Shape Optimisation of a Li-Ion Satellite Battery Structure and Mass-Spring-Damper

MECH5030M Team Project – Individual Report
Shape Optimisation of a Li-Ion Satellite Battery Structure and Mass-Spring-Damper
Author: Mathias D'souza 201003870
Supervisor: Dr. Greg de Boer
Industrial Mentor: John Dobson & Daniel Murphy
Examiner: Andrew Shires
Date: 27/04/2021

SCHOOL OF MECHANICAL ENGINEERING **UNIVERSITY OF LEEDS**

MECH5030M TEAM PROJECT 60 credits

TITLE OF PROJECT

Shape Optimisation of a Li-Ion Satellite Battery Structure and Mass-Spring-Damper

PRESENTED BY

Mathias D'souza

OBJECTIVES OF PROJECT

1) Conduct a literature review into shape optimisation; **2)** Carry out a PCA to identify the key parameters of the MSD-Battery; **3)** Establish a DOE; **4)** Build a validated metamodel from the DOE; **5)** Perform MOO on the metamodel; **6)** Use the optimisation framework to carry out shape optimisation of the baseline NARMER battery module; **7)** Deliver findings and methods for improvement.

IF THE PROJECT IS INDUSTRIALLY LINKED TICK THIS BOX AND PROVIDE DETAILS BELOW ☑

COMPANY NAME AND ADDRESS:
EnerSys ABSL
Cullham Science Centre
Abingdon
OX14 3ED

INDUSTRIAL MENTOR:
John Dobson & Daniel Murphy

THIS PROJECT REPORT PRESENTS OUR OWN WORK AND DOES NOT CONTAIN ANY UNACKNOWLEDGED WORK FROM ANY OTHER SOURCES.

SIGNED  DATE 27/04/2021

Table of Contents

Table of Contents iii
Acknowledgements v
Abstract vi
Nomenclature vii
Chapter 1 – Introduction 1
1.0 Introduction 1
1.1 Aims 2
1.2 Objectives 2
1.3 Project Report Layout 2
Chapter 2 – Literature Review 3
2.0 Introduction 3
2.1 Design of Experiments 3
2.1.1 Overview 3
2.1.2 Classical DOEs – Full/Fractional Factorial Designs 3
2.1.3 DOEs in Industrial Applications 4
2.1.4 Latin Hypercube DOEs in Computational Analysis 5
2.2 Principal Component Analysis 6
2.3 Metamodeling 6
2.3.1 Overview 6
2.3.2 Regression-Type Metamodels 7
2.3.3 Interpolation-Type Metamodels 7
2.3.4 Critical Comparison of Kriging, RBF and RSM in Engineering Optimisation 8
2.3.5 Metamodel Validation and Error Analysis 9
2.4 Multi-Objective Optimisation 9
2.4.1 Standard Multi-Objective Optimisation Model 9
2.4.2 Metamodel-Based Design Optimisation Algorithms 10
2.5 Software Integration 11
2.5.1 Abaqus Finite Element Methods for Modal Analysis 11
2.5.2 iSight Optimisation Software and Critical Review of Case Studies 11
2.6 Conclusions 12
Chapter 3 – Abaqus and iSight Setup 13
3.0 Introduction 13
3.1 Finite Element Analysis Model 13
3.1.1 Model Geometry 13
3.1.2 Meshing 13
3.1.3 Selection of Model Parameters 13
3.1.4 Boundary Conditions 14
3.1.5 Material Properties 14
3.1.6 Verification and Validation of the Models Used 15
3.2 iSight Optimisation Loop 15
3.2.1 Overview 15
3.2.2 Operation 15
Chapter 4 – Principal Component Analysis 16
4.1 Introduction 16
4.2 iSight PCA Component Flowchart 16
4.3 Choice of Screening Design 16
4.4 Objective Functions 17
4.4.1 Relative Improvement Function 17
4.4.2 Weighted Sum Function 17
4.4.3 Handling of Constraints 18
4.5 Statistical Methods 18
4.5.1 Regression Analysis (RSM metamodeling) 18

4.5.2 Pareto Plot 19
4.5.3 Main Effects Plot 19
4.6 Results and Discussion 19
4.6.1 Definitive Screening Design....19
4.6.2 RSM Screening Metamodels 20
4.6.3 Pareto Plots....21
4.6.4 Main Effects Plots....22
4.6.5 Principal Parameter Selection and Critical Analysis of Methods 23
Chapter 5 – Detailed DOE & Optimisation 24
5.1 Introduction 24
5.2 Detailed DOE & Optimisation Steps 24
5.3 Optimal Latin Hypercube Experimental Design (F) 24
5.4 Metamodel Building and Validation....25
5.4.1 Radial Basis Function Metamodel (G) 25
5.4.2 Leave-One-Out Cross-Validation and Sequential Sampling (G) 25
5.5 Multi-Objective Particle Swarm Optimisation (H) 26
5.5.1 Multi-Objective Particle Swarm Optimisation Flowchart (H) 26
5.5.2 Shape Optimisation Outputs (I) 26
5.5.3 Validation of the Predicted Optima (I) 26
5.6 Sensitivity Studies Overview and Results 27
5.6.1 OLHD DOE Points Sensitivity Study (E) 27
5.6.2 Optimisation Algorithm Sensitivity Studies (D) 27
5.7 Results 28
5.7.1 Optimal Latin Hypercube (Detailed) DOE 28
5.7.2 OLHD-RBF Metamodels....29
5.7.3 LOOCV and Error Analysis....29
5.7.4 Shape Optimisation on RBF Metamodel....30
5.8 Discussion....31
5.8.1 Interpretation of Optimum Solutions 31
5.8.2 Optimisation Algorithms....32
Chapter 6 – Baseline Battery Optimisation....33
6.1 Introduction 33
6.2 PCA, Detailed DOE and Optimisation Results....33
6.2.1 Principal Component Analysis....33
6.2.2 OLHD-RBF Metamodels....33
6.2.3 Optimisation of the RBF Metamodel....33
6.3 Conclusions....33
Chapter 7 – Conclusions....34
7.1 Achievements....34
7.2 Discussions....34
7.3 Conclusions....35
7.4 Future Works....35
References....36
Appendices....40

Acknowledgements

Firstly, I would like to thank my project supervisor, Dr Gregory de Boer, without his guidance and support this thesis would not have been possible.

I also would like to thank my physics teacher, Mr Shutler, who is the most influential person I have met during my time as a student, and who I never thanked enough for his charisma and encouragement throughout high school to pursue this degree.

Thank you to the University of Leeds, which has become my home for the past five years – I am extremely grateful for the opportunities and advice that I have received during my time studying here.

Finally, I would like to thank all my family and friends, who are too many to name, thank you for all the support and memories, which I will cherish for the rest of my life.

Abstract

High space-launch costs, coupled with the harsh dynamic environments experienced during rocket launch, often means a trade-off must be struck between reducing mass and maintaining sufficient stiffness of space components [1]. This work optimises the design of a satellite battery structure manufactured by ABSL-EnerySys, which has been redesigned to incorporate a Mass-Spring-Damper system (MSD) [2].

An automated shape optimisation approach was taken by incorporating the iSight software package, which allowed for cycle-time reduction of the design process [3]. An Optimal Latin Hypercube Design (OLHD) was used in conjunction with a Radial Basis Function (RBF), to build a metamodel for the First Natural Frequency (FNF) and mass of the redesigned 'MSD-Battery' structure.

The RBF metamodel was validated using Leave-One-Out Cross-Validation (LOOCV) to a maximum error of 3.95%. The optima were located on the metamodel using a Multi-Objective Particle Swarm (MOPS) algorithm. The Pareto set of solutions was found that allows ABSL to select their preferential design for a given mass and FNF requirement. Error analysis showed that the weighted sum optimum predicted from the RBF metamodel was accurate to 0.968%, as verified in Abaqus. The shape-optimised design improved the FNF by 18.12% whilst increasing the mass by 4.87%.

Through the design and optimisation of the MSD-Battery structure, the maximum launch acceleration experienced by the battery cells was reduced by 93.7% [2] resulting in greatly improved survivability of the cells at launch. This optimisation approach was further applied to the Baseline battery model without the MSD-system, which improved the FNF by 10.62% whilst increasing the mass by 1.57%. The results of both studies indicate that the shape optimisation methodology proposed can increase both the efficiency and accuracy of the optimisation works carried out by ABSL in the future.

Nomenclature

ANOVA – Analysis of Variance

BBD – Box-Behnken Design

BC – Boundary Condition

CAD – Computer Aided Design

CAE – Computer Aided Engineering

CCD – Central-Composite Design

DOE – Design of Experiments

DSD – Definitive Screening Design

EDO – Engineering Design Optimisation

ESE – Enhanced Stochastic Evolutionary Algorithm

FEA – Finite Element Analysis

FF – Full Factorial Design

FNF – First Natural Frequency

FrF – Fractional Factorial Design

GA – Genetic Algorithms

LHD – Latin Hypercube Design

LOOCV – Leave-One-Out Cross-Validation

MBDO – Metamodel-Based Design Optimisation

MMFD – Modified Method of Feasible Directions

MOGA – Multi-Objective Genetic Algorithm

MOPS – Multi-Objective Particle Swarm

MSD – Mass-Spring-Damper

OAT – One-At-A-Time Design

PCA – Principal Component Analysis

PB – Plackett-Burmann Design

OLHD – Optimal Latin Hypercube Design

PSO – Particle Swarm Optimisation

RBF – Radial Basis Function

MOO – Multi-Objective Optimisation

RI – Relative Improvement

RSM – Response Surface Models

SA – Simulated Annealing

V&V – Verification and Validation

WS – Weighted Sum

Chapter 1 – Introduction

1.0 Introduction

The growing commercialisation of the space sector (termed “New Space” [4]) has led to a rise of companies looking to take advantage of the growing demand for internet services, by developing large fleets of satellite constellations, capable of providing high-speed global bandwidth [5]. An example is the company OneWeb, which plans to launch 648 satellites by 2022 [6], and with launch costs being over £16,000 per kg [7], it is imperative for manufacturers to minimise the mass of satellites, to reduce the operational costs incurred during space launch.

Further to reducing mass, rocket launch generates extremely harsh environments for satellites due to vibroacoustic loads, and caution must be taken to design structures that avoid coupling between the launch vibrations and the structure’s natural frequency [8]. This is particularly important to prevent resonances that can reduce the operational lifespan of space components, such as the Lithium-Ion (Li-Ion) batteries that are used in satellites [9]. Two approaches that are commonly used in the space sector to minimise the vibrational damage to a structure are: **1)** Incorporating a Mass-Spring-Damper (MSD) system to reduce vibrational accelerations [10]; **2)** Maximising the First Natural Frequency (FNF) to avoid low-frequency resonance [11].

ABSL-EnerSys are a satellite battery manufacturer, and collaborating as part of a wider group project [2] have designed an MSD-system to integrate into their ‘NARMER’ battery module, to reduce the vibrational accelerations experienced by the Li-Ion cells during launch. Previous design studies conducted by ABSL [11] typically involved manually adjusting design variables on a Finite Element (FE) model to increase the FNF, resulting in improved vibrational performance. Despite this optimisation method being the traditional approach, it is limited by being time-consuming, computationally inefficient, and prone to errors as the number of design variables are increased [12].

A Request for Proposals (RFP) from ABSL looks to establish a framework that improves the accuracy and accelerates the design development, by performing shape optimisation on the ‘MSD-Battery’ structure. This entails following a Design of Experiments (DOE) and metamodeling approach that locates the best design methodically. The methods presented in this thesis removes the need for physical testing by using a verified FE model of the MSD-Battery structure for structural analysis in Abaqus CAE 2017. Moreover, the need for simulation is minimised by performing metamodel-based shape optimisation, to locate the design that maximises the FNF of the structure whilst minimising the increase in mass.

1.1 Aims

This thesis aims to perform shape optimisation on the 'NARMER' Battery Module – which has been redesigned to integrate an MSD system produced as part of a structural design study [13] – to maximise the structure's FNF whilst minimising its mass. From regular meetings with John Dobson and Daniel Murphy of ABSL-EnerSys, with who this project industrially collaborates, it was proposed that this thesis will also provide a framework for future shape optimisation. The shape optimisation tasks proposed in the CPP were elaborated into objectives that are listed in Section 1.2.

1.2 Objectives

1) Conduct a literature review into shape optimisation that explores the key concepts of Design of Experiments (DOE), Principal Component Analysis (PCA), Metamodeling, and Multi-Objective Optimisation (MOO).
2) Use a PCA study to identify the three key parameters of the battery structure.
3) Establish a DOE that explores the design space for the FE model; build a metamodel from the DOE results, and validate its accuracy for optimisation.
4) Perform shape optimisation on the metamodel using MOO and validate the result using FEA, to establish the shape-optimised design for the battery structure.
5) Use the optimisation framework created to carry out shape optimisation of the Baseline NARMER battery module, to verify the methods used.
6) Present the findings from shape optimisation alongside suggestions for improvements to the shape optimisation framework.

1.3 Project Report Layout

- Chapter 1 introduces the broad topic area being tackled through this report and discusses why the objectives of maximising FNF and minimising mass are important concerning the goals of ABSL.
- Chapter 2 discusses the literature underpinning shape optimisation – introducing key concepts and terminology relevant for DOE, PCA, Metamodeling and MOO.
- Chapter 3 outlines the implementation of the adapted battery model in Abaqus, as well as the integration of Abaqus in iSight, which forms the optimisation framework.
- Chapter 4 explains the methods used to carry out PCA and select the relevant design parameters, which are taken into Chapter 5 for the detailed DOE.
- Chapter 5 presents the sensitivity studies used to produce the detailed DOE, metamodel and MOO algorithm, and presents the final shape optimisation results.
- Chapter 6 overviews the shape optimisation performed on the baseline battery module, for verification of the methods used within this thesis.
- Chapter 7 discusses conclusions, achievements, and future works for the project.

Chapter 2 – Literature Review

2.0 Introduction

Shape optimisation is the process of varying the boundary of a structure to improve its performance [14] and has been applied through this thesis to ABSL's battery structure. This chapter introduces the literature and background underpinning shape optimisation by overviewing: **2.1)** Design of Experiments to collect data efficiently; **2.2)** Principal Component Analysis to identify key design variables; **2.3)** Metamodeling to build an approximation model for the design responses; **2.4)** Multi-objective optimisation to determine the optimal design(s) from the metamodel; **2.5)** Software Integration.

2.1 Design of Experiments

2.1.1 Overview

Oftentimes, an engineer may find that the input-output relationship between variables in a system is unknown to them. In this situation, the first port of call would be to analyse the theory that underpins the system – an engineer designing pipelines may look to Bernoulli's equation to understand the relationship between pipe area, velocity, and pressure. In this example, the mathematical theory has been extensively studied and rigorously tested in engineering applications. However, for more novel complex systems, a priori mathematical understanding may not be available, and a controlled experiment may be required to gain insight into the behaviour of the system. If the engineer in question is inexperienced, it is unlikely that they possess the statistical background to most effectively plan the experiment [15], and may resort to the common but dated method of varying each variable at a time whilst holding all other variables constant – termed the one-at-a-time (OAT) design.

There is a distinct difference between data and knowledge [16] and in any experiment, the knowledge gained on a system will not increase efficiently with the quantity of data, if the data is poorly planned. The more statistical approach would be to use a Design of Experiment (DOE), where selected values for the independent variables are used to plan each test in an experiment. The DOE formulation is given by the $N \times d$ matrix, $X = [x_1\ x_2 \dots x_N]^T$, where each row $x_i = [x_{i_1}\ x_{i_2} \dots x_{i_d}]$ is an experiment and d is the number of factors [17].

The following sections will introduce: **2.1.2)** Classical DOEs; **2.1.3)** DOEs in Industrial Applications; and **2.1.4)** Latin Hypercube DOEs for Computational Analysis.

2.1.2 Classical DOEs – Full/Fractional Factorial Designs

A Full Factorial (FF) DOE is a widely used classical method of experimental design where every possible combination of design variable is trialled, and a Fractional

Factorial (FrF) is a subset of an FF DOE. The example shown in Table 2.1 is an FF DOE with three design variables ($x_1, x_2\ and\ x_3$) and each of these variables can take two values, $x = -1 \text{ or } +1$ (called a two-level design):

Table 2.1: *Full-Factorial DoE (transposed)* [18]								
Variable	**Experiment Run #**							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
x_1	+1	+1	-1	-1	-1	-1	+1	+1
x_2	-1	-1	-1	+1	+1	-1	+1	+1
x_3	-1	+1	+1	-1	+1	-1	+1	-1

For an FF DOE, the number of runs in an m-level, d-factor experiment is given by $N = m^d$. Therefore, as the number of design variables (or levels) increases, the number of experiments required for an FF DOE rapidly rises. FrF designs only use a fraction of the experiments of an FF DOE, and as a result, are less costly to carry out.

The first papers on DOE methods were the work of R. Fisher [19] in 1922 who presented the use of FrF and FF DOEs for investigating the effect of fertilisers in agriculture. The challenge with interpreting the results were due to the large variations in the soil used, and the associated errors with data collection. Fisher's [19] pioneering work lead to the development of analysis of variance regression techniques (ANOVA) – where he fits 5th-degree polynomials to data for the annual yield of crops, which was used to interpret the significance of different factors and also provide an estimation of the errors in the results. Moreover, it was also identified that a key property of DOEs is orthogonality, which improves the separation of main effects in statistical analysis [20]. Orthogonal experiments have a cross-product of zero for each variable, as shown for the FF DOE in Table 2.1 where $[x_1] * [x_2] = [x_1] * [x_2] = [x_2] * [x_3] = 0$.

2.1.3 DOEs in Industrial Applications

A limitation of the classical methods presented by Fisher [19] is its unsuitability for engineering, as classical DOEs were developed to account for the variability present in biological systems that are far less prevalent in mechanical systems [16]. Recent theoretical developments have extended the industrial application of DOEs through key works such as the five-level Central-Composite Design (CCD) that have been used to investigate the interaction of process variables in the chemical industry [21]; the three-level Box-Behnken Design (BBD) that has been used in material manufacturing [22]; and the rise of non-classical designs such as the Latin Hypercube Design (LHD), which are the most widespread DOEs for computational simulation [17] (and are discussed in Section 2.1.4). Despite these developments in industrial applications, the adoption of DOE methods in structural engineering has been slow by comparison [23], so this thesis holds the potential to help fill a gap in an otherwise growing body of literature surrounding experimental design applications in engineering design.

2.1.4 Latin Hypercube DOEs in Computational Analysis

The Latin Hypercube Design (LHD) is constructed through splitting the d-dimensions of the design space into m-levels, which are orthogonal hypercubes of equal probability where each level contains a single sampling point (as shown in Figure 2.1b for $d = 2$, $m = 9$). The number of experiments is given by $N = m$, where the number of experiments required, N, can be freely selected by the experimenter. This provides added flexibility when compared to designing FrF experiments, where standard tables are typically used to determine the number of experiments required [24].

The assignment of sampling points to the hypercubes is a random procedure. This quality of the LHD allows for exploration of the design space by removing bias or conjectures involved with deterministic methods of selecting sampling points [25]. Although the orthogonal and exploratory aspects of the LHD are desirable traits, the volume of literature surrounding the optimisation of LHDs [17] indicates that their greatest limitation, when compared to classical designs, is in their reduced spatial coverage. This is illustrated in Figure 2.1, which shows a three-level, nine-run FF design (Figure 2.1a) alongside two nine-run LHDs (Figure 2.1b and 2.1c):

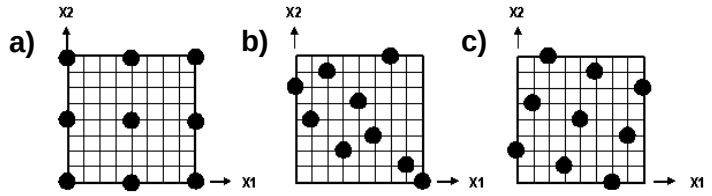


Figure 2.1: *For $d = 2, N = 9$: a) FF three-level design b) Standard LHD c) OLHD*

For the FF design in Figure 2.1a, there are three replicates of each design point at each factor level, which make it ideal for estimating error in biological systems such as in the experiments conducted by Fisher [19], however by removing replicates, LHDs can be made more suited to deterministic computational simulations by trading replication for vastly improved exploration of the design space.

It is generally accepted that computational DOEs should maximise spatial coverage [26] and the spatial coverage of LHDs can be greatly improved through optimisation – this sub-set of designs are called Optimised Latin Hypercube Designs (OLHDs) (Figure 2.1c). The optimisation of LHDs is most widely formulated as a minimisation of the spatial coverage criterion ϕ_p [17] and numerous algorithms have been employed to accomplish this such as simulated annealing (SA) [27]; genetic algorithms (GA) [28] and particle swarm optimisation [29]. A criticism of these papers is the focus on optimising a single measure of spatial coverage, and from a recent bibliographical review of OLHDs [17], only the Enhanced Stochastic Evolutionary (ESE) algorithm optimised the space-filling capacity of LHDs to three measures of spatial coverage: the ϕ_p-criteria, the potential energy U, and L2 discrepancy of the LHD. Therefore, the

robustness of the ESE algorithm, combined with its successful application in previous computational shape optimisation works [3,30,31] highlights its suitability for the computational works proposed in this thesis.

2.2 Principal Component Analysis

It has been suggested that before building a metamodel using a more detailed DOE, a Principal Component Analysis (PCA) – also known as a Screening Design – should be the first step [24]. A PCA is a low-experiment DOE, followed by regression analysis, to determine the most influential input parameters affecting the output response(s), to reduce the dimensionality, d, of the experiment by removing the less significant (non-active) design variables before performing a detailed DOE [32]. This is done to reduce experiments – for the FF example given in Table 2.1, if the number of factors increased to $d = 6$, the number of experiments required would rise to $N = 64$. Reducing experimental tests is critical in engineering since the use of experimentation, whether physical or computational, is synonymous with increased project costs [33].

From a 2008 review of PCA industry application, it was found that the most popular designs were two-level Taguchi and FrF/FF designs, which were used 31% and 30% of the time, respectively [23]. This contradicts the belief that a PCA should determine whether a system response is nonlinear with respect to the input parameters, which cannot be detected from two-level DOEs [34]. The large adoption of two-level designs may be attributed to the fact that for most engineering applications in literature, a previous understanding of the system already exists [23]. From a shape optimisation study of a steel wheel, it was also found that the two-level FrF PCA was inadequate to characterise the main effects, which subsequently led to the use of a three-level design that yielded better screening results, including quadratic effects [35]. Of the three-level screening designs in literature, the Definitive Screening Design (DSD) is the most robust method used for screening and characterisation [34,36] with various applications in shape optimisation [37,38] indicating its suitability for this thesis.

2.3 Metamodeling

2.3.1 Overview

From the DOE conducted, X, the experimenter would have collected results for the responses $F = [f_1\ f_2 \dots\ f_N]^T$ for each experimental run, often called the training data. Where optimisation is the goal, creating an approximation model for the responses F to the inputs X is often the next step. This process is termed metamodeling [12]. The resulting metamodel – also known as a response surface or approximation model – removes the need to perform further computational simulations for different design evaluations. As shown in a recent case of optimising a turbocharger radial turbine, where shape optimisation by direct computational simulation took ten days using 20

CPUs [39], optimising a simulation model directly can be extremely costly. Optimising a metamodel is far more efficient – particularly for the conceptual design stage [12]. Metamodels are split into two categories: interpolation-type and regression-type techniques, which are outlined and discussed in the following sections.

2.3.2 Regression-Type Metamodels

Regression-type metamodels (or Response Surface Models (RSM)) are ideal where the training data is subject to random error and uncertainties – as it does not force the response surface through each point and instead interprets the training data to build the response surface [12]. In fact, many of the early advancements in the field of regression analysis came from the experimentation conducted by Fisher [19], where variances in crop data was a pressing issue. RSMs are typically in the form of low-order polynomials and are described as local metamodeling strategies, owing to their validity over only a pre-determined region of the design space [40]. The most common regression model uses a least-squares regression matrix to fit the training data using either a linear or a quadratic polynomial. The RSM equation is given in (2.1) [41]:

$$\hat{y} = \beta_o + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_{i} \sum_{j} \beta_{ij} x_i x_j \qquad (2.1)$$

Where: $\beta_i, \beta_{ii}, \beta_{ij}$ are the regression coefficients; x_i are the input variables; x_i^2 are quadratic effects and $x_i x_j$ are interaction effects.

The number of data points required to fit an RSM is given by $N = (n+1)(n+2)/2$ and increases significantly with the polynomial order. This has led to arguments against RSM for multivariate design optimisation when compared to more efficient methods of metamodel building such as interpolation models [42]. A key benefit of RSM metamodels is the ability to interpret the parameter sensitivity based on the regression coefficients β obtained from the training data – this highlights the utility and robustness of RSM beyond approximation models but also for screening key parameters [43,44].

2.3.3 Interpolation-Type Metamodels

Interpolation-type metamodeling algorithms ensure that the response surface passes through each data point. Interpolation-type metamodels are suited to deterministic analysis, where the main source of error is numerical and systematic error, compared to the random error associated with physical experiments [42]. The two most common interpolation-type methods used in engineering optimisation, are Kriging and Radial Basis Functions (RBF) [42]. The equation for Kriging is given by (2.2) [41]:

$$\hat{y} = \sum_{j=1}^{k} \beta_j f_j(x) + Z(x) \qquad (2.2)$$

Where: $f_j(x)$ is a combination of known fixed functions; $Z(x)$ is the mean component.

The Kriging mean component $Z(x)$ allows for some degree of relaxation of the interpolation criteria through incorporating correlation functions into the formulation [45], however, this flexibility comes with the drawback of Kriging being a very computationally costly method of metamodeling [44].

RBF is another popular interpolation-type metamodel, and is given by (2.3) [41]:

$$\hat{y} = \sum_{i} a_i \|\mathbf{X} - \mathbf{X_{0i}}\| \tag{2.3}$$

Where: $\mathrm{a_i}$ is the coefficient of the expression including a basis function; $\mathbf{x_{0i}}$ is the input.

RBF uses a neural network to interpolate multivariate data using radial basis functions of the Euclidean distance [46]. Both RBF and Kriging are global metamodels that can be extended to a larger region of the design space when compared to RSM [40]. Since this thesis aims to optimise an enclosed region of the design space, as determined by previous work conducted by ABSL [11], both local and global methods were suitable candidates considered for metamodeling.

2.3.4 Critical Comparison of Kriging, RBF and RSM in Engineering Optimisation

Kriging, RSM and RBF metamodels have been the subject of many recent comparative studies, which can be attributed to the rise in computational simulation and growing demand for metamodels tailored to industrial applications. A limitation of early studies was the use of only a single test function as the focal point of comparison between metamodels [41]. So, a study consisting of 14 benchmark problems with varying degrees of nonlinearity was conducted, and it was found that RBF excelled for evaluating the functions across a range of sample sizes and dimensionalities, whereas Kriging was the worst-performing overall. A shortcoming of this study is that only a single sampling method was tested, which could instead suggest that the LHD sampling method used to collect the training data simply favoured RBF. A more rigorous study conducted in 2016 compared the use of RBF, Kriging and RSM against 9 engineering-specific optimisation problems, whilst also varying the sampling methods used [42]. This study reaffirmed the findings shown previously [41] that space-filling sampling methods such as OLHD combined with RBF metamodeling outperforms RSM and Kriging when it came to robustness, efficiency, and accuracy measures across a range of design problems and sampling methods. Despite this, RSM showed comparable accuracy to RBF for low-dimension problems, whilst offering supreme computational efficiency amongst all metamodeling techniques tested. From this, it can be concluded that no single metamodel is fit-for-all purposes and experts in the field [47] suggest that the metamodeling technique chosen for optimisation be based on **1)** The DOE sampling method used and **2)** The behaviour of the response being analysed such as the nonlinearity, number of design variables, and number of outputs [41,44].

2.3.5 Metamodel Validation and Error Analysis

Metamodels only offer an approximation to the underlying responses being modelled, and it is important to ensure adequate validation and error analysis has taken place before optimisation. Leave-One-Out Cross-Validation (LOOCV) has become commonplace in metamodeling [12]. LOOCV is where: **1)** A point is removed from the N-length DOE; **2)** The metamodel is built using $N-1$ points and used to predict the response at the removed point; **3)** The error is then calculated at this point; **4)** LOOCV is reiterated until all DOE points have been approximated [48]. A single error statistic cannot portray the accuracy of the entire metamodel [40], and the common R^2 statistic alone is insufficient as it does not indicate whether the model is overfitted [48]. For robustness, various papers suggest that R^2 should be paired with the absolute maximum error e_{max} to interpret both the metamodel global and local accuracy [12,41].

2.4 Multi-Objective Optimisation

2.4.1 Standard Multi-Objective Optimisation Model

After constructing the metamodel, optimisation can be performed. The standard formulation for multi-objective optimisation (MOO) is defined as follows in (2.4) [49]:

$$\begin{cases} Minimize/Maximise\ f(X) = [f_1(X), \dots, f_n(X)] \\ subject\ to: \\ g_j(X) \le 0, j = 1,2, \dots, J \\ h_k(X) = 0, k = 1, 2, \dots, K \\ where: \\ x_{iL} \le x_i \le x_{iU}\ or\ x_i \ge 0, i = 1\ to\ d \end{cases} \tag{2.4}$$

Where: $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{d})$ is a d-dimensional vector; $\mathrm{f(X)(i} = 1, 2, \dots, \mathrm{n})$ is an objective function; $\mathrm{g_j(X)} \le 0$ and $\mathrm{h_k(X)} = 0$ are the system constraints.

For MOO, it is unlikely to be able to optimise all objective functions simultaneously, therefore the Pareto set is typically the main result found [50]. This is a set of non-dominated solutions x*, where there does not exist any other point x that reduces at least one objective function without increasing another [51]. The Pareto set and concept of domination are illustrated for a MOO scenario below in Figure 2.2:

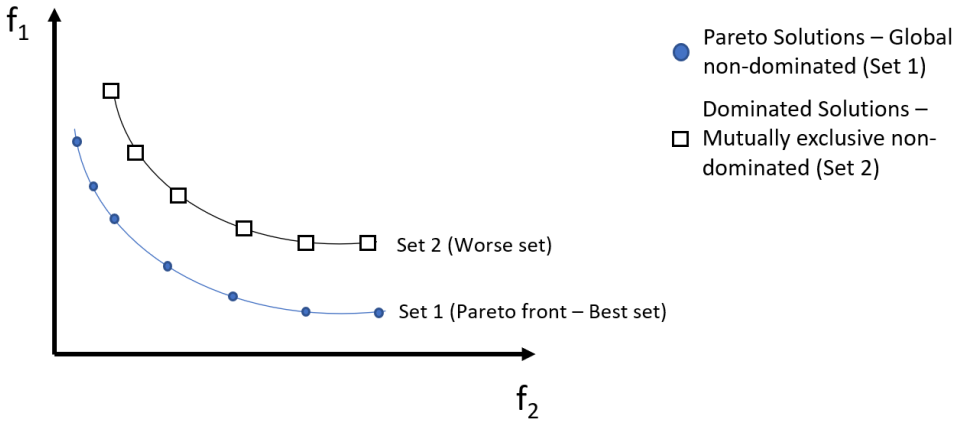


Figure 2.2: *Pareto set and concept of non-dominant sets for the minimisation of f_1 and f_2*

The Pareto front is shown in blue and is the global, optimal set of solutions. Set 2 is a set of solutions, sorted into a separate non-dominated set, which is a strategy used by MOO algorithms to evaluate optimality and is elaborated in Section 2.4.2.

2.4.2 Metamodel-Based Design Optimisation Algorithms

Metamodel-Based Design Optimisation (MBDO) algorithms are MOO methods that have been developed to be performed on multi-dimensional, nonlinear, and multi-objective design spaces [52]. Two main types of MBDO methods are typically used: direct search methods and gradient-based search methods [40]. The gradient-based search methods require the objective functions to be differentiable and are limited to continuous functions only. Since many metamodels are indifferentiable and noisy, gradient-based methods cannot provide the accuracy of direct search methods [44].

Direct search methods, as their name suggests, locates the optima through performing direct function evaluations of the metamodel. This makes them operable on any type of metamodel. From a review into MBDO algorithms used in engineering, it was found that Multi-Objective Genetic Algorithms (MOGA) were amongst the most widely used and researched, with Multi-Objective Particle Swarm (MOPS) being a rapidly growing development within the field [47]. Both algorithms are **1)** Metaheuristic – meaning that they cannot guarantee an exact solution and **2)** Stochastic – indicating an underlying use of randomness to generate design points, therefore the results cannot be exactly replicated each time the algorithm is used [53]. The popularity of these algorithms is attributed to their exploratory qualities, avoiding getting 'trapped' at local optima, particularly when compared to gradient-based methods [33].

MOGA is an evolutionary algorithm consisting of 5 main steps: **1)** Initialisation of a population of individuals; **2)** Evaluation of the fitness of each individual; **3)** Selection of the fittest individuals as parents for reproduction; **4)** Crossover and mutation that breeds new individuals; and **5)** Ranking and extermination of the weakest individuals [54], where 'individuals' refers to test points in the design space.

For MOPS, the two differences when compared to MOGA are: **1)** Where MOGA requires three steps for processing: selection of parents, selection of individuals, and crossover – MOPS only relies on using leaders to guide particles to the Pareto front; **2)** In MOPS the direction of movement is based on the individual's observed global best and personal best position only, constraining the particle's movement [50].

Differently from single-function optimisation, during the selection process MOGA and MOPS both sort the individuals (particles) into mutually exclusive non-dominated sets based on their fitness, illustrated in Figure 2.2 above. This allows for individuals to explore regions near the Pareto set and improves convergence [55]. At the time of writing, only two research papers exist comparing MOGA and MOPS in engineering

applications: **1)** A study conducted into flow line optimisation, which showed that MOPS outperformed MOGA for the number of Pareto solutions obtained and in processing time [56]; and **2)** A comparative study where MOPS outperformed MOGA for converging to the Pareto set during the shape optimisation of a centrifugal pump [50].

This suggests that MOPS is better suited for use in this thesis, however, this is not conclusive, owing to the limited studies in this field, and knowing that performance is highly metamodel dependant [44]. To select an optimisation algorithm, the suggested approach is to perform comparison studies, with the number of Pareto points located used as the distinguishing measure of performance between algorithms, to determine the most suitable algorithm for the metamodel being analysed [57].

2.5 Software Integration

2.5.1 Abaqus Finite Element Methods for Modal Analysis

When designing complex mechanical structures, the physical behaviour may be difficult to solve analytically as a continuous structure, moreover, it would be inefficient and costly to build each iteration of the design and test it experimentally. Finite Element Analysis (FEA) is used as a simulation tool to discretise structural problems into smaller nodes and elements, to be solved separately using polynomial shape functions, and then joined together through interpolation to model the entire system behaviour [58,59].

This thesis uses FEA to carry out modal analysis, and Abaqus CAE 2017 is used to determine the structure's eigenvalues (mode shapes) ϕ by solving Equation (2.5) [60]:

$$(\mu^2 M^{MN} + \mu C^{MN} + K^{MN})\phi^N = 0 \qquad (2.5)$$

Where: M^{MN}, $\mathrm{C}^{\mathrm{MN}}$and K^{MN} are mass, damping and stiffness matrices; ϕ^{N} is an eigenvector (modes); μ is a complex eigenvalue, and M and N are degrees of freedom.

From this equation it is clear to see the important aspects of FEA modelling: M^{MN} would depend on the material densities; C^{MN} and K^{MN} would depend on the material properties, and M and N would depend on the boundary conditions and constraints used [60]. To ensure representative values for these are used, validation is imperative.

Validation ensures that the model is representative of real-life [61] and is conducted by comparing FEA simulations against physical experimental results. Moreover, verification ensures that the Abaqus code is correct and that the numerical error from discretisation is minimised [59]. Verification is conducted by **1)** Comparing simulation results to benchmark tests – where Abaqus has been tested extensively for modal analysis [62], and **2)** Conducting a mesh convergence where the mesh size is increased until the numerical error converges to an insignificant value [61].

2.5.2 iSight Optimisation Software and Critical Review of Case Studies

iSight is an optimisation software that allows for seamless integration with Abaqus, with both tools developed by SIMULIA. A 2008 review into engineering optimisation

proposed that a hindrance to computational optimisation techniques was limited software integration [47]. Since then, the development of iSight has allowed for rapid integration between Computer-Aided Design (CAD) and Computer-Aided Engineering (CAE) tools, gaining widespread use in industry [63]. Two iSight studies are discussed:

Case Study 1) iSight was used in shape optimising a high-speed spindle, to minimise thermal deformation and maximise FNF by integrating ANSYS Workbench [64]. An FEA model was used to obtain a 100-point OLHD for four design factors that had been predetermined as main effects. The OLHD was used with RBF to build the metamodel, however, no validation was reported – and this may be due to an assumption that 100 points would give an accurate metamodel, based on previous studies. The optima were located by weighted sum based on expert opinion, which decreased the thermal deformation by 2.43% and increased FNF by 12.37%. Although 'expert-based optimisation' is common in the industry, it is constrained by expert biases [47] and can be improved by also obtaining the Pareto set of solutions.

Case Study 2) iSight was used to shape optimise the brake horsepower of a V-Twin engine, integrating the WAVE thermodynamic simulation software [3], and optimised with respect to 15 factors using an LHD and RSM. The modified method of feasible directions (MMFD) optimisation method converged to an optimal design at 68 LHD points, yielding a 4% improvement, however a limitation of this study was that no error analysis was included in the report, and the optima located was not verified in the WAVE software. The use of 15 factors for optimisation, compared to removing non-active factors through a PCA first, certainly added to the computations required for optimisation. The use of iSight was still said to have reduced the amount of routine experimental tasks from 80% to 20%, indicating the benefits of the software. From this, iSight has been shown to aid optimisation, however, the literature suggests that using the software may lead to less rigorous practices. To avoid this, all optimisation steps including PCA, validation, and error analysis were included in this thesis.

2.6 Conclusions

To conclude, the four key findings from this chapter taken forward in this thesis are: **1)** For PCA, a three-level design will allow for screening of main effects and nonlinearity using regression analysis (RSM); **2)** To build the metamodel, OLHD combined with RBF offers the most robust metamodeling method, and validation will be performed using LOOCV; **3)** To locate the optimal design, MOGA and MOPS are both strong candidates for MOO to identify the Pareto solutions, and the algorithm selection will be determined using comparative studies to ensure a suitable algorithm is chosen for the metamodels; **4)** Integrating Abaqus optimisation within the iSight software package will improve the efficiency of the framework proposed to ABSL.

Chapter 3 – Abaqus and iSight Setup

3.0 Introduction

The Baseline NARMER battery model [65] was adapted in this thesis, based on the MSD-Battery design proposal [13], to incorporate the 'MSD' system using FEA. This section introduces the MSD-Battery FEA model created for shape and topology optimisation, and further detail about the Baseline battery model and MSD design considerations can be found in the respective documents referenced above. This chapter also outlines the integration of Abaqus CAE 2017 into the iSight 2017 optimisation package, to deliver an intelligent and robust optimisation framework.

3.1 Finite Element Analysis Model

3.1.1 Model Geometry

The SOLIDWORKS MSD-Battery concept is shown in Figure 3.1a and the MSD-Battery FEA model created for use in this thesis is shown in Figure 3.1b:

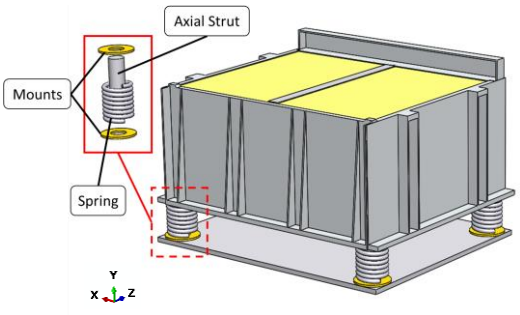


Figure 3.1a: *Solidworks MSD-Battery concept*

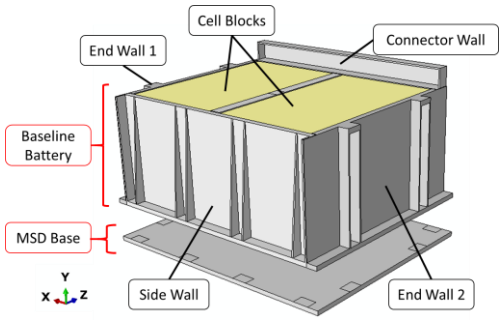


Figure 3.1b: *FEA model of the MSD-Battery*

This design concept is based on vibration isolators, which introduce two low-frequency modes shapes, that reduce the vibrational acceleration of the cell blocks [10,13].

The only addition made to the Baseline NARMER battery FEA model was the MSD base (Figure 3.1b). Removing the axial struts and representing these as constraints and springs reduced the FEA model complexity and aided computational convergence.

3.1.2 Meshing

C3D10M 10-node quadratic tetrahedron elements were used to mesh the MSD-Battery (Figure 3.2a). This allowed for fast meshing in iSight, and minimal element deformation compared to hexahedral elements. C3D10M elements were also complimentary to model the surface-surface tie constraints used between structural components [66].

3.1.3 Selection of Model Parameters

Key structural members were parameterised to allow for automatic updating of the design geometry during shape optimisation within iSight. Parameters were created for strut angle, feet thickness, flange thickness, wall thickness, strut thickness and base thickness. Figure 3.2b illustrates these parameters on the 'Side Wall' component:

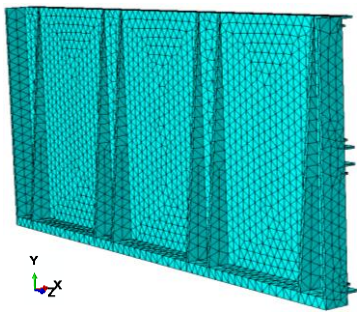


Figure 3.2a: *Side wall mesh*

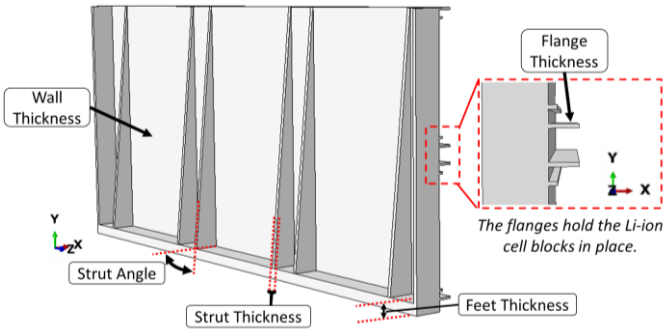


Figure 3.2b: *Overview of design parameters*

The MSD 'base thickness' is in the y-axis, as shown in Figure 3.1b.

The upper, mid and lower limits for the parameters, as well as their initial values from ABSL [67], are shown in Table 3.1:

Table 3.1: *Design parameters used for the MSD-Battery*

Design Parameter	Initial Value	Lower Limit (-1)	Upper Limit (+1)	Mid-Value (0)
Strut Angle (°)	84.5	84.5	89.5	87
Feet Thickness (mm)	6	3	12	7.5
Flange Thickness (mm)	1	0.5	2	1.25
Wall Thickness (mm)	0.95	0.5	2	1.25
Strut Thickness (mm)	2	1	4	2.5
Base Thickness (mm)	6	3	12	7.5

3.1.4 Boundary Conditions

There were two main changes in boundary conditions (BCs) from the Baseline model:

1) Moving the encastre BC from the feet to the MSD base (Figure 3.3a) replicating the same attachment as the validated BCs of the Baseline battery FE model.

2) Added constraints replicating four spring elements (total 70 kN/m) attached to four corners on the MSD and the battery structure (Figure 3.3b) which represents the axial strut attachments between the MSD-base and the Baseline battery structure.

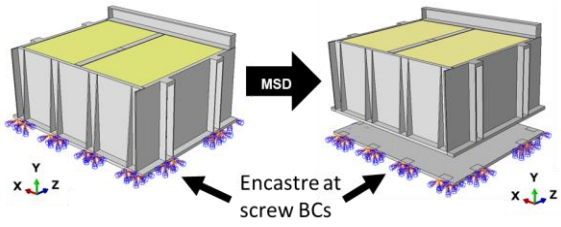


Figure 3.3a: *Battery-Satellite Attachment BCs*

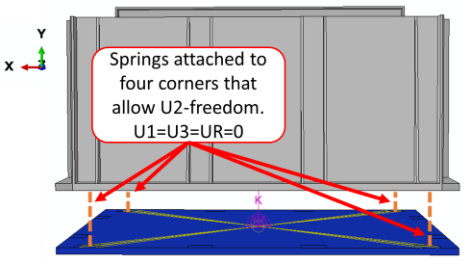


Figure 3.3b: *Base-Battery Constraints*

Tie constraints were also used to represent welding between structural components.

3.1.5 Material Properties

The same material properties were used as in the Baseline model for the aerospace aluminium: 66 GPa Young's Modulus and 0.33 Poisson's Ratio for all structural components, with structural damping of 0.0259 N/m as specified by ABSL. Smaller features such as screws, mounting holes, glue etc. were excluded from the FEA model

to aid meshing and computation. The densities of each structural component were adapted to incorporate the features that were excluded [65], which is the standard approach followed by ABSL in FEA modelling of their batteries [11].

3.1.6 Verification and Validation of the Models Used

The first effective frequency impacting the MSD-Battery cells is the FNF used in this study [13]. The MSD-Battery mesh was verified within 0.652% as shown in red in Table 3.2. Since experimental validation was not possible for the MSD, verification is shown in Table 3.3, by comparing the MSD-Battery FEA (red) with validated ABSL batteries:

Table 3.2: *MSD-Battery mesh convergence*

Mesh	# Elements	FNF (kHz)	% Error
1	148178	0.59105	-
2	**111790**	**0.59493**	**0.652**
3	94439	0.60737	2.048
4	85072	0.61257	0.849

Table 3.3: *MSD-Battery Verification*

Battery Model	FNF (kHz)
MSD-Battery	**0.59493**
NARMER [67]	0.56290
Triple Deck [11]	0.35635
Single Deck [11]	0.92796

Since the FNF was sufficiently converged and within the range of similar batteries, it was deemed sufficient verification and validation (V&V) for this optimisation study. Further V&V studies conducted for the Baseline battery are detailed by Asad [65].

The initial MSD-Battery FNF and mass are given, for comparison post-optimisation:

∴ Pre-optimised outputs for the MSD-Battery: FNF=0.5949kHz; Mass=23.2578kg.

3.2 iSight Optimisation Loop

3.2.1 Overview

Figure 3.4 displays the novel optimisation loop created in iSight to perform rapid and automated shape optimisation, without having to interface with Abaqus directly:

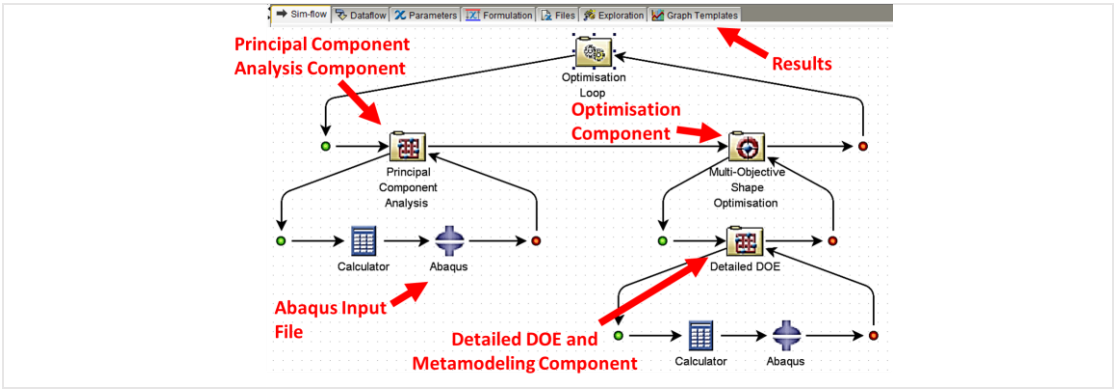


Figure 3.4: *iSight Shape Optimisation Loop*

3.2.2 Operation

Running each component is intuitive; to run the PCA, the user needs to right-click on the 'Principal Component Analysis component' and select 'run component'. This is the same for DOE, and optimisation components. As each runs, the 'Graph Templates' is populated with results. Chapters 4 and 5 outlines how the PCA, DOE, Metamodel and Optimisation results were generated and validated. Chapter 6 then verifies the robustness of the iSight framework by shape optimising the Baseline battery model.

Chapter 4 – Principal Component Analysis

4.1 Introduction

This section outlines the Principal Component Analysis (PCA) which aims to identify the three main design parameters impacting the first natural frequency (FNF) and the mass response of the MSD-Battery by using a low-run Design of Experiment (DOE) and regression analysis. First, the screening design will be outlined. Then, the objective functions and statistics for screening are presented alongside the rationale for their inclusion. Finally, the results and discussion are shown where the principal design parameters are selected, and the effectiveness of the methods used are discussed.

4.2 iSight PCA Component Flowchart

The PCA high-level flowchart in Figure 4.1 overviews the methods used in this chapter:

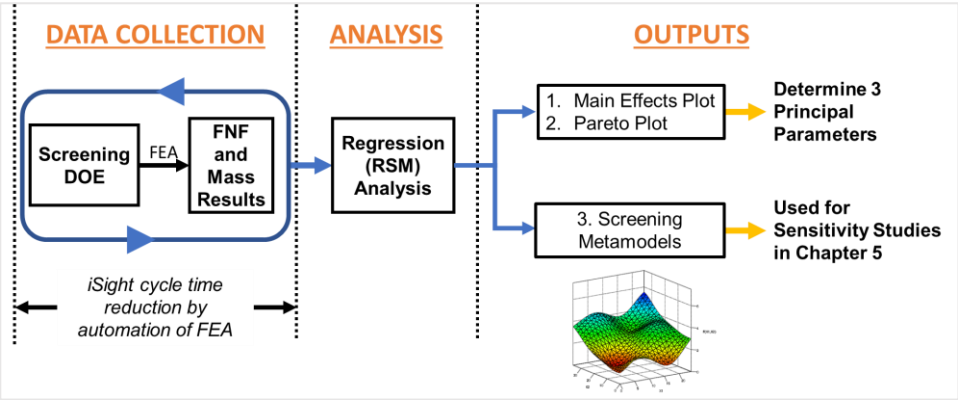


Figure 4.1: *Principal Component Analysis Flowchart*

There are two key results from this chapter: **1)** Determining the three principal parameters; and **2)** Obtaining a 'Screening Metamodel' – which is a low-resolution metamodel – used for sensitivity studies in Chapter 5.

4.3 Choice of Screening Design

The experimental design that was chosen for the PCA was the Definitive Screening Design (DSD). A key advantage of the DSD compared to FrF designs is its improved ability to use the three-level factors to detect quadratic effects between parameters, whilst also ensuring the main effects are orthogonal [36]. This was desirable since the characterisation of nonlinear effects would aid the selection of compatible metamodeling strategies in the following chapters. The number of runs r required by for a DSD is given by (4.1) [68]:

$$If\ n\ is\ odd: r = 2n + 3, \quad else, \quad If\ n\ is\ even: r = 2n + 1 \tag{4.1}$$

Where n is the number of design parameters. The DSD was constructed using conference matrices $[C]$ where Equation (4.2) is satisfied [69]:

$$[C]'[C] = (n-1)[I]_{n\times n} \tag{4.2}$$

And $[I]_{nxn}$ is a square identity matrix. The DSD is then obtained from (4.3):

$$[D] = \begin{bmatrix} C \\ -C \\ 0 \end{bmatrix} \tag{4.3}$$

The DSD matrix $[D]$ was computed using MATLAB and is shown in Section 4.6.1. This DOE was then imported into the iSight PCA component to run the design points in FEA.

4.4 Objective Functions

The maximisation of FNF and the minimisation of mass are conflicting objectives that required multi-objective problem formulation. This section outlines the two screening objective functions coded into the iSight PCA component: **1)** The Relative Improvement (RI) and **2)** the Weighted Sum (WS). These functions were computed for each DSD result, which was then used for regression analysis, outlined in Section 4.5.

4.4.1 Relative Improvement Function

RI has been used previously by ABSL for screening studies [11]. The function for RI, f_{RI}, is calculated using Equation (4.4):

$$Maximise\ f_{RI}(x_1 \dots x_{i=n}) \left(\frac{Hz}{kg}\right) = \frac{f_f(x_1 \dots x_n) - f_0}{f_m(x_1 \dots x_n) - m_0} \tag{4.4}$$

Where: x_i are the design parameters*;* f_f is the FNF function*;* f_m is the mass function*;* f_0 is the initial FNF (0.5949 kHz); m_0 is the initial mass *(*23.2578 kg).

RI allows smaller design features, that may have a small absolute impact on the FNF, to be screened by showing the improvement in FNF *per unit change in mass.*

4.4.2 Weighted Sum Function

WS is the most common approach in literature to multi-objective shape optimisation in metamodeling [30,51,64]. The function for WS, f_{WS}, is shown in Equation (4.5):

$$Maximise\ f_{WS}(x_1 \dots x_{n=6}) = \frac{w_f f_f(x_1 \dots x_n)}{SF_1} - \frac{w_m f_m(x_1 \dots x_n)}{SF_2} \tag{4.5}$$

Where: $\mathrm{w_f}$ is the weighting of $\mathrm{f_f}$ (between 0-1); $\mathrm{w_m}$ is the weighting of $\mathrm{f_m}$ (given by 1-$\mathrm{w_1}$); $\mathrm{SF_1}(=\mathrm{f_0})$ is the scale function for $\mathrm{f_f}$; $\mathrm{SF_2}(=\mathrm{m_0})$ is the scale function for $\mathrm{f_m}$.

The reason for subtracting f_m in (4.5) is because the minimisation of f_m is equal to the maximisation of $-f_m$. The scaling functions SF_1 and SF_2 are necessary to normalise the frequency and mass to the same order of magnitude. The weighting factors w_f and w_m determine the importance of the frequency maximisation objective, to the mass minimisation objective. The addition of these parameters makes the WS more flexible compared to RI since different weightings can be assigned to the mass and FNF. These values are typically chosen based on the needs of the company, however for screening, both f_f and f_m were given an equal weighting ($w_{f,m} = 0.5$), and this is

termed the “WS-Optimum”, $f_{WS_{w=0.5}}$. This output provides the best linear combination of increasing FNF whilst balancing the increase in mass.

A comparison will be made between RI and WS in the discussion to determine the most effective objective function for ABSL to use in future screening studies.

4.4.3 Handling of Constraints

There were no constraints to the mass or FNF given by ABSL since this thesis optimises a conceptual MSD-battery structure. Therefore, the shape optimisation performed was only constrained by the design parameter limits, as given in Table 3.1.

4.5 Statistical Methods

This section outlines the statistical methods used for the PCA. This section details the regression analysis that was performed on the DSD data obtained from FEA, for screening. Then, the Pareto and Main Effects plots are introduced, which were used for characterisation and screening of the MSD-Battery behaviour, respectively.

4.5.1 Regression Analysis (RSM metamodeling)

The iSight optimisation framework uses regression analysis (RSM) to determine the regression coefficients, which were then interpreted for screening, and also to produce a ‘rough’ metamodel for sensitivity studies (outlined in Figure 4.1). First, the second-order polynomial regression model was formulated using (4.6): [12,60]:

$$f(x) = \alpha_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \beta_{ii} x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \beta_{ij} x_i x_j \qquad (4.6)$$

Where: x_i is the input design parameter; x_i^2 is the quadratic (nonlinear) effect of design parameter $\mathrm{x_i}$; $x_i x_j$ is the interaction effects between design parameters $\mathrm{x_i}$ and $\mathrm{x_j}$; $\beta_i, \beta_{ii}, \beta_{ij}$ are the regression coefficients; and α_0 is a constant term.

This can be rewritten to form the fitted regression model $\hat{f}(x)$ as given in (4.7):

$$\hat{f}(x) = X\hat{\beta} \qquad (4.7)$$

Where: X is the matrix of design parameter input values and $\hat{\beta}$ is the least-squares regression coefficients matrix where $\hat{\beta} = (X^T X)^{-1} X^T f(x)$.

An assumption made in Equation (4.6) was that the higher-order terms describing interactions between three or more design parameters are negligible, which is a common assumption used to reduce the number of design evaluations required to build the regression model [41]. As discussed in Section 2.3.2, regression models relax the constraint that the model must pass through each design point evaluated from the DOE data, therefore there is an error associated with the polynomial regression model compared to the training data. To determine the accuracy of the fitted polynomial, the R^2 error is calculated and is as shown in Equation (4.8):

$$R^2 = 1 - \frac{\sum_i \left(f_{tested} - f_{approximated}\right)^2}{\sum_i (f_{tested} - f_{mean})^2} \quad (4.8)$$

Where: f_{tested} is the actual FEA function value; $f_{approximated}$ is the approximated value from the regression model; f_{mean} is the overall mean of the function.

The R^2 value has a range of 0-1 which describes how well the regression model predicts the function value, with $R^2 = 1.00$ corresponding to a perfectly fitted model. R^2 is used with the maximum error, e_{max}, to evaluate the model accuracy.

4.5.2 Pareto Plot

The purpose of the Pareto plot is to order the design parameters from the parameters with the greatest to the least impact on the objective functions. To produce the plot, the regression coefficients $\hat{\beta}$ were scaled with Equation (4.9) [60]:

$$Scaled\ coefficients\ \tilde{\beta}_i = \hat{\beta}_i \frac{S_{x_i}}{S_f} \quad (4.9)$$

Where: $\hat{\beta}_i$ is the design parameter *i* regression coefficient; S_f is the standard deviation of f; and S_{x_i} is the standard deviation of x_i.

These normalised and scaled $\tilde{\beta}_i$ coefficients were then ordered into a bar chart to compare the effects of the parameters on the WS and RI functions for screening.

4.5.3 Main Effects Plot

The Main Effects plot shows the mean effects of the design parameters on the functions. The first step was to standardize the parameters using Equation (4.10):

$$\bar{x} = 2\left(\frac{x - \mathrm{x}_{min}}{x_{max} - x_{min}}\right) - 1 \quad (4.10)$$

Where: x is the parameter value at a specific design point; x_{min} is the lower limit of the parameter; x_{max} is the upper limit of the parameter.

After standardisation, the design parameter values are unitless and centralised between -1 to 1. The Main Effects plot was then created by determining the mean effect of each design parameter at their high-level setting $\bar{f}(x_H)$ and the low-level setting $\bar{f}(x_L)$ for both the FNF and mass. This was used to characterise the behaviour of the MSD-battery model FNF and mass responses with respect to the design parameters by determining nonlinear effects, which was required to select a suitable 'detailed' DOE tailored to the response surface topography, for optimisation in the following chapter.

4.6 Results and Discussion

4.6.1 Definitive Screening Design

The number of FEA experimental runs required is given by evaluating Equation (4.1):

$$r = 2n + 1 = 2(6) + 1 = 13\ experiments$$

The DSD created for screening is shown in Table 4.1:

Table 4.1: *DSD used for Principal Component Analysis*						
	Design Parameter Level Setting (Outlined in Table 3.1)					
Run	***Feet***	***Flange***	***Strut***	***Strut Angle***	***Wall***	***Base***
1	0	+1	+1	+1	+1	+1
2	0	-1	-1	-1	-1	-1
3	+1	0	-1	+1	+1	-1
4	-1	0	1	-1	-1	+1
5	+1	-1	0	-1	+1	+1
6	-1	+1	0	+1	-1	-1
7	+1	+1	-1	0	-1	+1
8	-1	-1	+1	0	+1	-1
9	+1	+1	+1	-1	0	-1
10	-1	-1	-1	+1	0	+1
11	+1	-1	+1	+1	-1	0
12	-1	+1	-1	-1	+1	0
13	0	0	0	0	0	0

The DSD produced an experimental design where the number of lower (-1), mid (0) and upper (+1) level runs for each design parameter are equal – called a balanced design. This has many benefits, most notably: minimisation of confounding effects between factors, increasing the ability to detect interactions, maximum spatial coverage in the design space, and no selection bias impacting the results across [70]. The FEA results for FNF and mass for each run is presented in Figure 4.2:

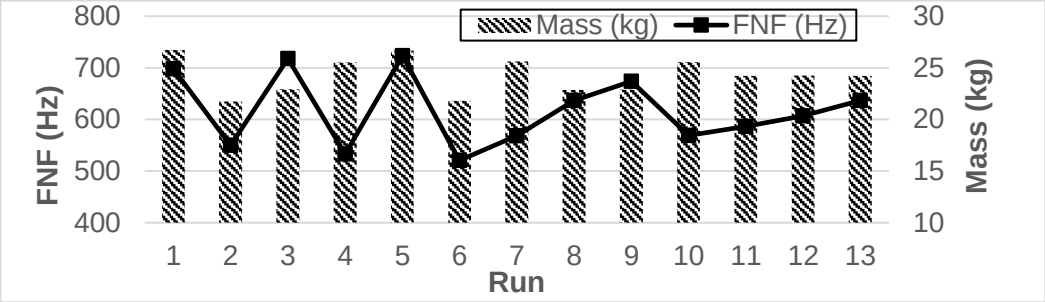


Figure 4.2*: FNF and mass results from the DSD for each run*

4.6.2 RSM Screening Metamodels

The Screening Metamodels are shown in Figure 4.3:

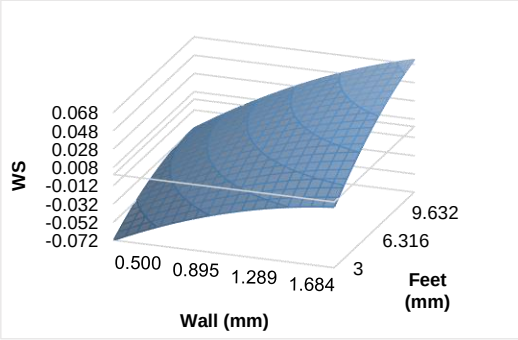


Figure 4.3a: *WS Screening Metamodel*

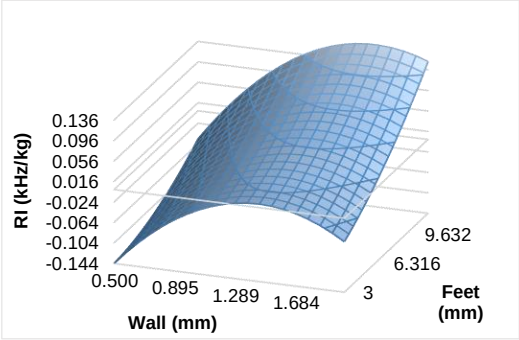


Figure 4.3b: *RI Screening Metamodel*

The Wall-Feet WS and RI metamodels are shown for visualisation, and these Screening Metamodels were used for sensitivity studies in Chapter 5.

The R^2 and maximum error e_{max} from the regression analysis is shown in Table 4.2:

Table 4.2: *Regression Analysis (RSM) Errors*

Metamodel	R^2	e_{max}
WS	0.975	10.003%
RI	0.893	29.241%
Mass	0.662	45.882%
FNF	0.946	11.086%

An R^2 of 0.9 is commonly used to determine whether the RSM model is well fitted for optimisation [12,30]. Table 4.2 shows that the WS, RI, mass and FNF outputs were within range of this measure. This indicates that the DSD is suited for the prediction of the MSD-Battery outputs, however, it is also known that R^2 alone is insufficient for determining the accuracy of a metamodel [71] since a high R^2 value may also indicate that the metamodel is overfitted to the data. In addition to global accuracy (from the R^2 value), local accuracy is confirmed by analysing e_{max}. To optimise the MSD-Battery FNF to 50 Hz accuracy as in previous studies by ABSL, it was required that $e_{max} < 5\%$. Therefore, these metamodels produced from the DSD were unsuitable for predicting the MSD-Battery response. This was an expected result, as screening designs are low-run and not used for prediction, but instead for estimation and interpretation of input-output relationships, for screening purposes [34]. Since it has been widely agreed that metamodeling settings are problem-specific [41,44], the approach taken was to use this 'rough' Screening Metamodel in the following chapter for sensitivity studies, to determine the ideal settings specific to the MSD-Battery, for building and optimising the detailed response surface to greater accuracy.

4.6.3 Pareto Plots

The Pareto plots are presented in Figure 4.4 below:

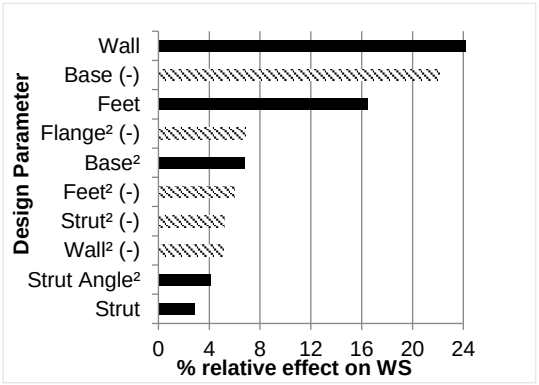


Figure 4.4a: *WS Pareto plot*

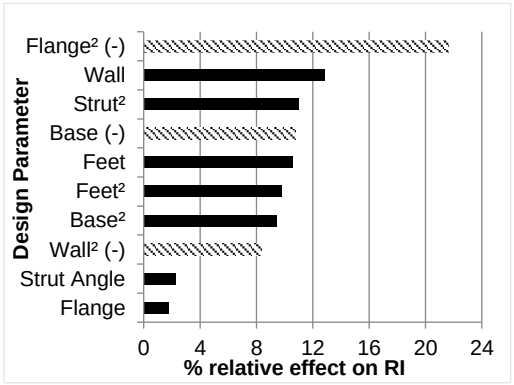


Figure 4.4b: *RI Pareto plot*

For the Pareto plots, the terms that had a positive effect on the outputs are indicated by solid black bars, and the terms with a negative effect on the outputs are indicated

by the dashed bars. The quadratic terms (x^2) show the *rate of change of the effect* of the design parameter on the outputs. For the WS, the $Flange^2$ term had a negative value of -6.88%, as shown in Figure 4.4a. This shows that as the flange thickness increases, the overall effect that the flange has on the WS decreases. However, since the first-order term for the flange was not present, the absolute impact of the flange on the output responses is insignificant [60].

The base thickness had the largest negative impact (-22.13% and -10.76%) on the WS and RI outputs, respectively (Figure 4.4a and 4.4b) highlighting the importance of the MSD-Satellite boundary condition on the FNF. As the base thickness is increased, the outputs decrease. Conversely, reducing the base thickness would increase the WS and RI, however, topology optimisation that follows shape optimisation was determined to be more suitable to develop the base. Uniformly decreasing the thickness through shape optimisation would lead to concentrated stresses at the screw location BCs (Figure 3.3a) and highlights a limitation of shape optimisation. As advised by ABSL, 6mm thickness is constrained at the screws as standard practice [67]. So, topology optimisation was performed while constraining 6mm thickness at this BC, with the results verifying the PCA results for the base thickness, as detailed by Taramona [72].

4.6.4 Main Effects Plots

The Main Effects plots are shown below in Figure 4.5a and Figure 4.5b:

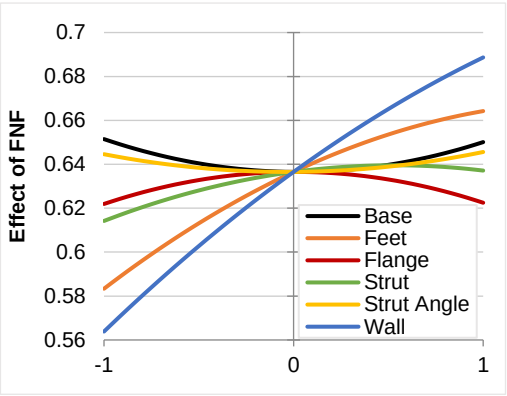


***Figure 4.5a**: FNF Main Effects plot*

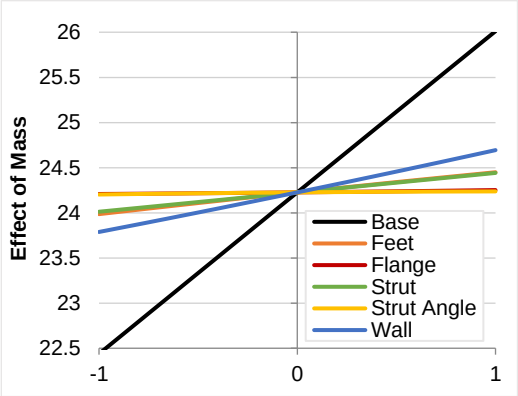


***Figure 4.5b:** Mass Main Effects plot*

The wall and feet thickness had the largest impact on the FNF, as shown in Figure 4.5, which showed agreement with the Pareto plot results. The use of a three-level design allowed for nonlinear effects to be predicted. Figure 4.5a shows that the behaviour for the FNF is nonlinear with respect to the input parameters. This was a key finding from the PCA, as it provided the insight that for the detailed DOE in Chapter 5, the metamodel-building algorithm should be suited to model nonlinearity by using suitable methods such as OLHD and RBF [42]. A drawback to the DSD came at the expense of computational efficiency since a two-level Plackett-Burman (PB) DOE can identify the first-order effects to a similar accuracy with 8-runs compared to 13 [18] particularly

for the linear mass response (Figure 4.5b). However, with no detailed prior knowledge of the MSD-battery's response, the ability to determine nonlinearity was required from the PCA and indicates the advantage to using a DSD in this thesis.

4.6.5 Principal Parameter Selection and Critical Analysis of Methods

From the Pareto plot for the WS output (Figure 4.4a), from the first-order terms, the two parameters with the greatest positive impact on the outputs were the wall and feet thicknesses. This was also verified from results from previous studies conducted on the Baseline battery model [51]. From the Pareto plot for RI (Figure 4.4b) the results were coherent with the conclusion that the wall and feet thicknesses are influential parameters, moreover, it highlighted the presence of quadratic effects which showed nonlinearity being present in the outputs. In the RI Pareto plot, seven terms were centralised between 8-13% (Figure 4.4b) and as a result, it was less apparent to distinguish the main effects. This result can be explained due to the normalisation in the calculation of RI, given in Equation (4.4), where the change in FNF was divided by the change in mass. A quality of screening designs is that as the number of significant (active) parameters in the design approaches the number of design parameters, the less accurate the separation of terms in the regression analysis – a behaviour termed "sparsity of effects" [36]. This is due to there being reduced degrees of freedom in the regression analysis to predict the quadratic and interaction terms, therefore more confounding of second-order factors occurs. The reduced ability to separate the main effects resulted in the regression analytics producing a poorer model for the RI output. This was confirmed by analysing the errors, where $R^2 = 0.975$ for the WS output, whereas for the RI output, $R^2 = 0.893$. Although both these values suggest that the models were well fitted to the training data, the e_{max} for WS and RI is 10.00% and 29.24%, respectively. This evidences that RI was overfitted and cannot provide local accuracy. From this, it is concluded that f_{ws} is better suited to screening studies compared to f_{RI}, due to the increased flexibility to assign company-specific weightings to each objective, and due to the improved accuracy in separating main effects.

Using the WS Pareto plot (Figure 4.4a), the final principal parameter was the strut thickness, as it had the third-largest positive first-order effect on the output [2].

When it comes to predicting the RI for different parameters, a DOE with a larger sample size design could provide an improvement in the output accuracy, however, due to the limited computational resources of this project, performing further screening analysis was not necessary, since the WS output also allows for far more flexibility and accuracy during optimisation than the RI output. Through this PCA, the dimensions of the design problem were reduced from six to three variables, which improved the computational efficiency for metamodeling and optimisation performed in Chapter 5.

Chapter 5 – Detailed DOE & Optimisation

5.1 Introduction

Having identified the principal parameters to be the wall, feet and strut thicknesses, this chapter establishes a validated metamodel that predicts the FNF and mass of the MSD-Battery within $e_{max} < 5\%$. Firstly, a detailed DOE is created using an Optimal Latin Hypercube DOE (OLHD). Secondly, using a Radial Basis Function (RBF), the metamodel is built and evaluated using Leave-One-Out Cross-Validation (LOOCV). Finally, the $f_{WS_{w=0.5}}$ solution is found using Multi-Objective Particle Swarm (MOPS) and the Pareto front is produced.

5.2 Detailed DOE & Optimisation Steps

Figure 5.1 below shows the steps taken to carry out the optimisation in this chapter:

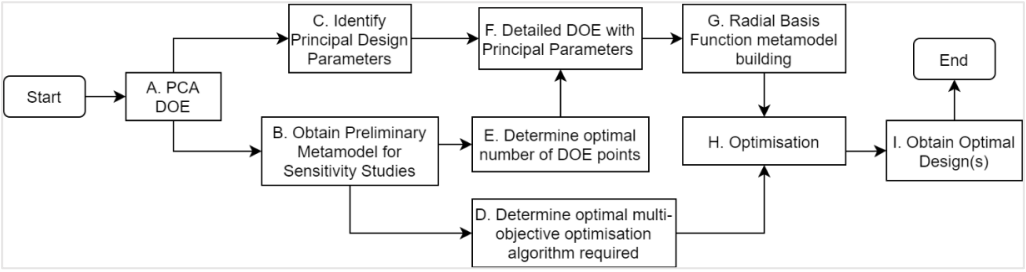


Figure 5.1: *Detailed DOE and Optimisation Flowchart Following PCA*

Each sub-heading for the methodology in this chapter is indicated (A-I) in parenthesis to highlight their position within this high-level optimisation flowchart, for clarity.

5.3 Optimal Latin Hypercube Experimental Design (F)

The design chosen for the Detailed DOE component within iSight was the Optimised Latin Hypercube design (OLHD). OLHDs are orthogonal and provide high spatial coverage [26] which made them well suited for the deterministic FEA computational simulations used in this thesis [26]. The optimisation method used within iSight to produce the OLHD is based on an enhanced stochastic evolutionary (ESE) algorithm (discussed in Section 2.1.4). Differently from PCA, the parameters were treated as continuous, instead of having three levels, to allow for greater exploration of the design space. In discussions with ABSL, it was confirmed that all structural parts are machined using a CNC machine, allowing for freedom of varying the parameters within the design range (compared to using standard aluminium sheet thicknesses to discretise the design parameters). The limits for each design parameter are given below in Table 5.1:

Table 5.1: *Design Parameter Limits for Detailed OLHD*

Design Parameter	**Design Limits (mm)**
Wall Thickness	$0.5 \leq x_{wall} \leq 2$
Feet Thickness	$3 \leq x_{feet} \leq 12$
Strut Thickness	$1 \leq x_{strut} \leq 4$

30 points were used for the Detailed DOE component, determined through a sensitivity study, which is outlined in Section 5.6.1.

5.4 Metamodel Building and Validation

5.4.1 Radial Basis Function Metamodel (G)

A Radial Basis Function (RBF) was used to build the metamodel, as this technique has been widely used in literature as the ideal follow-up algorithm for complementing an OLHD to produce an approximation model [12]. OLHD combined with RBF has been proven to outperform other methods of metamodeling for optimisation across a wide range of measures such as accuracy, robustness and computational efficiency [42].

RBF uses multi-dimensional data-mapping between the input design parameter data, a hidden layer, and the outputs. This technique is done using the interpolation function which takes the form as shown below in Equation (5.1) [12,60]:

$$F(x) = \sum_{i=1}^{N} \lambda_i \phi(\|x - x_i\|) + bx + c \tag{5.1}$$

Where: $\|x - x_i\|$ is the Euclidean norm; λ_i is the expansion coefficients for i^{th} basis function; $\phi(x)$ is the basis function where $\phi(x) = \|x - x_i\|^c$ for iSight $(0.2 < c < 3)$; N is the number of training data sampling points.

The unknown parameters are then solved using Equation (5.2) below:

$$\begin{Bmatrix} \lambda \\ a \end{Bmatrix} = \begin{bmatrix} \Phi & P \\ P^T & 0 \end{bmatrix}^{-1} \begin{Bmatrix} F \\ 0 \end{Bmatrix} \tag{5.2}$$

Where: λ is the expansion coefficient vector; a is the polynomial coefficient vector; P is the projection matrix [46]. The resulting λ and a vector is used to interpolate between the input parameters and the FNF and mass of the battery structure.

5.4.2 Leave-One-Out Cross-Validation and Sequential Sampling (G)

It was necessary to validate the resulting RBF metamodel for accuracy at various locations within the design space, and this was done using Leave-One-Out Cross-Validation (LOOCV). Automated sequential sampling was set up within the iSight Detailed DOE component to iteratively nest points in the design space at positions of maximum error if the results of the LOOCV produces a metamodel with $e_{max} < 5\%$ for the mass or FNF outputs. This validation procedure is shown below in Figure 5.2:

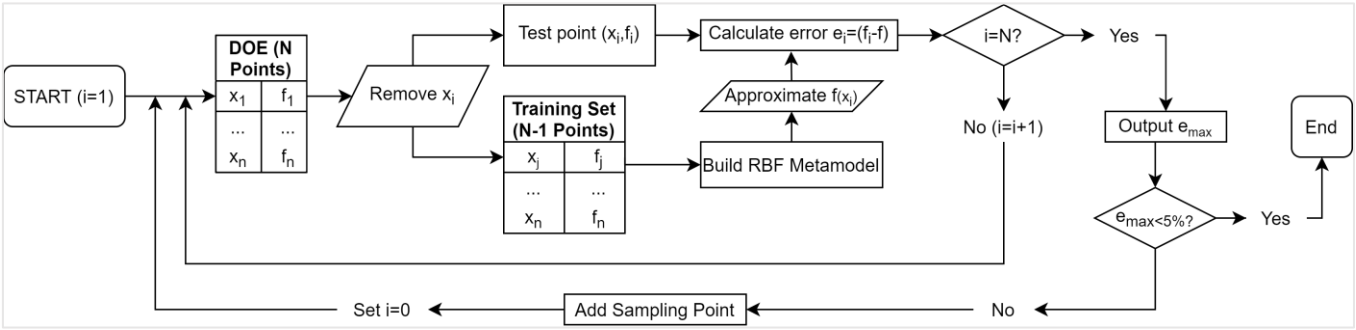


Figure 5.2: *LOOCV and Sequential Sampling Loop*

5.5 Multi-Objective Particle Swarm Optimisation (H)

Based on a sensitivity study comparing a multi-objective particle swarm (MOPS) algorithm and a multi-objective genetic algorithm (MOGA) within the iSight optimisation component, it was found that MOPS outperformed MOGA for locating the Pareto set of solutions on the Screening Metamodel (outlined in Section 5.6.2). From this, MOPS was selected to optimise the detailed metamodel. This section outlines the MOPS algorithm that was used in iSight to optimise the detailed metamodel, to find the WS-Optimum solution, $f_{WS_{w=0.5}}$, and the Pareto set of solutions.

5.5.1 Multi-Objective Particle Swarm Optimisation Flowchart (H)

The MOPS search algorithm replicates the behaviour of fish and birds that showcase the social principles of evaluation, comparison and imitation, which enables the non-dominated points to move towards the regions of higher fitness [73] (as further outlined in Section 2.4.2). The flowchart for MOPS is given below in Figure 5.3:

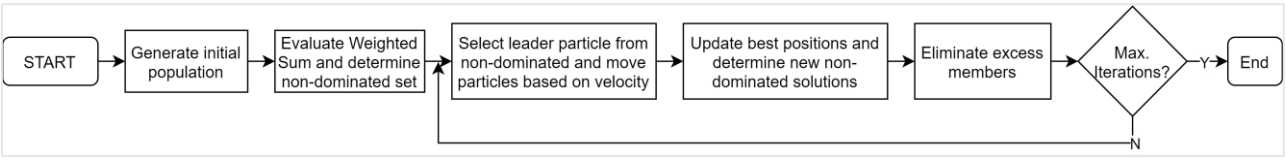


Figure 5.3: *MOPS Algorithm Flowchart*

5.5.2 Shape Optimisation Outputs (I)

The two main outputs obtained from the optimisation are outlined below:

1) $f_{WS_{w=0.5}}$: The WS-Optimum design was found as the main optima, to balance the improvements to the FNF with the changes in mass. This design was passed onto topology optimisation to reduce the mass whilst maintaining the improved FNF.
2) Pareto front: Each satellite manufacturer imposes their constraints on the mass budget for the satellite battery, therefore the ability to be able to identify the optimal design for a range of potential design masses was a critical outcome for ABSL – and this is exactly what the Pareto front delivers. The Pareto front is a set of non-dominated solutions within the design space (domination is explored in Section 2.4.1). Since this was found on the metamodel, it required very little computational power and time – 500 iterations of design points were evaluated, in <10 seconds, and the Pareto solutions identified. If this were to be determined through FEA alone, 500 test points would take 67 hours to produce on the hardware used (~8 minutes per simulation), and more to compile into a Pareto plot, and highlights the benefits to the metamodeling approach implemented into the iSight optimisation loop.

5.5.3 Validation of the Predicted Optima (I)

Once the $f_{WS_{w=0.5}}$ predicted optimum was located from the metamodel, the design was created and simulated in Abaqus to determine the error between the metamodel and the actual values for the MSD-Battery responses, to validate the result.

5.6 Sensitivity Studies Overview and Results

5.6.1 OLHD DOE Points Sensitivity Study (E)

As shown in the high-level flowchart in Figure 5.1 at Point E, a sensitivity study was conducted on the Screening Metamodel to determine the number of OLHD points that were required to build an RBF metamodel with an average error $e_{max} < 10\%$.

Four OLHDs were tested by increasing the number of design points $(N = 10, 20, 30, 40)$ tested and e_{max} outputted for the FNF and mass response. This was necessary to avoid long-running iterations of the LOOCV and Sequential Sampling loop. The result for this is shown below in Figure 5.4 and Table 5.2:

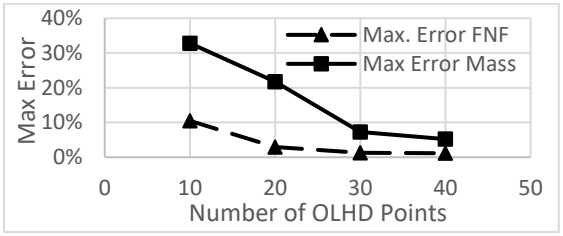


***Figure 5.4**: OLHD Points Sensitivity Study*

***Table 5.2:** OLHD Points SS*		
OLHD Points	**Max. Error e_{max}**	
	Mass	***FNF***
10	32.83%	10.49%
20	21.73%	2.95%
30	7.30%	1.26%
40	5.22%	1.16%

The errors for both FNF and mass converged below $e_{max} < 10\%$ at 30 design points, and the sequential sampling loop in Figure 5.2 would reduce this to $e_{max} < 5\%$. As a result of this sensitivity study for OLHD, 30 design points were selected.

5.6.2 Optimisation Algorithm Sensitivity Studies (D)

The optimisation algorithms considered in this sensitivity study was MOGA (outlined in Section 2.4.2) and MOPS (outlined in Section 5.5.1).

To determine the best algorithm for use on the OLHD-RBF metamodel, a sensitivity study was performed looking at the performance of both MOGA and MOPS on the Screening Metamodel. To ensure a fair comparison, the number of function evaluations for both optimisation algorithms were kept equal, with the number of function evaluations for MOGA and MOPS given by Equations (5.3) and (5.4) [50]:

$$Function\ Evaluations_{MOPS} = Swarm\ Size \times Iterations \quad (5.3)$$

$$Function\ Evaluations_{MOGA} = Population\ Size \times Generations \quad (5.4)$$

The number of function evaluations for both algorithms was limited to 500, to prevent a long-run problem [12] with the algorithm settings used displayed below in Table 5.2:

***Table 5.2:** Baseline Optimisation Algorithm Settings* [60]				
MOGA Algorithm Settings			**MOPS Algorithm Settings**	
Population Size	10		*# Particles*	10
# Generations	50		*# Iterations*	50
Crossover Probability	0.9		*Inertia*	0.9
Crossover Distribution Index	10		*Global Increment*	0.9
Mutation Distribution Index	20		*Particle Increment*	0.9
			Maximum Velocity	0.1

These solver settings were based on those found in literature for similar studies [50,56] and the ability of both algorithms to locate the optimal solutions was tested.

The WS-Optimum function, $f_{WS_{w=0.5}}$, was used to locate the main optima within the design space. From the results of this sensitivity study, both the MOGA and MOPS algorithms located $f_{WS_{w=0.5}}$on the Screening Metamodel, in 8 and 7 seconds respectively, for the weighted sum $f_{WS_{w=0.5}}$ (shown below in Table 5.3):

Table 5.3: *Screening Approximation Model Optimal Design Point*

Design Parameters (mm)			**Metamodel Outputs**		
Wall	***Feet***	***Strut***	***Mass (kg)***	***FNF (kHz)***	$f_{WS_{w=0.5}}$
2	12	1	22.9248	0.718051	0.11101

Since both algorithms located the optima, to select the best optimisation algorithm for the detailed RBF metamodel, the Pareto points identified by both algorithms within 500 function evaluations were also examined and are shown in Figure 5.5:

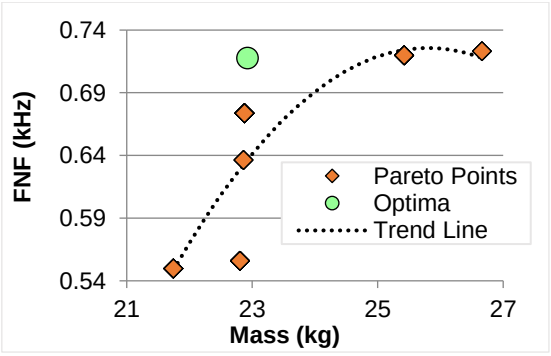


Figure 5.5a: *MOPS Pareto Points*

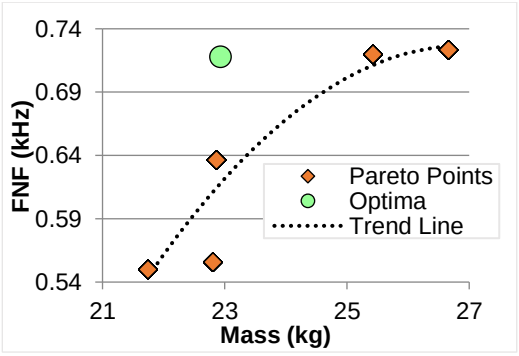


Figure 5.5b: *MOGA Pareto Points*

Using the number of Pareto points to compare MOO algorithms is the most widely used approach used in literature [50,57]. An assumption made was that the Screening Metamodel will be similar to the detailed RBF metamodel and that the erroneous e_{max} of the Screening model will make convergence to the non-dominated set more difficult in this sensitivity study, ensuring that the most robust algorithm was selected.

From the results, the MOPS algorithm located the $f_{WS_{w=0.5}}$ design, whilst also locating 16.7% more Pareto optimal points compared to MOGA (shown in Figure 5.5) which allowed for better extrapolation of the Pareto front. To verify these results, the experiment was repeated three times to ensure this was not due to the stochastic nature of the algorithms [53] – however in each experiment MOPS outperformed MOGA as shown above, and thus was selected for the iSight optimisation component.

5.7 Results

5.7.1 Optimal Latin Hypercube (Detailed) DOE

The non-principal parameters are kept at their baseline values; Flange thickness = 1mm, Strut Angle = 84.5°, Base thickness = 6mm. Figure 5.6 shows the 30-run OLHD generated (exact table of parameter values used are given in Appendix A):

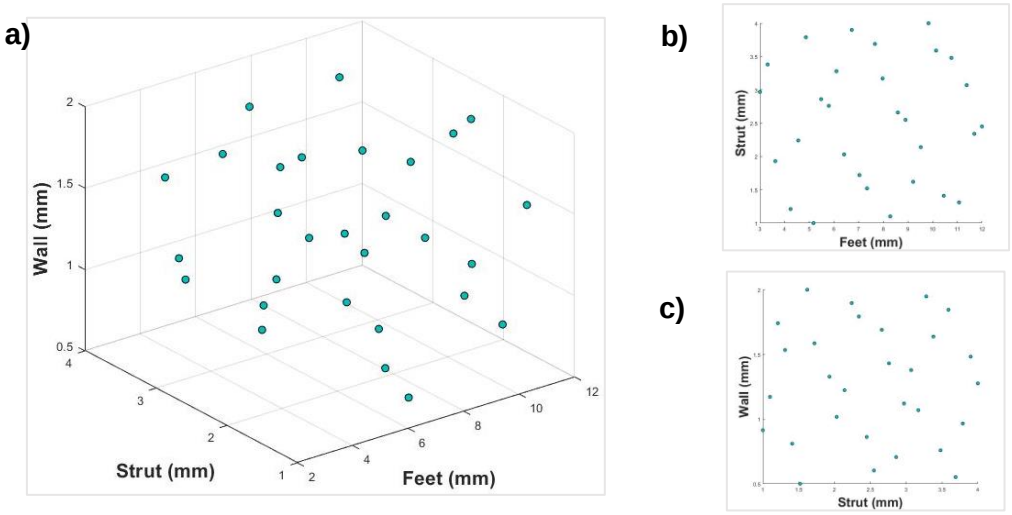


Figure 5.6: *OLHD Generated a) Isometric b) Strut-Feet c) Wall-Strut views*

5.7.2 OLHD-RBF Metamodels

The RBF metamodels created for the FNF and mass using the 30-run OLHD are illustrated below in Figure 5.7 and Figure 5.8, respectively:

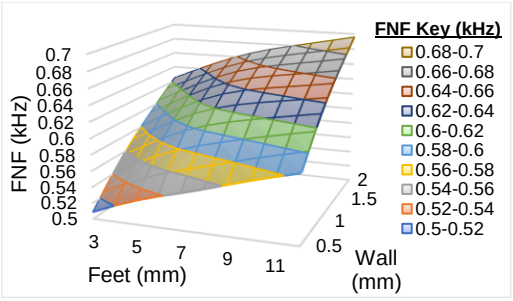


Figure 5.7a: *Wall & Feet thickness vs FNF*

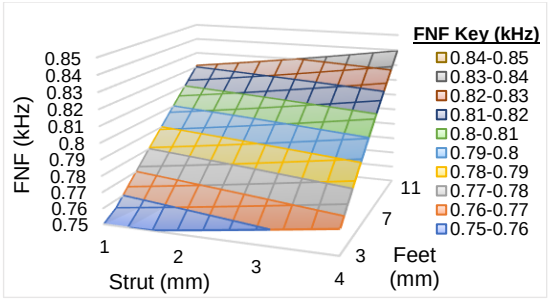


Figure 5.7b: *Feet & Strut thickness vs FNF*

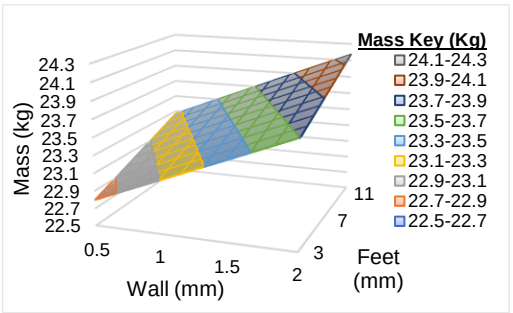


Figure 5.8a: *Wall & Feet thickness vs Mass*

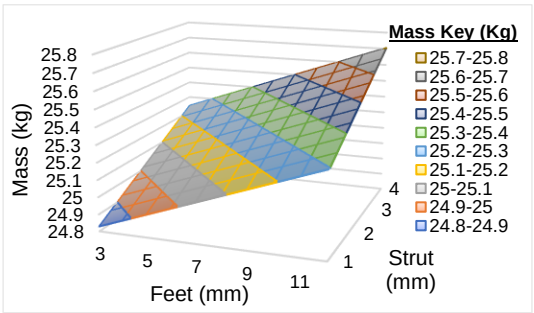


Figure 5.8b: *Feet & Strut thickness vs Mass*

The RBF metamodels displayed a more linear response when compared to the Screening Metamodels, which was attributed to the errors present in the low-run DSD – where the nonlinear nature of the metamodel was overestimated during RSM analysis. The RBF metamodels show that increasing the design parameter thickness increases the FNF and mass of the battery structure across the entire design space. This was an intuitive result, due to the associated increase in structural stiffness [1].

5.7.3 LOOCV and Error Analysis

The 10-run LOOCV study for the OLHD-RBF metamodel is shown in Table 5.4:

Table 5.4: *LOOCV Data and Error Analysis*							
	Parameter Thickness (mm)			**Mass (kg)**		**1st Mode (kHz)**	
LOOCV #	**Feet**	**Strut**	**Wall**	**Actual**	**Predicted**	**Actual**	**Predicted**
1	10.45	1.41	0.81	23.347	23.346	0.605	0.603
2	11.07	1.31	1.534	23.804	23.802	0.662	0.663
3	8.9	2.55	0.603	23.294	23.294	0.581	0.582
4	11.69	2.34	1.793	24.129	24.130	0.686	0.687
5	7.66	3.69	0.552	23.357	23.357	0.572	0.577
6	7.97	3.17	1.069	23.613	23.612	0.630	0.627
7	10.14	3.59	1.845	24.251	24.251	0.688	0.692
8	9.83	4.00	1.276	23.947	23.948	0.657	0.656
9	12	2.45	0.862	23.597	23.599	0.625	0.625
10	9.21	1.62	2.00	24.028	24.046	0.680	0.684
				Mass $R^2 = 0.99969$		**FNF $r^2 = 0.9959$**	
				Mass $e_{max} = 1.890\%$		**FNF $e_{max} = 3.949\%$**	

For the Screening Metamodel, $e_{max} = 45.88\%$, whereas for the detailed metamodel the error was reduced to $e_{max} = 3.95\%$ by using the 30-run OLHD and RBF. The R^2=0.99 for both the mass and the FNF and the maximum total error was found to be 3.95% for the FNF. This assured both the global and local accuracy of the RBF metamodel, which made it well-suited for optimisation to predict the optimum point [12].

5.7.4 Shape Optimisation on RBF Metamodel

Using the MOPS optimisation algorithm, $f_{WS_{w=0.5}}$ was found, repeating the result three times and cross-validated in Abaqus. The results are shown below in Table 5.5:

Table 5.5: *WS-Optimum Solution from Metamodel and Validation in FEA*								
Parameters (mm)			**Predicted (RBF)**		**Actual (FEA)**		**Error e**	
Feet	***Strut***	***Wall***	***Mass (kg)***	***FNF (kHz)***	***Mass (kg)***	***FNF (kHz)***	***Mass***	***FNF***
12	4	2	24.4959	0.7095	24.4100	0.7027	0.352%	0.968%

The Pareto front was obtained from the set of results and is shown in Figure 5.9 with $f_{WS_{w=0.5}}$ in green and the polynomial equation for the FNF vs mass:

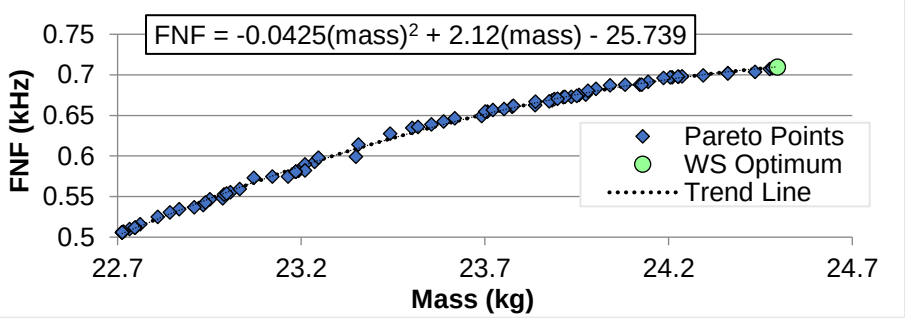


Figure 5.9*: RBF Pareto Points with $f_{WS_{w=0.5}}$ optimum located*

The Pareto optimal set provides the means to determine the exact FNF of the MSD-battery module for a range of design masses. The parameters for each Pareto optimal solution are given in Appendix B. Although the Pareto front showed some indication of the FNF levelling off, this was outside the design limits of the current NARMER design. The WS Optimum and Pareto solutions are discussed further in Section 5.8.

5.8 Discussion

5.8.1 Interpretation of Optimum Solutions

Differently from the MOPS Screening Pareto set (Figure 5.5a), the $f_{WS_{w=0.5}}$ optimum in the RBF Pareto set (Figure 5.9) was located at the edge of the design space where the parameter upper limits were an active constraint (Table 5.5). The false optimum from the Screening Metamodel was a result of the large errors present in the mass output ($R^2 = 0.66$ and $e_{max} = 45.88\%$ from Table 4.2) which made the mass more sensitive to design changes, as indicated by the exaggerated mass range in the x-axis.

Although the true $f_{WS_{w=0.5}}$ solution found was constrained on the RBF Pareto front (Figure 5.9), extending the design limits beyond the upper-limit values chosen would result in large element deformation and intersecting geometries within the current FEA model, moreover, searching beyond this range would not guarantee the presence of an unconstrained optimum, therefore it was not considered in these works. Instead, if FNF improvements are needed beyond what is shown, it would be required to overhaul the battery design, with the Baseline re-designed to incorporate thicker structures and greater parameter limits – which cannot be done in the current NARMER configuration.

An alternative solution is the 'Compromise' solution which was found by minimising the normalised Euclidean distance between the Pareto points and the Utopia point f_0 (theoretical best) in the criterion space [74]. This was located by setting the Utopia point as the target solution for MOPS within iSight, and is shown in Figure 5.10:

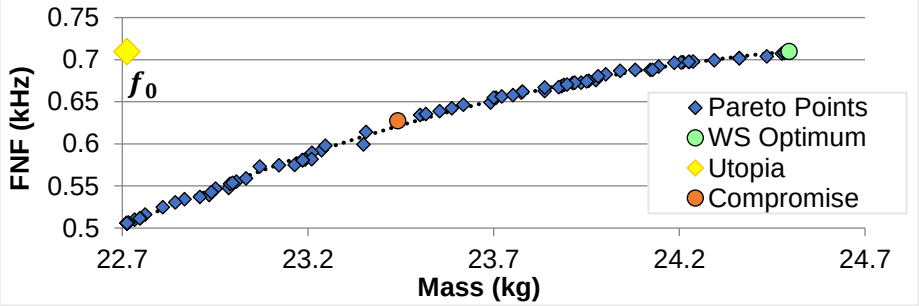


Figure 5.10: *Pareto front with the Utopia solution and compromise solution highlighted*

Both the Compromise and $f_{WS_{w=0.5}}$ solutions are equally Pareto optimal, and ABSL would typically inform the selected design. If shape optimisation was the only optimisation step for the MSD-Battery, the Compromise solution represents the design closest to the Utopia point in the criterion space. The Compromise solution was not selected as the candidate design to take forward for topology optimisation, as there was no maximum mass to constrain this study. Knowing that topology optimisation would follow shape optimisation, $f_{WS_{w=0.5}}$ was selected as it provided the best increase-in-FNF for an increase-in-mass within ABSL's parameter limits.

The WS-Optimum $f_{WS_{w=0.5}}$ shape-optimised geometry was passed onto topology optimisation, to reduce the mass whilst maintaining the vibrational improvements, with the performance of the resulting shape and topology optimised structure analysed by Taramona [72]. From the shape optimisation, the $f_{WS_{w=0.5}}$ increased the MSD-battery mass by 4.87% from 23.2578kg to 24.4100kg and the FNF by 18.12% from 594.9Hz to 702.7Hz. This was an improvement over previous optimisation efforts made on the Baseline model, where the FNF could only be optimised to a 1.22% increase for a 1.13% increase in mass [51]. This optimum was validated in Abaqus to an accuracy of 0.968% and highlights the improvements made to the framework delivered to ABSL in this thesis. The accuracy of the design predicted in FEA was significant – as it was located at a corner of the design space which is often the position of maximum error for an OLHD [17]. The lack of corner nodes for OLHDs, as can be seen in Figure 5.5, typically leads to greater maximum error when compared to FrF DOEs [75]. Despite this, the optimum was found to <1% accuracy at this position, so the OLHD point sensitivity study and the use of RBF were beneficial to the resulting accuracy.

5.8.2 Optimisation Algorithms

The best-performing optimisation algorithm was found to be MOPS, however, this was not a universal result for all engineering optimisation problems, as MOGA has also been shown to vastly outperform MOPS in discontinuous design spaces [56]. This demonstrates the importance of prior knowledge of the metamodel to be optimised, to allow for complementary methods of optimisation to be selected. In this study, OLHD and RBF metamodeling was chosen based on findings from the PCA. Moreover, performing sensitivity studies on the Screening Metamodel ensured that the ideal settings were used for the detailed metamodel. Various studies have compared MOPS and MOGA, and it was evidenced through this thesis as well as in literature, that for simply locating Pareto optimal solutions, MOPS can outperform MOGA, whilst also having improved computational efficiency [50,56]. A limitation of only using the $f_{WS_{w=0.5}}$ optimal solution for the MOO algorithm sensitivity study was that it only searches for a single position on the Pareto optimality set and may not hold true for searching for all Pareto solutions [49]. Future studies could extend these works to test the ability of different MOO algorithms to derive several specific positions on the Pareto front (For example $w_{f,m} = 0.2, 0.4, 0.6, 0.8$) however, as both algorithms used in this study located the same $f_{WS_{w=0.5}}$ optimum – which verified the required shape-optimised design – additional studies were unnecessary. For future works, gradient-based search optimisation could also further improve computational efficiency, due to the linearity of the metamodels (Figure 5.7 and 5.8) where local optima are not apparent [47].

Chapter 6 – Baseline Battery Optimisation

6.1 Introduction

Using the same methods as for the MSD-Battery, shape optimisation was performed on the Baseline battery to verify the robustness of the iSight optimisation framework to optimise the FNF and mass response of a structure. The key results are presented here as an executive summary, to demonstrate the use of the Pareto sets for selecting an optimum design. Additional detail can be referred to in Appendix C.

6.2 PCA, Detailed DOE and Optimisation Results

6.2.1 Principal Component Analysis

The results for the DSD correlated strongly with the results for the MSD-Battery, with the key parameters identified to be the wall, feet, and strut thicknesses.

6.2.2 OLHD-RBF Metamodels

A 30-run OLHD was used to obtain the RBF metamodels which were validated with a minimum $r^2 = 0.99885$ and maximum total error of 2.09% for the FNF response.

6.2.3 Optimisation of the RBF Metamodel

ABSL stated a maxi-mass of 21.4kg [67] for the Baseline, so the Pareto front was used to identify the appropriate solution as shown in Figure 6.1, and validated in Table 6.1:

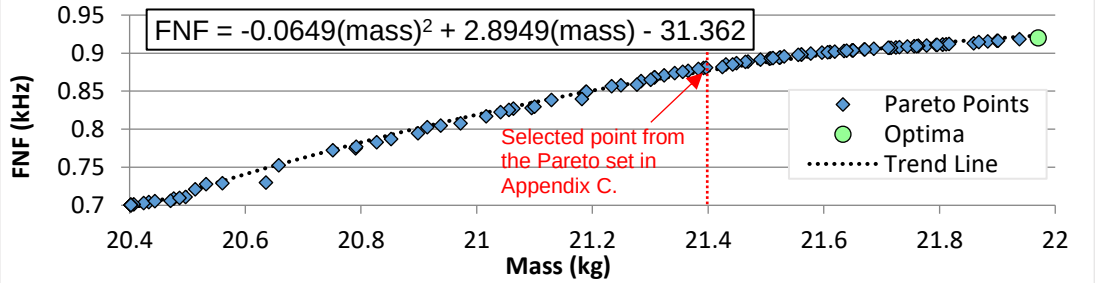


Figure 6.1: *Pareto optimal set and locating the 21.4kg Pareto optimum (in red)*

Table 6.1: *Validation of the closest Pareto optimal solution within the 21.4kg maxi-mass*

Parameters (mm)			Predicted (RBF)		Actual (FEA)		Error e	
Feet	***Strut***	***Wall***	***Mass (kg)***	***FNF (kHz)***	***Mass (kg)***	***FNF (kHz)***	***Mass***	***FNF***
5.986	1	2	21.392	0.880	21.309	0.876	0.390%	0.548%

6.3 Conclusions

This chapter verified the ability of the iSight optimisation framework to locate the Pareto set of solutions for the Baseline battery. Moreover, it showed how the mass constraints from a manufacturer can be used to locate the Pareto optimum. This resulted in optimising Baseline Battery FNF from 0.783kHz to 0.876kHz (10.62%), with only a 20.98kg to 21.31kg (1.57%) increase in mass, validated to 0.548% accuracy using FEA in Abaqus. The cycle-time reduction brought by the iSight framework improves upon previous works and make it suitable for future shape optimisation studies conducted.

Chapter 7 – Conclusions

7.1 Achievements

1) Conducted a literature review into shape optimisation, which underlined the development of the methods carried out for DOE, PCA, Metamodeling and MOO.
2) Implemented the MSD System onto the Baseline Battery Structure using Abaqus.
3) Carried out a PCA to determine the key parameters impacting the FNF and mass of the MSD-Battery, reducing the dimensionality of the design problem.
4) Conducted a detailed DOE using an OLHD, to build an accurate RBF metamodel for the FNF and mass response surfaces, which was validated using LOOCV.
5) Optimised the RBF metamodel using MOPS, producing the shape optimised design, which was validated in FEA to confirm the accuracy of the methods used.
6) Further verified and validated the iSight optimisation framework through repeating the shape optimisation for the Baseline NARMER Battery structure.

7.2 Discussions

In Chapter 2, from the literature review into shape optimisation methods, the most useful literature found for this report was the comparison of metamodeling algorithms for engineering-specific problems [41,42] which allowed RBF and OLHD to be identified as applicable techniques for the battery structures. Although the relative use of DOE and metamodeling methods in engineering is low [23] papers that target the application of statistical methods to the non-expert can help improve its accessibility to industrial applications [15,36] as was the case for this thesis.

In Chapter 4, the three-level DSD design was able to detect quadratic effects, which allowed for the selection of complimentary metamodeling techniques but compromised the ability to interpret the final principal parameter. With metamodel nonlinearity determined in detail through this thesis, future studies can focus on parameter screening. The two-level PB design has been evidenced to be as effective as DSDs for interpreting principal parameters in trade-off studies [36], whilst also requiring fewer FEA runs (8-runs) compared to the DSD (13-runs) [18].

Chapter 5 showed that the final RBF metamodel for the battery responses was mostly linear. Gradient-based search optimisation methods, such as gradient-based genetic algorithms, are a recent development that have shown improved convergence to constrained optima for response surfaces without local minima [76]. This could represent an improved algorithm for the metamodels obtained in this study, however, MOPS remains the most versatile in cases where the metamodel topography is not previously known. This is due to the balance struck between computational efficiency

and robustness, having been proven successful for locating the Pareto solutions in this thesis and also literature for engineering applications [50,56]. An improvement to optimisation formulation would integrate the outputs for the thermal, stress and vibrational acceleration in iSight to remove the need for post-optimisation analysis. This was not possible in this thesis due to the inability to access university facilities, which meant computational simulations had to be run using limited hardware. As a result, only the FNF and mass were used for the optimisation, but as shown in the wider group project [2], maximising FNF was successful at reducing the vibrational accelerations experienced by the Li-Ion cells and was sufficient for achieving the CPP's objectives.

Regarding the methods for optimisation formulation used, the computational cost of obtaining the Compromise solution is shown to rapidly increase with the dimensionality of the output responses compared to WS [74], moreover, WS allows for ABSL to assign specific function weightings and locate the Pareto set. Chapter 6, where the optimisation framework was verified on the Baseline structure, demonstrated the practicality of the Pareto sets. The Pareto set allowed the optimum design to be located using ABSL's Baseline mass constraint, further highlighting the robustness of the WS formulation, and making it the recommended formulation for future studies.

7.3 Conclusions

For both the MSD-Battery and Baseline structure, it was found that the principal parameters were the wall, feet, and strut thicknesses. The OLHD-RBF metamodels produced for the MSD-Battery and Baseline models were validated to $e_{max} = 0.968\%$ and 0.548%, respectively. The optimum design located by the MOPS algorithm improved the MSD-Battery structure's FNF by 18.12% with only a 4.87% increase in mass. The optimisation was repeated with the Baseline battery and increased the FNF by 10.62% with only a 1.57% increase in mass. The final shape-optimised solutions were then verified in FEA to e_{max}<1%, building confidence in the use of the iSight framework created in this thesis. Through the work outlined above, all the aims and objectives identified in the CPP were achieved for shape optimisation.

7.4 Future Works

1) Integrating the stress, acceleration, and thermal responses as constraints in the optimisation problem formulation to reduce the requirement for additional analysis on the optimum design located from the optimisation loop.
2) Use of a PB DOE for PCA to aid the interpretation of the principal parameters, and a comparison made between PB and DSD for screening studies.
3) A wider study of optimisation algorithms to determine the most cost-effective method of optimisation on the battery metamodels, comparing gradient-based methods to MOPS and MOGA on mostly linear response surfaces.

References

1. Zhu L, Li N, Childs PRN. Light-weighting in aerospace component and system design. Propuls Power Res. 2018;7(2):103–19.
2. D'souza M, Taramona Pérez M, Zziwa A, Asad T. DESIGN OPTIMISATION OF A MASS-SPRING-DAMPER SYSTEM FOR A SATELLITE BATTERY STRUCTURE. University of Leeds; 2021.
3. Alsharif F, McNaughton Jr JL. Optimization Analysis of a V-Twin Motorcycle Engine Using WAVE Cycle Analysis and an iSight Optimization Framework. Ricardo Softw e-Brochure v3. 2005;
4. Davidian K. Definition of NewSpace. New Sp [Internet]. 2020;8(2):53–5. Available from: https://doi.org/10.1089/space.2020.29027.kda
5. Daehnick C, Klinghoffer I, Maritz B, Wiseman B. Large LEO satellite constellations: Will it be different this time? [Internet]. McKinsey. 2021 [cited 2021 Apr 24]. Available from: https://www.mckinsey.com/industries/aerospace-and-defense/our-insights/large-leo-satellite-constellations-will-it-be-different-this-time#
6. Etherington D. OneWeb launches 36 satellites to join its global broadband constellation on orbit [Internet]. Techcrunch. 2020 [cited 2021 Apr 6]. Available from: https://techcrunch.com/2020/12/18/oneweb-launches-36-satellites-to-join-its-global-broadband-constellation-on-orbit/
7. Walker W, Yayathi S, Shaw J, Ardebili H. Thermo-electrochemical evaluation of lithium-ion batteries for space applications. J Power Sources [Internet]. 2015;298:217–27. Available from: http://dx.doi.org/10.1016/j.jpowsour.2015.08.054
8. Jorge P, Ravi N. Noise and Vibration of Spacecraft Structures. Ingeniare Rev Chil Ing. 2006;
9. Uno M, Ogawa K, Takeda Y, Sone Y, Tanaka K, Mita M, et al. Development and on-orbit operation of lithium-ion pouch battery for small scientific satellite "rEIMEI." J Power Sources [Internet]. 2011;196(20):8755–63. Available from: http://dx.doi.org/10.1016/j.jpowsour.2011.06.051
10. Furger S. Analysis and Mitigation of the Cubesat Dynamic Environment. California Polytechnic State University; 2013.
11. Woods J, Ng KL, T C, Curzon D. Optimisation of Space Battery Structural Analysis Report. Oxford; 2016.
12. Park HS, Dang XP. Structural optimization based on CADCAE integration and metamodeling techniques. CAD Comput Aided Des [Internet]. 2010;42(10):889–902. Available from: http://dx.doi.org/10.1016/j.cad.2010.06.003
13. Zziwa A. Structural Design of a Mass-Spring-Damper System for a Satellite Battery Structure. University of Leeds; 2021.
14. Haftka RT, Grandhi R V. Structural shape optimization-A survey. Comput Methods Appl Mech Eng. 1986;57(1):91–106.
15. Tanco M, Viles E, Pozueta L. Comparing Different Approaches for Design of Experiments (DoE). In: Ao S-I, Gelman L, editors. Advances in Electrical Engineering and Computational Science [Internet]. Dordrecht: Springer Netherlands; 2009. p. 611–21. Available from: https://doi.org/10.1007/978-90-481-2311-7_52
16. Niedz RP, Evens TJ. Design of experiments (DOE)—history, concepts, and relevance to in vitro culture. Vitr Cell Dev Biol - Plant [Internet]. 2016;52(6):547–62. Available from: http://dx.doi.org/10.1007/s11627-016-9786-1
17. Viana FAC. Things You Wanted to Know About the Latin Hypercube Design and Were Afraid to Ask. 10th World Congr Struct Multidiscip Optim. 2013;1(1):1–9.
18. JMP® 15.2.1. Orthogonal Designs [Internet]. Orthogonal designs. 2021 [cited 2021 Mar 30]. Available from: https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/doe/supporting-topics/basics/orthogonal-designs/
19. Yates F. Sir Ronald Fisher and the Design of Experiments Author (s): F . Yates Reviewed work (s): Source : Biometrics , Vol . 20 , No . 2 , In Memoriam : Ronald Aylmer Fisher , 1890-1962 (Jun ., Published by : International Biometric Society Stable URL : http. Biometrics. 1964;20(2):307–21.
20. Fisher J. R.A. fisher and the design of experiments, 1922-1926. Am Stat. 1980;34(1):1–7.

21. Savic I, Gajic D, Stojiljkovic S, Savic I, Gennaro S di. Modelling and optimization of methylene blue adsorption from aqueous solution using bentonite clay [Internet]. Vol. 33, Computer Aided Chemical Engineering. Elsevier; 2014. 1417–1422 p. Available from: http://dx.doi.org/10.1016/B978-0-444-63455-9.50071-4
22. Vinayagamoorthy R. Parametric optimization studies on drilling of sandwich composites using the Box–Behnken design. Mater Manuf Process [Internet]. 2017;32(6):645–53. Available from: http://dx.doi.org/10.1080/10426914.2016.1232811
23. Ilzarbe L, Álvarez MJ, Viles E, Tanco M. Practical applications of design of experiments in the field of engineering: A bibliographical review. Vol. 24, Quality and Reliability Engineering International. 2008. p. 417–28.
24. NIST/SEMATECH. e-Handbook of Statistical Methods [Internet]. e-Handbook of Statistical Methods. 2006 [cited 2021 Mar 30]. Available from: https://www.itl.nist.gov/div898/handbook/
25. Davey KR. Latin Hypercube Sampling and Pattern Search in Magnetic Field Optimization Problems. IEEE Trans Magn. 2008 Jun;44(6):974–7.
26. Jin R, Chen W, Sudjianto A. An efficient algorithm for constructing optimal design of computer experiments. J Stat Plan Inference [Internet]. 2005;134(1):268–87. Available from: https://proceedings.asmedigitalcollection.asme.org
27. Morris MD, Mitchell TJ. Exploratory designs for computational experiments. J Stat Plan Inference. 1995;43(3):381–402.
28. Bates SJ, Sienz J, Toropov V V. Formulation of the optimal latin hypercube design of experiments using a permutation genetic algorithm. Collect Tech Pap - AIAA/ASME/ASCE/AHS/ASC Struct Struct Dyn Mater Conf. 2004;7(March 2015):5217–23.
29. Chen RB, Hsieh DN, Hung Y, Wang W. Optimizing Latin hypercube designs by particle swarm. Stat Comput. 2013;23(5):663–76.
30. Shi L, Sun B. Modeling and optimization of vibration response characteristics of the orbital sander based on surrogate model. Struct Multidiscip Optim. 2018;57(6):2259–71.
31. You YM. Optimal design of PMSM based on automated finite element analysis and metamodeling. Energies. 2019 Dec 9;12(24).
32. Box GEP, Meyer RD. Finding the Active Factors in Fractionated Screening Experiments. J Qual Technol [Internet]. 1993;25(2):94–105. Available from: https://doi.org/10.1080/00224065.1993.11979432
33. Yang XS, Koziel S, Leifsson L. Computational optimization, modelling and simulation: Past, present and future. Procedia Comput Sci [Internet]. 2014;29(Coms):754–8. Available from: http://dx.doi.org/10.1016/j.procs.2014.05.067
34. Jones B, Nachtsheim CJ. A Class of Three-Level Designs for Definitive Screening in the Presence of Second-Order Effects A Class of Three-Level Designs for Definitive Screening in the Presence of Second-Order Effects. J Qual Technol. 2017;43(1):1–15.
35. Schäfer C, Finke E. Shape optimisation by design of experiments and finite element methods-an application of steel wheels. Struct Multidiscip Optim. 2008 Nov;36(5):477–91.
36. Santos CP, Rato TJ, Reis MS. Design of Experiments: A comparison study from the non-expert user's perspective. J Chemom. 2019 Jan 1;33(1).
37. Mohamed OA, Masood SH, Bhowmik JL. Influence of processing parameters on creep and recovery behavior of FDM manufactured part using definitive screening design and ANN. Rapid Prototyp J [Internet]. 2017 Jan 1;23(6):998–1010. Available from: https://doi.org/10.1108/RPJ-12-2015-0198
38. Mohamed OA, Masood SH, Bhowmik JL. Modeling, analysis, and optimization of dimensional accuracy of FDM-fabricated parts using definitive screening design and deep learning feedforward artificial neural network. Adv Manuf [Internet]. 2021;9(1):115–29. Available from: https://doi.org/10.1007/s40436-020-00336-9
39. Mueller L, Alsalihi Z, Verstraete T. Multidisciplinary Optimization of a Turbocharger Radial Turbine. J Turbomach. 2012;135(2):1–9.
40. Barton RR. Simulation optimization using metamodels. Proc - Winter Simul Conf. 2009;(June):230–8.
41. Jin R, Chen W, Simpson TW. Comparative studies of metamodelling techniques under multiple modelling criteria. Struct Multidiscip Optim. 2001;23(1):1–13.
42. Liu H, Xu S, Wang X. Sampling strategies and metamodeling techniques for

engineering design: Comparison and application. Proc ASME Turbo Expo. 2016;2C-2016:1–12.
43. Madu CN. Simulation in manufacturing: A regression metamodel approach. Comput Ind Eng. 1990;18(3):381–9.
44. Wang GG, Shan S. Review of metamodeling techniques in support of engineering design optimization. J Mech Des Trans ASME. 2007;129(4):370–80.
45. Simpson TW, Mauery TM, Korte JJ, Mistree F. Comparison of response surface and kriging models for multidisciplinary design optimization. 7th AIAA/USAF/NASA/ISSMO Symp Multidiscip Anal Optim. 1998;(October):381–91.
46. Orr MJL. Introduction to Radial Basis Function Networks. University of Edinburgh. University of Edinburgh; 1996.
47. Roy R, Hinduja S, Teti R. Recent advances in engineering design optimisation: Challenges and future trends. CIRP Ann - Manuf Technol. 2008;57(2):697–715.
48. Loweth EL, De Boer GN, Toropov V V. Practical recommendations on the use of moving least squares metamodel building. Proc 13th Int Conf Civil, Struct Environ Eng Comput. 2011;(September).
49. Zhang J, Zhu H, Yang C, Li Y, Wei H. Multi-objective shape optimization of helico-axial multiphase pump impeller based on NSGA-II and ANN. Energy Convers Manag [Internet]. 2011;52(1):538–46. Available from: http://dx.doi.org/10.1016/j.enconman.2010.07.029
50. Nourbakhsh A, Safikhani H, Derakhshan S. The comparison of multi-objective particle swarm optimization and NSGA II algorithm: Applications in centrifugal pumps. Eng Optim. 2011;43(10):1095–113.
51. Sinclair J. Meta-modelling within the Design Optimisation of a Li-Ion Satellite Battery Structure. University of Leeds; 2020.
52. Barton RR, Meckesheimer M. Chapter 18 Metamodel-Based Simulation Optimization. Vol. 13, Handbooks in Operations Research and Management Science. Elsevier; 2006. p. 535–74.
53. Glynn PW. Stochastic Optimization. Encycl Actuar Sci. 2004;1–20.
54. Chang Y, Bouzarkouna Z, Devegowda D. Multi-objective optimization for rapid and robust optimal oilfield development under geological uncertainty. Comput Geosci [Internet]. 2015;19(4):933–50. Available from: http://dx.doi.org/10.1007/s10596-015-9507-6
55. Deb K, Goel T. Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence. In: Zitzler E, Thiele L, Deb K, Coello Coello CA, Corne D, editors. Evolutionary Multi-Criterion Optimization. Berlin, Heidelberg: Springer Berlin Heidelberg; 2001. p. 67–81.
56. Ghodratnama A, Jolai F, Tavakkoli-Moghaddam R. Solving a new multi-objective multi-route flexible flow line problem by multi-objective particle swarm optimization and NSGA-II. J Manuf Syst [Internet]. 2015;36:189–202. Available from: http://dx.doi.org/10.1016/j.jmsy.2014.06.009
57. Van der Velden A, Koch P. Isight Design Optimization Methodologies. ASM Handb. 2010;22B(1).
58. Pavlou DG. Chapter 1 - An Overview of the Finite Element Method. In: Pavlou DG, editor. Essentials of the Finite Element Method [Internet]. Academic Press; 2015. p. 1–18. Available from: https://www.sciencedirect.com/science/article/pii/B9780128023860000013
59. Zienkiewicz OC. The Finite Element Method: Its Basis and Fundamentals. The Finite Element Method: its Basis and Fundamentals. 2013. iii.
60. SIMULIA. iSight (2017) Documentation. Providence, RI, USA: Dassault Systèmes; 2017.
61. Schwer LE. An overview of the PTC 60/V&V 10: Guide for verification and validation in computational solid mechanics: Transmitted by L. E. Schwer, Chair PTC 60V&V 10. Eng Comput. 2007;23(4):245–52.
62. SIMULIA. ABAQUS (2017) Benchmark Manual. Providence, RI, USA: Dassault Systèmes; 2017.
63. Tiwari A, Hoyos PN, Hutabarat W, Turner C, Ince N, Gan XP, et al. Survey on the use of computational optimisation in UK engineering companies. CIRP J Manuf Sci Technol [Internet]. 2015;9:57–68. Available from: http://dx.doi.org/10.1016/j.cirpj.2015.01.003

64. Park SS, Chung WJ, Ahn JS, Kim ST, Lee SJ, Choi DB. Application of iSIGHT ® (OLH & RBF Modules) to Optimal Design of a Dynamical System with High Speed Spindle considering Thermal Behavior and Natural Frequency. 2021;
65. Asad T. Verification and Validation of Satellite Battery Structures. University of Leeds; 2021.
66. OPTIMEC. Tetrahedral Elements available in ABAQUS for structural analysis? [Internet]. 2021 [cited 2021 Apr 16]. Available from: http://optimec.ca/news/tetrahedral-elements-available-abaqus-structural-analysis-use/
67. Dobson J, Davidson A, Lomas S. User Manual & Design Description. Oxford: ABSL Space Products; 2019.
68. JMP® 15.2.1. Overview of Definitive Screening Design [Internet]. 2021 [cited 2021 Apr 1]. Available from: https://www.jmp.com/support/help/en/15.2/index.shtml#page/jmp/overview-of-definitive-screening-design.shtml%23
69. Xiao L, Lin DKJ, Bai F. Constructing definitive screening designs using conference matrices. J Qual Technol. 2012;44(1):2–8.
70. Ma L, Fong J, Lane B, Moylan S, Filliben J, Heckert A, et al. Using design of experiments in finite element modeling to identify critical variables for laser powder bed fusion. Proc - 26th Annu Int Solid Free Fabr Symp - An Addit Manuf Conf SFF 2015. 2020;219–28.
71. Hawkins DM. The Problem of Overfitting. Vol. 44, Journal of Chemical Information and Computer Sciences. 2004. p. 1–12.
72. Taramona Pérez M. Topology Optimisation of a Mass-Spring-Damper System for a Lithium-Ion Satellite Battery Structure. University of Leeds; 2021.
73. Yang G. A modified particle swarm optimizer algorithm. 2007 8th Int Conf Electron Meas Instruments, ICEMI. 2007;2675–9.
74. Awan F. Weighted Scalarization versus Compromise Solution in Multi-Objective Economic Dispatch for Microgrids. McGill University; 2016.
75. Toropov V, Bates SJ, Querin O. Generation of Extended Uniform Latin Hypercube Designs of Experiments. In: Proceedings of the ninth international conference on the application of artificial intelligence to civil, structural and environmental engineering. 2007.
76. Angelo GD, Palmieri F. GGA : A modified genetic algorithm with gradient-based local search for solving constrained optimization problems. Inf Sci (Ny) [Internet]. 2021;547(1):136–62. Available from: https://doi.org/10.1016/j.ins.2020.08.040

Appendices

Appendix A – MSD-Battery Detailed DOE

	Parameter Thickness (mm)		
Run	***Feet***	***Strut***	***Wall***
1	9.21	1.62	2
2	11.38	3.07	1.379
3	12	2.45	0.862
4	4.55	2.24	1.897
5	5.17	1	0.914
6	3	2.97	1.121
7	3.31	3.38	1.638
8	4.24	1.21	1.741
9	8.9	2.55	0.603
10	11.07	1.31	1.534
11	9.83	4	1.276
12	7.66	3.69	0.552
13	8.59	2.66	1.69
14	4.86	3.79	0.966
15	10.14	3.59	1.845
16	5.48	2.86	0.707
17	10.45	1.41	0.81
18	8.28	1.1	1.172
19	10.76	3.48	0.759
20	11.69	2.34	1.793
21	6.41	2.03	1.017
22	7.03	1.72	1.586
23	7.34	1.52	0.5
24	5.79	2.76	1.431
25	6.72	3.9	1.483
26	6.1	3.28	1.948
27	7.97	3.17	1.069
28	9.52	2.14	1.224
29	3.62	1.93	1.328
30	10.45	1.41	0.81

Appendix B – MSD-Battery Pareto Optimal Points

Feet mm	Strut mm	Wall mm	Mass (kg)	FNF (kHz)
5.008068	1.846781	0.884168	23.16364	0.57468
5.690645	1.85671	0.870521	23.19236	0.581421
7.776978	1.362339	1.069384	23.35577	0.614103
5.751925	1.477727	0.832669	23.12125	0.574617
8.428978	1.235631	2	23.93521	0.672746
7.614989	1.054108	1.293983	23.44182	0.627206
11.7067	3.124119	2	24.36095	0.701759
11.60188	3.163748	2	24.36095	0.701409
8.266452	2.217773	1.391298	23.69119	0.649161
7.022781	1.611206	0.882602	23.23625	0.592479
8.708687	1.006288	1.646742	23.70639	0.654755
7.571029	1.252876	0.5	22.98608	0.547662
8.045353	1.487798	1.400792	23.58704	0.642263
7.448349	1	1.506719	23.55409	0.638686
9.706041	1.01759	1.546352	23.70031	0.654533
8.031517	1	2	23.88252	0.668198
8.640866	1.4454	2	23.9746	0.675586
8.986876	1	1.649049	23.7217	0.656396
8.611268	1	2	23.91315	0.672146
11.18276	1.717349	2	24.14418	0.691423
10.31737	1.180936	1.579222	23.77426	0.660668
8.750605	1.261432	2	23.95564	0.674956
8.702088	1	1.309587	23.50188	0.63435
9.849928	1	1.665446	23.77728	0.661854
6.741521	1.387138	0.97434	23.24668	0.597733
5.684942	1	0.860694	23.07032	0.572959
9.391762	1	1.274901	23.5175	0.635471
8.560399	1	1.963717	23.8886	0.669974
3.93613	1	0.5	22.76133	0.515817
8.931835	2.112357	0.790598	23.34879	0.599039
12	4	2	24.49594	0.709515
9.669016	1.462661	1.536917	23.75193	0.65769
4.655432	1	1.182651	23.21015	0.58947
10.82701	1.385847	2	24.08137	0.687731
12	1.965537	1.837167	24.12159	0.687845
8.617668	1	0.5	23.00737	0.555076
9.633823	1	1.420151	23.61803	0.64637
7.55692	1	0.5	22.95136	0.546607
5.963306	1	0.5	22.86745	0.534325
3	1	0.559134	22.74814	0.511728
11.02565	1	2	24.04014	0.6868
8.28245	1	0.5	22.98967	0.552371
6.397904	1.318884	0.5	22.93329	0.53921
8.386982	2.123591	1.64236	23.83675	0.662537
12	4	1.899696	24.43524	0.703596
5.482663	1	0.5	22.84222	0.530388
10.3146	1.928695	2	24.12718	0.687879
3	1	0.5	22.71258	0.505674
4.848105	1	0.5	22.80898	0.524779
10.9442	1.663635	1.709376	23.94958	0.674009
3	1	0.5	22.71258	0.505674
8.417315	1	0.5	22.99679	0.553458
3	1	0.877135	22.93941	0.542593
3.38982	1	0.5	22.73286	0.509994

Feet mm	Strut mm	Wall mm	Mass (kg)	FNF (kHz)
3	1	0.5	22.71258	0.505674
9.097181	1	0.5	23.03269	0.558943
12	1.185021	0.5	23.20981	0.581823
6.1262	1.241096	0.5	22.90853	0.536798
3	1	0.5	22.71258	0.505674
7.89537	1	2	23.87533	0.667226
12	1.88931	2	24.20965	0.696822
3	1	0.5	22.71258	0.505674
11.59968	4	2	24.47539	0.707208
3	1	0.5	22.71258	0.505674
12	2.017598	2	24.22676	0.697437
12	1	0.5	23.18504	0.580536
3	1	0.5	22.71258	0.505674
8.696526	1	2	23.91765	0.672703
12	1.066316	1.562964	23.83726	0.666844
3	1	0.5	22.71258	0.505674
12	2.092175	2	24.23672	0.697798
12	1	0.5	23.18504	0.580536
11.74047	4	2	24.48261	0.708014
3	1	0.5	22.71258	0.505674
8.313713	1	2	23.89743	0.670156
12	1.847112	2	24.20403	0.696621
12	1	0.5	23.18504	0.580536
3	1	0.5	22.71258	0.505674
3	1	0.5	22.71258	0.505674
10.29654	1	2	24.00191	0.682518
12	1.712999	2	24.18617	0.695986
12	2.624086	1.976034	24.29371	0.69916
3.072073	1	0.5	22.71633	0.506482
12	2.006723	2	24.22531	0.697384
3	1	0.556188	22.74637	0.511427
3	1	0.5	22.71258	0.505674
9.890694	1	2	23.98058	0.680107

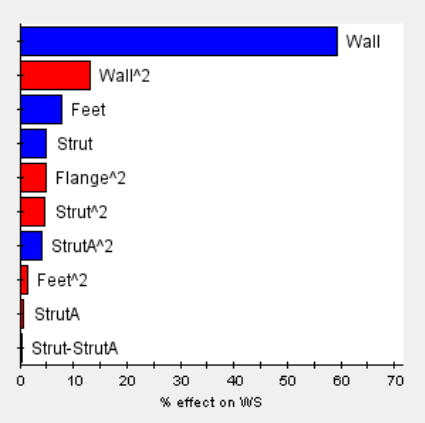
(Baseline values used for: Flange thickness = 1mm, Strut Angle = 84.5°, Base thickness = 6mm.)

Appendix C – Baseline NARMER Battery Optimisation

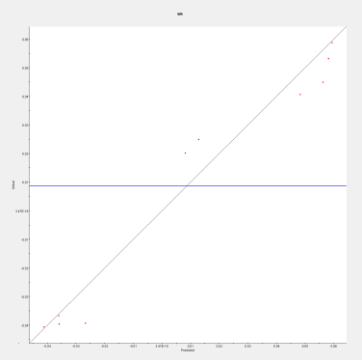
DSD:

	Design Parameter Level Setting				
Run	**Feet**	**Flange**	**Strut**	**Strut Angle**	**Wall**
1	0	1	1	1	1
2	0	-1	-1	-1	-1
3	1	0	-1	1	1
4	-1	0	1	-1	-1
5	1	-1	0	-1	1
6	-1	1	0	1	-1
7	1	1	-1	0	-1
8	-1	-1	1	0	1
9	1	1	1	-1	0
10	-1	-1	-1	1	0
11	1	-1	1	1	-1
12	-1	1	-1	-1	1
13	0	0	0	0	0

PCA Weighted Sum Graph and Error Analysis:



Error Analysis:



$R^2 = 0.95897$

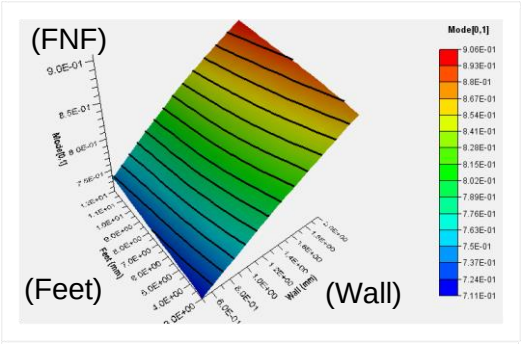
$e_{max} = 12.585\%$

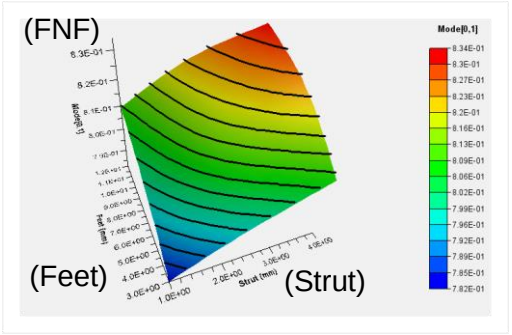
OLHD:

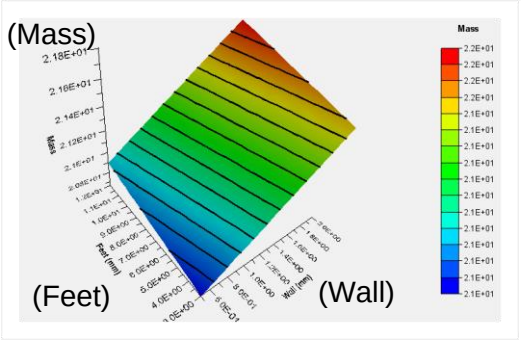
	Parameter Thickness (mm)		
Run	**Feet**	**Strut**	**Wall**
1	6.1	2.45	1.897
2	3	2.03	1.69
3	5.79	2.86	0.655
4	4.86	3.9	0.81
5	9.21	2.66	0.552
6	9.52	2.97	2
7	8.28	3.69	0.759
8	11.07	2.24	1.017
9	10.45	3.79	1.638
10	4.55	1.21	1.276
11	7.97	1.41	1.431
12	3.62	3.17	1.741
13	6.41	1.72	0.5
14	11.38	1.62	1.586
15	4.24	4	1.379

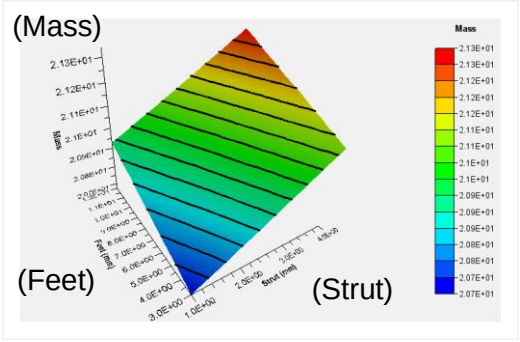
16	3.93	1.93	0.862
17	8.59	2.55	1.483
18	10.76	1.1	1.121
19	8.9	1.83	1.948
20	11.69	3.48	0.707
21	6.72	3.38	1.224
22	3.31	3.07	1.069
23	12	2.76	1.534
24	5.48	1.31	1.845
25	7.34	1	0.914
26	7.66	2.14	0.966
27	10.14	3.28	1.172
28	7.03	3.59	1.793
29	5.17	2.34	1.328
30	9.83	1.52	0.603

RBF Metamodels:









LOOCV:

	Parameter Thickness (mm)			Mass (kg)		1st Mode (kHz)	
LOOCV #	**Feet**	**Strut**	**Wall**	**Actual**	**Predicted**	**Actual**	**Predicted**
1	8.590	2.550	1.483	21.364	21.364	0.854	0.854
2	9.520	2.970	2.000	21.761	21.760	0.904	0.907
3	11.070	2.240	1.017	21.109	21.110	0.805	0.803
4	5.790	2.860	0.655	20.830	20.830	0.751	0.750
5	4.550	1.210	1.276	20.942	20.942	0.805	0.803
6	7.030	3.590	1.793	21.654	21.654	0.884	0.884
7	11.690	3.480	0.707	21.109	21.110	0.774	0.776
8	8.280	3.690	0.759	21.077	21.077	0.778	0.777
9	6.410	1.720	0.500	20.595	20.596	0.718	0.722
10	10.140	3.280	1.172	21.320	21.320	0.829	0.828
				Mass $R^2 = 1$		**FNF $R^2 = 0.99885$**	
				Mass $e_{max} = 0.114\%$		**FNF $e_{max} = 2.09\%$**	

Baseline Battery Pareto Optimal Points:

Feet (mm)	Strut (mm)	Wall (mm)	Mass (kg)	FNF (kHz)
12	3.45675	2	21.89591	0.915959
12	3.257615	2	21.86848	0.914517
7.690812	1	1.470964	21.12105	0.834478
8.196232	1.105136	1.214628	20.99498	0.809192
5.972248	1	1.794468	21.26748	0.860669
12	1.663622	2	21.65065	0.903757
3	1.446307	0.5	20.46282	0.705191
12	1.688533	2	21.65402	0.903943
5.797319	1	1.639703	21.16908	0.845044
7.067455	1	1.452198	21.09202	0.83061
5.70103	1	0.5	20.47748	0.708839
6.803631	1	1.389047	21.04642	0.82308
12	2.435901	2	21.75572	0.909052
4.689508	1	1.463534	21.03114	0.822819
7.756744	1	2	21.44167	0.886153
3.01979	1.183437	0.676526	20.53388	0.724973
12	1.547356	2	21.6349	0.902873
3.061239	1	1.098465	20.76465	0.77484
3	1.51346	0.5	20.47204	0.705909
7.30761	1	0.5	20.52289	0.713571
5.234632	1	0.68972	20.5791	0.732448
12	1.973545	2	21.6927	0.905986
4.762832	1	2	21.35721	0.875685
5.624408	1	2	21.38164	0.878963
12	3.042898	2	21.83894	0.913018
5.736822	1	1.131449	20.86051	0.790264
5.735582	1	0.5	20.47845	0.70894
5.985626	1	2	21.39185	0.880274
12	1.271555	2	21.5976	0.900663
8.066505	1	2	21.45037	0.887104
3	1	1.036277	20.72541	0.767217
3	1	0.5	20.40179	0.700498
8.132557	1	1.996563	21.45015	0.886979
6.497324	1	2	21.40627	0.882063
12	3.121205	2	21.84971	0.913557
9.299039	1	2	21.48501	0.890767
6.326827	1.657575	1.121306	20.95976	0.799814
5.405974	1.16678	1.418161	21.0467	0.823155
6.60376	1.156534	1.042023	20.85224	0.784513
6.740723	1	2	21.41312	0.882887
12	3.737984	2	21.93473	0.91805
3.091626	1	2	21.30976	0.868854
3	1.3101	0.5	20.44415	0.703747
5.680806	1	1.02855	20.79673	0.777558
5.038626	1	2	21.36504	0.876756
12	1	2	21.56089	0.898333
9.792815	1	2	21.4989	0.892201
3	1.505078	0.5	20.47089	0.705819
9.749283	1	2	21.49767	0.892076
6.346603	1	2	21.40202	0.881544
12	1	2	21.56089	0.898333
6.630959	1	2	21.41003	0.882517
4.592503	1.607147	1.414144	21.08079	0.823916
8.329058	1	2	21.45775	0.887898
6.937204	1	2	21.41864	0.883539
12	1	2	21.56089	0.898333
8.189271	1	2	21.45382	0.887476
6.524614	1	2	21.40704	0.882156
4.735132	1.41113	1.189275	20.92254	0.798189
5.264327	1	2	21.37144	0.877618
12	2.253193	2	21.73078	0.907867
10.78336	1.796848	2	21.63455	0.901141
12	1	2	21.56089	0.898333
11.22652	1	2	21.5392	0.896251
3.222593	1.391262	1.124862	20.83835	0.78312
3	1	0.869434	20.62474	0.746999
3.587974	1.429131	0.5	20.47677	0.706932
7.223312	1	2	21.42669	0.88447
6.559895	1	2	21.40803	0.882276
3	1.151149	2	21.32772	0.869675
12	1	2	21.56089	0.898333
9.945386	1	2	21.50319	0.892641
12	1	2	21.56089	0.898333
10.43681	1	2	21.51701	0.894045

Feet (mm)	Strut (mm)	Wall (mm)	Mass (kg)	FNF (kHz)
12	1	2	21.56089	0.898333
12	1	2	21.56089	0.898333
9.087641	1	2	21.47907	0.890148
3.72645	1.768456	1.328571	21.0268	0.812119
3	1.444049	1.441732	21.03018	0.816939
4.553368	1.409061	0.5	20.50084	0.709631
3	1	2	21.30717	0.868468
5.365618	1.049894	1.684355	21.19051	0.848244
8.58419	1	2	21.46491	0.888661
12	2.876434	2	21.81608	0.9119
3	1.059367	2	21.31524	0.868947
12	1	2	21.56089	0.898333
9.026179	1	2	21.47734	0.889968
3	1.114801	0.5	20.41745	0.701696
12	1	2	21.56089	0.898333
7.833504	1	2	21.44383	0.88639
12	1	2	21.56089	0.898333
12	1	2	21.56089	0.898333
11.01658	1	2	21.5333	0.895672
3	1	2	21.30717	0.868468
7.234121	1	2	21.42699	0.884505
12	2.66543	2	21.78714	0.910526
12	1	2	21.56089	0.898333
7.470234	1	2	21.43362	0.885257
3.193732	1	1.987288	21.30496	0.868037
3	1	0.736344	20.54442	0.730429
5.824176	1.275995	0.5	20.51825	0.71183
12	1	2	21.56089	0.898333
3	1.089892	2	21.31939	0.869191
9.893084	1	2	21.50172	0.892491
12	1.150057	2	21.58117	0.899639
3	1	2	21.30717	0.868468
12	1	2	21.56089	0.898333
11.67417	1	2	21.55176	0.897466
3	1	0.5	20.40179	0.700498
3	1	2	21.30717	0.868468
12	4	2	21.97097	0.920021
9.75233	1	2	21.49776	0.892084
12	1.612282	2	21.64369	0.90337
12	1	2	21.56089	0.898333
8.131506	1	2	21.4522	0.887301
12	1	2	21.56089	0.898333
3	1	2	21.30717	0.868468
9.442273	1	2	21.48904	0.891185
12	3.362397	2	21.88291	0.91527
12	1	2	21.56089	0.898333
12	1	2	21.56089	0.898333

Appendix D – Meeting Log for Mathias D'souza

Date	(Host) Summary of Discussion		(Performed by) Objectives for Next Meeting	
21/10/20	MT	Introduction to project and deliverables, discussion about role responsibilities.	MD	Read about project and roles. Select roles
27/10/20	MT	Research questions to ask ABSL based on given resources.	MD	Prepare questions to ask ABSL representatives.
28/10/20	ABSL	Preparation for CPP. Introductory session setting the preparatory work for project goals.	MD	Revise CPP Guidelines
04/11/20	GD	Reoccurring meeting request for supervisory project meetings.	MD	Make CPP Template for team
09/11/20	MT	Agreement on management structure and working hours. Agreement on fixed time and date for weekly recurring meetings, as a minimum. Tutorial on Notion for use as a meeting log platform. Individual task allocation for CPP document.	MD	Create management structure.
12/11/20	TA	Communicate individual work completed thus far for the CPP draft. Create a logical optimisation loop. Decide on further objectives and tasks.	MD	Discuss shape optimisation software options w/ Greg. Breakdown work into ordered structure.
13/11/20	AZ	Discussion of MSD system and implementation. Discussion of DoE process, regarding shape/topology optimisation. Decision regarding shape topology optimisation software Sharing Notion & OneDrive folders w/ Greg.	MD	Select software for shape optimisation.
16/11/20	MD	Finalise the CPP draft. Send CPP draft to Greg. Set deadlines for CPP presentation completion.	MD	Create CPP work plan.

19/11/20	MT	Discuss topology optimisation work allocation. Discuss thermal analysis considerations w/ Greg.	MD	Finalise CPP.
23/11/20	TA	Setup presentation and script creation.	MD	Email CPP and presentation to Andrew Shires.
26/11/20	AZ	Brief overview of individual objectives, both completed and outstanding. Discuss CPP submission date. Update Gantt chart with finalised project tasks.	MD	Remove all animations/transitions from presentation. Finalise script length & content for a mock presentation. Continue progress with individual literature review.
29/11/20	MT	Discuss further presentation changes.	MD	Review each other's individual scripts, checking for overlap. Create PowerPoint without script for sharing externally.
30/11/20	TA	Review script changes. Place scripts into PowerPoint. Practice presentation sharing within Microsoft Teams. Practice presentation in general.	MD	Script formatting.
01/12/20	MD	Final presentation practice run, recorded for later review. Final submission of presentation and CPP document.	MD	Submit relevant CPP documents.
07/12/20	AZ	Decide work to be completed over Christmas. Shorten CPP presentation for upcoming AIAA presentation.	MD	Preliminary software (iSight) setup. DoE and PCA completion. Condense presentation for next meeting. Reassess Gantt chart progress. Continue relevant literature research.
10/12/20	MT	Discuss how to condense CPP presentation into a 5-min. presentation, with additional questions. Discuss preparation of new scripts, w/ a 90-sec. maximum talk time. Set up a meeting in Microsoft Teams, 15 mins before the IAB meeting for preparatory purposes.	MD	Condense each section of the CPP document
11/12/20	GD	Meeting for the University of Leeds Aeronautical and Aerospace Engineering Industrial Advisory Board (IAB).	MD	Same as before.
14/12/20	MD		MD	Create individual slides for ABSL presentation.

		Discuss updates on work to be completed before Christmas break. Preparation for upcoming ABSL meeting.		Produce questions to ask ABSL before holiday break.
16/12/20	ABSL	Discussion on the extent of battery model simplification. Discussion on topology optimisation, and how to improve the loading. Discussion of the process for designing the MSD system; what does it do exactly?	MD	Research iSight integration.
21/12/20	AZ	Discussion of progress on V&V of simplified model. Discussion of individual research progress; updates on literature review. Acknowledgement of issues regarding nature of initial candidate MSD design.	MD	Implement shape optimisation example in iSight.
28/12/20	TA	Update on ongoing processes mentioned in last meeting. Discuss previous year's work in terms of V&V; how does this relate to our work? Creating of future tasks and deadlines.	MD	Complete iSight validation study and PCA
04/01/21	MT	Check progress on objectives set in last meeting: Battery structural validation SolidWorks MSD model iSight validation study	MD	Plan PCA to start on adapted battery-MSD model.
19/01/21	MD	Progress update post-exam period.	MD	Work on PCA.
25/01/21	AZ	Discuss usage of HPC suite. Find out how to obtain mass properties from model (Tauseef). Discuss thermal analysis implementation and its impact. Discuss optimisation problem formulation. Ask Greg about previous group's individual reports pertaining to optimisation.	MD	Produce individual components for upcoming poster.
27/01/21	GD	Meeting with Greg to give update on Christmas work from each team member.	MD	Continue objectives from previous meeting.

31/01/21	TA	Discuss current progress on the poster. Discuss what is needed to finalise poster. Set deadlines for poster scripts and audio recordings.	MD	Create audio recordings to be embedded into poster.
01/02/21	MD	Discussion around audio for poster and request for feedback from Greg.	MD	Send poster to Greg for feedback.
08/02/21	AZ	Check updated battery modal mass. Discussion around mass budget for new MSD structure. Selection of damping and stiffness coefficients in Abaqus (units?) Combination of new MSD with battery in Abaqus for meshing.	MD	Parameterisation of CAD model
15/02/21	MT	Update on progress regarding MSD implementation in Abaqus. Discuss creation of PowerPoint presentation to show ABSL the progress to date.	MD	Create individual sections for presentation
17/02/21	TA	Final touches on ABSL presentation. Sharing and discussion of questions to take to ABSL.	MD	Finalise presentation scripts
18/02/21	GD, ABSL	Catch-up meeting with ABSL and Greg; discussion revolving around pre-submitted questions for ABSL.	MD	Make changes suggested by ABSL in the design approach
22/02/21	MD	General progress updates: Modal mass changes Optimal MSD coefficients Final baseline model	MD	Conduct FEA on MSD concepts
25/02/21	MT	Discussion around outstanding project-critical tasks.	MD	Finalise FEA on MSD concepts
27/02/21	MT	Continuation of discussions from previous meeting.	MD	Same as previous.
02/03/21	AZ	Discuss analysis of Excel data for conventional MSD case; is there a link across GA and FMR? Evaluation of design feasibility; remodelling to determine new model effectiveness HPC suite access, or lack thereof (Anthony)	MD	Help with vibrational analysis.
04/03/21	MD	Check mass participation graphs (Tauseef) Assessment of findings from random response analysis. Determination of relevant field outputs in Abaqus.	MD	Start finalising the final concept design
06/03/21	MD	Check conclusions from MSD studies. Confirmation of final candidate model.	MD	.

		Collect baseline model from V&V. Assess next steps for project.		Finish presentation on MSD response.
15/03/21	MT	Update regarding latest thermal analysis, together w/ views from ABSL's response. Team input on latest results from thermal analysis. Presentation of changes made to the baseline battery. Discussion about next steps and short-term deadlines.	MD	Creation of updated baseline model with reduced elements for HPC. Finalise results from optimisation.
24/03/21	MT	Briefly present each skeleton for each individual report. Prepare for meeting with Greg at 11am.	MD	Ask Greg whether each other's individual contributions can be cross-referenced. Discuss availability during Easter break.
29/03/21	AZ	Assess progress on Chapter 1 of the team report. Set tasks/timeline for team report progression. Discussion revolving around any critical outstanding tasks.	MD	Obtain shape-optimised result for MSD-Structure
01/04/21	TA	Discussion regarding progress .	ALL	Finalise literature review section for the group report
05/04/21	MD	Progress update from last meeting's set objectives. Updates on results obtained from topology optimisation.	MD	Integration of methodology sections into group report.
08/04/21	MT	Review existing methodology section in group report. Status update on expected completion date. Review final topology optimisation results. Review final design interpretation.	MD	Continue progress on individual reports and team report contributions.
12/04/21	AZ	Check progress of team report, namely the methodology section. Routine update on individual reports. Discussion around the creation of final model to be presented.	MD	Add methodology and results sections to team report template.
15/04/21	TA	Review of team report progress. Outline any pressing changes to be made. Issue short-term deadlines for the upcoming days.	MD	Finalise results section of group report
16/04/21	MD	Checking changes to group report (figures, methods, removal of literature review).	MD	Continued insertion of corresponding sections to team report template.
17/04/21	MT	Check progress on group report. Ensuring all team members acted on comments made on shared team report.	MD	Complete a final draft of the team report

18/04/21	AZ	Finalising group report. Ensuring individual reports corroborate with each other's work.	MD	Continue finalising team report draft
19/04/21	TA	Review changes made from previous meeting.	MD	Submit team report by next week