johansson, J. (2008). *Efficient Information Visualization of Multivariate and Time-Varying Data*. Linkoping University.

Johnson-Laird, P. . (2010). *Mental Models and Human Reasoning*. Princeton.

Johnson, K., Murphy, S., O'Hara, C., & Shirey, K. (2015). *Four Phases of the Engineering Design Process in Math and Science Classrooms*.

Kaliszewski, I., & Michalowski, W. (1999). Searching for Psychologically Stable Solutions of Multiple Criteria DEcision Problems. *European Journal of Operational Research*, *118*, 549–562.

Kapoor, A., Lee, B., Tan, D., & Horvitz, E. (2010). Interactive optimization for steering machine classification. *CHI Conference*, 1343–1352.

Kann, V. (1992). *On the approximability of NP-complete optimization problems*. Stockholm: PhD Thesis.

Karray, F. et al., 2008. Human-Computer Interaction: Overview on State of the Art. *International Journal on Smart Sensing and Intelligent Systems*, 1(1), pp.137–159.

Keane, A. ., & Nair, P. . (2005). *Computational Approaches for Aerospace Design: The Pursuit of Excellence*. John Wiley & Sons.

Keim, D. . (1997). Visual Techniques for Exploring Databases. *International Conference on KNowledge Discovery and Data Mining Tutorial Program*.

Keim, D. et al., 2008. Visual Analytics : Definition , Process , and Challenges. , pp.154–175.

Keim, D.A. et al., 2006. Challenges in Visual Data Analysis. *Proceedings of Information Visualization, IEEE*, IV, pp.9–16.

Keim, Daniel et al., 2010. *Mastering the Information Age: Solving Problems with Visual Analytics*, Mansmann.

Kerren, A., Ebert, A. & Meyer, J., 2006. *Human-Centered Visualization Environments*, Springer.

Khan, M. & Khan, S., 2011. Data and information visualization methods, and interactive mechanisms: A survey. *International Journal of Computer Applications*, 34(1), pp.1–14.

Kim, S., Alonso, J.J. & Jameson, A., 2004. Multi-Element High-Lift Configuration Design Optimization Using Viscous Continuous Adjoint Method. *Journal of Aircraft*, 41(5), pp.1082–1097.

Kinnear, D., Atherton, M., Collins, M., Dokhan, J., & Karayiannis, T. (2006). *Colour in Visualisation for Computational Fluid Dynamics*. UK.

Kipouros, T. (2012). Stochastic Optimisation in Computational Engineering Design. *EVOLVE- A Bridge between Probability, AISC*, *1*(175), 475–490.

Kipouros, T. (2003). Multi-Objective Aerodynamic Design Optimisation. *PhD Thesis 1st Year Report*, 152.

Kipouros, T., 2006. Multi-Objective Aerodynamic Design Optimisation. *PhD Thesis*, pp.81–87.

Kipouros, T., Inselberg, A., Parks, G., & Savill, A. M. (2013). Parallel Coordinates in Computational Engineering Design. *AIAA Multidisciplinary Design Optimization Specialists*, *1750*, 1–11. http://doi.org/10.2514/6.2013-1750

Kipouros, T., & Lewis, A. (2014). Understanding the complexities in Turbomachinery design by visualizing the discontinuous areas in the design parameter space. *OPT-I*.

Kipouros, T., Molinari, M., Dawes, W. N., Parks, G. T., Savill, M., & Jenkins, K. W. (2007). An Investigation of the Potential for Enhancing the Computational Turbomachinery Design Cycle Using Surrogate Models and High Performance Parallelisation. *Proceedings of ASME Turbo Expo 2007*, 1415–1424.

Kirwan, B. (1994). *A Guide to Practical Human Reliability Assessment*. London. Taylor & Francis Publication.

Kishi, Y., Kanazaki, M., Makino, Y., & Matusushima, K. (2016). Planform Dependency on Airfoil Design Results for Supersonic and Transonic Wing. *Journal of Flow Control, Measurement & Visualization*, 10.

Klamroth, K. & Miettinen, K., 2008. Integrating Approximation and Interactive Decision Making in Multicriteria Optimization. *Operations Research*, 56(56).

Klau, G., Lesh, N., & Marks, J. (2002). The HuGS platform: A toolkit for interactive optimization. *Proceedings of the …*, 4–10.

Kleine, H., & Groenig, H. (1981). Color Schlieren Methods in Shock Waves. *Shock Waves, Band 1*.

Knezevic, J., Frisch, J., Mundani, R.-P., & Rank, E. (2011). Interactive Computing Framework for Engineering Applications. *Journal of Computer Science*, *7*(5), 591–599.

Koksalan, M., & Karahan, I. (2010). An interactive territory defining evolutionary algorithm: ITDEA. *IEEE Transactions on Evolutionary Computation*, *14*(5), 702–722.

Kosara, R., Hauser, H., & Gresh, D. L. (2003). An Interaction View on Information Visualization. *Eurographics*, 123–137.

Krause, J., Perer, a, & Bertini, E. (2014). INFUSE: Interactive Feature Selection for Predictive Modeling of High Dimensional Data. *Visualization and Computer Graphics, IEEE Transactions on*, *PP*(99), 1.

Kriwaczek, F., & Rustem, B. (2000). *Interactive Multiple Objective Decision Making Based On Quadratic Programming*.

Larichev, O.I., 1992. Cognitive Validity in Design of Decision-Aiding Techniques. *Journal of Multi-Criteria Decision Analysis*, 1, pp.127–138.

Lane, D. . (1995). *Visualization of Numerical Unsteady Fluid Flows*. CA.

Laramee, R., & Hauser, H. (2005). *Geometric Flow Visualization Techniques for CFD Simulation Data*.

Legriel, J., Le-Guernic, C., Cotton, S., & Maler, O. (2010). Approximating the Pareto Front of Multi-Criteria Optimization Problems. *Tools and Algorithms for the Construction and Analysis of Systems*, *6015*, 69–83.

Luque, M. et al., 2007. Equivalent Information for Multiobjective Interactive Procedures. *Management Science*, 53, pp.125–134.

Luque, M., Ruiz, F., & Miettinen, K. (2007). GLIDE: General Formulation for Interactive Multiobjective Optimization. *Helsinki School of Economics*, (W-432).

MacEahren, A. M. (1992). Visualizing Uncertain Information. *Catographic Perspective*, *13*, 10–19.

Martinelli, L. & Jameson, A., 2013. Computational Aerodynamics : Solvers and Shape Optimization. , 135(January), pp.1–9.

Marks, P. (2016). The Shape of Wings to Come.

Mavris, D.N. et al., 2010. Systems Design and Modeling : A Visual Analytics Approach. *27th Congress of International Council of the Aeronautical Sciences ICAS*, pp.1–28.

Martinelli, L., & Jameson, A. (2013). Computational Aerodynamics : Solvers and Shape Optimization, *135*(January), 1–9. http://doi.org/10.1115/1.4007649

Mason, W. H. (2006). *Transonic Aerodynamics of Airfoils & Wings*.

McCormick, B. ., DeFanti, T. ., & Brown, M. . (1987). *Visualization in Scientific Computing*. USA.

McLean, D., (2013 Boeing). Understanding Aerodynamics: Arguing from the Real Physics, Wiley Publication.

McLoughlin, T., Laramee, R., Peikert, R., Post, F., & Chen, M. (2010). Over Two Decades of Integration-Based, Geometric Flow Visualization. *Computer Graphics Forum*.

Meignan, D., Knust, S., & Gaud, N. (2015). A Review and Taxonomy of Interactive Optimization Methods in Operations Research. *ACM Transactions On Interactive Intelligent Systems*, *5*(3).

Miettinen, K., 1999. *Nonlinear Multiobjective Optimization*, Springer.

Moggridge, B. (2007). *Designing Interactions*. The MIT Press.

Murphy, V. (2016). The DIKW Model for Knowledge Management.

Nakayama, H. & Sawaragi, Y., 1984. Satisfacting Trade-Off Method for Multiobjective Programming. *Interactive Decision Analysis*, pp.113–122.

Narzisi, G. (2008). *Multi-Objective Optimization*.

Nielsen, J., 1994. *Usability Engineering*, Kaufmann.

Norman, D. . (2002). *The Design of Everyday Things*. Basic Books.

Oberkampf, W. ., & Trucano, T. (2002). *Verification and Validation in Computational Fluid Dynamics*.

Pahl, G., & Beitz, W. (1996). *Engineering Design- A Systematic Approach*. London.

Palmer, M.T. et al., 1995. A Crew-Centered Flight Deck Design Philosophy for HIgh-Speed Civil Transport (HSCT) Aircraft. *NASA Tech Memo 109171*.

Parashar, M., & Hariri, S. (2005). Autonomic Computing: An Overview. In *Unconventional Programming Paradigms* (pp. 257–269). Springer Berlin Hiedelberg.

Parasuraman, R. (1997). Humans nad Automation: Use, Misuse, Disuse, Abuse. *Human Factors*, *39*(2), 230–253.

Parthasarthy, G., Shishira, N. ., & Dinesh, K. . (2015). Computational Fluid Analysis for High Lift on Multi-Element Airfoil Used in Civilian Aircraft. *International Journal of Science and Research*, *4*(7).

Pascal, B. (1670). *Pensées, Decision Theory*.

Paulus, M., Hozack, N., Zauscher, B., McDowell, B., Frank, J., Brown, L., & Braff, G. (2001). Pre-frontal, parietal, and temporal cortex networks underlie decision making in the presence of uncertainty. *NeuroImage*, *13*, 91–100.

Peng, W., Ward, M., & Rundensteiner, E. (2004). Clutter Reduction in Multi-Dimensional Data Visualization Using Dimension Reordering. *INFOVIS, IEEE Symposium on Information Visualization*, 89–96.

Phillips, B. ., & McQuarrie, E. . (2004). *Beyond Visual Metaphor: A New Typology of Visual Rhetoric In Advertising*.

Poles, S. (2008). *Multi-Objective Optimization and Decision Making Process in Engineering Design*.

Post, F. ., & Van Walsum, T. (1993). Fluid Flow Visualisation. *Focus on Scientific Visualization*.

Rawlins, T., Lewis, A., Hettenhausen, J., & Kipouros, T. (n.d.). Enhancing ANN-guided MOPSO through Active Learning.

Rawlins, T., Lewis, A., Hettenhausen, J., & Kipouros, T. (2014). Enhancing MOPSO through the guidance of ANNs. *2014 International Joint Conference on Neural Networks (IJCNN)*, 4003–4010.

Reason, J. (1990). *Human Error*. Cambridge University Press.

Reason, J.T., 1997. *Managing the risks of Organizational Accidents*, Ashgate.

Reckzeh, D. (2004). Aerodynamic Design of Airbus High-Lift Wings In A MultiDisciplinary Environment. *European Congress on Computational Methods in Applied Sciences and Engineering*.

Redmill, F. & Rajan, J., 1997. *Human Factors: In Safety-Critical Systems*, Routledge.

Reehuis, E., Olhofer, M., Sendhoff, B., & Baeck, T. (2013). Learning Guided Exploration in Airfoil Optimization. *IDEAL*.

Ren, D., & Tobias, H. (2014). Expressive Interactive Design of Information Visualizations. *IEEE Computer Graphics and Applications*, *20*(12), 2092–2101.

Ries, E., & Hickman, L. (2012). *The Lean Startup*. London.

Rini, DP., Shamsuddin, SM., Yuhaniz, SS., Particle Swarm Optimization: Technique, System and Challenges. (2011). International Journal of Computer Applications (0975-8887), volume 14-No.1.

Roberts, B. ., & Del Vecchio, W. . (2000). The Rank-Order Consistency of Personality Traits from Childhood to Old Age: A Quantitative Review of Longitudinal Studies. *Psychological Buletin*, *126*(1).

Rothlauf, F. (2011). *Design of Modern Heuristics*. *Springer Publication*.

Royal Academy of Engineering, May 2014. *Thinking Like An Engineer: Implications For The Education System,* UK

Rumsey, C. ., Long, M., Stuever, R. ., & Wayman, T. . (2010). *Summary of the First AIAA CFD HIgh Lift Prediction Workshop*. Chicago.

Sadraey, M. (2012). *Aircraft Design: A Systems Engineering Approach*. Wiley Publications.

Sakawa, M., 1982. Interactive Multiobjective Decision Making by the Sequential Proxy Optimization Technique. *European Journal of Operational Research*, 9.

Salim, S. ., Ong, K. ., & Cheah, S. . (2011). Comparison of RANS, URANS and LES in the Prediction of Airflow and Pollutant Dispersion. *Proceedings of the World Congress on Engineering and Computer Science*, *2*.

Samtaney, R. (1999). *Visualization, Extraction and Quantification of Discontinuities in Compressible Flows*. Moffett Field, CA.

Sandberg, R. (2010). Aerofoil Noise.

Schneiderman, B. & Plaisant, C., 2009. Treemaps for Space Constrained Visualization of

Hierarchies.

Schulz, H.-J., Nocke, T., Heitzler, M., & Schumann, H. (2013). A Design Space of Visualization Tasks. *IEEE Computer Society*.

Schwartz, B. (2015). *Why We Work*. Simon and Schuster/ TED.

Scott, S. D., Lesh, N., & Klau, G. W. (2001). Investigating Human-Computer Optimization. *CHI Conference*.

Shan, S., & Wang, G. G. (2010). Metamodeling for High Dimensional Simulation-Based Design Problems. *Journal of Mechanical Design*, *132*(May 2010), 051009.

Shaffer, C. ., Knill, D. ., & Watson, L. . (1998). Visualization for Multiparameter Aircraft Designs. *IEEE*.

Shapiro, & Stockman. (2000). Perceiving 3D from 2D Images. In *Computer Vision*.

Sharp, H., Rogers, Y., & Preece, J. (2007). *Interaction Design: Beyond Human-Computer Interaction*. John Wiely & Sons.

Shimoyama, K., Kawai, S., & Alonso, J. J. (2013). Dynamic Adaptive Sampling Based on Kriging. *AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1–28.

Sinek, S. (2009). *Start With Why- How Great Leaders Inspire Everyone To Take Action*. Penguin.

Smythe, J. (2012). *Roulette Wheel Particle Swarm Optimization*. University of Georgia.

Soo Yi, J., Kang, Y., Stasko, J., & Jacko, J. (2007). Toward a Deeper Understanding of the Role of Interaction in Information Visualization. *IEEE*.

SSC. (2000). *Informative Presentation of Tables, Graphs and Statistics*. Reading.

Steinhauer, G. . (1986). *Artificial Behaviour: Computer Simulation of Psychological Processes*. Prentice Hall.

Tabors, R. ., & Steinberg, A. (2000). *Trade-Off Analysis*.

Tan, D., & Nijholt, A. (2010). Brain-Computer Interfaces and Human-Computer Interaction. In *Brain-Computer Interfaces* (1st ed.). London: Springer-Verlag London.

Tan, P.-N., Steinbach, M. & Kumar, V., 2005. *Introduction to Data Mining*, Addison Wesley Longman.

Tarvainen, K., 1984. On the Implementation of the Interactive Surrogate Worth Trade Off (ISWT) Method. *Interactive Decision Analysis*, pp.154–161.

Thakur, T. (2015). *Requirement Engineering in HCI for Human Error Prevention*.

Thomas, J. ., & Cook, Kristin, A. (2005). *Illuminating the Path: The Research and Development Agenda for Visual Analytics*.

Thomas, J. J., & Cook, K. (2005). Illuminating the Path: The R&D Agenda for Visual Analytics. *IEEE*.

Tilocca, G. (2016). *Interactive Optimisation for Aircraft Applications, MSc Thesis*. Cranfield.

TKU. (1995). *Airfoil Design*. Taiwan.

Trapani, G., 2009. Multi-Objective Optimization of 2D High-Lift Airfoil Configurations using Tabu Search. *MSc Thesis*.

Tufte, E.R., 1983. *The Visual Display of Quantitative Information*, Graphics Press.

Tusar, T., & Filipic, B. (2014). Visualization of Pareto Front Approximations in Evolutionary Multiobjective Optimization: A Critical Review and the Prosection Method. *IEEE Transactions on Evolutionary Computation*.

UK, R. C. (2014). The Enhanced Fidelity Transonic Wing (EFT).

Van Dam, C. . (2002). The Aerodynamic Design of Multi-Element High-Lift Systems for Transport Airplanes. *Progress in Aerospace Sciences*, *38*, 101–144.

van Wijk, J. J. (2002). Image Based Flow Visualization. *ACM Transactions on Graphics*.

Versteeg, H.K. & Malalasekera, W., 2007. An introduction to computational fluid dynamics: the finite volume method. *McGraw-Hill, Inc.*

Vicente, K.H., 1999. Cognitive Work Analysis: Towards Safe, Productive and Healthy Computer-Based Work. *Mahwah, NJ*.

Viel, M. (2016). Adaptation of the DIKW Pyramid by US Army Knowledge Managers.

Wallenius, J. (1975). Comparative Evaluation of Some Interactive Approaches to Multi-Criterion Optimization. *Managment Sciences*, *21*(12), 1387–1396.

Wang, G. G., & Shan, S. (2007). Review of Metamodeling Techniques in Support of Engineering Design Optimization. *Journal of Mechanical Design*, *129*(4), 370.

Ware, C., 2000. *Information Visualization: Perception for Design*, Morgan Kaufman.

Wierzbicki, A.P., 1997. On the Role of Intuition in Decision Making and some Ways of Multi Criteria Aid of Intuition. *Journal of Multi-Criteria Decision Analysis*, (6), pp.65–78.

Wierzbicki, A.P., 1999. Reference Point Approaches. *Journal of Multi-Criteria Decision Analysis*, pp.9–39.

Will, J., & Perng, Y. (2011). *Optimization in ANSYS Workbench*.

Wing-Yi Chan, W. (2006). *A Survey of Multi-Variate Data Visualization*. Hong Kong University of Science & Technology.

Witowski, K., Liebscher, M., & Goel, T. (2009). Decision Making in Multi-Objective Optimization for Industrial Applications- Data MIning and Visualization of Pareto Data. *7th European LS-DYNA Conference*.

Wittgenstein, L. (1958). *Philosophical Investigations*. Prentice Hall.

Wong, K.-C., Wu, C.-H., Mok, R. K. ., & Peng, C. (2011). Evolutionary Multi-Modal Optimization Using the Principle of Locality. *Information Sciences*, *194*.

Yang, J.B. & Li, D., 2002. Normal Vector Identification and Interactive Tradeoff Analysis Using Minimax Formulation in Multiobjective Optimization. *IEEE Transactions on Systems, Man & Cybernetics*, (32), pp.305–319.

Zenithair., 2013, STOL CH701 Design, Leading edge wing slats, Internet page.

Zhou, H., Yuan, X., Qu, H., Cui, W., & Chen, B. (2008). Visual Clustering in Parallel Coordinates. *IEEE-VGTC Symposium on Visualization*, *27*.

Ziemkiewicz, C., Ottley, A., Crouser, J., Chauncey, K., & Su, S. . (2012). *Understanding Visualization by Understanding Individual Users*. NL.

Zionts, S. & Wallenius, J., 1976. An Interactive Programming Method for Solving the Multiple Criteria Problem. *Managment Sciences*, (22), pp.652–663.

Appendix

A. Conference Paper & Posters

“Aircraft Design Optimisation with its Designer at Heart”

2016 Applied Aerodynamics Conference: Evolution & Innovation Continues- The next 150 years of concepts, design and operations.
Royal Aeronautical Society, 150th anniversary year | 19-21 July 2016, Bristol, United Kingdom

“Aircraft Design Optimisation and the Impact of its Designers in Driving Development and Innovation”

Paper was shortlisted for ‘Go for Gold’ Challenge 2016 organised by RAeS for participants under 30 to mark 150th Anniversary.

Poster: *“Interactive Optimisation for High Lift Design”*

ERCOFTAC Osborne Reynolds Day 2016
The University of Manchester, United Kingdom | July 2016
Note: Poster Finalist

Poster: *“Aircraft Design Optimisation and the Impact of its Designers in Driving Development and Innovation”*

The Airbus Flight Physics Distributed Partnership R&T, DiPaRT 2016
Centre for Modelling & Simulation (CFMS), Bristol & Bath Science Park | 21-23 November 2016, Bristol, United Kingdom

----

B. Pictures

Following are pictures that were generated or used as part of the research work in various intermediate presentations and reports but were not added to the primary content of the thesis. They are fairly self-explanatory and support various sections presented across this document.

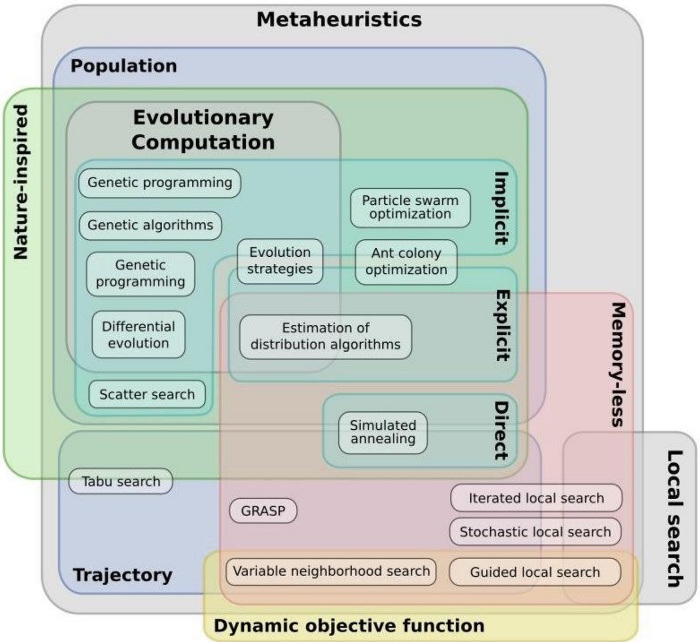


Figure B1: Different classification of metaheuristics; various high level strategies that guide a set of simpler search processes (Source: Image by Nojhan)

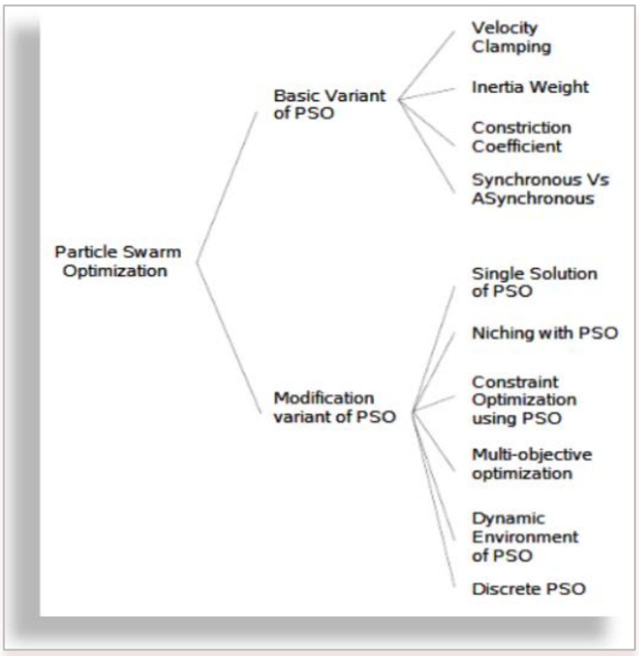


Figure B2: Variants of Particle Swarm Optimisation (Source: Rini et al., 2011)

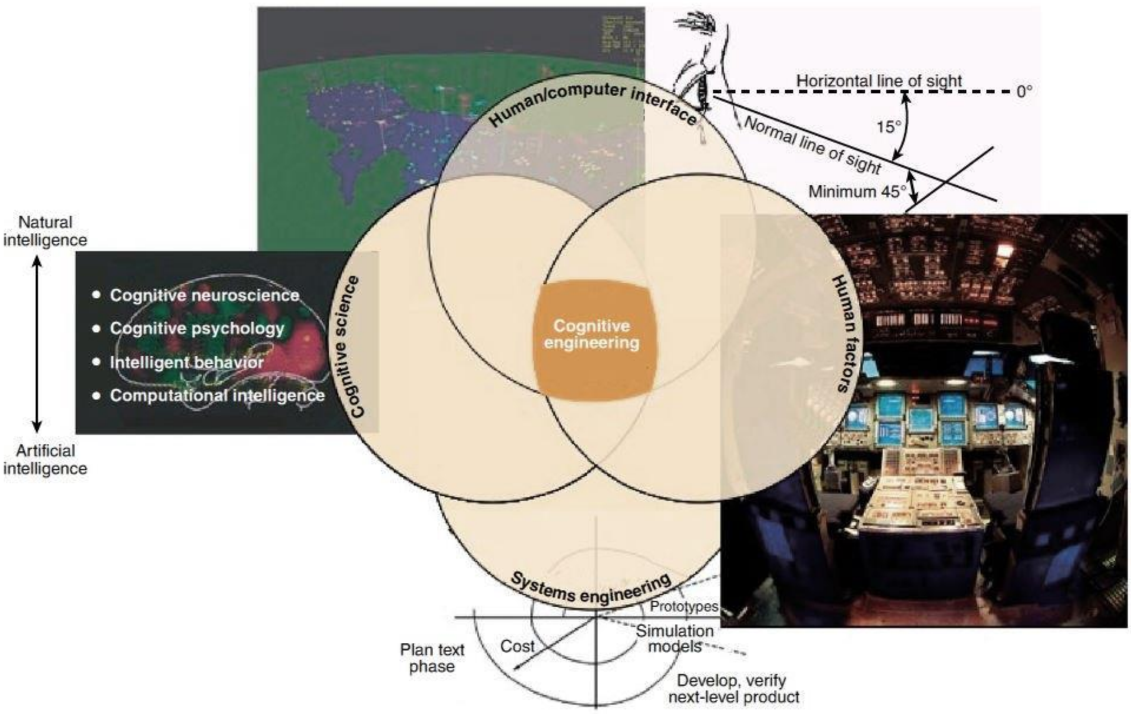


Figure B3: Overview of cognitive science and engineering Source: (Gersh et al. 2005)

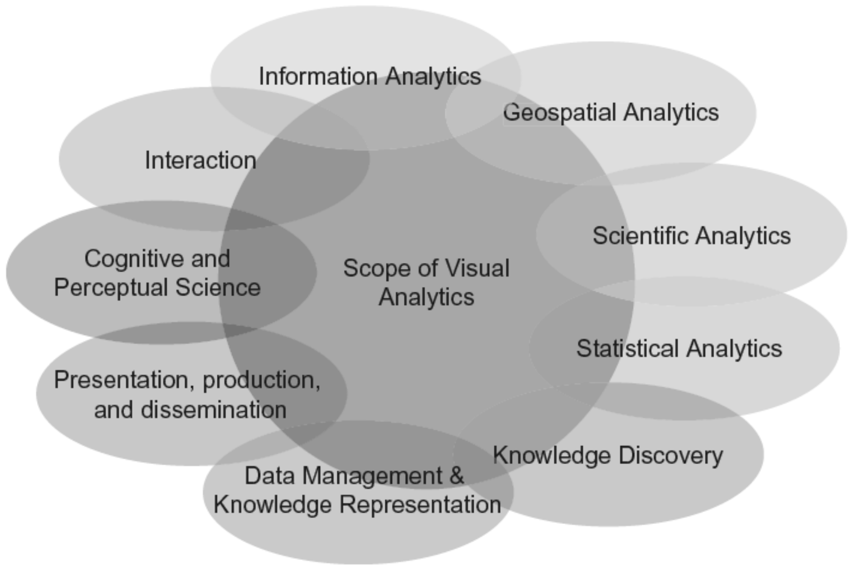


Figure B4: Interdisciplinary fields of visual analytics (Source: Keim et al. 2008 and Keim et al. 2006)

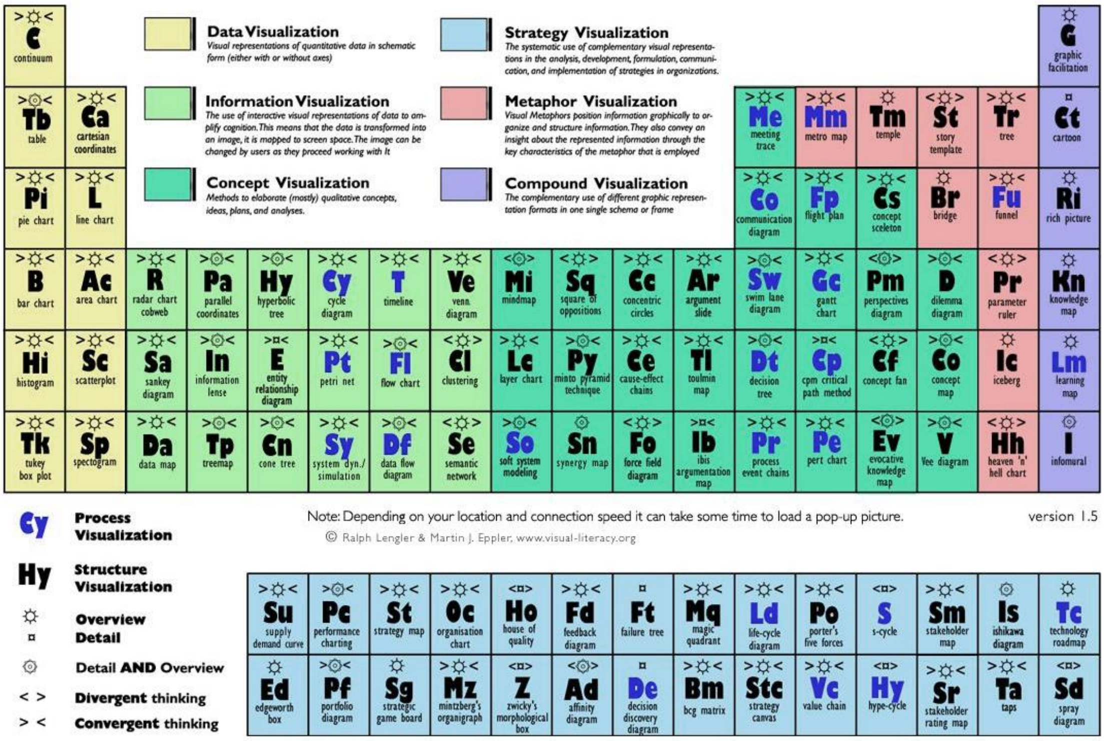


Figure B5: A periodic table of visualisation methods (source: Visual Literacy)

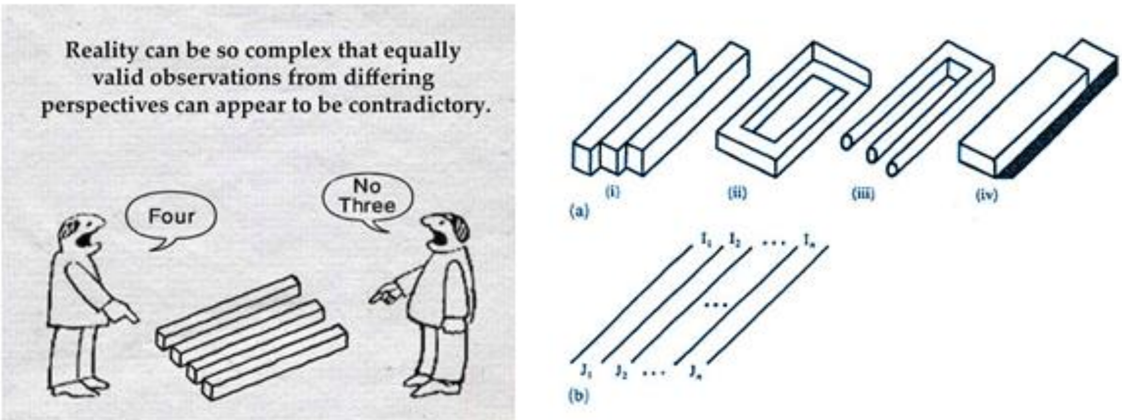


Figure B6: Visual Illusions: three or four bars? (author unknown) | Striped figures (source: abc-people)

Figure B7: Various steps of a typical computational aerodynamics' trade study (source: Airbus, 2014)

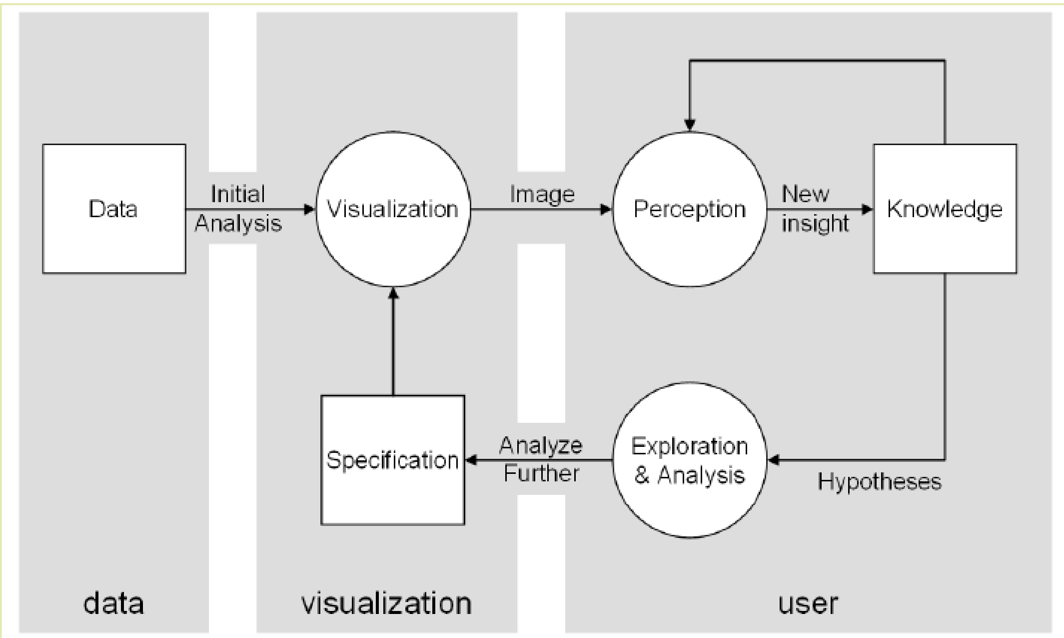


Figure B8: The sense-making loop for visual analytics based on the simple model of visualization by Wijk42 (source: adapted from Keim et al., 2008)

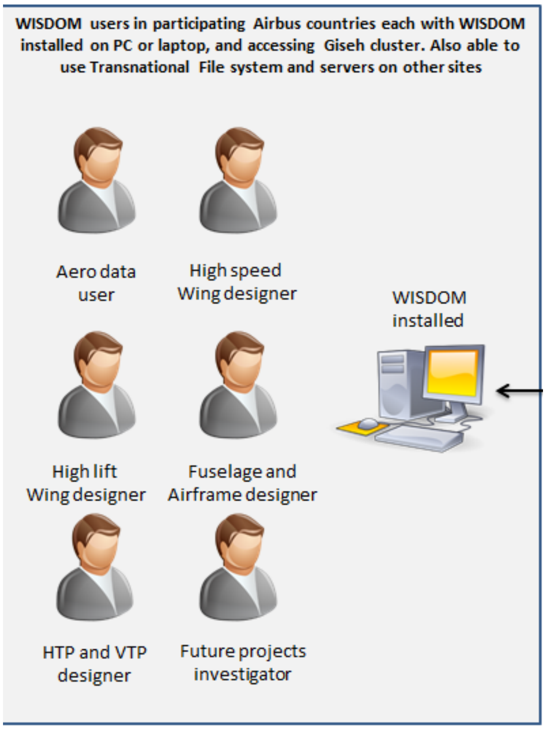


Figure B9 (above): WISDOM tool various users' architecture (source: Airbus)

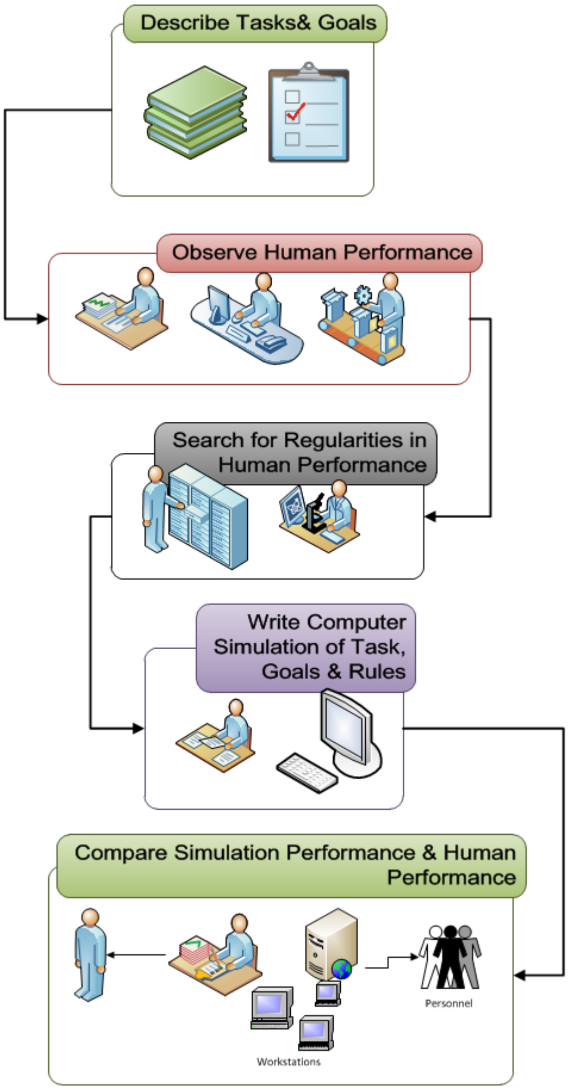


Figure B10 (right): Five general steps in developing Artificial Intelligence and Artificial Behaviour computer programs. Steps 2 to 5 define a loop that is repeated until the computer functions at some predefined level of accuracy (Steinhauer, 1986)

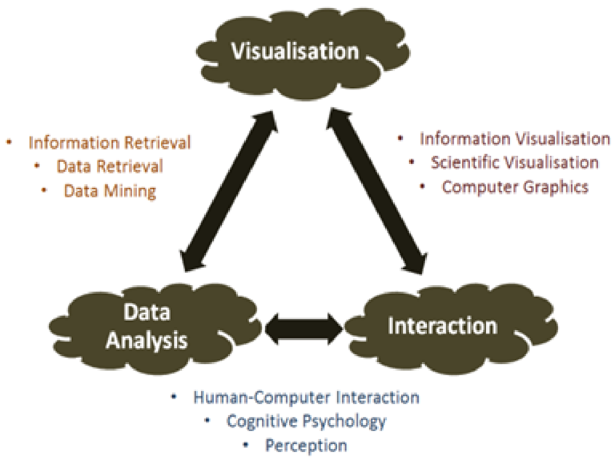
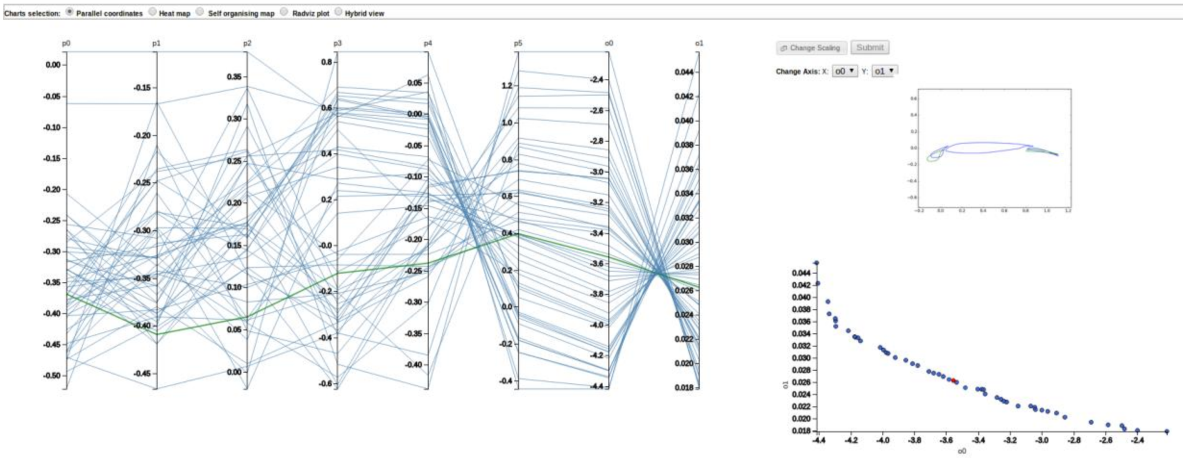
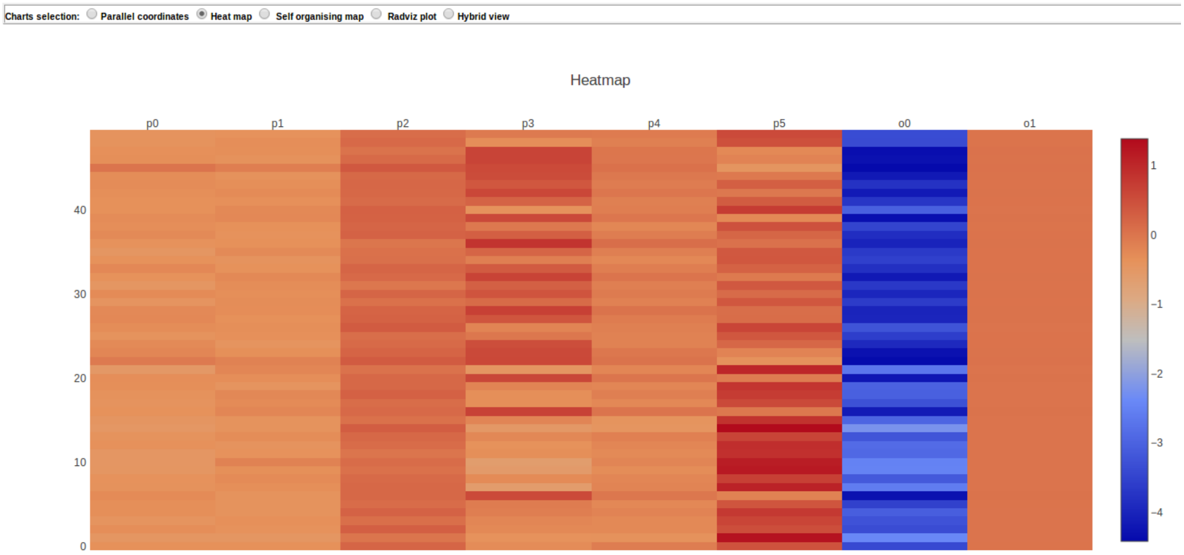


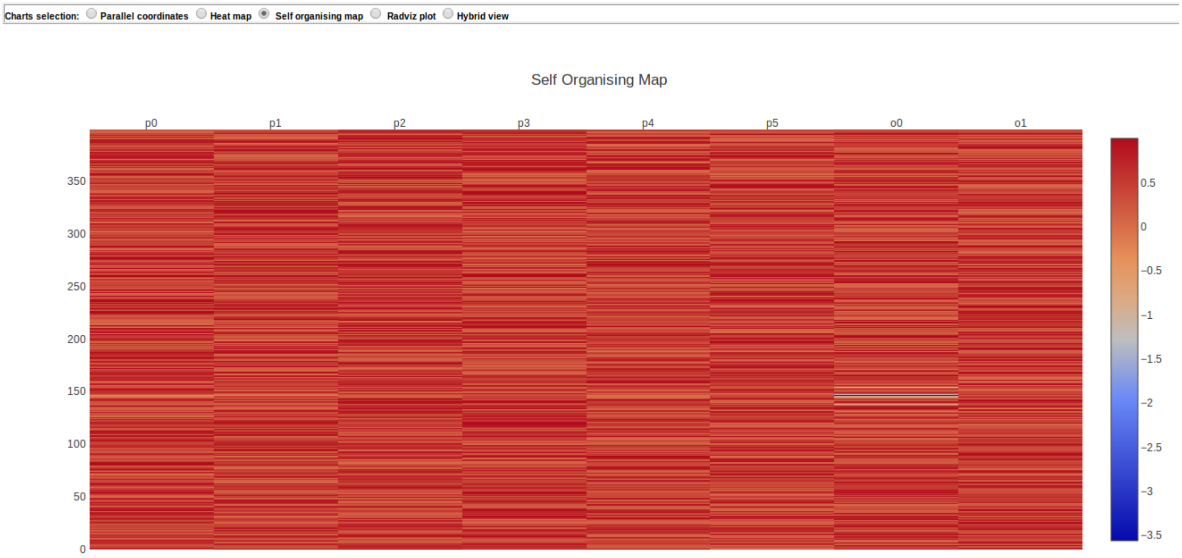
Figure B11: Automated data analysis: user-in-the-loop and visual data exploration (source: Keim et al., 2008, adapted picture)

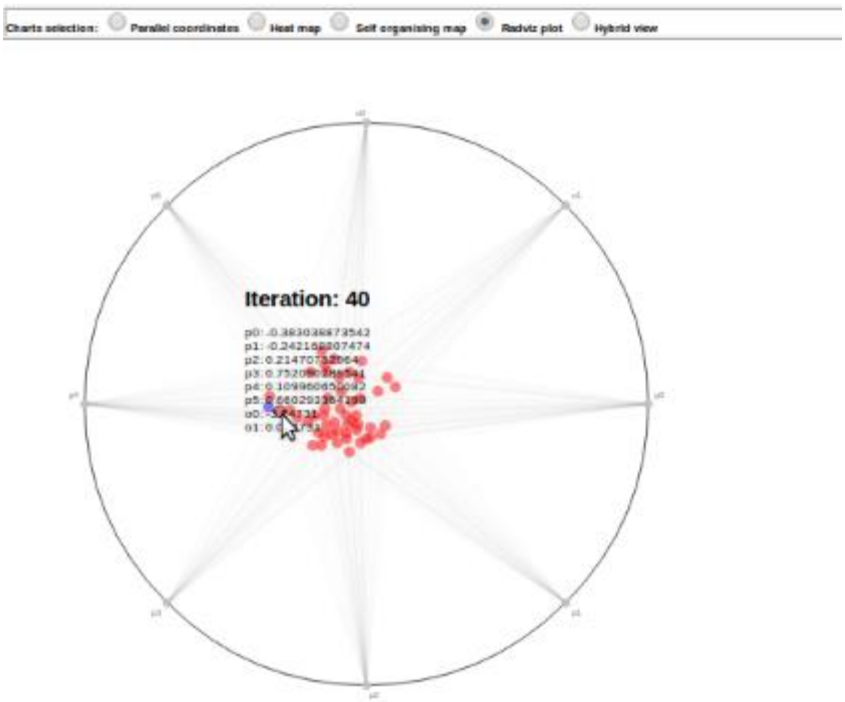
C. Test Runs Extra

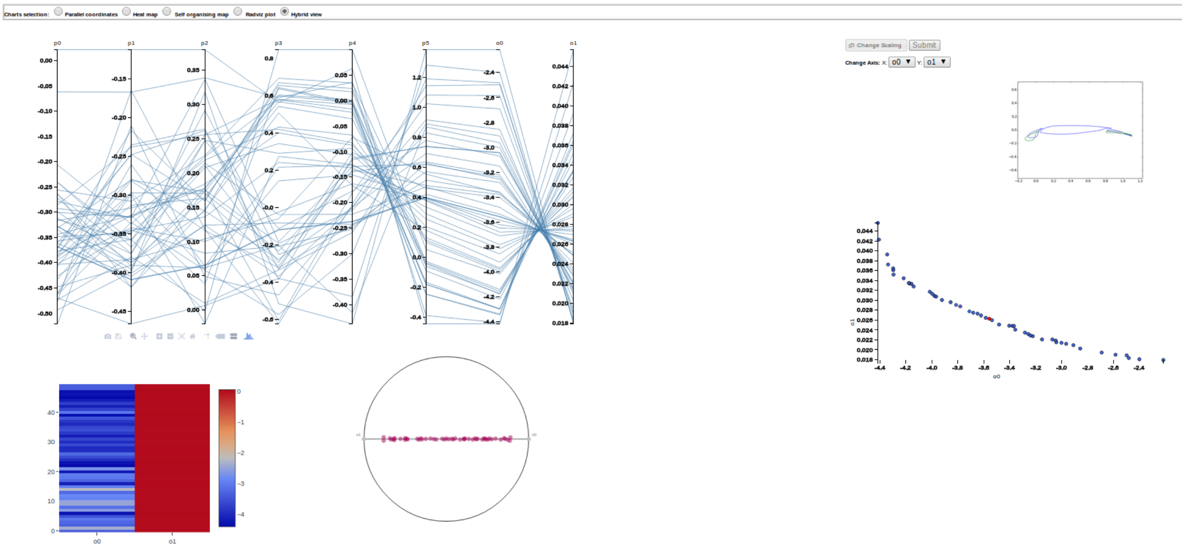
Figures C1: Garteur aerofoil section: 100 iterations, 60 particles, with an interval of 5, interaction starting at 20. The results were generated for maximum lift according to the optimiser's configuration.



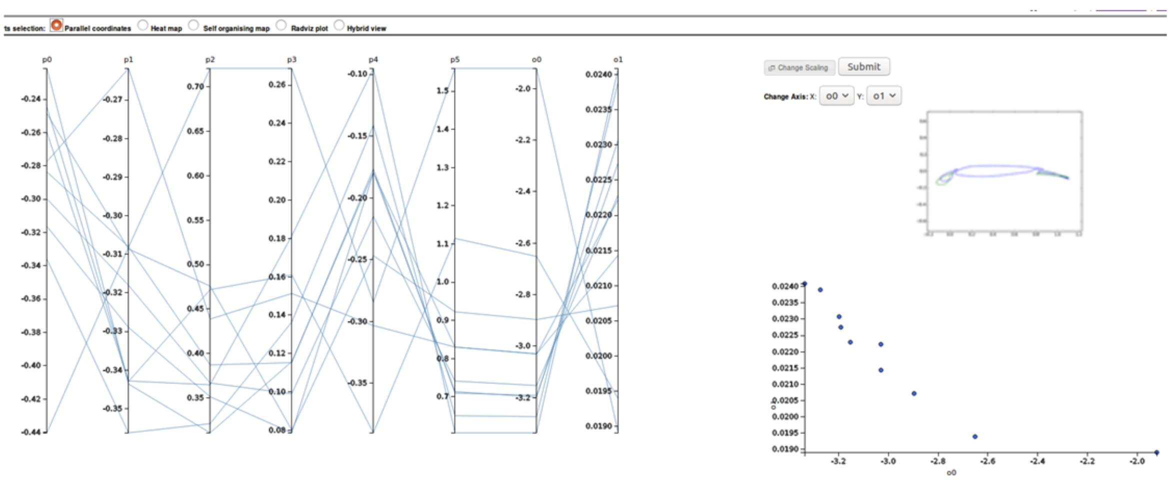


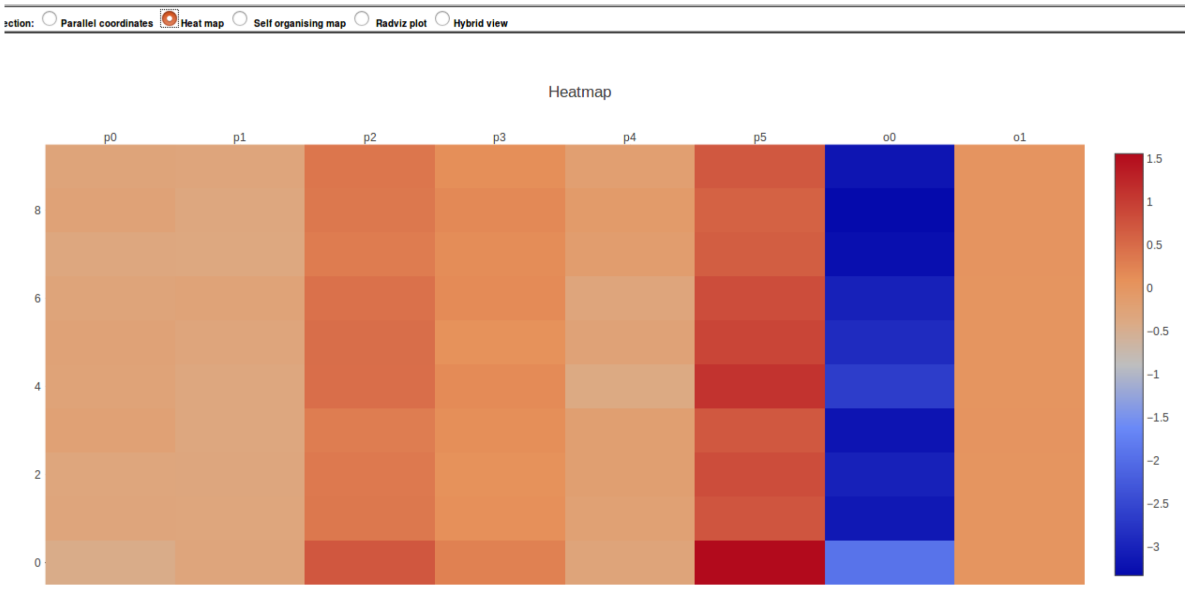

Charts selection: Parallel coordinates Heat map Self organising map Radviz plot Hybrid view
Self Organising Map
p0
p1
p2
p3
p4
p5
o0
o1
350
300
250
200
150
100
50
0
0.5
0
−0.5
−1
−1.5
−2
−2.5
−3
−3.5

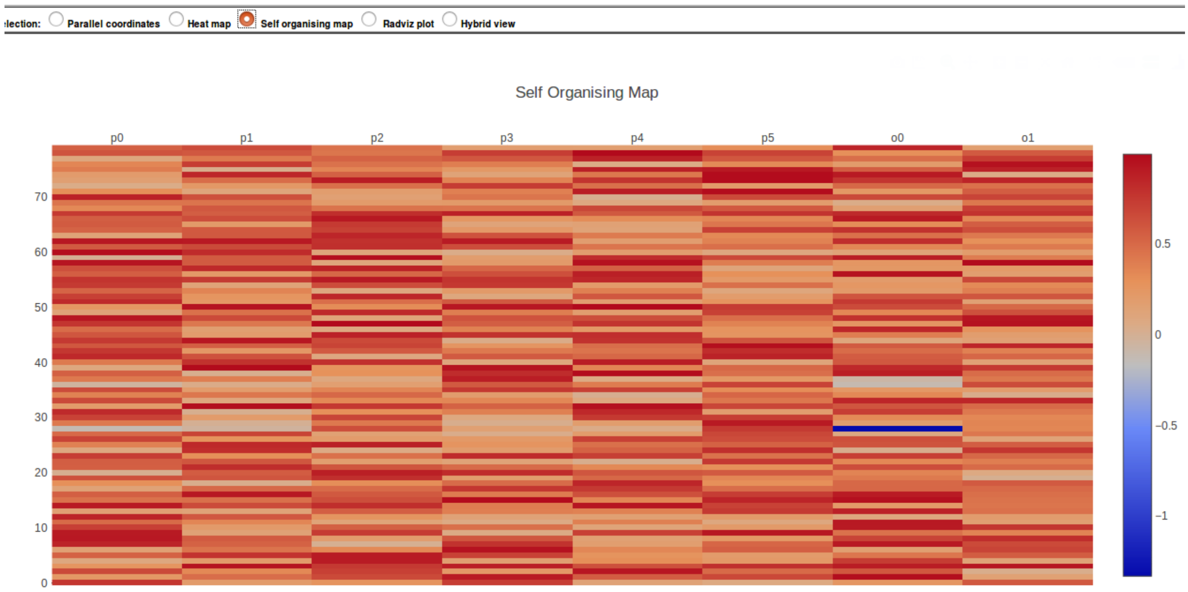

Charts selection: Parallel coordinates Heat map Self organising map Radviz plot Hybrid view
Iteration: 40

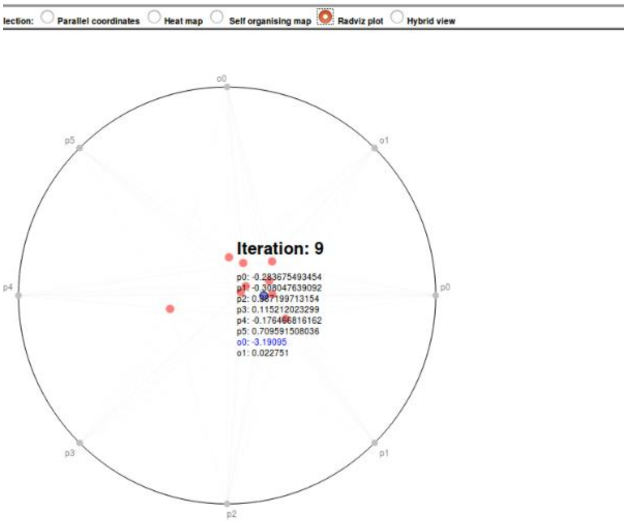


Figures C2: SC2-0610 aerofoil section: 50 iterations, 25 particles, with an interval of 5, interaction starting at 10. The results were generated for maximum lift according to the optimiser's configuration.




ection: Parallel coordinates Heat map Self organising map Radviz plot Hybrid view
Heatmap
p0 p1 p2 p3 p4 p5 o0 o1
8
6
4
2
0
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−2.5
−3


election: Parallel coordinates Heat map Self organising map Radviz plot Hybrid view
Self Organising Map
p0 p1 p2 p3 p4 p5 o0 o1
70
60
50
40
30
20
10
0
0.5
0
−0.5
−1


Parallel coordinates
Heat map
Self organising map
Radviz plot
Hybrid view
Iteration: 9
p0: -0.283675493454
p1: -0.308047639092
p2: 0.387199713154
p3: 0.115212023299
p4: -0.176466816162
p5: 0.709591508036
o0: -3.19095
o1: 0.022751