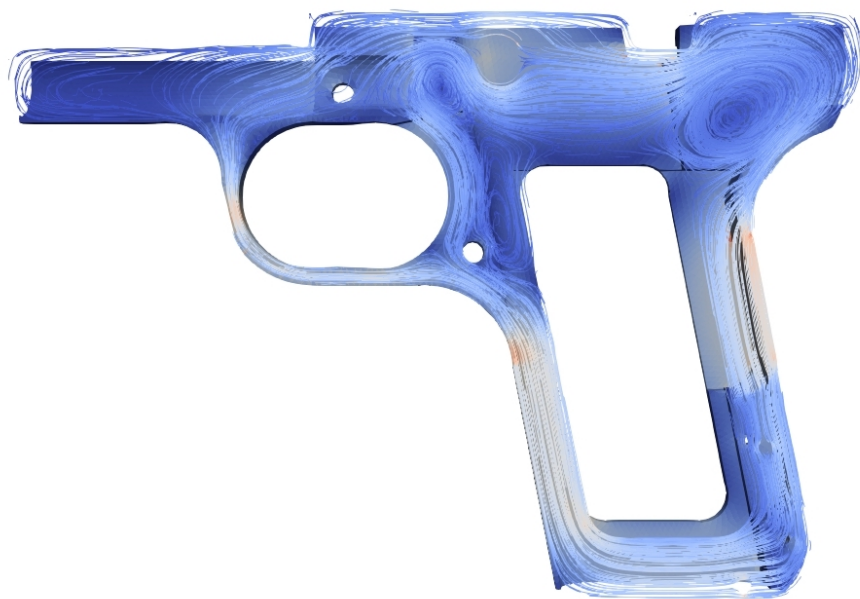# Characterisation and Classification of Hidden Conducting Security Threats using Magnetic Polarizability Tensors

Ben Alexander Wilson

Submitted to Swansea University in fulfilment of the
requirements for the Degree of
Doctor of Philosophy

20 January 2022

Abstract

The early detection of terrorist threat objects, such as guns and knives, through improved metal detection, has the potential to reduce the number of attacks and improve public safety and security. Walk through metal detectors (WTMDs) are commonly deployed for security screening purposes in applications where these attacks are of particular concern such as in airports, transport hubs, government buildings and at concerts. However, there is scope to improve the identification of an object's shape and its material properties. Using current techniques there is often the requirement for any metallic objects to be inspected or scanned separately before a patron may be determined to pose no threat, making the process slow. This can often lead to build ups of large queues of unscreened people waiting to be screened which becomes another security threat in itself. To improve the current method, there is considerable potential to use the fields applied and measured by a metal detector since, hidden within the field perturbation, is object characterisation information. The magnetic polarizability tensor (MPT) offers an economical characterisation of metallic objects and its spectral signature provides additional object characterisation information. The MPT spectral signature can be determined from measurements of the induced voltage over a range of frequencies for a hidden object. With classification in mind, it can also be computed in advance for different threat and non-threat objects, producing a dataset of these objects from which a machine learning (ML) classifier can be trained. There is also potential to generate this dataset synthetically, via the application of a method based on finite elements (FE). This concept of training an ML classifier trained on a synthetic dataset of MPT based characterisations is at the heart of this work.
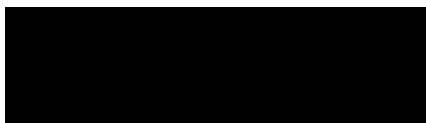
In this thesis, details for the production and use of a first of its kind synthetic dataset of realistic object characterisations are presented. To achieve this, first a review of recent developments of MPT object characterisations is provided, motivating the use of MPT spectral signatures. A problem specific, $H(\text{curl})$ based, hp-finite element discretisation is presented, which allows for the development of a reduced order model (ROM), using a projection based proper orthogonal decomposition (PODP), that benefits from a-posteriori error estimates. This allows for the rapid production of MPT spectral signatures the accuracy of which is guaranteed. This methodology is then implemented in Python, using the `NGSolve` finite element package, where other problem specific efficiencies are also included along with a series of additional outputs of interest, this software is then packaged and released as the open source `MPT-Calculator`. This methodology and software are then extensively tested by application to a series of illustrative examples. Using this software, MPT spectral signatures are then produced for a series of realistic

threat and non-threat objects, creating the first of its kind synthetic dataset, which is also released as the open source `MPT-Library` dataset. Lastly, a series of ML classifiers are documented and applied to several supervised classification problems using this new synthetic dataset. A series of challenging numerical examples are included to demonstrate the success of the proposed methodology.

Declarations

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.
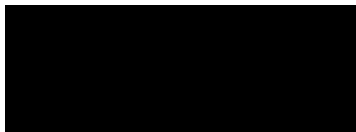
Signed: ███████████████   Date: 20/01/2022

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.
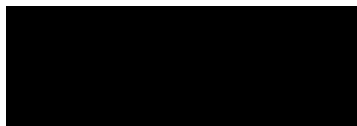
Signed: ███████████████   Date: 20/01/2022

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed: ███████████████   Date: 20/01/2022

The University's ethical procedures have been followed and, where appropriate, that ethical approval has been granted.

Signed: ███████████████   Date: 20/01/2022

Acknowledgements

Firstly, I would like to thank my supervisor Prof. Ledger for his help and guidance throughout the past three years, without this help I would not have known where to start, never mind finish. His professionalism and passion for the subject has driven and motivated me to rise to the challenge of a PhD research project. Further to this I would like to thank those involved with the Reducing Threat to Public Safety project for the many useful discussions at the research meetings, with a special mention of Prof. Lionheart whose discussions have been particularly informative. Additionally, I would like to thank the Zienkiewicz Centre for Computational Engineering and College of Engineering as a whole for the support and opportunities they have provided me with over the last 3 years. I would also like to acknowledge the support the Engineering and Physical Sciences Research Council have provided via the DTP studentship with the project reference number 2129099 for which I am grateful.

A PhD, is an individual professional endeavour, however, it is impossible to complete without the help and support of friends and family. The adversities faced, both professionally and personally, are simply insurmountable without the help of those around you. To all of the friends who have supported me throughout my PhD, thank you. In particular: the colleagues of room A130, for the many coffee breaks that were both distracting yet informative; the boys, for always being on the other end of the phone, no matter where we are. Lastly, yet most importantly, I would like to thank my family for always being there for me, in particular my parents, for supporting me throughout my education, from helping me with homework as a child up to now, helping proof papers and even this thesis. I could not have done it without you.

Contents

III   Object Characterisation   79

6   Decomposing the Inverse Problem to one of Object Location and Object Classification   81

7   Real World Object Characterisation Dataset   89

IV   Object Classification   119

8   Probabilistic and Non-Probabilistic Machine Learning Classifiers   121

List of Figures

List of Tables

Table of symbols

Symbol	Meaning	Unit
α, β, γ	Euler angles	Radians
α	Object size B	m
B	Unit sized object placed at the origin	-
B_α	Physical object	m^3
B_α^c	Unbounded region of free space outside the physical object	m^3
∂B	Boundary of B	-
Ω	Computational domain	-
$\boldsymbol{x}$	Position vector	m
$\boldsymbol{z}$	Object location vector	m
$\boldsymbol{n}$	Unit normal vector	-
$\boldsymbol{\xi}$	Position vector for transmission problem	-
$\boldsymbol{e}_j$	Orthonormal unit vector	-
i	Complex unit $\sqrt{-1}$	-
σ	Electric conductivity	S/m
σ_*	Conductivity of the object	S/m
ω	Angular frequency	rad/s
ϵ	Permitivity	F/m
μ	Magnetic permeability	H/m
μ_0	Magnetic permeability of a vacuum	H/m
μ_*	Magnetic permeability of the object	H/m
μ_r	Relative magnetic permeability of the object	-
ν	$\alpha^2 \mu_0 \sigma_* \omega$	HSrad/s
$\boldsymbol{E}$	Electric field intensity vector	V/m
$\boldsymbol{E}_0$	Electric background field	V/m
$\boldsymbol{E}_\alpha$	Electric interaction field	V/m
$\boldsymbol{H}$	Magnetic field intensity vector	A/m
$\boldsymbol{H}_0$	Magnetic background field	A/m
$\boldsymbol{H}_\alpha$	Magnetic interaction field	A/m
$\boldsymbol{J_0}$	External current source	A/m^2

$G(\boldsymbol{x}, \boldsymbol{z})$	Laplace free space Green's function	m^{-1}
$\boldsymbol{D}_x^2 G(\boldsymbol{x}, \boldsymbol{z})$	Hessian of G	m^{-3}
$\mathcal{M}$	Magnetic polarizability tensor (MPT)	m^3
$\mathcal{N}^0$	Magnetostatic response of the MPT	m^3
$\tilde{\mathcal{R}}$	Real part of the MPT	m^3
$\mathcal{I}$	Imaginary part of the MPT	m^3
$\boldsymbol{\theta}_i^{(0)}, \boldsymbol{\theta}_i^{(1)}$	Solutions to the transmission problems	-
ε	Regularisation parameter	-
h	Mesh size	m
p	Polynomial degree	-
Δ	a-posteriori error estimate	-
$\lambda(\cdot)$	Eigenvalue	-
δ_{ij}	Kronecker delta	-
$\mathbf{R}$	Orthonormal rotation/reflection matrix	-
$\mathbf{Q}$	Matrix of eigenvectors	-
$\boldsymbol{\Lambda}$	Matrix of eigenvalues	-
a	Bilinear form of the weak from presented in (3.3)	-
r	Linear form of the weak from presented in (3.3)	-
$\mathbf{A}$	System matrix of the weak from presented in (3.3)	-
$\mathbf{A}^M$	Reduced system matrix	-
$\boldsymbol{r}$	Right hand side of discretised (3.3)	-
$\boldsymbol{r}^M$	Right hand side of reduced system	-
N_d	Number of degrees of freedom	-
$\boldsymbol{N}^{(k)}(\boldsymbol{\xi})$	Basis functions	-
$\mathbf{N}(\boldsymbol{\xi})$	Matrix constructed with $\mathbf{N}^{(k)}(\boldsymbol{\xi})$ as columns	-
$\mathbf{q}$	Solution of discretised system (3.3)	-
$\mathbf{D}$	Matrix constructed with $\mathbf{q}(\omega)$ as columns	-
$\mathbf{U}$	Left singular vectors	-
$\boldsymbol{\Sigma}$	Singular values	-
$\mathbf{V}$	Right singular vectors	-
TOL	Tolerance of the truncated singular value decomposition	-
$NDOF$	Number of degreees of freedom	-
$I(\cdot)$	Principal tensor invariants	-
$J(\cdot)$	Deviatoric tensor invariants	-
$\mathcal{Z}$	Commutator of $\tilde{\mathcal{R}}$ and $\mathcal{I}$	-

C_k	Class k	-
$G^{(k)}$	Number of distinct geometries in class k	-
$V^{(k)}$	Number of variations for each geometry in class k	-
$P^{(k)}$	Number of different samples in class k	-
x	Feature vector	-
$p(C_k|\mathbf{x})$	Probability of C_k given $\mathbf{x}$	-
$\gamma_k(\mathbf{x})$	Approximate posterior probability of class C_k given $\mathbf{x}$	-
κ	Cohen's kappa score	-

Part I

Preliminaries

Chapter 1

Introduction

1.1 Metal Detectors

Metal detectors have many uses from hobbyist treasure hunting and assisting with archeological digs [29] to more safety critical applications such as security screening [87, 82] and humanitarian demining [7].

Security screening is required in many situations with obvious applications in airports and other transport hubs, however, these systems are also deployed in scenarios such as stadiums, concert halls and some museums. Due to the number of people that need to be processed in these applications and the safety critical nature of the work, there is a requirement for speed without undermining the accuracy of the process. With increased gun and knife crime in many countries (e.g. U.K. knife crime has seen a significant increase in the last 8-9 years [38]), the early detection of such weapons has the potential to reduce the number of related attacks and improve safety and security.

Similarly, the identification of hidden anti-personal landmines and unexploded ordnance (UXO) in areas of former conflict is vital, with over 7,000 recorded casualties due to landmines in 2020 alone, 80% of which were civilians and of these civilian casualties half were children [126]. At current rates, it is estimated that it would take 1,100 years to remove all of the world's active anti-personnel landmines, this is provided no new mines are laid [90], sadly this is not the case, with new landmines being deployed in Afghanistan, Colombia, India, Myanmar, Nigeria, and Pakistan in 2020 [126]. One other consideration, is the remoteness of the affected areas [48], this presents significant difficulties in the requirements for the system to be used, due to lack of high powered computing available in such remote areas a solution with low computational overhead is required.

Metal detection is a low frequency electromagnetic technique that operates typically between 0.5-500kHz [9]. A metal detector works by passing a low frequency electric current through a coil, which induces a magnetic field. When this magnetic field interacts with a highly conducting body, eddy currents are generated. The generation of eddy currents perturbs the magnetic field. A second set of coils senses the perturbation as a voltage. Many commercial metal detectors sense metal objects based on simple thresholding and give an audible signal when the threshold is above a certain level. While metal detectors

do detect metal objects, they are often not able to distinguish between different shapes, different materials and small objects close to coils and larger objects further away.

Magnetic induction tomography (MIT) uses a similar physical process to metal detection. It is based on placing a series of coils around a conducting object and takes measurements of induced voltages in the coils generated by magnetic induction to recover images of conductivity (and permeabilty). Typical operating frequencies are 5-500kHz [80]. A review of the technique is provided by Griffiths [46]. Images are often produced by voxelting the imaging region and minimising a cost functional to update conductivity values in the voxels. It has been used for industrial process modelling (e.g. molten metal [81, 119]) and has been proposed as a novel medical imaging modality (e.g. for strokes in the brain [142]). Related to MIT is electrical impedance tomography (EIT), which relies on low-frequency currents and measures the voltages induced by the presence of low conducting objects [59, 106]. While EIT still operates at low-frequncies, the physical process is no longer magnetic induction but instead electrical in nature and is commonly used for medical imaging [23].

Ground penetrating radar (GPR) is another imaging based technique, it is based on high frequency electromagnetic pulses typically in the range or 10-5000MHz [67]. GPR like metal detection is used for locating objects underground. GPR provides images of the sub-surface unlike in low-frequency metal detection and MIT, GPR works with electromagnetic fields that propagate as waves. When these waves interact with buried objects they may be partially absorbed, reflected and scattered. The time of travel and energy of the reflected waves is used to build up an image of the subsurface. One advantage of GPR is its ability to go beyond detection of metal alone, and can detect non-permeable, non-conducting objects. Proposals for a method combining GPR and classical metal detection have also been proposed [88].

Metal detection is a low-cost modality, which provides a fast and efficient method of detecting metal objects. However, there is scope to better understand the information contained in signals measured by metal detectors and how these can be utilised to better identify the location of hidden metallic objects and information about their shape, size and material properties. Thereby solving the metal detection inverse problem.

1.1.1 Approaches to Solving the Metal Detection Inverse Problem

As discussed previously, the simplest approach to metal detection is based on thresholding [136]. The signals measured by a metal detector are in the form of voltages in a set of measurement coils. In principal, one could use knowledge of the currents induced in the exciting fields and the voltages measured in the measurement coils as a basis for solving an electromagnetic inverse problem akin to what is done in MIT and the related EIT. If sufficiently many exciting coils and measurement coils are available, a cost functional approach can be prescribed for the difference between the measured voltages and the voltages obtained by solving the eddy current approximation to Maxwell's equations with a given conductivity distribution. Given a voxelated grid, and a numerical approximation to the partial differential equations, the functional then needs to be minimised to find the

optimal set of conductivity values for the voxels.

Unfortunately, the solution for conductivity distributions in MIT (and EIT) is severely ill-posed. An ill-posed problem differs from a well-posed problem [52], since: A solution may not exist; The solution may not be unique; The solution may not change continuously with the data. The small number of measurements also makes the problem very difficult. This means that the discrete non-linear system that needs to be solved is ill-conditioned. For this reason regularisation needs to be added to the functional to be minimised, Tikhonov regularisation is the most widely used [41]. Adding too little regularisation means the ill-conditioning dominates. Adding too much means that any interesting features are smoothed out and any useful information in the image is lost. There are automated methods for choosing the amount of regularisation to be added which include the discrepancy principal [51, 50] and L-curve methods [53, 25].

Common approximate approaches for solving the set of partial differential equations include either finite element methods or finite difference methods [18, 78, 124]. In the case of the finite element method, non-standard vectoral edge elements must be applied to approximate the eddy current approximation of the Maxwell system as the use of a nodal finite element approximation will lead to the generation of spurious modes and the continuity of fields at material interface conditions would be incorrectly imposed by a nodal scheme [75, 103]. Such finite elements are now widely used in electromagnetics and high order versions are available e.g. [77]. Finite difference approximations typically use rectangular grids and some authors have adopted the staggered grids used in finite difference time domain (FD-TD) [124] to the frequency domain.

Some examples of this approach for imaging of conductivities relevant to both MIT and metal detection are [34, 121]. As well as imaging conductivity, some approaches image both conductivity and permeability [120].

Additionally, instead of using a voxelated grid, one could use a level-set approach to represent the hidden conducting objects, this approach attempts to reconstruct the interface between regions of differing conductivity [118]. To overcome the issues associated with the ill-posed nature of the inverse conductivity problem and the small number of measurements that can be made in metal detection, an alternative approach, which attempts to characterise hidden conducting objects by a small number of parameters using a magnetic polarizability tensor has gained popularity. This may have considerable advantages for example in airport security scanning [82] and landmine detection [96].

1.2 Characterising Objects by Magnetic Polarizability Tensors

In contrast to building a 'picture' of the domain, significant work has been done in recent years to characterise hidden conducting objects using a small number of parameter through a magnetic polarizability tensor (MPT). The MPT is a complex symmetric rank 2 tensor, which has 6 independent coefficients. Its coefficients are a function of the exciting frequency, the object's size, its shape as well as its conductivity and permeability. Initially

the MPT was used based on engineering intuition [86, 94]. But, following the derivation of the leading order term in an asymptotic expansion of the perturbed magnetic field by Ammari, Chen, Chen, Garnier and Volkov [13], Ledger and Lionheart showed that the result reduces to an object characterisation using a rank 2 complex symmetric tensor, thus, providing the first explicit formula for the MPT [69]. In the subsequent works, Ledger and Lionheart derived alternative formulations, which although equivalent for exact continuous fields, offer advantages for their explicit computation and for investigating their properties [70, 71, 72, 73]. In [72] Ledger and Lionheart have extended Ammari et al's [13] to provide a complete asymptotic expansion of the perturbed magnetic field and introduce the concept of generalised MPTs (GMPTs). The rank 2 MPTs have been validated against exact solutions for a conducting sphere and against measurements of MPTs for different objects [70, 71].

Also, Ledger and Lionheart have proven the way in which the tensor coefficients vary with the exciting frequency [73] and explained the improved object characterisation this offers. The connection of the MPT to simpler rank 2 Póyla-Szegö tensor is considered in [70, 73]. In [69] Ledger and Lionheart explain how objects with rotational or reflectional symmetries have a reduced number of independent coefficients and describe how these can be determined from the object symmetries. In addition to these theoretical advancements, apparatus have been developed to measure MPTs in a lab based setting [96] and are becoming ever more accurate. As well as these lab based settings signals from commercial metal detectors have also been used to recover MPT coefficients [86, 87, 82, 83]. In contrast to determining conductivity distributions in MIT, determining the MPT coefficients is easier and can be achieved by simple linear least squares, by solving an over-determined linear system of equations, which is much better conditioned. Minimal regularisation may be added in the case of noisy measurements.

From previous studies of the simpler Póyla-Szegö tensor characterisation of an object for a fixed conductivity contrast in electrical impedance tomography (EIT), it is known that shape and material contrast information cannot be separated [15]. In [15] it is also shown, for a two dimensional object, that a unique equivalent ellipse can be constructed that has the same Póyla-Szegö tensor if the object's conductivity contrast is known. For a three-dimensional object, a numerical procedure has been proposed for computing an equivalent ellipsoid that has the same Póyla-Szegö tensor, although it is not clear if the resulting ellipsoid is unique [64]. Given this, it is generally accepted that using an MPT characterisation at a single frequency also only provides limited information. In addition, for electro-sensing, where it has been postulated that electric fish characterise objects by Pólya-Szegö tensors [65, 63], spectral information, with frequency dependent contrasts, have been used for the successful classifications of objects [11]. There has been success in small scale measured MPT spectral signatures for object classification [14, 74, 82, 87, 62, 101] and a theoretical study of the MPT spectral signature [73] has been undertaken.

One approach for the automated computation of the MPT spectral signature is to post-process finite element method (FEM) solutions to eddy current problems obtained using commercial packages (e.g. with ANSYS as in [100]), however, improved accuracy, and a better understanding, can be gained by using the available explicit expressions for MPT

6

coefficients, which rely on computing finite element (FE) approximations to a transmission problem [69, 71, 73]. In common with other approaches to solving metal detection problems and inverse problems in MIT, the vectoral nature of the transmission problems dictates that vectoral edge based finite elements be used. Appropriate formulations for solving the transmission problem needed for computing the MPT have already been presented in the context of solving eddy current problems [77].

A further alternative would be to use the boundary element method (BEM) to discretise the transmission problem, which only requires discretisation of the conductor's surface for a homogenous conductor and hence fewer degrees of freedom. However, unlike the sparse matrices in FEM, BEM results in fully populated matrices, and, for general inhomogeneous objects, requires discretisation of the conductor's volume and coupling with FEM. Appropriate types of BEM for the transmission problem are described in [56, 57].

1.3 Classification and Machine Learning

Machine learning (ML) has been around for a long time, with the concept of a logic based neural network being postulated about as far back as 1944 by McCulloch and Pitts [104]. More classical techniques, date back far further, with the term regression coined by Galton in the late 19th century [39], although the concept of a least squares fit was developed for astronomical applications at the start of the 19th century by Gauss and Legendre independently [115]. The theory and mathematical reasoning behind these techniques and many more is well understood and has been well documented in literature [20, 54, 66]. However, it is only due to recent advances in personal computing and the wide spread availability of machine learning libraries such as Scikit learn, Pytorch and Tensor Flow, that the full capabilities of these techniques have been able to be realised.

Of particular interest for the work in this thesis are the ML methods used for classification. ML classifiers are capable of finding patterns in data and making predictions from this data that humans are simply incapable of. Once trained, in an off-line process known as training, they are capable of processing large quantities of data extremely efficiently, making them well suited to the task of metal detection, showing particular promise when combined with an approach based on MPT spectral signatures. Some of the well known classification algorithms that are commonly applied in data science applications as well as in science and engineering include decision trees, random forests, gradient boosting, support vector machines, neural networks and logistic regression [20, 54, 66].

The MPT spectral signature has been exploited as an object feature description in a range of simple classification algorithms including simple library classification for homogeneous [14] and inhomogeneous objects [74], a k nearest neighbours (KNN) classification algorithm [82] and other machine learning approaches [127]. The MPT classification of objects has already been applied to a range of different applications including airport security screening [87, 82], waste sorting [62] and anti-personnel landmine detection [101]. The aforementioned *supervised* classification techniques rely on a library of MPT spectral signatures to *learn* how to classify the objects. Previous practical applications have used

dictionaries of measured MPT coefficients for known objects, relevant for the application under consideration, for example, [83] considered between 3-13 classes of objects with up to 200 samples for the classification. But, such dictionaries are subject to errors and have unavoidable noise, as described above, and hence, this may limit the performance of classification of unknown objects if it is then used in a metal detection system with less noise (and/or reduced errors) compared to the system used for the measuring the dictionary. Despite this existing work there is considerable scope for using alternative ML classifiers and larger dictionaries or simulated (instead of measured) object characterisations to be harnessed in metal detection object classification.

1.4   Aim and Objectives

The aim of this thesis is to harness the state-of-the-art mathematical developments in MPT object characterisation by Ledger and Lionheart in a series of papers [69, 70, 71, 73] and to apply recent developments in ML classification to the metal detection problem. This will centre around the creation and use of a synthetic database of conducting object characterisations, with the application of hidden conducting security threat identification in mind. The thesis covers: the development of a problem specific FE implementation that benefits from the application of a reduced order model (ROM) allowing for the fast computation of MPT spectral signatures; the creation of a synthetic database of realistic conducting object characterisations; and application of supervised classification methods. To achieve this aim, the next specific objectives are considered:

1. Provide a review of the recent work and developments of theory surrounding MPTs.

2. Develop a robust and accurate computational methodology for the solution of eddy-current problems using finite elements software.

3. Create an efficient ROM that allows for the fast computation of MPT spectral signatures across many frequencies.

4. Demonstrate the accuracy and efficiency offered by this computational methodology.

5. Apply the method to a series of geometrically realistic objects to create a database of threat and non-threat object characterisations.

6. Document and implement a series of supervised ML classifiers to the database of object characterisations.

The research carried out in order to achieve these objectives is split into several stages, which are described in the outline of the thesis. This research was done as part of the Reducing the threat to public safety project, in collaboration with Keele University, Swansea University, The University of Manchester and University College London with the industrial partners Rapiscan Systems, MT Safeline and Defence Science Technology Labratory

1.5   Outline of the Thesis

This thesis is formed of 10 chapters and is complemented by an appendix, and is partitioned into 6 parts. The organisation of these is as follows:

—PART I—
Preliminaries

- **Chapter 1: (Introduction)** presents a brief introduction to metal detection, discussing critical safety and security based applications, providing motivation for the work conducted throughout the thesis. An overview of current methodologies is provided, this covers related imaging based methods such as MIT and GPR. Then a review of current work on MPTs is presented, touching on subjects such as the short comings of single frequency characterisations and difficulties in measuring MPTs plus the advantages of generating MPTs using a computational approach. Recent work using MPT coefficients and eigenvalues as features in a ML scheme are then discussed. The aim and objectives of this thesis are then stated.

—PART II—
MPT Spectral Signatures

- **Chapter 2: (Eddy-Current Model and Object Characterisation)** achieves three things, firstly it sets the scene for other chapters in the thesis, introducing and discussing important topics such as the formulation of the time harmonic eddy current model, basic metal detector designs and current methodology, the difference between the forward and inverse problems, the asymptotic expansion and MPT. Secondly, it gives an overview of the MPT, discussing limiting cases which provide a fundamental insight into the behaviour of the MPT and highlight the important role object topology plays. Lastly, it provides motivation for the use of the MPT spectral signature as opposed to the MPT at a single frequency. The chapter also contains previously undocumented scaling results relating to object size and conductivity with frequency.

- **Chapter 3: (Finite Element Discretisation and the Reduced Order Model)** focuses on the development of an efficient FE based method for the rapid computation of MPT spectral signatures. This involves reducing the computational cost associated with solving the transmission problems discussed in Chapter 2, the development of a reduced order model involving projection based proper orthogonal

decomposition (PODP) to further reduce the cost of the spectral aspect of the MPT spectral signature. For the full order solution, a hp-FE discretisation is used for the weak forms associated with the transmission problems documented in Chapter 2. The ROM benefits from a-posteriori error estimates, which provide an upper bound on the error in the MPT coefficients obtained with the reduced order model compared to those obtained from a full order solution.

- **Chapter 4: (Implementation Details of `MPT-Calculator` Software)** builds on the work presented in Chapter 3, focusing on the implementation for the computation of MPT coefficients and the development of open source software `MPT-Calculator` created by the author. The chapter discusses the reasons behind the choice of FE framework, the computational implementation using this FE framework, structure of the software and the problem specific efficiencies that have been achieved in the implementation used. The chapter also documents the support offered for the software in the form of a user guide and a YouTube video series.

- **Chapter 5: (Numerical Results for the Reduced Order Model)** presents a series of illustrative examples to demonstrate the implementation and performance of the ROM, with a-posteriori error estimates, proposed in Chapters 3 and 4. The chapter covers many important topics such as linearly vs logarithmically placed snapshots for the ROM, accuracy of the methods, additional snapshot placement and time savings offered by the methods. The chapter also contains examples of the application of the original scaling results documented in Chapter 2.

—PART III—
Object Characterisation

- **Chapter 6: (Decomposing the Inverse Problem to one of Object Location and Object Classification)** describes the decomposition of the inverse problem, to one of object location and object classification. In this chapter a brief review of methods to obtain the location of a hidden target is presented with the methodology of the MUSIC algorithm documented. Object classification will be based on object characterisations, with those that are invariant under position and rotation being advocated for due to their obvious advantages. A series of applicable invariants that can be obtained from the MPT coefficients are then presented, these include eigenvalues, principal and deviatoric tensor invariants along with the eigenvalues of the commutator of the real and imaginary tensor also being proposed.

- **Chapter 7: (Real World Object Characterisation Dataset)** focuses on the development of a dataset of realistic object characterisations that could be used in a detection device in a security screening environment. The dataset, called the `MPT-Library`, created by the author is the first of its kind and has been made publicly available. The chapter provides a series of illustrative examples to demonstrate how the ROM approach described in Chapter 3 can be applied to a set of real-

istic geometries and combined with an appropriate choice of eigenvalues or tensor invariants in Section 6.3 to form a realistic dataset for object classification.

—PART IV—
Object Classification

- **Chapter 8: (Probabilistic and Non-Probabilistic Machine Learning Classifiers)** builds on the work presented in Chapter 7, documenting and comparing probabilistic and non-probabilistic ML classifiers that are appropriate for classifying objects when the features are MPT invariants, with the goal of developing an ML classifier trained on the `MPT-Library` that could be deployed in a walk through metal detector. The chapter starts by documenting the capability of currently deployed walk through metal detectors and lab based solenoid measurement systems. The chapter proposes a method for adding noise to the MPT coefficients that simulates the noise associated with measuring MPT coefficients in real world setups. Then a review of candidate ML classifiers are documented with references provided for those less familiar with the methods. Finally metrics that will be used to validate the results are discussed.

- **Chapter 9: (Machine Learning Classification Results)** presents results for the application of the ML classifiers discussed in Chapter 8 applied to two problems, a coin classification problem and a multi-class problem. The former is used as a demonstrative example where various intricacies about the problem are considered such as a sampling method based on the scaling results derived in Chapter 2 used to enrich the dataset, the effect the level of noise has on the problem, and the effect the number of frequencies measured has on the accuracy of the created classifier. The multi-class problem is then considered with classes of similar shaped objects being grouped together. The chapter culminates with an investigation into removing specific knife geometries out of the training set and the effect it has on the classifier's ability to detect these threat objects.

—PART V—
Conclusions

- **Chapter 10: (Conclusions and Future Work)** presents an overview of the work and the achievements of this thesis. The conclusions extracted from each chapter are stated. Furthermore, a list of recommendations for future research, that could further improve methods and results are provided.

—PART IV—
Appendices

- **Appendix A:** builds on the description of the `MPT-Calculator` software provided in Chapter 4, presenting further details of the contents of the files.

1.6 Research Outcomes

This section presents a list of the research outcomes from the work performed as part of this thesis.

1.6.1 Journal Publications

- **B. A. Wilson** and P. D. Ledger. Efficient computation of the magnetic polarizability tensor spectral signature using POD. *International Journal for Numerical Methods in Engineering*, 122(8), 1940-1963, 2021.

- P. D. Ledger, **B. A. Wilson**, A. A. S. Amad, and W. R. B. Lionheart. Identification of metallic objects using spectral magnetic polarizability tensor signatures: Object characterisation and invariants. *International Journal for Numerical Methods in Engineering*, 122(15), 3941-3984, 2021.

- **B. A. Wilson**, P. D. Ledger, and W. R. B. Lionheart. Identification of metallic objects using spectral magnetic polarizability tensor signatures: Object classification. (accepted) 2022, DOI: 10.1002/nme.6927.

1.6.2 Conference Papers and Presentations

- **B. A. Wilson** and P. D. Ledger. Hidden security threat identification: A reduced order model for the rapid computation of object characterisations. *UK Association for Computational Mechanics Conference,* Loughborough, United Kingdom. April 2020.

- **B. A. Wilson** and P. D. Ledger. Hidden security threat identification: Computing an ML library using FEM accelerated by a reduced order model. *UK Association for Computational Mechanics Conference,* Loughborough, United Kingdom. April 2021.

- **B. A. Wilson** and P. D. Ledger. Hidden security threat identification: A library of MPT spectral signature object characterizations for ML. *ECCOMAS Young Investigators Conference,* Valencia, Spain. July 2021.

- **B. A. Wilson** and P. D. Ledger. Hidden security threat identification: Applying ML classifiers to a computationally created library of realistic threat objects. *International Workshop on Optimisation and Inverse Problems in Electromagnetics,* Szczecin, Poland. September 2021. Awarded prize for third best presentation by a PhD student.

1.6.3 Research Posters

- **B. A. Wilson** and P. D. Ledger. Machine Learning for the Improvement of Metal Detection *ZCCE workshop,* Swansea, United Kingdom. January 2019.

- **B. A. Wilson** and P. D. Ledger. Machine Learning for the Improvement of Metal Detection *SIAM-IMA Student Chapter's inaugural Early Career Researcher conference,* Liverpool, United Kingdom. April 2019.

- **B. A. Wilson** and P. D. Ledger. Machine Learning for the Improvement of Metal Detection *Siam UKIE National Student Chapter conference,* Manchester, United Kingdom. June 2019. Awarded prize for best poster presentation.

- **B. A. Wilson** and P. D. Ledger. Computing Characterisations of Conducting Objects using NG-Solve for Metal Detection *NGSolve User Meeting,* Vienna, Austria. July 2019.

- **B. A. Wilson** and P. D. Ledger. Improved Metallic Object Characterisation, Location and Detection *ZCCE workshop,* Swansea, United Kingdom. January 2020.

- **B. A. Wilson** and P. D. Ledger. Machine Learning for the Improvement of Metal Detection *ZCCE workshop,* Swansea, United Kingdom. January 2021.

1.6.4 Software Packages and Datasets

The methodology described in this thesis has resulted in a new open source software, the `MPT-Calculator` [134], which allows for the rapid production of MPT spectral signatures. Along with the software itself there is considerable support offered around the software, in the form of a user guide and a YouTube video series. In addition to this software, a dataset, of realistic object characterisations, the `MPT-Library` [132], which is the first of its kind, has be produced and made publicly available. This dataset has been viewed 51 times and downloaded 12 times at the time of writing, showing it is being well utilised already.

Part II

MPT Spectral Signatures

Chapter 2

Eddy-Current Model and Object Characterisation

2.1 Introductory Remarks

The aim of this chapter is three fold, firstly to set the scene for other chapters in the thesis, introducing and discussing important topics such the formulation of the time harmonic eddy current model, basic metal detector designs and methodology, the difference between the forward and inverse problems, the asymptotic expansion and MPT. Secondly, to give an overview of the MPT, discussing limiting cases which provide a fundamental insight into the behaviour of the MPT and highlight the important role object topology plays. Lastly, to provide motivation for the use of the MPT spectral signature as opposed to the MPT at a single frequency.

The chapter is based on a combination of the work presented by the author in [133, 76], along with results first presented in [69, 70].

The main novelties of the chapter are the collation of recent results that have been proved by Ledger and Lionheart in [69, 70, 71] on the characterisation of objects by MPTs that will be relevant for the remaining chapters of the thesis. These include Ledger and Lionheart's real-imaginary split symmetric form of the MPT presented in (2.5), their limiting frequency approach for the eddy-current model for which a numerical implementation is employed in Chapters 3 and 4, and finally results proved by Ledger on the scaling of the MPT under the parameters of object size and conductivity.

The structure of the chapter is as follows in Section 2.2 the time harmonic eddy current model, along with the forward and inverse problems are discussed. In Section 2.3 the asymptotic expansion and real-imaginary symmetric form of the MPT is presented along with discussing the limiting frequency of the eddy-current model. Next, in Section 2.4 a series of limiting cases for the MPT are considered. In Section 2.5 the extent to which an MPT at a fixed frequency uniquely characterises an object is examined. In Section 2.6 an examination into the improvements to MPT object characterisation given spectral data. In Section 2.7 scaling results under changes in object size and conductivity parameters are derived. The chapter finishes with concluding remarks in Section 2.8.

2.2 Eddy Current Model

The eddy current model is a low frequency approximation of the Maxwell system that neglects the displacement currents, it is commonly accepted to be valid if the quasi-static assumption applies (dimension $D \approx \alpha$ of the object B_α is small compared to the wavelength) and the conductivities are high ($\sigma_{max} \gg \omega\epsilon_{max}$, where ϵ_{max} denotes the object's maximum permittivity, here assumed to be $\epsilon_{max} = \epsilon_0 \approx 8.854 \times 10^{-12}$F/m). A rigorous justification of the model involves the topology of the conducting body [12]. The eddy current model is described by the time-harmonic system

$$\nabla \times \boldsymbol{E}_\alpha = \mathrm{i}\omega\mu\boldsymbol{H}_\alpha, \tag{2.1a}$$

$$\nabla \times \boldsymbol{H}_\alpha = \boldsymbol{J}_0 + \sigma\boldsymbol{E}_\alpha, \tag{2.1b}$$

where $\boldsymbol{E}_\alpha$ and $\boldsymbol{H}_\alpha$ are the electric and magnetic interaction fields, respectively, $\boldsymbol{J}_0$ is an external current source, $\mathrm{i} := \sqrt{-1}$, ω is the angular frequency, μ is the magnetic permeability and σ is the electric conductivity.

2.2.1 Metal Detection

In metal detection, the conducting body is assumed to be contained in some medium $B_\alpha^c := \mathbb{R}^3 \backslash \bar{B}_\alpha$, where the overbar denotes the closure of B_α. The overbar to denote the complex conjugate, but it should be clear from the context as to which definition applies. This medium is assumed to have a vastly lower conductivity $\sigma \approx 0$ and have a permeability $\mu = \mu_0 := 4\pi \times 10^{-7}$H/m. A current $\boldsymbol{J}_0$ is passed through a series of exciter coils which each generate a background field $\boldsymbol{H}_0$. When this time varying background field interacts with a highly conducting object it generates eddy currents. The eddy currents generated lead to a $\boldsymbol{H}_0$ being perturbed to $\boldsymbol{H}_\alpha$. A sequence of measurement coils at position $\boldsymbol{x}$ then sense the perturbation $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ in the form of a voltage. The situation of a single measurement and exciter coil is shown in Figure 2.1.
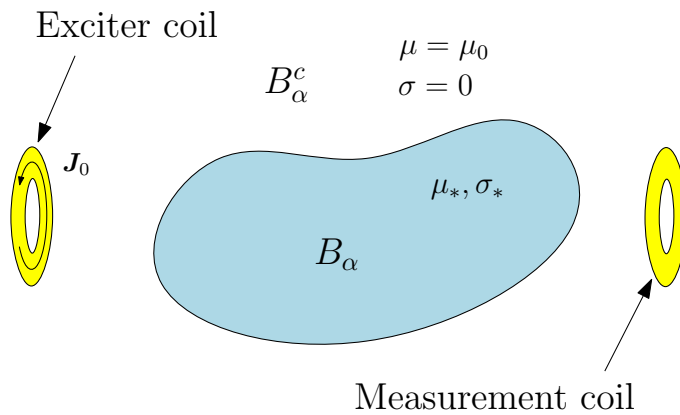
Figure 2.1: A diagram showing a conducting object B_α, in a non-conducting medium B_α^c, with a current $\boldsymbol{J}_0$ passing through an exciter coil and an associated measurement coil.

The model shown in Figure 2.1, is described by the system (2.1), with

$$\mu(\boldsymbol{x}) = \begin{cases} \mu_* & \boldsymbol{x} \in B_\alpha \\ \mu_0 & \boldsymbol{x} \in B_\alpha^c \end{cases}, \qquad \sigma(\boldsymbol{x}) = \begin{cases} \sigma_* & \boldsymbol{x} \in B_\alpha \\ 0 & \boldsymbol{x} \in B_\alpha^c \end{cases}. \tag{2.2}$$

The regions B_α and B_α^c are coupled by the transmission conditions

$$[\boldsymbol{n} \times \boldsymbol{E}_\alpha]_{\Gamma_\alpha} = [\boldsymbol{n} \times \boldsymbol{H}_\alpha]_{\Gamma_\alpha} = \boldsymbol{0}, \tag{2.3}$$

which hold on $\Gamma_\alpha := \partial B_\alpha$. In the above, $[u]_{\Gamma_\alpha} := u|_+ - u|_-$ denotes the jump, the $+$ refers to just outside of B_α and the $-$ to just inside and $\boldsymbol{n}$ denotes a unit outward normal to Γ_α. The electric interaction field is non-physical in B_α^c and, to ensure uniqueness of this field, the condition $\nabla \cdot \boldsymbol{E}_\alpha = 0$ is imposed in this region. Furthermore, we also require that $\boldsymbol{E}_\alpha = O(1/|\boldsymbol{x}|)$ and $\boldsymbol{H}_\alpha = O(1/|\boldsymbol{x}|)$ as $|\boldsymbol{x}| \to \infty$, denoting that the fields go to zero at least as fast as $1/|\boldsymbol{x}|$, although, in practice, this rate can be faster.

Hand-held metal detector: Hand-held metal detectors are used in a variety of situations for a variety of different purposes, including, anit-personnel landmine clearance [7] and archeological digs [29], with additional applications in prospecting. The concealed object B_α is assumed to be buried in relatively non-conducting, $\sigma \approx 0$, non-permeable, $\mu = \mu_0$, soil. The search coil assembly of the detector contains the coils of wires which make up the excitor and measurement coils, that the current source $\boldsymbol{J}_0$ passes though and take the measurements $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$, respectively. Note that practical metal detectors measure a voltage perturbation, which corresponds to $\int_S \boldsymbol{n} \cdot (\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$ over an appropriate surface S [71]. For very small coils, this voltage perturbation is approximated by $\boldsymbol{m} \cdot (\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ where $\boldsymbol{m}$ is the magnetic dipole moment of the coil [71]. These coils are supported by the operator in the non-conducting, non-permeable, air above the soil and object, B_α^c consists of the combination of soil and air. The detector can then be moved through the air above the object to take measurements $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ for different positions $\boldsymbol{x}$. A Diagram of the set up can be seen in Figure 2.2.

Walk-through metal detector: Walk-through metal detectors, which are commonly used in airport security screening [87, 82], are another example of detecting hardware that rely upon the eddy current principal. The concealed object B_α passes through the detector as the person walks through the detector, with the air, person's clothing and body making up the non-conducting, non-permeable region B_α^c. The arch design of the detector allows for multiple exciting and measurement coils to be housed in the panels of the detector, enabling greater precision in measurements by using combinations of coil pairs. These coil pairs are chosen in turn and take measurements with the excitor coil having a current source $\boldsymbol{J}_0$ pass through it and the measurement coil taking measurements $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$. A diagram of the set up can be seen in Figure 2.3. In metal detection the location of the object and it materials are unknown. We shall introduce this as an inverse problem in the next section. The corresponding direct forward problem relates to the solution of (2.1) where the location, shape of the object and its materials properties are known and the task is determine the fields $\boldsymbol{E}_\alpha$ and $\boldsymbol{H}_\alpha$ in and around the object from knowledge of $\boldsymbol{J}_0$.
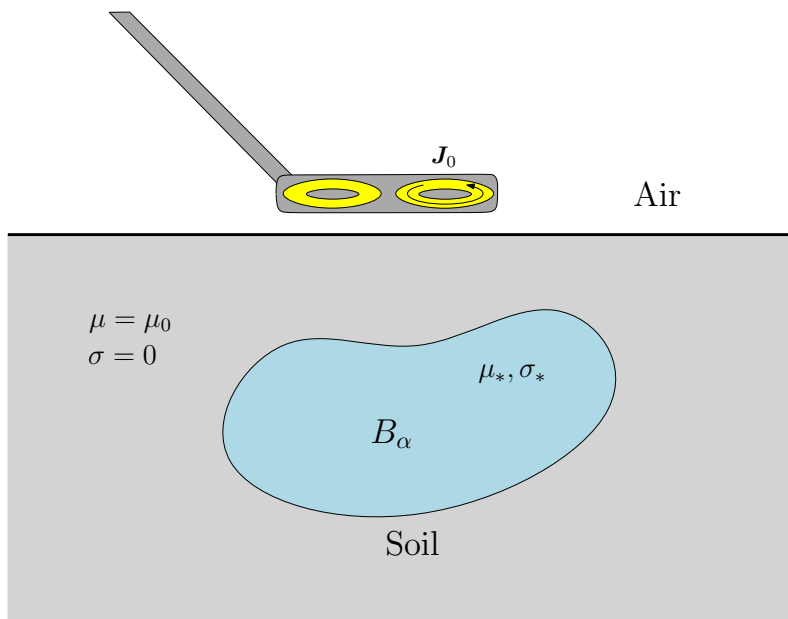
Figure 2.2: A diagram showing a hidden conducting object B_α, buried in soil, with an excitor and measurement coil located in the air above.
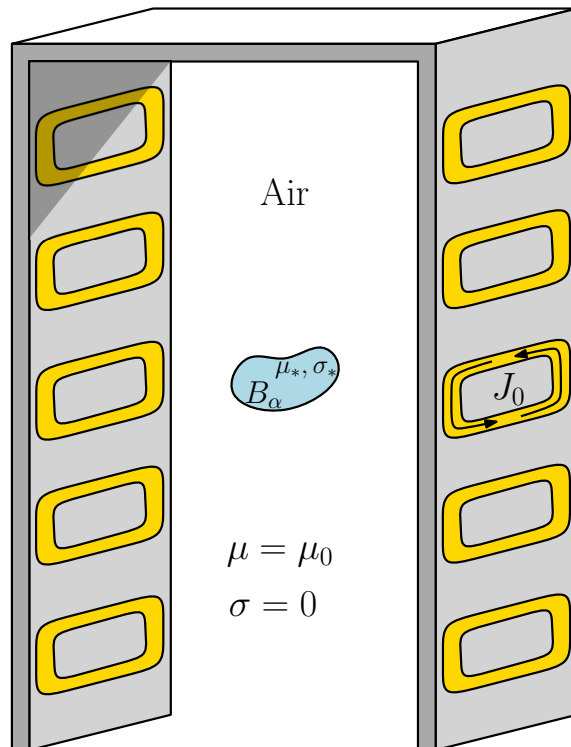
Figure 2.3: A diagram showing a concealed conducting object B_α, with a current source passing through a coil located in one side of the detector, and measurement coils located throughout the detector.

2.2.2 Inverse Problem

The inverse problem is to determine the location, shape and material properties (σ_* and μ_*) of the conducting object B_α from measurements of $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ taken at a range of locations $\boldsymbol{x}$. Here, $\boldsymbol{H}_0$ denotes the background magnetic field and $\boldsymbol{E}_0$ and $\boldsymbol{H}_0$ are the solutions of (2.1) with $\sigma = 0$ and $\mu = \mu_0$ in $\mathbb{R}^3$. Similar to above, the decay conditions $\boldsymbol{E}_0 = O(1/|\boldsymbol{x}|)$ and $\boldsymbol{H}_0 = O(1/|\boldsymbol{x}|)$ as $|\boldsymbol{x}| \to \infty$ are required. Additionally, as described in the introduction and will be further justified in Section 2.6 there are considerable advantages in using spectral data, i.e. additionally measuring $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ over a range of frequencies ω, within the limit of the eddy current model.

A traditional approach to the solution of this inverse problem involves creating a discrete set of voxels, each with unknown σ and μ, and posing the solution to the inverse problem as an optimisation process in which σ and μ are found through minimisation of an appropriate functional e.g. [122]. From the resulting images of σ and μ one then attempts to infer the shape and position of the object. However, this problem is highly ill-posed [24] and presents considerable challenges mathematically and computationally in the case of limited noisy measurement data.

Preference is given to an alternative approach, where the object will be characterised by a small number of coefficients that can be obtained from the measurements. In the next section this characterisation is introduced.

2.3 The Asymptotic Expansion and MPT Description

Following the definition made in [13, 69] where $B_\alpha := \alpha B + \boldsymbol{z}$ where B is a unit size object with Lipschitz boundary [31], that is that for each point on ∂B, B is locally the set of points located above the graph of some Lipschitz function, α is the object size and $\boldsymbol{z}$ is the object's translation from the origin as shown in Figure 2.4.
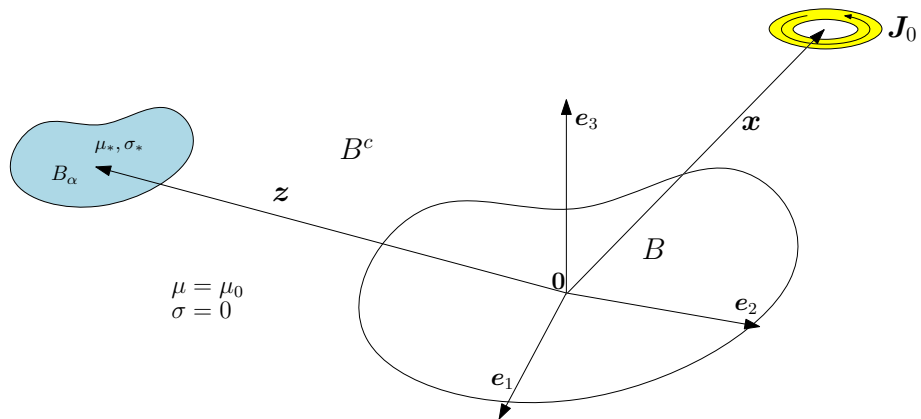
Figure 2.4: A diagram showing the physical description of B_α with respect to the coordinate axes.

Then, using the asymptotic formula obtained by Ammari, Chen, Chen, Garnier and Volkov [13], Ledger and Lionheart [69] have derived the simplified form

$$(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})_i = (\boldsymbol{D}_{\boldsymbol{x}}^2 G(\boldsymbol{x}, \boldsymbol{z}))_{ij}(\mathcal{M})_{jk}(\boldsymbol{H}_0(\boldsymbol{z}))_k + O(\alpha^4), \qquad (2.4)$$

which holds as $\alpha \to 0$ and makes the MPT explicit. The relationship between the leading order term in the above to the dipole expansion of $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ is discussed in [71]. In the above, $G(\boldsymbol{x}, \boldsymbol{z}) := 1/(4\pi|\boldsymbol{x} - \boldsymbol{z}|)$ is the free space Laplace Green's function, $\boldsymbol{D}_x^2 G$ denotes the Hessian of G, Einstein summation convention of the indices is implied and $\mathcal{M} = (\mathcal{M})_{jk}\boldsymbol{e}_j \otimes \boldsymbol{e}_k$, where $\boldsymbol{e}_i$ denotes the ith orthonormal unit vector, is the MPT. Unlike a matrix, due to $\mathcal{M}$ being a rank 2 tensor, its coefficients transform as the coefficients of a rank 2 tensor under rotation of the coordinate axes, i.e. $(\mathcal{M})'_{ij} = (\mathbf{R})_{ip}(\mathbf{R})_{jq}(\mathcal{M})_{pq}$ where $\mathbf{R} = \mathbf{R}_1(\alpha)\mathbf{R}_2(\beta)\mathbf{R}_3(\gamma)$ where α, β and γ are the Euler angles and $\mathbf{R}$ is an orthogonal rotation matrix describing the coordinate rotation, an example of such rotation can be seen in Figure 2.5.
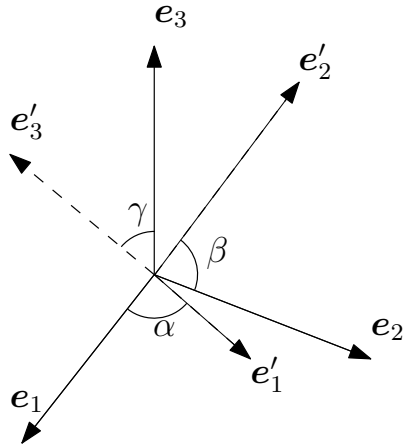
Figure 2.5: A diagram showing rotated coordinate axes $\boldsymbol{e}'_1, \boldsymbol{e}'_2$ and $\boldsymbol{e}'_3$, which are obtained by applying the angles α, β and γ of rotation for each of the coordinate axes.

This expansion has a series of properties which we summarise below:

- There exist different, yet equivalent formulae, for computing the tensor coefficients, these forms may be exploited in various situations, this document makes use of the formulation derived by Ledger and Lionheart in Theorem 5.1 of [73], however, other formulations derived by Ledger and Lionheart in [69, 70, 71] which are equivalent for exact fields.

- It is an asymptotic expansion as $\alpha \to 0$, this means that the residual becomes increasingly small for small objects. In practice, provided that $\alpha < 1$m the leading order term provides a reasonable approximation. Further improvements can be obtained using generalised magnetic polarizability tensors (GMPTs) [72].

- The MPT is a rank 2 complex symmetric tensor, i.e. its coefficients satisfy $(\mathcal{M})_{ij} = (\mathcal{M})_{ji}$, and it, therefore, transforms as a rank 2 tensor, i.e. $(\mathcal{M})'_{ij} = (\mathbf{R})_{ip}(\mathbf{R})_{jq}(\mathcal{M})_{pq}$, this was proved by Ledger and Lionheart in Theorem 3.2 of [69].

- The MPT is invariant under position shown, i.e. $(\mathcal{M})_{ij} = (\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ is a function of $\alpha B, \omega, \sigma_*, \mu_r$, where $\mu_r := \mu_*/\mu_0$, but is independent of position $\mathbf{z}$, this was shown by Ammari *et al.* in Theorem 3.2 of [14].

- The asymptotic expansion (2.4) allows the object position $\mathbf{z}$ to be separated from $\alpha B, \omega, \sigma_*, \mu_r$. But, it is not possible to separate $\alpha B, \omega, \sigma_*, \mu_r$ at a single frequency. this is explained further in Section 2.5.1.

- The MPT's dependence on ω is called its spectral signature. This will allow further object characterisation to be captured. This is explained further in Section 2.6.

- Higher order GMPTs and a complete asymptotic expansion of $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ as $\alpha \to 0$ have been derived by Ledger and Lionheart in [72], these provide additional object characterisation information, this is outside the scope of the investigation and will therefore not be considered in this thesis.

On occasion, the MPT will be written as $\mathcal{M}[\alpha B, \omega]$ to emphasise its dependence on αB and ω, the MPT will also be denoted as $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$, to emphasise its dependence also on μ_r and σ_*. The above formulation, and the definition of $\mathcal{M}$ below, are presented for the case of a single homogenous object B, the extension to multiple inhomogeneous objects can be found in [74, 73].

Using the derivation in [73], the explicit formulae for the computation of the coefficients of $\mathcal{M}$ are stated, which are particularly well suited to a FEM discretisation, that will be discussed in more detail in Chapter 3. The splitting $(\mathcal{M})_{ij} := (\mathcal{N}^0)_{ij} + (\mathcal{R})_{ij} + \mathrm{i}(\mathcal{I})_{ij}$ obtained in [73] is used, with

$$(\mathcal{N}^0[\alpha B, \mu_r])_{ij} := \alpha^3 \delta_{ij} \int_B (1 - \tilde{\mu}_r^{-1}) \mathrm{d}\boldsymbol{\xi} + \frac{\alpha^3}{4} \int_{B \cup B^c} \tilde{\mu}_r^{-1} \nabla \times \tilde{\boldsymbol{\theta}}_i^{(0)} \cdot \nabla \times \tilde{\boldsymbol{\theta}}_j^{(0)} \, \mathrm{d}\boldsymbol{\xi},$$
$$(2.5\text{a})$$

$$(\mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} := -\frac{\alpha^3}{4} \int_{B \cup B^c} \tilde{\mu}_r^{-1} \nabla \times \boldsymbol{\theta}_j^{(1)} \cdot \nabla \times \overline{\boldsymbol{\theta}_i^{(1)}} \, \mathrm{d}\boldsymbol{\xi}, \qquad (2.5\text{b})$$

$$(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} := \frac{\alpha^3}{4} \int_B \nu \left(\boldsymbol{\theta}_j^{(1)} + (\tilde{\boldsymbol{\theta}}_j^{(0)} + \boldsymbol{e}_j \times \boldsymbol{\xi}) \right) \cdot \left(\overline{\boldsymbol{\theta}_i^{(1)} + (\tilde{\boldsymbol{\theta}}_i^{(0)} + \boldsymbol{e}_i \times \boldsymbol{\xi})} \right) \mathrm{d}\boldsymbol{\xi},$$
$$(2.5\text{c})$$

where $B^c := \mathbb{R}^3 \backslash \overline{B}$ and $\mathcal{N}^0[\alpha B, \mu_r]$, $\mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r]$, $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are real symmetric rank 2 tensors, which each have real eigenvalues. Note that sometimes the notation $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} = (\mathcal{N}^0[\alpha B, \mu_r])_{ij} + (\mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ will be used so that $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} = \mathrm{Re}(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ indicate the real coefficients of the MPT and $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} = \mathrm{Im}(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ indicate the imaginary coefficients

of the MPT. In the above, $\boldsymbol{\theta}_i^{(0)} = \boldsymbol{\theta}_i^{(0)}(\alpha B, \mu_r)$, $\boldsymbol{\theta}_i^{(1)} = \boldsymbol{\theta}_i^{(1)}(\alpha B, \omega, \sigma_*, \mu_r)$, $\boldsymbol{\xi}$ is chosen to be measured from an origin inside B,

$$\tilde{\mu}_r(\boldsymbol{\xi}) := \begin{cases} \mu_r & \boldsymbol{\xi} \in B \\ 1 & \boldsymbol{\xi} \in B^c \end{cases},$$

and $\nu := \alpha^2 \omega \mu_0 \sigma_*$, δ_{ij} is the Kronecker delta. The computation of (2.5) rely on the real solution $\boldsymbol{\theta}_i^{(0)}(\boldsymbol{\xi})$ of the transmission problem [73, 69, 13]

$$\begin{align}
\nabla \times \tilde{\mu}_r^{-1} \nabla \times \boldsymbol{\theta}_i^{(0)} &= \boldsymbol{0} & &\text{in } B \cup B^c, & (2.6a) \\
\nabla \cdot \boldsymbol{\theta}_i^{(0)} &= 0 & &\text{in } B \cup B^c, & (2.6b) \\
[\boldsymbol{n} \times \boldsymbol{\theta}_i^{(0)}]_\Gamma &= \boldsymbol{0} & &\text{on } \Gamma, & (2.6c) \\
[\boldsymbol{n} \times \tilde{\mu}_r^{-1} \nabla \times \boldsymbol{\theta}_i^{(0)}]_\Gamma &= \boldsymbol{0} & &\text{on } \Gamma, & (2.6d) \\
\boldsymbol{\theta}_i^{(0)} - \boldsymbol{e}_i \times \boldsymbol{\xi} &= \boldsymbol{O}(|\boldsymbol{\xi}|^{-1}) & &\text{as } |\boldsymbol{\xi}| \to \infty, & (2.6e)
\end{align}$$

where $\Gamma := \partial B$ and the complex solution $\boldsymbol{\theta}_i^{(1)}(\boldsymbol{\xi})$ of the transmission problem

$$\begin{align}
\nabla \times \mu_r^{-1} \nabla \times \boldsymbol{\theta}_i^{(1)} - \mathrm{i}\nu(\boldsymbol{\theta}_i^{(0)} + \boldsymbol{\theta}_i^{(1)}) &= \boldsymbol{0} & &\text{in } B, & (2.7a) \\
\nabla \times \nabla \times \boldsymbol{\theta}_i^{(1)} &= \boldsymbol{0} & &\text{in } B^c, & (2.7b) \\
\nabla \cdot \boldsymbol{\theta}_i^{(1)} &= 0 & &\text{in } B^c, & (2.7c) \\
[\boldsymbol{n} \times \boldsymbol{\theta}_i^{(1)}]_\Gamma &= \boldsymbol{0} & &\text{on } \Gamma, & (2.7d) \\
[\boldsymbol{n} \times \tilde{\mu}_r^{-1} \nabla \times \boldsymbol{\theta}_i^{(1)}]_\Gamma &= \boldsymbol{0} & &\text{on } \Gamma, & (2.7e) \\
\boldsymbol{\theta}_i^{(1)} &= \boldsymbol{O}(|\boldsymbol{\xi}|^{-1}) & &\text{as } |\boldsymbol{\xi}| \to \infty. & (2.7f)
\end{align}$$

Note also the choice to introduce $\tilde{\boldsymbol{\theta}}_i^{(0)}(\boldsymbol{\xi}) := \boldsymbol{\theta}_i^{(0)}(\boldsymbol{\xi}) - \boldsymbol{e}_i \times \boldsymbol{\xi}$, which can be shown to satisfy the same transmission problem as (2.6) except with a non-zero jump condition for $[\boldsymbol{n} \times \tilde{\mu}_r^{-1} \nabla \times \tilde{\boldsymbol{\theta}}_i^{(0)}]_\Gamma$ and the decay condition $\tilde{\boldsymbol{\theta}}_i^{(0)}(\boldsymbol{\xi}) = \boldsymbol{O}(|\boldsymbol{\xi}|^{-1})$ as $|\boldsymbol{\xi}| \to \infty$. The computational treatment of (2.6) and (2.7) will be discussed in Chapter 3.

2.3.1 Reductions in the Number of Independent Coefficients

As discussed in Section 2.3, the MPT is a rank 2 complex symmetric tensor, in the case of a general object (with no symmetries) this therefore means that it has 6 independent complex coefficients. However, Ledger and Lionheart show in [69] that this number can be reduced if the object has rotational or reflectional symmetries. The basis for this is that if the object is the same under a rotation or reflection then so will be its MPT. They use these arguments to derived the MPTs of some standard object shapes in Table 1 of [69].

Reflectional symmetry: In the case of an object B_α has reflectional symmetry in some plane with unit normal vector $\boldsymbol{n}$ the transformation tensor $\mathbf{R}$ can be defined as

$$(\mathbf{R})_{ij} := \delta_{ij} - 2n_i n_j, \tag{2.8}$$

where, the transformed tensor $(\mathcal{M})'_{ij}$ is defined to be

$$(\mathcal{M})'_{ij} := (\mathbf{R})_{il}(\mathbf{R})_{jm}(\mathcal{M})_{lm}. \qquad (2.9)$$

In the case where $\boldsymbol{n} = \boldsymbol{e}_3$, that is to say that the object is symmetric in the $\boldsymbol{x}_3$ direction through the plane $\boldsymbol{x}_3 = 0$, the following equalities can be found, $(\mathcal{M})'_{ii} = (\mathcal{M})_{ii}$ for $i = 1, 2, 3$, $(\mathcal{M})'_{12} = (\mathcal{M})_{12}$ and $(\mathcal{M})'_{21} = (\mathcal{M})_{21}$, $(\mathcal{M})'_{i3} = -(\mathcal{M})_{i3} = 0$ for $i = 1, 2$ and , $(\mathcal{M})'_{3i} = -(\mathcal{M})_{3i} = 0$ for $i = 1, 2$, this leads to a tensor in the form

$$\mathcal{M} = \begin{pmatrix} \mathcal{M}_{11} & \mathcal{M}_{12} & 0 \\ \mathcal{M}_{21} & \mathcal{M}_{22} & 0 \\ 0 & 0 & \mathcal{M}_{33} \end{pmatrix}, \qquad (2.10)$$

where, $\mathcal{M}_{12} = \mathcal{M}_{21}$ due to the symmetric property of the MPT.

Rotational symmetry: In the case where the object B_α has rotational symmetries about some axis $\boldsymbol{e}_i$ we may define some transformation tensor $\mathbf{R}$ which defines the rotation. In the case that the object B_α has rotational symmetries about the axis $\boldsymbol{e}_3$, a rotation around $\boldsymbol{e}_3$ by α in the anti-clockwise direction can be defined as

$$\mathbf{R} := \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad (2.11)$$

then once again defining the transformation as,

$$(\mathcal{M})'_{ij} := (\mathbf{R})_{il}(\mathbf{R})_{jm}(\mathcal{M})_{lm}. \qquad (2.12)$$

If B_α is invariant under an anti-clockwise rotation around $\boldsymbol{e}_3$ with $\psi = \pi/2$ the following equalities can be found, $(\mathcal{M})'_{11} = (\mathcal{M})_{22} \implies (\mathcal{M})_{11} = (\mathcal{M})_{22}$, $(\mathcal{M})'_{33} = (\mathcal{M})_{33}$ for the on diagonal coefficients of the tensor and, using the fact that the tensor is symmetric, $(\mathcal{M})'_{12} = -(\mathcal{M})_{12} = 0$, $(\mathcal{M})'_{21} = -(\mathcal{M})_{21} = 0$, and $(\mathcal{M})'_{13} = (\mathcal{M})_{23}$, $(\mathcal{M})'_{23} = -(\mathcal{M})_{13}$, which implies $(\mathcal{M})'_{13} = (\mathcal{M})_{23} = 0$ and $(\mathcal{M})'_{23} = -(\mathcal{M})_{13}$ $(\mathcal{M})'_{31} = (\mathcal{M})_{31}$, $(\mathcal{M})'_{32} = -(\mathcal{M})_{31}$, which implies $(\mathcal{M})'_{31} = (\mathcal{M})_{32} = 0$, this leads to a tensor in the form

$$\mathcal{M} = \begin{pmatrix} \mathcal{M}_{11} & 0 & 0 \\ 0 & \mathcal{M}_{22} & 0 \\ 0 & 0 & \mathcal{M}_{33} \end{pmatrix}, \qquad (2.13)$$

where $\mathcal{M}_{11} = \mathcal{M}_{22}$.

Limiting case of a sphere: In the limiting case of where B_α is a sphere, there are infinitely many rotational and reflectional symmetries, this leads to an MPT which is diagonal with only a single independent coefficient. An analytical solution for a conducting sphere in a time varying magnetic field was derived by Wait [128]. In the case of a sphere of unit radius the formulation in (2.4) provides an approximation of the form

$$M\delta_{ij} - (\mathcal{M})_{ij} = O(\alpha^4),$$

25

where M is the exact diagonal coefficient for the sphere, which are defined as

$$M := 2\pi\alpha^3 \frac{(2\mu_* + \mu_0)vp - (\mu_0(1 + v^2) + 2\mu_*)q}{(\mu_* - \mu_0)vp + (\mu_0(1 + v^2) - \mu_*)q}, \tag{2.14}$$

where $v = \alpha\sqrt{\mu_*\sigma\omega\mathrm{i}}$, $p = \sqrt{2v/\pi}\cosh(v)$, $q = \sqrt{2v/\pi}\sinh(v)$ and α is the radius of B_α in the case of B being a sphere of unit radius. An equivalent formulation using Bessel functions was presented by Wait [128]. An example of how the response changes with frequency for the case where B_α is a permeable conducting sphere of unit radius, with $\alpha = 0.01$ m, a permeability $\mu_r = 2$ and conductivity $\sigma_* = 10^6$ S/m is shown in Figure 2.6. This analytical solution will be exploited later in Chapter 5 when calculating associated error with the FE simulations.

$(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$ $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$
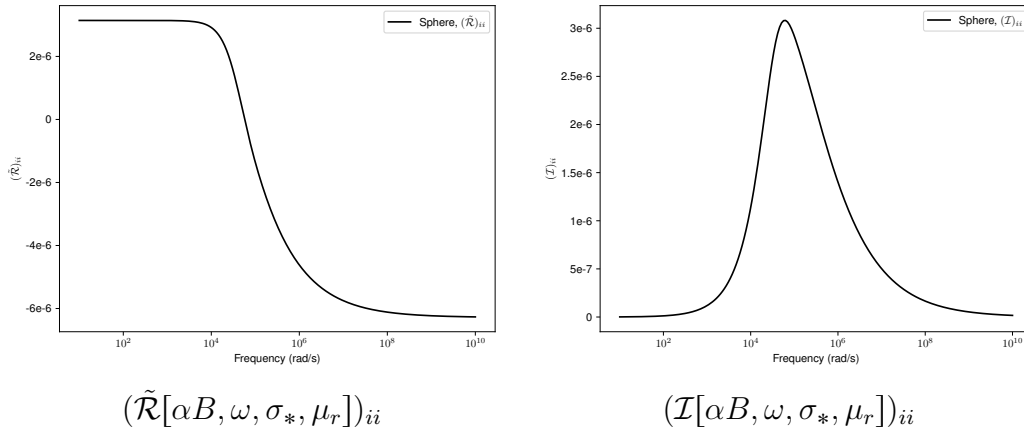
Figure 2.6: Permeable conducting sphere of unit radius B, $\alpha = 0.01$ m $\mu_r = 2$ and $\sigma_* = 10^6$ S/m: Comparison of the on diagonal coefficients $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$ and $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$.

2.4 Limiting Cases

In this section we present previously documented results pertaining to limiting cases of the MPT, these give some interesting insights into the behaviour of the MPT for specific use cases.

2.4.1 Limiting Magnetostatic Case and the Pólya-Szegö Tensor

When considering the limiting case of $\omega = 0$, $(\mathcal{R}[\alpha B, \omega = 0, \sigma_*, \mu_r])_{ij} = (\mathcal{I}[\alpha B, \omega = 0, \sigma_*, \mu_r])_{ij} = 0$, thus, $(\mathcal{M}[\alpha B, \omega = 0, \sigma_*, \mu_r])_{ij} = (\mathcal{N}^0[\alpha B, \mu_r])_{ij}$. The coefficients of $\mathcal{N}^0[\alpha B, \mu_r]$ have been further simplified by Ledger and Lionheart in Theorem 9 of [70] to the coefficients of the Pólya-Szegö tensor $\mathcal{T}[\alpha B, \mu_r]$, where the coefficients $(\mathcal{T}[\alpha B, \mu_r])_{ij}$ are defined as

$$(\mathcal{T}[\alpha B, \mu_r])_{ij} := \alpha^3\left(\delta_{ij}(\mu_r - 1)|B| + (\mu_r - 1)^2 \int_\Gamma \boldsymbol{n}^- \cdot (\nabla_{\boldsymbol{\xi}}\phi_i)\xi_j \mathrm{d}\boldsymbol{\xi}\right), \tag{2.15}$$

26

where $|B|$ is the volume of B and ϕ_i, $i = 1, 2, 3$, can be found by solving a scalar transmission problem,

$$\nabla^2 \phi_i = 0 \qquad \qquad \text{in } B \cup B^c, \qquad (2.16a)$$

$$[\phi_i]_\Gamma = 0 \qquad \qquad \text{on } \Gamma, \qquad (2.16b)$$

$$[\boldsymbol{n} \cdot \tilde{\mu}_r \nabla \phi_i]_\Gamma = \boldsymbol{n} \cdot \nabla \xi_i \qquad \qquad \text{on } \Gamma, \qquad (2.16c)$$

$$\phi_i \to 0 \qquad \qquad \text{as } |\boldsymbol{\xi}| \to \infty. \qquad (2.16d)$$

Following the notation presented in [15], with $\mathcal{T}[\alpha B, k]$ where k is referred to as the contrast, the tensor is simpler than the MPT and characterises small homogeneous conducting objects with shape B in electrical impedance tomography ($k = \sigma_*/\sigma_0$) and small permeable homogeneous objects with shape B in magnetostatics ($k = \mu_r = \mu_*/\mu_0$), it is symmetric and is positive (negative) definite provided that the contrast $k > 1$ ($0 \leqslant k < 1$) [15]. This formulation will be used in Section 2.5 when considering the MPT object characterisation at limiting frequencies.

2.4.2 Limiting Case of $\nu \to \infty$ and $\sigma_* \to \infty$

Considering an object B and the limiting case of $\nu = \alpha^2 \mu_0 \sigma_* \omega \to \infty$. Although the choice of either ω fixed and $\sigma_* \to \infty$ or σ_* fixed and $\omega \to \infty$ are both valid. In reality, while this MPT can be computed, the limiting case of $\omega \to \infty$ does not hold in practice as the eddy current model is a low frequency approximation to the Maxwell system. This was studied by Ledger and Lionheart in Lemma 4.1 of [73] the coefficients of $(\mathcal{M}[\alpha B, \omega \to \infty, \sigma_*, \mu_r])_{ij}$ simplify to

$$(\mathcal{M}[\alpha B, \omega \to \infty, \sigma_*, \mu_r])_{ij} = \alpha^3 \delta_{ij} |B| - \frac{\alpha^3}{4} \int_{B^c} \nabla \times \tilde{\boldsymbol{\theta}}_i^{(\infty)} \cdot \nabla \times \tilde{\boldsymbol{\theta}}_j^{(\infty)} \, \mathrm{d}\boldsymbol{\xi}, \qquad (2.17)$$

where $\boldsymbol{\theta}^{(\infty)}$ is the real vector field solution to

$$\nabla \times \nabla \times \boldsymbol{\theta}_i^{(\infty)} = \boldsymbol{0} \qquad \qquad \text{in } B^c, \qquad (2.18a)$$

$$\nabla \cdot \boldsymbol{\theta}_i^{(\infty)} = 0 \qquad \qquad \text{in } B^c, \qquad (2.18b)$$

$$\boldsymbol{n} \times \nabla \times \boldsymbol{\theta}_i^{(\infty)}|_+ = -2\boldsymbol{n} \times \boldsymbol{e}_i \qquad \qquad \text{on } \Gamma, \qquad (2.18c)$$

$$\boldsymbol{\theta}_i^{(\infty)} = O(|\boldsymbol{\xi}|^{-1}) \qquad \qquad \text{as } |\boldsymbol{\xi}| \to \infty. \qquad (2.18d)$$

An example of this for the permeable conducting sphere is presented in Figure-fig:limitingsphereexample.

27
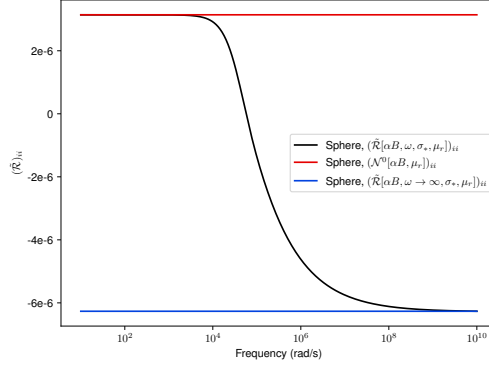
Figure 2.7: Permeable conducting sphere of unit radius B, $\alpha = 0.01$ m $\mu_r = 2$ and $\sigma_* = 10^6$ S/m: Comparison of the on diagonal coefficients of the $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$, $(\mathcal{N}^0[\alpha B, \mu_r])_{ii}$ and $(\tilde{\mathcal{R}}[\alpha B, \omega \to \infty, \sigma_*, \mu_r])_{ii}$.

Additionally, restricting the problem to the limiting case of $\sigma_* \to \infty$, for an object B, which is simply connected, with the first betti number of B, $\beta_1(B) = \beta_1(B^c) = 0$. Recall that $\beta_0(B)$ indicates the number of connected parts of B, which for a bounded connected region in $\mathbb{R}^3$ is always 1, $\beta_1(B)$ is the genus and is equal to the number of holes in B and $\beta_2(B)$ is equal to the number of voids in B. For further details on Betti numbers and their implications for MPTs see [70] and references therein. The coefficients of $(\mathcal{M}[\alpha B, \omega \to \infty, \sigma_*, \mu_r])_{ij} = (\mathcal{T}[\alpha B, \mu_r = 0])_{ij}$ as was shown by Ledger and Lionheart in Theorem 11 of [70], with

$$(\mathcal{T}[\alpha B, \mu_r = 0])_{ij} = -\alpha^3\big(|B|\delta_{ij} + \int_\Gamma \boldsymbol{n}^- \cdot \boldsymbol{e}_i \psi_j d\boldsymbol{\xi}\big), \tag{2.19}$$

where ψ_j solves

$$\nabla^2 \psi_j = 0 \qquad\qquad \text{in } B^c \tag{2.20a}$$
$$\boldsymbol{n} \cdot \nabla \psi_j|_+ = \boldsymbol{n} \cdot \nabla \boldsymbol{\xi} \qquad\qquad \text{on } \Gamma \tag{2.20b}$$
$$\psi_j \to 0 \qquad\qquad \text{as } |\boldsymbol{\xi}| \to \infty. \tag{2.20c}$$

2.4.3 Limiting Frequency of the Eddy Current Model

To improve on the approximate conditions for the validity of the eddy current model stated in Section 2.2 understanding the role the topology of the object on the limiting frequency of the eddy current model is important. Schmidt, Sterz and Hiptmair [107] have obtained the following estimates

$$C_1 \epsilon_{max} \mu_{max} \omega^2 D^2 \ll 1, \tag{2.21a}$$
$$C_2 \epsilon_{max} \omega \sigma_{min}^{-1} \ll 1, \tag{2.21b}$$

that are required to hold to ensure the validity of the eddy current model. In the above, $C_1 = C_1(B)$ and $C_2 = C_2(B)$ are constants that depend on the object's topology and

28

$(|B|\alpha^3)^{(1/3)}$ is the diameter of the computational domain.

In particular, an object with a long thin extension or with a small gap (eg hoarse-shoe shaped conductor) lead to capacitive coupling and have large $C_1(B)$ and $C_2(B)$ limiting the frequency ω_{limit} at which the MPT spectral signature remains valid compared to using the quasi-static and high conductivity conditions alone. Schmidt *et al.* describe a numerical procedure that allows the constants to be estimated numerically for different objects by solving a low-dimensional eigenvalue problem. Once the constants have been found, the limiting the frequency ω_{limit} can be estimated from (2.21). This procedure is applied to the numerical examples in this work.

2.5 MPT Object Characterisation at a Fixed Frequency

The notion that an object can be uniquely characterised is fundamental to the idea of object identification, for if an object cannot be uniquely characterised, identification would be impossible. In this section, the extent to which a MPT at a fixed frequency uniquely characterises an object is examined, this section is key in motivating the use of MPT spectral signatures. This section, follows the arguments presented by Ledger, Wilson, Amad and Lionheart in [76]. First, eigenvalue decompositions of the real and imaginary parts of the MPT for a fixed frequency is considered and, secondly, a proof of how equivalent ellipsoids can be constructed at limiting frequencies is derived.

2.5.1 Eigenvalue Decomposition at Fixed Frequency

Consider the characterisation of an object $B_\alpha = \alpha B$ by an MPT (recall the characterisation is independent of z), which can be expressed by the splitting

$$
\begin{aligned}
\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r] &= \mathcal{N}^0[\alpha B, \mu_r] + \mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r] + i\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r], \\
&= \tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r] + i\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r], \quad (2.22)
\end{aligned}
$$

where the interest in this section is for a fixed frequency $0 \leqslant \omega = \Omega < \infty$. For simplicity in this section, the assumption is made that the parameter dependent coefficients of the rank 2 tensors are arranged as 3×3 matrices and use the same notation for both. Thus, the associated matrices $\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]$ are symmetric, $\mathcal{N}^0[\alpha B, \mu_r]$ is positive definite if $\mu_r > 1$, $\mathcal{R}[\alpha B, \Omega, \sigma_*, \mu_r]$ is negative definite and $\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]$ is positive definite and they can be diagonalised as follows

$$
\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r] = \mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]} \boldsymbol{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]} (\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]})^T, \quad (2.23a)
$$

$$
\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r] = \mathbf{Q}^{\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]} \boldsymbol{\Lambda}^{\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]} (\mathbf{Q}^{\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]})^T, \quad (2.23b)
$$

where $\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]}$ is an orthogonal matrix whose columns are the eigenvectors of $\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]$ and $\boldsymbol{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]}$ is a diagonal matrix whose diagonal entries are the eigenvalues of $\tilde{\mathcal{R}}[\alpha B, \Omega, \sigma_*, \mu_r]$ and T denotes the transpose. The matrices $\mathbf{Q}^{\mathcal{I}[\alpha B, \Omega, \sigma_*, \mu_r]}$

and $\mathbf{\Lambda}^{\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]}$ contain the eigenvectors and eigenvalues of $\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]$, respectively. Furthermore, if the object has reflectional or rotational symmetries, the number of independent coefficients in $\tilde{\mathcal{R}}[\alpha B,\Omega,\sigma_*,\mu_r]$ and $\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]$ are reduced, which was discussed in Section 2.3.1. In the case that $\tilde{\mathcal{R}}[\alpha B,\Omega,\sigma_*,\mu_r]$ and $\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]$ have at most 3 independent coefficients, this reduction means that $\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B,\Omega,\sigma_*,\mu_r]} = \mathbf{Q}^{\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]} = \mathbf{Q}(B)$ where the emphasis is made that $\mathbf{Q}$ only depends on B. Moreover, when $\tilde{\mathcal{R}}[\alpha B,\Omega,\sigma_*,\mu_r]$ and $\mathcal{I}[\alpha B,\Omega,\sigma_*,\mu_r]$ are diagonal, due to canonical choice of B and the object's reflectional and rotational symmetries [69], then $\mathbf{Q} = \mathbb{I}$ is the identity matrix.

2.5.2 Equivalent Ellipsoids

For an ellipsoidal object $E_\alpha = \alpha E$ of size α with material parameters μ_r, σ_* and aligned with coordinate axes such that E is defined by

$$\left(\frac{\xi_1}{a}\right)^2 + \left(\frac{\xi_2}{b}\right)^2 + \left(\frac{\xi_3}{c}\right)^2 = 1, \tag{2.24}$$

with $a \geqslant b \geqslant c$ then, for a fixed frequency $\omega = \Omega$, its MPT $\mathcal{M}[\alpha E,\Omega,\sigma_*,\mu_r]$, as well as its real and imaginary parts, are diagonal.

Next, considering the frequencies $\omega = 0$ or $\omega \to \infty$ the work presented in [79] is reviewed, which shows that equivalent ellipsoids $E(0)$ and $E(\infty)$ can be found, that have the same MPT as $\mathcal{M}[\alpha B,0,\sigma_*,\mu_r]$ and $\mathcal{M}[\alpha B,\infty,\sigma_*,\mu_r]$, respectively, a comment is also made on the construction of ellipsoids for other fixed frequencies.

2.5.3 Equivalent Ellipsoid $E(0)$

For the limiting case of $\omega = 0$, as discussed in Section 2.4.1 it is know that

$$\mathcal{M}[\alpha B,\omega = 0,\sigma_*,\mu_r] = \mathcal{N}^0[\alpha B,\mu_r] = \mathcal{T}[\alpha B,\mu_r], \tag{2.25}$$

This tensor is simpler than the MPT and characterises small homogeneous conducting objects with shape B in electrical impedance tomography ($k = \sigma_*/\sigma_0$) and small permeable homogeneous objects with shape B in magnetostatics ($k = \mu_r = \mu_*/\mu_0$), it is symmetric and is positive (negative) definite provided that the contrast $k > 1$ ($0 \leqslant k < 1$) [15]. Next consideration is given to the extent of which $\mathcal{M}[\alpha B,\omega = 0,\sigma_*,\mu_r]$ uniquely determines the object B. Before, presenting the results, recall that for the ellipsoid defined by (2.24) an analytical expression is available for $\mathcal{T}[\alpha E,k]$ in the form

$$\mathcal{T}[\alpha E,k] = \alpha^3 (k-1)|E| \begin{pmatrix} \frac{1}{1-A_1+kA_1} & 0 & 0 \\ 0 & \frac{1}{1-A_2+kA_2} & 0 \\ 0 & 0 & \frac{1}{1-A_3+kA_3} \end{pmatrix}, \tag{2.26}$$

where $|E| := \frac{4}{3}\pi abc$ is the volume of the ellipsoid and A_1, A_2, A_3 are the elliptical integrals

$$A_1 := \frac{bc}{a^2} \int_1^\infty \frac{1}{t^2 \sqrt{t^2 - 1 + \left(\frac{b}{a}\right)^2} \sqrt{t^2 - 1 + \left(\frac{c}{a}\right)^2}} \mathrm{d}t,$$

$$A_2 := \frac{bc}{a^2} \int_1^\infty \frac{1}{\left(t^2 - 1 + \left(\frac{b}{a}\right)^2\right)^{3/2} \sqrt{t^2 - 1 + \left(\frac{c}{a}\right)^2}} \mathrm{d}t,$$

$$A_3 := \frac{bc}{a^2} \int_1^\infty \frac{1}{\sqrt{t^2 - 1 + \left(\frac{b}{a}\right)^2} \left(t^2 - 1 + \left(\frac{c}{a}\right)^2\right)^{3/2}} \mathrm{d}t,$$

that are a function of a, b, c [15]. These integrals can also be shown to be equivalent to the alternative expressions in terms of incomplete elliptic integrals given by Osborn [95], which can be computed using standard libraries and they satisfy $A_1 + A_2 + A_3 = 1$ [95]. From the above, the following results, which were proved by Ledger *et al* in [76], about the MPT characterisation of B at $\omega = 0$ are obtained.

Lemma 2.5.1. *[Ledger [76]] Given α and $0 < \mu_r < \infty$, $\mu_r \neq 1$, an equivalent ellipsoid $E(0)$ for an object B can be found such that*

$$\mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, 0, \sigma_*, \mu_r]} = \tilde{\mathcal{R}}[\alpha E(0), 0, \sigma_*, \mu_r] = \mathcal{T}[\alpha E(0), \mu_r],$$
$$\mathbf{\Lambda}^{\mathcal{I}[\alpha B, 0, \sigma_*, \mu_r]} = \mathcal{I}[\alpha E(0), 0, \sigma_*, \mu_r] = 0,$$

holds and its radii a, b and c can uniquely be determined from $\mathcal{M}[\alpha B, \omega = 0, \sigma_*, \mu_r] = \tilde{\mathcal{R}}[\alpha B, 0, \sigma_*, \mu_r] + i\mathcal{I}[\alpha B, 0, \sigma_*, \mu_r]$.

Corollary 2.5.2 (Ledger [76]). *An important corollary of Lemma 2.5.1 is that $\mathcal{M}[\alpha B, \omega = 0, \sigma_*, \mu_r]$ does not provide a unique object characterisation as there is an equivalent ellipsoid $E(0)$ that has the same MPT.*

Remark 2.5.3 (Ledger [76]). *Proving analytically that the map $(A_2/A_1, A_3/A_1) \to (b/a, c/a)$ is injective is an open question. Nonetheless, the numerical justification provided in Figure 2 of [76] shows that this map is injective in practice.*

Remark 2.5.4 (Ledger [76]). *There are an infinite number of ways to choose $\alpha \ll 1$ and B that result in the same product αB. If $|B|$ is chosen such that $\det(\mathcal{T}[B, \mu_r]) = 1$ then α can be recovered from $\det(\mathcal{T}[\alpha B, \mu_r])$ and Lemma 2.5.1 only requires knowledge of μ_r to determine the equivalent ellipsoid $E(0)$.*

A numerical approach to finding the equivalent ellipsoid has been previously proposed by [64] and involves solving the minimisation problem

$$\min_{\mathbf{u}} \left| \sum_{i=1}^3 (\mathbf{\Lambda}^{\mathcal{N}^0[\alpha B, \mu_r]})_{ii} - (\mathcal{T}[\alpha E(0, \mathbf{u}), \mu_r])_{ii} \right|^2, \tag{2.27}$$

for $\mathbf{u} := (a, b, c)$ so as to find the equivalent ellipsoid $E(0)$. Since $a > 0$, $b > 0$ and $c > 0$ it can be shown that the associated non-linear system is continuous with respect to

the unknown variables, hence the approximate solution of this non-linear system is well-posed [64], but it was unclear if the solution a, b and c exists and is unique. Lemma 2.5.1 addresses this issue and, if desired, the steps in the proof can be followed as alternative approach for determining a, b and c. In practice, applying (2.27) gives the same result as the procedure for the examples presented in Section 2.5.5.1.

2.5.4 Equivalent Ellipsoid $E(\infty)$

Recalling from Section 2.4.2 that for the case of $\nu \to \infty$ which implies either $\sigma_* \to \infty$ or $\omega \to \infty$ it is known that [73]

$$\lim_{\sigma_* \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r] = \mathcal{M}^\infty[\alpha B] \equiv \lim_{\omega \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r], \qquad (2.28)$$

where, importantly, the latter equivalence must be viewed with care as $\omega \to \infty$ would violate the eddy current assumption and, instead, this limit should be viewed as the limiting frequency for which the eddy current model is valid. Again restricting the topology of B to the case that its Betti number $\beta_1(B) = 0$ then [73]

$$\lim_{\sigma_* \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r] = \mathcal{M}^\infty[\alpha B] = \mathcal{T}[\alpha B, 0] \equiv \lim_{\omega \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]. \quad (2.29)$$

The following results, which were proved by Ledger *et al* in [76], on the characterisation of B as $\omega \to \infty$ (up to the limit of the eddy current model) can be proved in an analogous way to Lemma 2.5.1.

Lemma 2.5.5. *[Ledger [76]] Given α, an equivalent ellipsoid $E(\infty)$ independent of μ_r can be found for an object B with $\beta_1(B) = 0$ such that*

$$\lim_{\omega \to \infty} \mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]} = \lim_{\omega \to \infty} \tilde{\mathcal{R}}[\alpha E(\infty), \omega, \sigma_*, \mu_r] = \mathcal{T}[\alpha E(\infty), 0],$$
$$\lim_{\omega \to \infty} \mathbf{\Lambda}^{\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]} = \lim_{\omega \to \infty} \mathcal{I}[\alpha E(\infty), \omega, \sigma_*, \mu_r] = 0,$$

holds (up to the limit of the eddy current model) and its radii a, b and c can uniquely be determined from
$\lim_{\omega \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r] = \lim_{\omega \to \infty} \left(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r] + i\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r] \right).$

Corollary 2.5.6 (Ledger [76]). *In a similar manner to Lemma 2.5.1, Lemma 2.5.5 shows that $\lim_{\omega \to \infty} \mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ (up to the limit of the eddy current model) does not provide a unique object characterisation as there is an equivalent ellipsoid $E(\infty)$ that has the same MPT.*

Remark 2.5.7. *[Ledger [76]] The equivalent ellipsoid $E(\infty)$ is independent of μ_r since $\lim_{\omega \to \infty} \mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]}$ is independent of μ_r [73]. However, in general, $E(0)$ is dependent on μ_r as $\mathcal{M}[\alpha B, \omega = 0, \sigma_*, \mu_r]$ is dependent on μ_r. Additionally, it is only guaranteed that $E(0)$ is independent of μ_r if B is an ellipsoid and in this case $B = E(0) = E(\infty)$.*

2.5.5 Equivalent Ellipsoid $E(\Omega)$

For the non-limiting fixed frequency case $\omega = \Omega$, $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ is non-vanishing and $\boldsymbol{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, 0, \sigma_*, \mu_r]}$ can no longer be expressed in terms of $\mathcal{T}[\alpha E, k]$. However, a semi-analytical solution is available for the MPT of an ellipsoid [45, 19, 17], which would allow an analogous numerical procedure to (2.27) to be applied to find an equivalent ellipsoid. But, given the non-explicit nature of this solution, it is not possible to show existence or uniqueness in this case. Nonetheless, the conjecture is made for $\beta_1(B) = 0$ such an equivalent ellipsoid exists so that the MPT characterisation at a fixed frequency does not uniquely characterise the object's shape and materials. To be able do this alot more data would be required.

2.5.5.1 Equivalent Ellipsoids at Fixed Frequency for an Irregular Tetrahedron

Ledger and Lionheart in [76] discuss how from Lemma 2.5.1 and 2.5.5 can be used to find an equivalent ellipsoid can be found that has the same MPT as the object B at $\omega = 0$ and for $\omega \to \infty$ (up to the limit of the eddy current model). To illustrate that the spectral signature of the MPT for an object contains richer information than the spectral signature of an equivalent ellipsoid obtained at a fixed frequency, a comparison is made of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{M}[\alpha E(0), \omega, \sigma_*, \mu_r]$ in Figure 2.8 using method summarised in Chapter 3 for the case where B is the irregular tetrahedron as described in Section 2.4.3. The resulting equivalent ellipsoid $E(0)$ has $a = 1.4426$, $b = 1.8797$ and $c = 2.4243$ (to 4dp). The observation is made that the eigenvalues of the real and imaginary parts of $\mathcal{M}[\alpha E(0), \omega, \sigma_*, \mu_r]$ agree well with those of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for small ω, but the spectral signature differs for large ω.

$$\lambda_i(\tilde{\mathcal{R}}) \qquad\qquad \lambda_i(\mathcal{I})$$
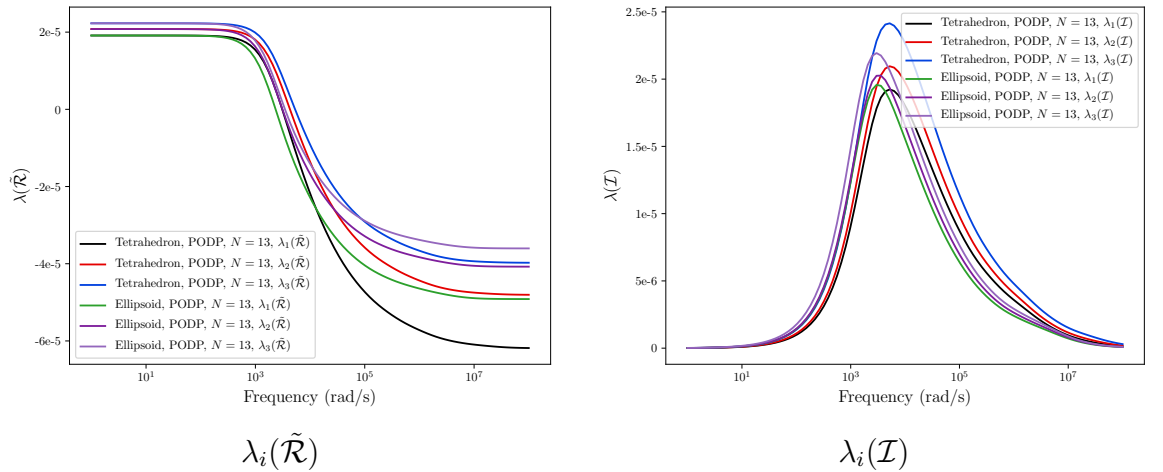
Figure 2.8: Irregular tetrahedron B with vertices as stated in (5.1), $\alpha = 0.01$ m, $\mu_r = 2$ and $\sigma_* = 5.96 \times 10^6$ S/m. Comparison of $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\tilde{\mathcal{R}}[\alpha E(0), \omega, \sigma_*, \mu_r])$ as well as $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha E(0), \omega, \sigma_*, \mu_r])$ using an equivalent ellipsoid $E(0)$.

The corresponding results comparing $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{M}[\alpha E(\infty), \omega, \sigma_*, \mu_r]$ are

shown in Figure 2.9, where the equivalent ellipsoid $E(\infty)$ is defined by $a = 1.3693$, $b = 1.9090$ and $c = 2.9404$ (to 4dp). The observation is made that the eigenvalues of the real and imaginary parts of $\mathcal{M}[\alpha E(\infty), \omega, \sigma_*, \mu_r]$ agree well with those of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for small and large ω, but the spectral signature differs considerably for other ω.

$$\lambda_i(\tilde{\mathcal{R}}) \qquad\qquad \lambda_i(\mathcal{I})$$
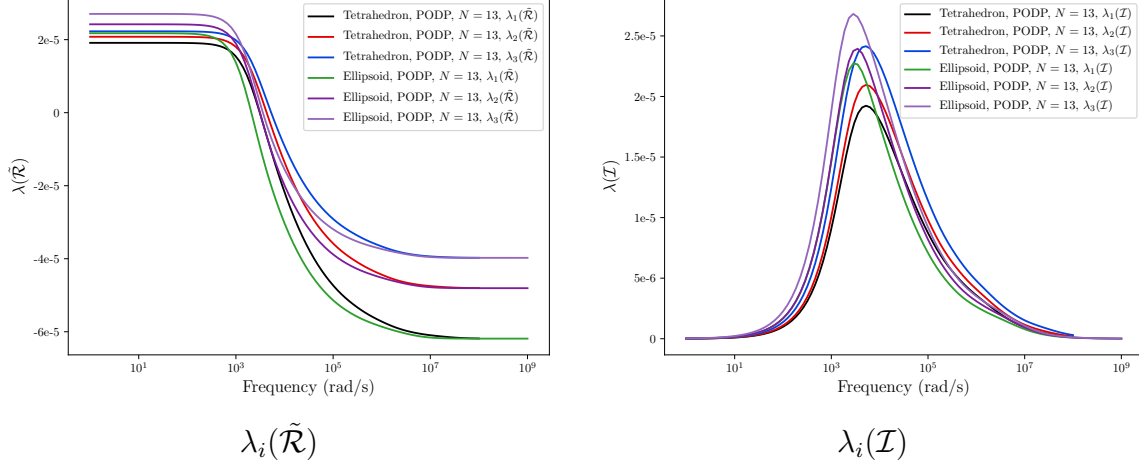
Figure 2.9: Irregular tetrahedron B with vertices as stated in (5.1), $\alpha = 0.01$ m $\mu_r = 2$ and $\sigma_* = 5.96 \times 10^6$ S/m: Comparison of $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\tilde{\mathcal{R}}[\alpha E(\infty), \omega, \sigma_*, \mu_r])$ as well as $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha E(\infty), \omega, \sigma_*, \mu_r])$ using an equivalent ellipsoid $E(\infty)$.

These results motivate the advantages of using spectral MPT signature over using MPT information from a single frequency, since, rather than a single equivalent ellipsoid, there are multiple equivalent ellipsoids corresponding to different ω providing greater information for the classification. In the next section further theoretical insights in to the object characterisation using spectral data is provided.

2.6 MPT Object Characterisation from Spectral Data

In this section, an examination into the improvements to MPT object characterisation given spectral data. Firstly, eigenvalue decompositions of the real and imaginary parts of the MPT for spectral data are considered and then, secondly, the arguments presented by Ledger and Lionheart in [76] for a theoretical justification for the observations seen in Figures 2.8 and 2.9 are recalled.

2.6.1 Eigenvalue Decomposition in the Spectral Case

Consider the case of the characterisation of an object by an MPT with varying frequency ω so that the real and imaginary parts of an MPT expressed by (2.22) are available continuously as a function of ω. A similar decomposition to (2.23) again applies, except $\mathbf{Q}^{\tilde{\mathcal{R}}}$, $\mathbf{\Lambda}^{\tilde{\mathcal{R}}}$, $\mathbf{Q}^{\mathcal{I}}$ and $\mathbf{\Lambda}^{\mathcal{I}}$ are functions of ω. If B has reflectional or rotational

symmetries, $\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}^{\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}(B)$ and, in the limiting case where $\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]$ are diagonal, $\mathbf{Q} = \mathbb{I}$ is the identity matrix. If $\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}^{\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}(B)$, the only dependence of the MPT's coefficients on α, ω, μ_r and σ_* is through $\mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]}$ and $\mathbf{\Lambda}^{\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]}$.

2.6.2 Understanding the Spectral Signature of MPTs

Using the results above, an explanation of the results observed in Figures 2.8 and 2.9 is given in the following results which were proved by Ledger *et al* in [76].

Lemma 2.6.1. *[Ledger [76]] If B is not an ellipsoid, the coefficients of $\mathcal{M}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{M}[\alpha E(0),\omega,\sigma_*,\mu_r]$ are different away from $\omega = 0$.*

Lemma 2.6.2. *[Ledger [76]] If B is not an ellipsoid, the coefficients of $\mathcal{M}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{M}[\alpha E(\infty),\omega,\sigma_*,\mu_r]$ are different away from $\omega \to \infty$ (upto the limit of the eddy current model).*

Remark 2.6.3 (Ledger [76])**.** *Provided that B is not an ellipsoid, Lemma 2.6.1 has shown that the coefficients of $\mathcal{M}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{M}[\alpha E(0),\omega,\sigma_*,\mu_r]$ are different away from $\omega = 0$ and Lemma 2.6.2 has shown that the coefficients of $\mathcal{M}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{M}[\alpha E(\infty),\omega,\sigma_*,\mu_r]$ are different away from $\omega \to \infty$ (up to the limit of the eddy current model). Furthermore, as discussed in Remark 2.5.7, unless B is an ellipsoid, $E(0) \neq E(\infty)$. Indeed, the conjecture is made that if the the real and imaginary parts of an MPT expressed by (2.22) are available at discrete frequencies $0 \leqslant \omega_m < \infty$, $m = 1\ldots,M$ then there are M different equivalent ellipsoids $E(\omega_n)$, $n = 1,\ldots,M$. Thus, the MPT spectral signature provides considerable advantages over MPT characterisation at a single fixed frequency and, hence, forms the basis of object characterisation strategy for the rest of the thesis.*

2.7 Scaling of the MPT Under Parameter Changes

Two results that aid the computation of the frequency sweep of an MPT for an object with scaled conductivity and an object with a scaled object size from an already known frequency sweep of the MPT for the same shaped object were derived by Ledger in [133] and a novel implementation of these results has been made.

Lemma 2.7.1. *[Ledger [133]] Given the MPT coefficients for an object αB with material parameters μ_r and σ_* at frequency $s\omega$, the coefficients of the MPT for an object, which has the same B, α and μ_r, but with conductivity $s\sigma_*$, at frequency ω, are given by*

$$(\mathcal{M}[\alpha B,\omega,s\sigma_*,\mu_r])_{ij} = (\mathcal{M}[\alpha B,s\omega,\sigma_*,\mu_r])_{ij}, \tag{2.30}$$

where $(\mathcal{M}[\alpha B,s\omega,\sigma_*,\mu_r])_{ij}$ denote the coefficients of the original MPT at frequency $s\omega$.

Lemma 2.7.2. *[Ledger [133]] Given the MPT coefficients for an object αB with material parameters μ_r and σ_* at frequency $s^2\omega$, the coefficients of the MPT for an object $s\alpha B$, which is the same as B apart from having size $s\alpha$, at frequency ω, are given by*

$$(\mathcal{M}[s\alpha B, \omega, \sigma_*, \mu_r])_{ij} = s^3(\mathcal{M}[\alpha B, s^2\omega, \sigma_*, \mu_r])_{ij}, \tag{2.31}$$

where $(\mathcal{M}[\alpha B, s^2\omega, \sigma_*, \mu_r])_{ij}$ denote the coefficients of the original MPT at frequency $s^2\omega$.

2.8 Chapter Summary

This chapter presented material designed to set the scene for subsequent chapters covering topics such as the formulation of the time harmonic eddy current model, basic metal detector designs and methodology, the difference between the forward and inverse problems, the asymptotic expansion and MPT. The chapter also gives an overview of the MPT and discussing limiting cases which provide a fundamental insight into the behaviour of the MPT and highlight the important role object topology plays. Lastly, the chapter provides motivation for the use of the MPT spectral signature as opposed to the MPT at a single frequency.

The main novelties of the chapter are the collation of recent results that have been proved by Ledger and Lionheart in [69, 70, 71] on the characterisation of objects by MPTs that will be relevant for the remaining chapters of the thesis. These include their real-imaginary split symmetric form of the MPT presented in (2.5), their approach to finding limiting frequency of the eddy-current model a numerical implementation is employed in Chapters 3 and 4, their investigation for MPT object characterisations at fixed frequencies which documents a novel method for determining the best fitting ellipsoid for a given object at a given frequency, with an example to an irregular tetrahedron and finally Ledger's scaling of the MPT under the parameters of object size and conductivity. A novel implementation of Ledger's scaling result has been made and results of this are presented in Section 5.6.

Chapter 3

Finite Element Discretisation and the Reduced Order Model

3.1 Introductory Remarks

This chapter focuses on the development and implementation of a ROM, for the rapid computation of the MPT spectral signatures. This involves reducing the computational cost associated with solving (2.7) for different ω. The ROM benefits from a-posteriori error estimates, which provide an upper bound on the error in the MPT coefficients obtained with the reduced order model compared to those obtained from a full order solution. For the full order solution, a hp-FE discretisation is used of the weak forms associated with the transmission problems (2.6) and (2.7).

The chapter is based on an extended version of the work presented by the author in [133]. The main novelties of the chapter are the implementation of regularised weak forms of (2.6) and (2.7), using the open source FE library `NGSolve` for the efficient solutions of these full order problems, the application of projection based proper orthogonal decomposition (PODP) to generate a ROM and the implementation of a-posterori error estimates to certify the MPT coefficients obtained with the ROM.

The structure of the chapter is as follows: First in Section 3.2.1, the weak forms of the problems are discussed, also stating the FE discretisation implemented in Section 3.2.2. In Section 3.3 a ROM based on PODP is derived. Finally, in Section 3.3.3 a-posteriori error estimates for the MPT coefficients obtained by the ROM are derived. The chapter finishes with concluding remarks in Section 3.4.

3.2 Full Order Model

To approximate the solutions to the transmission problems (2.6) and (2.7) the unbounded domain B^c is truncated at a finite distance from the object B and create a bounded domain Ω containing B. On $\partial\Omega$, the decay conditions (2.6e) and (2.7f) are approximated by $\boldsymbol{n} \times \tilde{\boldsymbol{\theta}}_i^{(0)} = \boldsymbol{n} \times (\boldsymbol{\theta}_i^{(0)} - \boldsymbol{e}_i \times \boldsymbol{\xi}) = \boldsymbol{0}$ and $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(1)} = \boldsymbol{0}$, respectively. On this finite domain, the associated weak variational statements to these problems are approximated

using FEM with a $\boldsymbol{H}(\mathrm{curl})$ conforming discretisation with mesh spacing h and order elements p where

$$\boldsymbol{H}(\mathrm{curl}) := \left\{\boldsymbol{u} : \boldsymbol{u} \in (L^2(\Omega))^3, \ \nabla \times \boldsymbol{u} \in (L^2(\Omega))^3\right\}, \qquad (3.1)$$

and $L^2(\Omega)$ denotes the standard space of square integrable functions. In Section 3.2.1 their weak formulations are provided, their discretisation are provided in Section 3.2.2. It is clear that the eddy current model (2.1), the asymptotic formula (2.4), the domain truncation to form Ω and the steps that follow to form a discrete FE model all involve approximations. However, the goal is to accurately and efficiently obtain the MPT coefficients in (2.5) and, to distinguish the coefficients obtained using FEM from those obtained from the further reduced ROM, the FEM approximation is referred to as the full order model.

3.2.1 Weak Formulation of the Problem

Following the approach advocated in [77] for magnetostatic and eddy current problems, a regularisation term $\varepsilon \int_\Omega \tilde{\boldsymbol{\theta}}_i^{(0)} \cdot \boldsymbol{\psi} \mathrm{d}\boldsymbol{\xi}$ is added, where ε is a small regularisation parameter, to the weak variational statement of (2.6), written in terms of $\tilde{\boldsymbol{\theta}}_i^{(0)}$, in order to circumvent the Coulomb gauge $\nabla \cdot \tilde{\boldsymbol{\theta}}_i^{(0)} = 0$. For details of the small error induced by this approximation see [77, 140]. Then, choosing an appropriate set of $\boldsymbol{H}(\mathrm{curl})$ conforming FE functions in $W^{(hp)} \subset \boldsymbol{H}(\mathrm{curl})$, the following discrete regularised weak form for (2.6) has been obtained by Ledger [68]: Find real solutions $\tilde{\boldsymbol{\theta}}_i^{(0,hp)} \in Y \cap W^{(hp)}$ such that

$$\int_\Omega \tilde{\mu}_r^{-1} \nabla \times \tilde{\boldsymbol{\theta}}_i^{(0,hp)} \cdot \nabla \times \boldsymbol{\psi}^{(hp)} \mathrm{d}\boldsymbol{\xi} + \varepsilon \int_\Omega \tilde{\boldsymbol{\theta}}_i^{(0,hp)} \cdot \boldsymbol{\psi}^{(hp)} \mathrm{d}\boldsymbol{\xi}$$
$$= 2 \int_B (1 - \mu_r^{-1}) \boldsymbol{e}_i \cdot \nabla \times \boldsymbol{\psi}^{(hp)} \mathrm{d}\boldsymbol{\xi}, \qquad (3.2)$$

for all $\boldsymbol{\psi}^{(hp)} \in Y \cap W^{(hp)}$, where

$$Y := \left\{\boldsymbol{u} \in \boldsymbol{H}(\mathrm{curl}) : \boldsymbol{n} \times \boldsymbol{u} = \boldsymbol{0} \text{ on } \partial\Omega\right\}.$$

In a similar manner, the discrete weak variational statement of (2.7) has been obtained by Ledger [68]: Find complex solutions $\boldsymbol{\theta}_i^{(1,hp)} \in Y \cap W^{(hp)}$ such that

$$\int_\Omega \left(\mu_r^{-1} \nabla \times \boldsymbol{\theta}_i^{(1,hp)}\right) \cdot \left(\nabla \times \overline{\boldsymbol{\psi}^{(hp)}}\right) \mathrm{d}\boldsymbol{\xi} - \mathrm{i} \int_B \nu \boldsymbol{\theta}_i^{(1,hp)} \cdot \overline{\boldsymbol{\psi}^{(hp)}} \mathrm{d}\boldsymbol{\xi}$$
$$+ \varepsilon \int_{\Omega \backslash B} \boldsymbol{\theta}_i^{(1,hp)} \cdot \overline{\boldsymbol{\psi}^{(hp)}} \mathrm{d}\boldsymbol{\xi} = \mathrm{i} \int_B \nu \boldsymbol{\theta}_i^{(0,hp)} \cdot \overline{\boldsymbol{\psi}^{(hp)}} \mathrm{d}\boldsymbol{\xi}, \qquad (3.3)$$

for all $\boldsymbol{\psi}^{(hp)} \in Y \cap W^{(hp)}$.

For what follows it is beneficial to restate (3.3) in the following form: Find $\boldsymbol{\theta}_i^{(1,hp)} \in Y \cap W^{(hp)}$ such that

$$a\left(\boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)}; \boldsymbol{\omega}\right) = r\left(\boldsymbol{\psi}^{(hp)}; \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\omega}\right), \qquad (3.4)$$

for all $\boldsymbol{\psi}^{(hp)} \in Y \cap W^{(hp)}$ where

$$a\big(\boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)}; \boldsymbol{\omega}\big) := \Big\langle \tilde{\mu}^{-1} \nabla \times \boldsymbol{\theta}_i^{(1,hp)}, \nabla \times \boldsymbol{\psi}^{(hp)} \Big\rangle_{L^2(\Omega)}$$

$$- \mathrm{i} \Big\langle \nu \boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)} \Big\rangle_{L^2(B)}$$

$$+ \varepsilon \Big\langle \boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)} \Big\rangle_{L^2(\Omega \backslash B)}, \tag{3.5a}$$

$$r\big(\boldsymbol{\psi}^{(hp)}; \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\omega}\big) := \mathrm{i} \Big\langle \nu \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\psi}^{(hp)} \Big\rangle_{L^2(B)}, \tag{3.5b}$$

$\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{L^2(\Omega)} := \int_\Omega \boldsymbol{u} \cdot \overline{\boldsymbol{v}} \mathrm{d}\boldsymbol{\xi}$ denotes the L^2 inner product over Ω and $\boldsymbol{\omega}$ indicates the list of the problem parameters $(\omega, \sigma_*, \mu_r, \alpha)$ that one might wish to vary. Note that $r\big(\boldsymbol{\psi}^{(hp)}; \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\omega}\big)$ is a function of μ_r as $\boldsymbol{\theta}_i^{(0,hp)}$ depends on μ_r.

3.2.2 Finite Element Discretisation

For the implementation of (3.2) and (3.4), NGSolve [4, 109, 108, 140] is used along with the hierarchic set of $\boldsymbol{H}(\mathrm{curl})$ conforming basis functions proposed by Schöberl and Zaglmayr [110], which are available in this software. Although there are many available choices for software, such as deal.ii, FENicS and COMSOL, the choice to use NGSolve is made for several key factors: the ease of use provided by the python interface; the unstructured tetrahedral meshes; the efficient implementation of hierarchical p-version $H(\mathrm{curl})$ conforming elements; and the availability of robust preconditioners which benefit from parallelisation. This choice is further discussed in Chapter 4. In the following, for simplicity, focus is given to the treatment of $\boldsymbol{\theta}_i^{(1,hp)}$ and drop the index i as each direction can be computed in a similar way (as can $\tilde{\boldsymbol{\theta}}_i^{(0,hp)}$). The basis functions are denoted by $\boldsymbol{N}^{(k)}(\boldsymbol{\xi}) \in W^{(hp)}$ leading to the expression of the solution function along with the weighting functions

$$\boldsymbol{\theta}^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) := \sum_{k=1}^{N_d} \boldsymbol{N}^{(k)}(\boldsymbol{\xi}) \mathrm{q}_k(\boldsymbol{\omega}), \tag{3.6a}$$

$$\boldsymbol{\psi}^{(hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) := \sum_{k=1}^{N_d} \boldsymbol{N}^{(k)}(\boldsymbol{\xi}) \mathrm{l}_k(\boldsymbol{\omega}), \tag{3.6b}$$

where N_d is the number of degrees of freedom. Here, and in the following, the bold italic font denotes a vector field and the bold non-italic Roman font represents a matrix (upper case) or column vector (lower case). With this distinction, (3.6) is rewritten in matrix form as

$$\boldsymbol{\theta}^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) = \mathbf{N}(\boldsymbol{\xi})\mathbf{q}(\boldsymbol{\omega}), \tag{3.7a}$$

$$\boldsymbol{\psi}^{(hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) = \mathbf{N}(\boldsymbol{\xi})\mathbf{l}(\boldsymbol{\omega}), \tag{3.7b}$$

where $\mathbf{N}(\boldsymbol{\xi})$ is the matrix constructed with the basis vectors $\boldsymbol{N}^k(\boldsymbol{\xi})$ as its columns, i.e.

$$\mathbf{N}(\boldsymbol{\xi}) := \big[\boldsymbol{N}^{(1)}(\boldsymbol{\xi}), \boldsymbol{N}^{(2)}(\boldsymbol{\xi}), ..., \boldsymbol{N}^{(N_d)}(\boldsymbol{\xi})\big].$$

With this, (3.4) is also rewritten as follows

$$\sum_{i=1}^{N_d} \sum_{j=1}^{N_d} \overline{l_i(\boldsymbol{\omega})} a\big(\boldsymbol{N}^{(j)}(\boldsymbol{\xi}), \boldsymbol{N}^{(i)}(\boldsymbol{\xi}); \boldsymbol{\omega}\big) q_j(\boldsymbol{\omega}) = \sum_{i=1}^{N_d} \overline{l_i(\boldsymbol{\omega})} r\big(\boldsymbol{N}^{(i)}(\boldsymbol{\xi}); \boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}\big), \quad (3.8)$$

and, with a suitable choice of $l_i(\boldsymbol{\omega})$, (3.8) is rewritten as the linear system of equations

$$\mathbf{A}(\boldsymbol{\omega})\mathbf{q}(\boldsymbol{\omega}) = \mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}), \quad (3.9)$$

where the coefficients of $\mathbf{A}(\boldsymbol{\omega})$ and $\mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega})$ are defined to be

$$(\mathbf{A}(\boldsymbol{\omega}))_{ij} := a\big(\boldsymbol{N}^{(j)}(\boldsymbol{\xi}), \boldsymbol{N}^{(i)}(\boldsymbol{\xi}); \boldsymbol{\omega}\big), \quad (3.10a)$$

$$(\mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}))_i := r\big(\boldsymbol{N}^{(i)}(\boldsymbol{\xi}); \boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}\big). \quad (3.10b)$$

`NGSolve` offers efficient approaches for the computational solution to (3.9) using pre-conditioned iterative solvers [140, 77]. The combination of Balancing Domain Decomposition by Constraints (BDDC) preconditioner [84] along with the conjugate gradients solver was found to work well for the problem, with (3.9) being a complex symmetric matrix which is positive definite. Following the solution of (3.9), $\boldsymbol{\theta}^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega})$ is obtained using (3.7) and, by repeating the process for $i = 1, 2, 3$, $\boldsymbol{\theta}_i^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega})$ is obtained. Then $(\mathcal{M}[\alpha B, \omega])_{ij}$, for the full order model, is found by using (2.5).

3.3   Reduced Order Model (ROM)

A traditional (full order model) approach for the computation of the MPT spectral signature, i.e. the variation of the coefficients of $\mathcal{M}[\alpha B, \omega]$ with frequency, would involve the repeated solution of the N_d sized system (3.9) for different ω. To reduce the computational cost of this, it is desirable to apply a ROM in which the solution of (3.9) is replaced by a surrogate problem of reduced size. Thus, reducing both the computation cost and time to produce a solution for each new ω. In particular, in Section 3.3.1, a ROM based on the POD method [26, 8, 55, 112] is described and, in Section 3.3.2, the variant called projection based POD (denoted by PODP) is applied, this has already been shown to outperform interpolation based POD in the analysis of an eddy current problem as part of magneto-mechanical coupling applied to MRI scanners [112], hence, we only consider the prior. To emphasise the generality of the approach, the formulation is presented for an arbitrary list of problem parameters denoted by $\boldsymbol{\omega}$. In Section 3.3.3 a procedure for computing certificates of accuracy on the ROM solutions with negligible additional cost is derived.

3.3.1   Proper Orthogonal Decomposition

Following the solution of (3.9) for $\mathbf{q}(\boldsymbol{\omega})$ for different values of the set of parameters, $\boldsymbol{\omega}$, a matrix $\mathbf{D} \in \mathbb{C}^{N_d \times N}$ is constructed, with the vector of solution coefficients as its columns in the form

$$\mathbf{D} := \big[\mathbf{q}(\boldsymbol{\omega}_1), \mathbf{q}(\boldsymbol{\omega}_2), ..., \mathbf{q}(\boldsymbol{\omega}_N)\big]. \quad (3.11)$$

Note that $\mathbf{q}(\boldsymbol{\omega}_i)$ denotes the vector of coefficients, which, when combined with (3.7a), produces a representative full order model solution snapshot for the ith set of parameters $\boldsymbol{\omega}_i$ and $N \ll N_d$ denotes the number of such snapshots. Application of a singular value decomposition (SVD) e.g. [21, 52, 42] gives

$$\mathbf{D} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H, \tag{3.12}$$

where $\mathbf{U} \in \mathbb{C}^{N_d \times N_d}$ and $\mathbf{V} \in \mathbb{C}^{N \times N}$ are unitary matrices and $\boldsymbol{\Sigma} \in \mathbb{R}^{N_d \times N}$ is a diagonal matrix enlarged by zeros so that it becomes rectangular. In the above, $\mathbf{V}^H = \overline{\mathbf{V}}^T$ is the Hermitian of $\mathbf{V}$.

The diagonal entries $(\boldsymbol{\Sigma})_{ii} = \sigma_i$ [1] are the singular values of $\mathbf{D}$ and they are arranged as $\sigma_1 > \sigma_2 > ... > \sigma_N$. Based on the sparse representation of the solutions to (2.7) as function of ν, and hence ω, (and hence also the sparse representation of the MPT) found in [73], these are expected to decay rapidly towards zero, which motivates the introduction of a truncated singular value decomposition (TSVD) e.g. [21, 52]

$$\mathbf{D} \approx \mathbf{D}^M = \mathbf{U}^M \boldsymbol{\Sigma}^M (\mathbf{V}^M)^H, \tag{3.13}$$

where $\mathbf{U}^M \in \mathbb{C}^{N_d \times M}$ are the first M columns of $\mathbf{U}$, $\boldsymbol{\Sigma}^M \in \mathbb{R}^{M \times M}$ is a diagonal matrix containing the first M singular values and $(\mathbf{V}^M)^H \in \mathbb{C}^{M \times N}$ are the first M rows of $\mathbf{V}^H$. The computation of (3.13) constitutes the off-line stage of the POD. Using (3.13) it is possible to recover an approximate representation for each of the solution snapshots as follows

$$\mathbf{q}(\boldsymbol{\omega}_j) \approx \mathbf{U}^M \boldsymbol{\Sigma}^M ((\mathbf{V}^M)^H)_j, \tag{3.14}$$

where $((\mathbf{V}^M)^H)_j$ refers to the jth column of $(\mathbf{V}^M)^H$.

3.3.2 Projection Based Proper Orthogonal Decomposition (PODP)

Two common POD based methods are interpolation based proper orthogonal decomposition and projection based proper orthogonal decomposition. Implementations for both methods were considered due to the work presented by Seoane [112] where both methods were considered to tackle a related eddy current problem. However, after some initial investigations were conducted, PODP was found to work better. In the online stage of PODP, $\mathbf{q}^{PODP}(\boldsymbol{\omega}) \approx \mathbf{q}(\boldsymbol{\omega})$ is obtained by taking a linear combination of the columns of $\mathbf{U}^M$ where the coefficients of this projection are contained in the vector $\mathbf{p}^M(\boldsymbol{\omega})$. An approximation of $\mathbf{l}(\boldsymbol{\omega})$ is produced in a similar way so that

$$\boldsymbol{\theta}^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) \approx (\boldsymbol{\theta}^{(1,hp)})^{\mathrm{PODP}}(\boldsymbol{\xi}, \boldsymbol{\omega}) := \mathbf{N}(\boldsymbol{\xi})\mathbf{q}^{PODP}(\boldsymbol{\omega}) = \mathbf{N}(\boldsymbol{\xi})\mathbf{U}^M\mathbf{p}^M(\boldsymbol{\omega}) \in Y^{(PODP)}, \tag{3.15a}$$

$$\boldsymbol{\psi}^{(hp)}(\boldsymbol{\xi}, \boldsymbol{\omega}) \approx (\boldsymbol{\psi}^{(1,hp)})^{\mathrm{PODP}}(\boldsymbol{\xi}, \boldsymbol{\omega}) := \mathbf{N}(\boldsymbol{\xi})\mathbf{l}^{PODP}(\boldsymbol{\omega}) = \mathbf{N}(\boldsymbol{\xi})\mathbf{U}^M\mathbf{o}^M(\boldsymbol{\omega}) \in Y^{(PODP)}, \tag{3.15b}$$

---

[1]Note that σ_* is used for conductivity and σ_i for a singular value, however, it should be clear from the application as to which definition applies

where $\in Y^{(PODP)} \subset Y \cap W^{(hp)}$. Substituting these lower dimensional representations in to (3.8) the following is obtained

$$\sum_{i=1}^{M}\sum_{j=1}^{M} \overline{o_i^M(\boldsymbol{\omega})} a\big(\boldsymbol{N}^{(j)}(\boldsymbol{\xi})(\mathbf{U}^M)_j, \boldsymbol{N}^{(i)}(\boldsymbol{\xi})(\mathbf{U}^M)_i; \boldsymbol{\omega}\big) p_j^M(\boldsymbol{\omega})$$

$$= \sum_{i=1}^{M} \overline{o_i^M(\boldsymbol{\omega})} r\big(\boldsymbol{N}^{(i)}(\boldsymbol{\xi})(\mathbf{U}^M)_i; \boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}\big),$$

$$(\mathbf{o}^M(\boldsymbol{\omega}))^H ((\mathbf{U}^M)^H \mathbf{A}(\boldsymbol{\omega})\mathbf{U}^M) \mathbf{p}^M(\boldsymbol{\omega}) = (\mathbf{o}^M(\boldsymbol{\omega}))^H (\mathbf{U}^M)^H \mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}). \qquad (3.16)$$

Then, choosing $\mathbf{o}^M(\boldsymbol{\omega})$ appropriately, to obtain the linear system

$$\mathbf{A}^M(\boldsymbol{\omega}) \mathbf{p}^M(\boldsymbol{\omega}) = \mathbf{r}^M(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}), \qquad (3.17)$$

which is of size $M \times M$ where $\mathbf{A}^M(\boldsymbol{\omega}) := (\mathbf{U}^M)^H \mathbf{A}(\boldsymbol{\omega})\mathbf{U}^M$ and $\mathbf{r}^M(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega}) := (\mathbf{U}^M)^H \mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega})$. Note, since $M < N \ll N_d$, this is significantly smaller than (3.9) and, therefore, substantially computationally cheaper to solve. After solving this reduced system, and obtaining $\mathbf{p}^M(\boldsymbol{\omega})$, an approximate solution for $\boldsymbol{\theta}^{(1,hp)}(\boldsymbol{\xi}, \boldsymbol{\omega})$ is obtained using (3.15).

Focusing on the particular case where $\boldsymbol{\omega} = \omega$, it is observed that $\mathbf{A}$ and $\mathbf{r}$ can be expressed as the simple sums

$$\mathbf{A}(\omega) = \mathbf{A}^{(0)} + \omega \mathbf{A}^{(1)},$$
$$\mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \omega) = \omega \mathbf{r}^{(1)}(\boldsymbol{\theta}^{(0,hp)}),$$

where the definitions of $\mathbf{A}^{(0)}$, $\mathbf{A}^{(1)}$ and $\mathbf{r}^{(1)}(\boldsymbol{\theta}^{(0,hp)})$ are defined as,

$$(\mathbf{A}^{(0)})_{ij} := a^{(0)}\big(\boldsymbol{N}^{(j)}(\boldsymbol{\xi}), \boldsymbol{N}^{(i)}(\boldsymbol{\xi})\big),$$
$$(\mathbf{A}^{(1)})_{ij} := a^{(1)}\big(\boldsymbol{N}^{(j)}(\boldsymbol{\xi}), \boldsymbol{N}^{(i)}(\boldsymbol{\xi})\big),$$
$$(\mathbf{r}^{(1)}(\boldsymbol{\theta}^{(0,hp)}))_i := r^{(1)}\big(\boldsymbol{N}^{(i)}(\boldsymbol{\xi}); \boldsymbol{\theta}^{(0,hp)}\big),$$

where $a^{(0)}$, $a^{(1)}$ and $r^{(1)}$ are defined as,

$$a^{(0)}\big(\boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)}; \boldsymbol{\omega}\big) := \left\langle \tilde{\mu}^{-1}\nabla \times \boldsymbol{\theta}_i^{(1,hp)}, \nabla \times \boldsymbol{\psi}^{(hp)} \right\rangle_{L^2(\Omega)}$$
$$+ \varepsilon \left\langle \boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)} \right\rangle_{L^2(\Omega \setminus B)},$$
$$a^{(1)}\big(\boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)}; \boldsymbol{\omega}\big) := -\mathrm{i} \left\langle \alpha^2 \mu_0 \sigma_* \boldsymbol{\theta}_i^{(1,hp)}, \boldsymbol{\psi}^{(hp)} \right\rangle_{L^2(B)},$$
$$r^{(1)}\big(\boldsymbol{\psi}^{(hp)}; \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\omega}\big) := \mathrm{i} \left\langle \alpha^2 \mu_0 \sigma_* \boldsymbol{\theta}_i^{(0,hp)}, \boldsymbol{\psi}^{(hp)} \right\rangle_{L^2(B)}.$$

Then, by computing and storing $(\mathbf{U}^M)^H \mathbf{A}^{(0)} \mathbf{U}^M$, $(\mathbf{U}^M)^H \mathbf{A}^{(1)} \mathbf{U}^M$, $(\mathbf{U}^M)^H \mathbf{r}^{(1)}(\boldsymbol{\theta}^{(0,hp)})$, which are independent of ω, it follows that $\mathbf{A}^M(\omega)$ and $\mathbf{r}^M(\boldsymbol{\theta}^{(0,hp)}, \omega)$ can be efficiently calculated for each new ω from the stored data. In a similar manner, by precomputing appropriate data, the MPT coefficients in (2.5) can also be rapidly evaluated for each new ω using the PODP solutions. This leads to further considerable computational savings. Emphasis is placed on the fact that the PODP is only applied to obtain ROM solutions for $\boldsymbol{\theta}^{(1)}(\boldsymbol{\xi}, \omega)$ and not to $\boldsymbol{\theta}^{(0)}(\boldsymbol{\xi})$, which does not depend on ω.

3.3.3 An a-posterori Error Estimate to Certify the PODP Output

Ledger [133] has derived an a-posterori error estimate on the MPT coefficients obtained with PODP, with respect to those obtained with full order model, as a function of ω. His approach builds on the procedure described in [55] for obtaining a-posteriori error estimates on outputs of interest from ROM solutions and the approach is summarised here. Importantly, this estimate can be computed at negligible additional cost during the online stage of PODP allowing it to be used to certify the MPT coefficients obtained using the ROM and to check their accuracy is within acceptable limits. To do this, he sets $\boldsymbol{\epsilon}_i(\omega) = \boldsymbol{\theta}_i^{(1,hp)}(\omega) - (\boldsymbol{\theta}_i^{(1,hp)})^{\text{PODP}}(\omega) \in Y^{(hp)}$, where the subscript i has been reintroduced, as the cases $i = 1, 2, 3$ need to be distinguished between. Although $\boldsymbol{\epsilon}_i$ also depends on $\boldsymbol{\xi}$, it has been chosen here, and in the following, to only emphasise its dependence on ω. Introducing $Y^{(hp)} = Y \cap W^{(hp)}$ for simplicity of notation, and noting that this error satisfies

$$a(\boldsymbol{\epsilon}_i(\omega), \boldsymbol{\psi}; \omega) = r(\boldsymbol{\psi}; \boldsymbol{\theta}_i^{(0,hp)}, \omega) \qquad \forall \boldsymbol{\psi} \in Y^{(hp)}, \tag{3.19}$$

which is called the error equation [55] and

$$a(\boldsymbol{\epsilon}_i(\omega), \boldsymbol{\psi}; \omega) = 0 \qquad \forall \boldsymbol{\psi} \in Y^{(PODP)}, \tag{3.20}$$

which is called Galerkin orthogonality [55]. The Riesz representation [55] of $r(\cdot; \boldsymbol{\theta}_i^{(0,hp)}, \omega)$ denoted by $\hat{\boldsymbol{r}}_i(\omega) \in Y^{(hp)}$ is such that

$$(\hat{\boldsymbol{r}}_i(\omega), \boldsymbol{\psi})_{Y^{(hp)}} = r(\boldsymbol{\psi}; \boldsymbol{\theta}_i^{(0,hp)}, \omega) \qquad \forall \boldsymbol{\psi} \in Y^{(hp)}, \tag{3.21}$$

so that

$$a(\boldsymbol{\epsilon}_i(\omega), \boldsymbol{\psi}; \omega) = (\hat{\boldsymbol{r}}_i(\omega), \boldsymbol{\psi})_{Y^{(hp)}} \qquad \forall \boldsymbol{\psi} \in Y^{(hp)}. \tag{3.22}$$

Then, by using the alternative set of formulae for the tensor coefficients [73]

$$\begin{aligned}
(\mathcal{R}[\alpha B, \omega])_{ij} &= -\frac{\alpha^3}{4} \int_B \nu \text{Im}(\boldsymbol{\theta}_j^{(1,hp)}) \cdot \boldsymbol{\theta}_i^{(0,hp)} \mathrm{d}\boldsymbol{\xi} \\
&= -\frac{\alpha^3}{4} \left\langle \nu \text{Im}(\boldsymbol{\theta}_j^{(1,hp)}), \boldsymbol{\theta}_i^{(0,hp)} \right\rangle_{L^2(B)}, \tag{3.23a}
\end{aligned}$$

$$\begin{aligned}
(\mathcal{I}[\alpha B, \omega])_{ij} &= \frac{\alpha^3}{4} \left(\int_B \nu \text{Re}(\boldsymbol{\theta}_j^{(1,hp)}) \cdot \boldsymbol{\theta}_i^{(0,hp)} \mathrm{d}\boldsymbol{\xi} + \int_B \nu \boldsymbol{\theta}_j^{(0,hp)} \cdot \boldsymbol{\theta}_i^{(0,hp)} \mathrm{d}\boldsymbol{\xi} \right) \\
&= \frac{\alpha^3}{4} \left(\left\langle \nu \text{Re}(\boldsymbol{\theta}_j^{(1,hp)}), \boldsymbol{\theta}_i^{(0,hp)} \right\rangle_{L^2(B)} + \left\langle \nu \boldsymbol{\theta}_j^{(0,hp)}, \boldsymbol{\theta}_i^{(0,hp)} \right\rangle_{L^2(B)} \right), \tag{3.23b}
\end{aligned}$$

which are written in terms of the full order solutions, Ledger's a-posteriori error estimate for the tensor entries computed using PODP is stated in the lemma below which was proved by Ledger in [133]. Note that the formulae stated in (2.5) are used for the actual POD computation of $(\mathcal{R}^{PODP}[\alpha B, \omega])_{ij}$ and $(\mathcal{I}^{PODP}[\alpha B, \omega])_{ij}$, but the form in (3.23) is useful for obtaining the error estimate. Also, as $(\mathcal{N}[\alpha B])_{ij}$ is independent of ω the equality $(\mathcal{N}^{0,PODP}[\alpha B])_{ij} = (\mathcal{N}^0[\alpha B])_{ij}$ holds, leading to the equality $\mathcal{M}^{PODP}[\alpha B, \omega] = \mathcal{N}^{0,PODP}[\alpha B] + \mathcal{R}^{PODP}[\alpha B, \omega] + \mathrm{i}\mathcal{I}^{PODP}[\alpha B, \omega]$ for the MPT obtained by PODP.

Lemma 3.3.1. *[133] (note, this was contributed by Ledger) An a-posteriori error estimate for the tensor coefficients computed using PODP is*

$$\left|(\mathcal{R}[\alpha B, \omega])_{ij} - (\mathcal{R}^{PODP}[\alpha B, \omega])_{ij}\right| \leqslant (\Delta[\omega])_{ij}, \tag{3.24a}$$

$$\left|(\mathcal{I}[\alpha B, \omega])_{ij} - (\mathcal{I}^{PODP}[\alpha B, \omega])_{ij}\right| \leqslant (\Delta[\omega])_{ij}, \tag{3.24b}$$

where

$$(\Delta[\omega])_{ij} := \frac{\alpha^3}{8\alpha_{LB}} \left(\|\hat{\boldsymbol{r}}_i(\omega)\|_{Y^{(hp)}}^2 + \|\hat{\boldsymbol{r}}_j(\omega)\|_{Y^{(hp)}}^2 + \|\hat{\boldsymbol{r}}_i(\omega) - \hat{\boldsymbol{r}}_j(\omega)\|_{Y^{(hp)}}^2 \right), \tag{3.25}$$

and α_{LB} is a lower bound on a stability constant.

Remark 3.3.2. *Ledger's a-posteriori error estimate in Lemma 3.3.1 allow the coefficients of $\mathcal{M}^{PODP}[\alpha B, \omega]$ obtained by PODP to be certified at low-computational cost during the on-line stage of the ROM using the procedure described below. The bound does not give an explicit dependence on N or ω, but, if desired, could be used as part of an iterative procedure to choose additional candidate ω values for the representative full order model solution snapshots in a similar manner to that described in [55]. The (spectral) behaviour $\mathcal{M}[\alpha B, \omega]$ with ω has been considered in [73], where results on the functions that characterise the spectral signature of the MPT are provided.*

A novel efficient evaluation of (3.24) has been made and it follows the approach presented in [55, pg. 52-54], adapted to complex matrices and with the simplification that a Riesz representation $\hat{\boldsymbol{r}}_i(\omega) \in Y^{(h0)}$ is computed using the lowest order elements for computational efficiency. The computations are split in to those performed in the off-line stage and those in the on-line stage as follows.

In the off-line stage, the following $(2M+1) \times (2M+1)$ Hermitian matrices are computed

$$\mathbf{G}^{(i,j)} = \left(\mathbf{W}^{(i)}\right)^H \mathbf{M}_0^{-1} \mathbf{W}^{(j)}, \tag{3.26}$$

where, since $\mathbf{G}^{(j,i)} = (\mathbf{G}^{(i,j)})^H$, it follows that, in practice, only the 3 matrices $\mathbf{G}^{(1,1)}$, $\mathbf{G}^{(2,2)}$ and $\mathbf{G}^{(3,3)}$ are required for computing the certificates on the diagonal entries of the tensors, and the further 3 matrices $\mathbf{G}^{(1,2)}$, $\mathbf{G}^{(1,3)}$ and $\mathbf{G}^{(2,3)}$ are needed for the off-diagonal terms. In the above, $(\mathbf{M}_0)_{ij} = \left\langle \boldsymbol{N}^{(i)}, \boldsymbol{N}^{(j)} \right\rangle_{L^2(\Omega)}$ are the coefficients of a real symmetric FEM mass matrix for the lowest order, with $\boldsymbol{N}^{(i)}, \boldsymbol{N}^{(j)} \in W^{(h0)}$ being typical lowest order basis functions, and

$$\mathbf{W}^{(i)} := \mathbf{P}_0^p \left(\begin{array}{ccc} \mathbf{r}^{(1)}(\boldsymbol{\theta}_i^{(0)}) & \mathbf{A}^{(0)}\mathbf{U}^{(M,i)} & \mathbf{A}^{(1)}\mathbf{U}^{(M,i)} \end{array} \right), \tag{3.27}$$

where $\mathbf{P}_0^p$ is a projection matrix of the FEM basis functions from order p to the lowest order 0, $\mathbf{U}^{(M,i)}$ is the $\mathbf{U}^M$ obtained in (3.13) for the ith direction. The stability constant $\alpha_{LB} = \lambda_{min}\min(1, \frac{\omega}{\omega'})$ is obtained from the smallest eigenvalue of an eigenvalue problem [55, pg56], which, in practice, is only performed once for smallest frequency of interest ω'.

In the on-line stage,

$$\|\hat{\boldsymbol{r}}_i(\omega)\|_{Y^{(hp)}}^2 = \left((\mathbf{w}^{(i)}(\omega))^H \mathbf{G}^{(i,i)}(\mathbf{w}^{(i)}(\omega))\right)^{1/2}, \tag{3.28a}$$

$$\|\hat{\boldsymbol{r}}_i(\omega) - \hat{\boldsymbol{r}}_j(\omega)\|_{Y^{(hp)}}^2 = \left(\|\hat{\boldsymbol{r}}_i(\omega)\|_{Y^{(hp)}}^2 + \|\hat{\boldsymbol{r}}_j(\omega)\|_{Y^{(hp)}}^2 - 2\mathrm{Re}((\mathbf{w}^i(\omega))^H \mathbf{G}^{(i,j)}(\mathbf{w}^{(j)}(\omega)))\right)^{1/2}, \tag{3.28b}$$

is evaluated, for each ω by updating the vector

$$\mathbf{w}^{(i)}(\omega) = \begin{pmatrix} \omega \\ -\mathbf{p}^{M,(i)}(\omega) \\ -\omega\mathbf{p}^{M,(i)}(\omega) \end{pmatrix}, \tag{3.29}$$

where $\mathbf{p}^{M,(i)}$ refers to $\mathbf{p}^M$ for the ith direction. Then (3.24) is applied to obtain the a-posteriori error estimate.

3.4   Chapter Summary

This chapter has presented a novel method for applying PODP to produce a ROM for the efficient computation of MPT spectral signatures. Original, regularised, weak forms of the transmission problems (2.6) and (2.7) were stated and discretised using $\boldsymbol{H}(\mathrm{curl})$ conforming elements. In the offline stage of the method, several snapshot full order problems are solved, using the `NGSolve` FE library, to which a truncated singular value decomposition is then applied. This allows for the projection of the problem from the full order space to a reduced order space, within which they can be solved in the online stage, at a greatly reduced computational cost [133]. Additionally the method presented also benefits from a-posteriori error estimates for the MPT coefficients derived by Ledger [133], which can be computed at a small additional computational cost, and, if desired, these error estimate could be used to drive an adaptive procedure for choosing new full order snapshots to further reduce the error. Further details on the novel computational implementation of these will follow in Chapter 4. Results pertaining to the computational savings, along exploratory results obtained using the method are documented in Chapter 5 and are also presented by the author in [133].

Chapter 4

Implementation Details of `MPT-Calculator` Software

4.1 Introductory Remarks

This chapter focuses on the implementation of the computation of MPT coefficients which has been made openly available in a software package called the `MPT-Calculator`. The chapter covers topics such as the choice of FE framework, computational implementation and support offered for the `MPT-Calculator`.

This chapter has been constructed from various sources and includes adapted code snippets from the author [134] and the `NGSolve` documentation [4]; adapted explanations from the documentation provided by the author [134]; and screenshots and analytics from the authors' YouTube channel [130].

The chapter provides justification of the choice to implement the $\boldsymbol{H}(\mathrm{curl})$ conforming basis functions offered by `NGSolve`, documenting the key features of `NGSolve` which make it the preferred FE framework to use in the production of the `MPT-Calculator`. The chapter presents the main aims and objectives of the `MPT-Calculator`, documenting the numerical efficiencies which have been implemented to speed up production of MPT spectral signatures. The chapter also covers the available support offered in the documentation and corresponding `MPT-Calculator` YouTube channel. The main novelties of the chapter are: the implementation of novel weak forms in `NGSolve`; the efficient implementation of the ROM discussed in Section 3.3, along with associated a-posteriori error certificates derived in Section 3.3.3; and the production and support offered in the open source software the `MPT-Calculator`.

The chapter is organised as follows: First in Section 4.2 the justification of the choice to use `NGSolve` as the FE framework the `MPT-Calculator` is established. In Section 4.2.1, an example of the implementation for the solution and post-processing of the problem (3.2) with a procedural outline to the solution of this problem is provided. Procedures for obtaining solutions to (3.4) are also documented in Section 4.2.2. In Section 4.3, an overview of the `MPT-Calculator` is provided documenting the key aims for developing the software, the structure available outputs and the numerical efficiencies

implemented. In Section 4.4, the support offered in the way of documentation and videos provided on the `MPT-Calculator` YouTube channel, documenting the topics covered by the videos. The chapter finishes with concluding remarks in Section 4.5.

4.2 `NGSolve`

Key to the approximate computation of $\boldsymbol{\theta}_i^{(0)}$ and $\boldsymbol{\theta}_i^{(1)}$, required for the computation of the MPT coefficients, is to discretise weak forms, such as (3.2) and (3.4), using the FE method leading to linear systems, such as (3.9) in the latter case. Details of FE discretisations can be found in many texts [27, 111, 93] and will not be repeated here. Rather than implement a new FEM procedures, it was decided to make use of an existing FE framework. Due to non-standard nature of the problem, there exists a requirement of user defined weak forms. Although there are many FE packages which allow for this, such as `COMSOL` [1], `deal.ii` [2] and `FEniCS` [3], `NGSolve` [4, 109, 108, 140] has been chosen, based on several key attributes of `NGSolve` when compared to other FE packages previously mentioned.

The ease of use provided by the Python interface: This is important for two reasons, firstly, this allows for the creation of the ROM, presented in Section 3.3, to be done in Python as opposed to inside the `NGSolve` library, allowing for a rapid and swift code development. Secondly, since part of this project involves producing a large dataset this benefits greatly by using scripting over of using a graphical user interface, which would be very slow for large numbers of objects. This allows for the production of more data with less human interaction. `FEniCS` also has a Python interface allowing similar rapid development. On the other hand `deal.ii` is entirely in C++. Although `COMSOL` has a similar interface, `COMSOL`'s interface uses Java which is generally harder and more unforgiving than Python as a language.

The unstructured tetrahedral meshes: Automatic procedures are available for generating unstructured tetrahedral meshes around complex geometries and these are implemented in `NetGen`, the mesher which forms part of `NGSolve`. Unstructured meshes are also possible in `FEniCS` and `COMSOL`. On the other hand, `deal.ii` uses hexahedral meshes and it is much harder, and not always possible, to generate meshes around complex configurations. In such cases, a commercial mesh generator is also needed. The tetrahedral elements provided in `NGSolve` also benefit from geometric polynomial enrichment allowing for greater precision in the approximation of geometry.

The efficient implementation of hierarchical p-version $\boldsymbol{H}(\text{curl})$ conforming elements: $\boldsymbol{H}(\text{curl})$ conforming elements are an appropriate choice for the problem since using these avoid the possibility of spurious modes, the use of these was advocated in [77] for magnetostatic and eddy current problems, this also leads to well conditioned matrices. With `NGSolve` providing a hierarchical implementation of the basis functions proposed by

Zaglmayr and Schöberl [110] which allows for the possibility of using p-refinement for greater fidelity in calculations. This basis also explicitly splits the basis in to gradients and non-gradient functions. As discussed in [77] this allows the introduction to efficient preconditioning for the linear system such as in (3.9) (see also below). While COMSOL and FEniCS also offer higher order $\boldsymbol{H}$(curl) basis functions, those proposed by Zaglmayr and Schöberl [110, 140] lead to better conditioned matrices.

The availability of robust preconditioners which benefit from parallelisation: NGSolve offers a number of robust and efficient preconditioners along with a range of iterative solvers, the combination of Balancing Domain Decomposition by Constraints (BDDC) preconditioner [84] along with the conjugate gradients solver, which is appropriate for systems with a stiffness matrix which is positive definite and (Hermitian) symmetric, was found to work well for the problem. The BDDC preconditioner also benefits from being highly parallelizable which is efficiently done using NGSolve's shared memory pipeline.

These factors, together with previous successful implementations of this basis [27, 69] in the research group meant NGSolve was selected as the library of choice.

In addition to the reasons discussed above, using NGSolve has some additional efficiencies which can be implemented. As p increases the number of degrees of freedom increases rapidly. However, of these, the interior degrees of freedom, which grow at rate $O(p^3)$, are independent between each element and can be eliminated using a process known as static condensation. This is achieved by eliminating element-internal unknowns of higher order FEs via a Schur complement factorisation, therefore, reducing the size of the linear system in (3.9) and reducing solution times. Secondly, due to construction of the $\boldsymbol{H}$(curl) basis in NGSolve, where gradients and non-gradients are made explicit. The transmission problem in (2.6) and (2.7) and their associated weaks forms in (3.2) and (3.3) together with construction of the basis, allows the gradient basis function to be skipped in $\Omega \backslash B$. This yet again reduces the number of degrees of freedom decreasing solution time.

4.2.1 Implementation in **NGSolve** for the Approximate Solution of $\boldsymbol{\theta}_1^{(0,hp)}$ to Compute $(\mathcal{N}^0)_{11}$

In this section an example of an implementation in NGSolve is given for the approximate solution of $\boldsymbol{\theta}_1^{(0,hp)}$ and the computation of $(\mathcal{N}^0)_{11}$. The first stage is to generate an unstructured tetrahedral discretisation Ω, which consists of both the object B and the truncated region of free space surrounding it to a truncation boundary. In the case where B is made up of multiple materials, B is replaced by $\boldsymbol{B}$ and the approach in [74] is followed. In this section it is assumed that such a discretisation is available in .vol file where each region $B^{(n)}$ in $\boldsymbol{B}$ is tagged $\mu_r^{(n)}$ values (additionally also with with $\sigma_*^{(n)}$ and

values when working out the full MPT) which the region outside is tagged as free space. The first step in the process is to import the Python modules which will be used[1].

```
01 from ngsolve import *
02 import netgen.meshing as ngmeshing
```

The next step is to load and initialise the mesh,

```
03 ngmesh = ngmeshing.Mesh(dim=3)
04 ngmesh.Load(``Object.vol'')
05 mesh = Mesh(``Object.vol'')
06 mesh.Curve(5)
```

these four lines are vital and each have a distinct job, line 03 first defines that the mesh which will be used is a 3D mesh, line 04 loads the mesh to line 05 creates the `mesh` instance in Python, finally line 06 uses curved FEs at curved boundaries and material interfaces to better approximate the object's shape, in this case this is achieved with 5th order polynomials. The next step is to define the region specific parameters

```
07 mu_coef = [ mur[mat] for mat in mesh.GetMaterials() ]
08 mu_r = CoefficientFunction(mu_coef)
09 inout_coef = [inorout[mat] for mat in
mesh.GetMaterials() ]
10 inout = CoefficientFunction(inout_coef)
```

for each parameter being defined there are 2 steps the first, which is done on lines 07, 09, creates lists of the values of each parameter corresponding to each element in the mesh from Python dictionaries `mur`, `sig` and `inorout`. The second step converts these lists to a coefficient functions which `NGSolve` can use quickly and efficiently. Note the addition of the parameter `inout`, this corresponds to elements which are contained in B and takes a value of $1 \in B$ and $0 \in B^c$. It is at this point the FE space is created and some problem specific variables are defined.

```
11 fes = HCurl(mesh, order=4, dirichlet=``outer'', flags
= { ``nograds'' :  True })
12 theta = GridFunction(fes)
13 theta.Set((0,0,0), BND)
14 u, v = fes.TnT()
15 e = CoefficientFunction( (1,0,0) )
```

---

[1]Although it is generally bad programming practice to use `from ...  import *` we use this to reduce the amount of code and improve code readability, the `NGSolve` functions used throughout this section start with a capital letter. Additionally all classes, methods and instances used through out the code are `NGSolve` functions with the exception of lines 04 and 05.

```
16 epsi = 10**-10
```

Here the FE space, solution vector, test and trial functions and a coefficient function for the forcing vector for the right hand side are all initialised. In line 11 we define an FE space which is based on order $p = 4$ $\boldsymbol{H}$(curl) conforming elements, which skip the gradient functions in all of the domain Ω, which is appropriate for problem (3.2), as set by `flags = { ''nograds'' : True }`. In addition the outer boundary has been set to have dirichlet boundary conditions and, in this case, $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(0,hp)} = \boldsymbol{0}$ on $\partial\Omega$ is an appropriate boundary condition for this problem and allows the otherwise unbounded domain to be replaced by a finite domain, as described in Section 3.2. Finally, the coefficient function `e` has been defined, these are the unit vector $\boldsymbol{e}_1$ which will be used as a forcing term along with a small regularisation parameter `epsi`, (set to 10^{-10}), which was mentioned in Section 3.2.1 which is required to circumvent the Coulomb gauge. It is at this point that the bilinear and linear forms in (3.2) are prescribed.

```
17 a = BilinearForm(fes, symmetric=True, condense=True)
18 a += SymbolicBFI((mu_r**(-1))*(curl(u)*curl(v)))
19 a += SymbolicBFI(epsi*(u*v))
20 f = LinearForm(fes)
21 f += SymbolicLFI(inout*(2*(1-mu_r**(-1)))) *
InnerProduct(e,curl(v)))
```

in line 17 the bilinear form is initialised, the inclusion of the two flags `symmetric=True` and `condense=True` these are both included for numerical efficiency. The first, setting `symmetric=True` can be done due to the problem having a symmetric stiffness matrix, this reduces memory consumption and wall clock time as when this is set `NGSolve` only operates on half of the stiffness matrix enforcing symmetry with the other. The second, setting `condense=True` informs `NGSolve` of the intention to apply static condensation as discussed above. Next the BDDC preconditioner is defined, matrices assembled and inverse operator defined.

```
22 c = Preconditioner(a,"bddc")
23 a.Assemble()
24 f.Assemble()
25 c.Update()
26 inverse= CGSolver(a.mat, c.mat, precision=10**-10,
maxsteps=1000)
```

The final step in obtaining a solution to the problem is done in the next lines.

```
27 f.vec.data += a.harmonic_extension_trans * f.vec
28 theta.vec.data = inverse * f.vec
29 theta.vec.data += a.harmonic_extension * Theta.vec
```

```
30 theta.vec.data += a.inner_solve * f.vec
```

Lines 29-30 implement both the static condensation procedure and iterative solution technique using the operators defined above. For further details see [4]. The final step in the process is to post-process the solution which is done in the next lines.

```
31 alpha = 0.001
32 N0_11 = (alpha**3)*(Integrate(1-mu_r**(-1),mesh) +
(1/4)*(Integrate(mu_r**(-1)*(InnerProduct(curl(theta),
curl(theta))),mesh)))
```

Untill this point the solution obtained has been dimensionless, α defines (in meters) the size of the units used in the mesh ($\alpha = 0.001$(m) the units used in `Object.vol` are in mm). Line 32 is a direct implementation of (2.5a) for $i = j = 1$. The above process can then be easily extended to handle $i, j = 1, 2, 3$. The full procedure can be summarised in a simple algorithmic format shown in Algorithm 1, which calls Algorithm 2 to compute the approximative $\boldsymbol{\theta}_i^{(0,hp)}$ solutions.

Algorithm 1 Calculation of $(\mathcal{N}^0)_{ij}$

Define object geometry B, μ_r and α.

Choose a suitable large domain Ω such that $B \subset \Omega$ and prescribe boundary conditions $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(0,hp)} = \boldsymbol{0}$ on $\partial\Omega$.

Define spacing and generate a mesh.

Choose a $\boldsymbol{H}(\text{curl})$ conforming FE space and choose element order.

SOLVE $\boldsymbol{\theta}_i^{(0,hp)}(B, \mu_r, \text{mesh})$ using function in Algorithm 2.

for $i, j = 1, 2, 3$ **do**

    Implement (2.5a) to compute $(\mathcal{N}^0)_{ij}$.

end for

Algorithm 2 Function for approximate solution of $\boldsymbol{\theta}^{(0,hp)}$

function SOLVE $\boldsymbol{\theta}_i^{(0,hp)}(B, \mu_r, \text{mesh})$

    for $i = 1, 2, 3$ **do**

        Choose $\boldsymbol{e}_i$ to be the ith column of the identity matrix.

        Prescribe the bilinear forms and linear forms in (3.2).

        Setup the preconditioner and assemble matrix.

        Solve linear system for coefficients of $\boldsymbol{\theta}_i^{(0,hp)}$.

    end for

end function

In an analogous algorithm to that in Algorithm 2, a similar function can be defined for obtaining an approximate solutions for $\boldsymbol{\theta}^{(1,hp)}$, and this is done in Algorithm 3. Note the inclusion of α and σ_* in the list of dependencies for the function which was not required previously, are needed to compute $\boldsymbol{\theta}^{(1,hp)}$.

Algorithm 3 Function for approximate solution of $\boldsymbol{\theta}^{(1,hp)}$
---
   function SOLVE $\boldsymbol{\theta}_i^{(1,hp)}(B, \alpha, \omega, \sigma_*, \mu_r, \boldsymbol{\theta}_i^{(0,hp)}$,mesh)

      for $i = 1, 2, 3$ **do**

         Prescribe the bilinear forms and linear forms in (3.4).

         Setup the preconditioner and assemble matrix.

         Solve linear system for coefficients of $\boldsymbol{\theta}_i^{(1,hp)}$.

      end for

   end function

4.2.2 Implementation and Methods for Computing $(\mathcal{M})_{ij}$

In a manner similar to the implementation provided for $(\mathcal{N}^0)_{ij}$ in the previous section, procedures can be defined for the different algorithms which calculate $(\mathcal{M})_{ij}$ which are implemented in the MPT-Calculator. There are two distinct algorithms[2] to calculate $(\mathcal{M})_{ij}$, the first which repeatedly solves the problem defined in (3.4) for different frequencies of ω_f, $f = 1, 2, ..., F$[3] this procedure is summarised in Algorithm 4.

Algorithm 4 Full order solution of $\boldsymbol{\theta}^{(1,hp)}$ and calculation of $(\mathcal{M})_{ij}$

   Define object geometry B, μ_r, σ_*, and α.

   Choose a suitable large domain Ω such that $B \subset \Omega$ and prescribe boundary conditions $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(0,hp)} = \boldsymbol{0}$ and $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(1,hp)} = \boldsymbol{0}$ on $\partial\Omega$.

   Define spacing and generate a mesh.

   Choose a $\boldsymbol{H}$(curl) conforming FE space and choose element order.

   SOLVE $\boldsymbol{\theta}_i^{(0,hp)}(B, \mu_r$,mesh) using function in Algorithm 2.

   for $i, j = 1, 2, 3$ **do**

      Implement (2.5a) to compute $(\mathcal{N}^0)_{ij}$.

   end for

   for $\omega = \omega_1, \omega_2, ..., \omega_F$ **do**

      SOLVE $\boldsymbol{\theta}_i^{(1,hp)}(B, \alpha, \omega, \sigma_*, \mu_r, \boldsymbol{\theta}_i^{(0,hp)}$,mesh) using function in Algorithm 3.

      for $i, j = 1, 2, 3$ **do**

         Implement (2.5b,c) to compute $(\mathcal{R})_{ij}$ and $(\mathcal{I})_{ij}$ and therefore $(\mathcal{M})_{ij}$.

      end for

   end for

The second which employs the use of a ROM, which was discussed in Section 3.3, this procedure, summarised in Algorithm 5, solves the problem defined in (3.4), N (which is less than F) times and uses these solutions to produce F surrogate problems which can then be solved at a reduced computational cost.

---

[2]In practice there are three separate solver functions in the MPT-Calculator with the additional function handling the case when a single MPT is produced. This implementation is, in part, due to the additional outputs which are available for a single MPT output.

[3]Note, in Chapters 8 and 9 F will be re-used as the number of features although this should be clear from context.

Algorithm 5 Reduced order solution of $(\boldsymbol{\theta}^{(1,hp)})^{PODP}$ and calculation of $(\mathcal{M})_{ij}$

Define object geometry B, μ_r, σ_*, and α.

Choose a suitable large domain Ω such that $B \subset \Omega$ and prescribe boundary conditions $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(0,hp)} = \mathbf{0}$ and $\boldsymbol{n} \times \boldsymbol{\theta}_i^{(1,hp)} = \mathbf{0}$ on $\partial\Omega$.

Define spacing and generate a mesh.

Choose a $\boldsymbol{H}(\text{curl})$ conforming FE space and choose element order.

SOLVE $\boldsymbol{\theta}_i^{(0,hp)}(B, \mu_r,\text{mesh})$ using function in Algorithm 2.

for $i, j = 1, 2, 3$ **do**

    Implement (2.5a) to compute $(\mathcal{N}^0)_{ij}$.

end for

for $\omega = \omega_1, \omega_2, ..., \omega_N$ **do**

    SOLVE $\boldsymbol{\theta}_i^{(1,hp)}(B, \alpha, \omega, \sigma_*, \mu_r, \boldsymbol{\theta}_i^{(0,hp)},\text{mesh})$ using function in Algorithm 3.

end for

Construct $\mathbf{D}$ using solutions $\boldsymbol{\theta}_i^{(1,hp)}$ as stated in (3.11).

Compute TSVD (3.13), to obtain $\mathbf{U}^M$, $\boldsymbol{\Sigma}^M$ and $(\mathbf{V}^M)^H$.

OBTAIN $(\Delta[\omega])_{ij}$ OFF-LINE using function in Algorithm 6 (if required).

for $\omega = \omega_1, \omega_2, ..., \omega_F$ **do**

    Solve reduced linear systems (3.17) for ω.

    OBTAIN $(\Delta[\omega])_{ij}$ on-line using function in Algorithm 7 (if required).

    Project $\mathbf{p}^M$ to full order space using (3.15a) to obtain $(\boldsymbol{\theta}_i^{(1)})^{PODP}$.

    for $i, j = 1, 2, 3$ **do**

        Implement (2.5b,c) with $(\boldsymbol{\theta}_i^{(1)})^{PODP}$ to compute $(\mathcal{R})_{ij}$ and $(\mathcal{I})_{ij}$ and therefore $(\mathcal{M})_{ij}$.

    end for

end for

Algorithm 6 Off-line stage of function to produce a-posterori error estimates
---
function OBTAIN $(\Delta[\omega])_{ij}$ OFF-LINE($\mathbf{U}^M, \mathbf{A}^{(0)}, \mathbf{A}^{(1)}, \mathbf{r}^{(1)}$)

    Obtain the stability constant α_{LB} by solving an eigenvalue problem [55, pg56].

    for $i = 1, 2, 3$ **do**

        Construct the matrix $\mathbf{W}^{(i)}$ from (3.27).

    end for

    for $i, j = 1, 2, 3$ **do**

        Obtain $\mathbf{G}^{(i,j)}$ from (3.26).

    end for

end function

Algorithm 7 On-line stage of function to produce a-posterori error estimates
___
   function OBTAIN $(\Delta[\omega])_{ij}$ ON-LINE($\mathbf{G}^{(i,j)}, \mathbf{p}^M$)
       for $i = 1, 2, 3$ **do**
          Construct the vector $\mathbf{w}^{(i)}$ from (3.29).
          Calculate $||\hat{\mathbf{r}}_i||^2_{Y^{hp}}$ from (3.28a).
       end for
       for $i, j = 1, 2, 3$ **do**
          Calculate $||\hat{\mathbf{r}}_i - \hat{\mathbf{r}}_j||^2_{Y^{hp}}$ from (3.28b).
          Calculate $(\Delta[\omega])_{ij}$ from (3.25).
       end for
   end function
___

4.3 `MPT-Calculator` Overview

The `MPT-Calculator` is a wrapper built around the framework provided by `NGSolve`, it provides a command line interface which allows for the repeated, post-processed solution of (3.2) and (3.4), to obtain the coefficients of the MPT (2.5) in a fast, efficient and automated manner. The key features of the `MPT-Calculator` are:

- The seamless integration with `NGSolve` which allows the user to produce an object of interest, defined in the `.geo` file format provided by `NGSolve`, a desired MPT spectral signature can then be produced from this input.

- The additional outputs offered, such as the error certificates, discussed in Section 3.3.3, produced for the ROM, the object specific frequency limit of the eddy current model, discussed in Section 2.4.3, or the `.vtk` output available for the eddy currents at single frequency.

- The reduction in solution time and ease of use when compared with previous codes which also calculate MPTs using an FEM approach.

- The supporting material which is included with the software, including the in-depth documentation detailing all aspects of the software and YouTube video series which provides a walk-through of several demonstrative examples.

We will briefly discuss several aspects of the `MPT-Calculator`.

4.3.1 Structure of the Code

The user is expected interact with four files `main.py`, `Settings.py`, `PlotterSettings.py` and a geometry file, `.geo` file, these files allow the user to produce an array of different MPT spectral signatures for many different objects. In this section we discuss the layout of the folder system in place, how each of the input

files can used and edited by the user to produce a frequency sweep along with how and where the results are saved. The structure of the code can be seen in Figure 4.1.

```
MPT-Calculator/
├── Functions
│       ├── Checkvalid.py
│       ├── FullSolvers.py
│       ├── MPTFunctions.py
│       ├── MeshCreation.py
│       ├── PODFunctions.py
│       ├── PODPlotEditor.py
│       ├── PODPlotEditorWithErrorBars.py
│       ├── PODSolvers.py
│       ├── PlotEditor.py
│       ├── PlotEditorWithErrorBars.py
│       ├── Plotters.py
│       ├── ResultsFunctions.py
│       └── SingleSolve.py
├── GeoFiles
│       ├── Coin.geo
│       ├── Cylinder.geo
│       ├── DualBar.geo
│       ├── Spheroid.geo
│       ├── Steelgun.geo
│       ├── Tetra.geo
│       ├── Torus.geo
│       ├── rifle.geo
│       └── sphere.geo
├── LICENSE
├── README.md
├── Results
│       └── vtk_output
├── Settings
│       ├── PlotterSettings.py
│       └── Settings.py
├── VolFiles
└── main.py
```

Figure 4.1: Image displaying the structure of the main folder of the `MPT-Calculator`.

This structure is used to produce an MPT spectral signature in the procedure summarised by the pipeline laid out in Figure 4.2.

In Figure 4.2 the four input files are denoted in green, the functions that are associated with obtaining a solution, i.e. that mesh, solve and save are denoted in blue and the functions that produce an output are denoted in red, the only exception to this is `SingleSolve.py` which in addition to producing a solution for the MPT at a single frequency also exports a vector field for the eddy currents in the `.vtk` file format. More detail about the contents of each file in Appendix A.

4.3.2 Algorithm Selection and Available Outputs

As mentioned in Section 4.2.2 there are three algorithms, two of which are related, that are available in the `MPT-Calculator`, these correspond to when a single MPT, a full order or reduced order MPT spectral signature are to be produced[4]. The chosen algorithm is

---

[4]Although the single MPT is a full order MPT spectral signature at a single frequency the differentiation is made here due to the separate handling of the cases in the code.
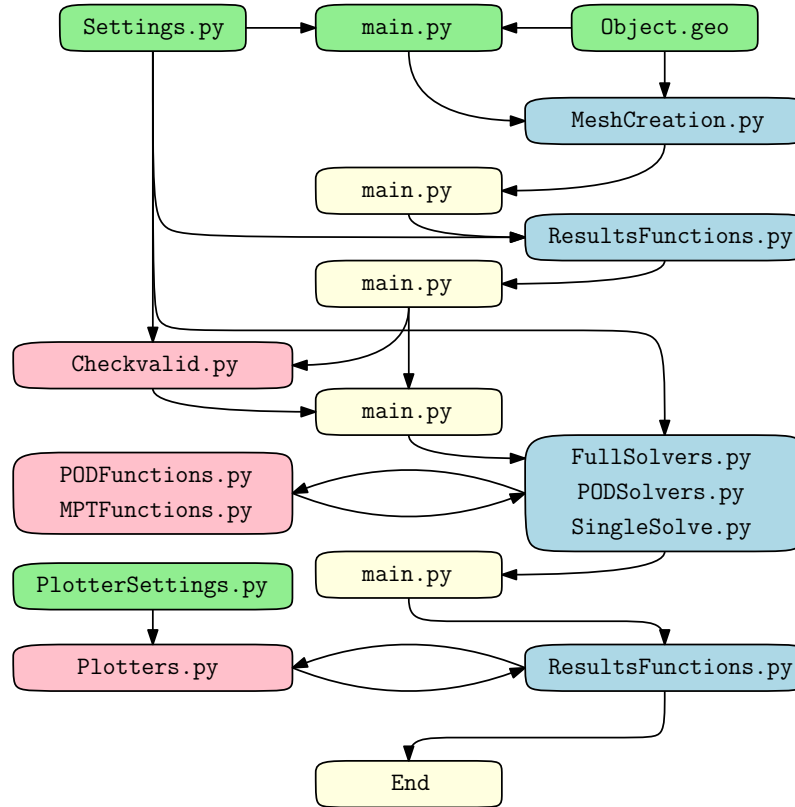
Figure 4.2: Pipeline of the `MPT-Calculator`.

selected within the `main.py` file, this selection determines which of the solver functions `FullSovers.py`, `PODSolvers.py` and `SingleSolve.py` are called and in turn which output is produced.

Single MPT: The case when a single MPT is to be produced is a special case of the procedure presented in Algorithm 4, with the number of frequencies at which to produce an MPT is $F = 1$. The reason for separating the case of a single MPT and a full order MPT spectral signature is due to the differences in the outputs produced. When this procedure is chosen the output folder produced has the structure that can be seen in Figure 4.3. This output shown in Figure 4.3 differs from the full order MPT spectral signature in 2 key ways, firstly there is no graphical output produced for the single MPT, this is due to the representation being constructed of a single point for each MPT coefficient. The second reason is due to the additional `.vtk` output which is available for the single MPT, when selected, this exports the field of eddy-currents in a `.vtk` file format for later visualisation in Pararview. This option is only available for the single MPT due to the `.vtk` files being very large in size. Finally, the inclusion of the file `Eddy-current_breakdown.txt` is noted, this is an additional output available in all of the procedures discussed and stores the object specific eddy-current limit which was discussed in Section 2.4.3.

```
OutputFolder/
├── Data
│   ├── Eddy-current_breakdown.txt
│   ├── Eigenvalues.csv
│   ├── MPT.csv
│   └── N0.csv
└── Input_files
    ├── Settings.py
    ├── main.py
    ├── sphere.geo
    └── sphere.zip
```
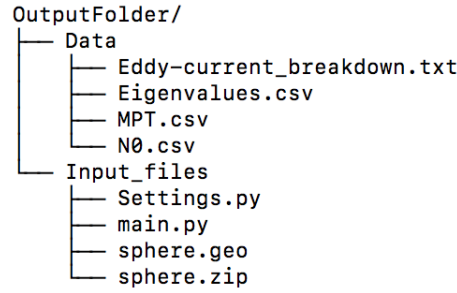
Figure 4.3: Image displaying the structure of the output folder produced for the case of a single MPT.

```
OutputFolder/
├── Data
│   ├── Eddy-current_breakdown.txt
│   ├── Eigenvalues.csv
│   ├── Frequencies.csv
│   ├── N0.csv
│   └── Tensors.csv
├── Functions
│   └── Plotters.py
├── Graphs
│   ├── ImaginaryEigenvalues.pdf
│   ├── ImaginaryTensorCoeficients.pdf
│   ├── RealEigenvalues.pdf
│   └── RealTensorCoeficients.pdf
├── Input_files
│   ├── Settings.py
│   ├── main.py
│   ├── sphere.geo
│   └── sphere.zip
├── PlotEditor.py
└── PlotterSettings.py
```
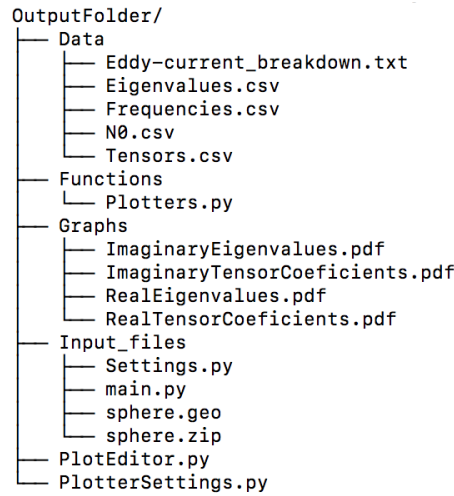
Figure 4.4: Image displaying the structure of the output folder produced for the case of a full order MPT spectral signature.

Full order MPT spectral signature: The case when a full order MPT spectral signature is to be produced involves the direct implementation of the procedure presented in Algorithm 4. Although F can be any positive integer, significant cost savings in the calculation of the MPT spectral signature are obtained if the reduced order method is applied for $F > 10$. An example of this is presented in Chapter 5 with the cost breakdown documented in Figure 5.6. When selected the output folder produced has the structure which can be seen in Figure 4.4. When compared to the outputs which are found in the case of a single MPT, the full order MPT spectral signature has several additional outputs, firstly the graphical outputs which show how $(\mathcal{M})_{ij}$ and $\lambda_i(\mathcal{M})$ changes as a function of ω are included. There are also several files `Plotters.py`, `PlotEditor.py` and `PlotterSettings.py`, which relate to re-plotting of these graphs and allow for the easy visualisation of the data.

Reduced order MPT spectral signature: As mentioned above, when an MPT spectral signature is required at a large number of frequencies $F > 10$, there are computational

```
OutputFolder/
├── Data
│       ├── Eddy-current_breakdown.txt
│       ├── Eigenvalues.csv
│       ├── ErrorBars.csv
│       ├── Frequencies.csv
│       ├── N0.csv
│       ├── PODEigenvalues.csv
│       ├── PODFrequencies.csv
│       ├── PODTensors.csv
│       └── Tensors.csv
├── Functions
│       └── Plotters.py
├── Graphs
│       ├── ImaginaryEigenvalues.pdf
│       ├── ImaginaryTensorCoeficients.pdf
│       ├── RealEigenvalues.pdf
│       └── RealTensorCoeficients.pdf
├── Input_files
│       ├── Settings.py
│       ├── main.py
│       ├── sphere.geo
│       └── sphere.zip
├── PODPlotEditor.py
├── PODPlotEditorWithErrorBars.py
├── PlotEditor.py
├── PlotEditorWithErrorBars.py
└── PlotterSettings.py
```

Figure 4.5: Image displaying the structure of the output folder produced for the case of a reduced order MPT spectral signature.

savings offered by employing the ROM technique described in Section 3.3 for which the procedure is outlined in Algorithm 5. When selected the output folder produced has the structure that can be seen in Figure 4.5. When compared to the outputs which are found in the case of a full order MPT spectral signature, the reduced order MPT spectral signature has several additional outputs, firstly the additional `.csv` files located in the `Data` folder, the file `ErrorBars.csv` contains the error certificates derived in Section 3.3.3, the files `PODEigenvalues.csv`, `PODFrequencies.csv` and `PODTensors.csv` contain the post-processed full order solutions which were used to create the ROM. The other difference is the inclusion of the additional plotter functions `PODPlotEditor.py`, `PODPlotEditorWithErrorBars.py` and `PlotEditorWithErrorBars.py`, these give more functionality when re-plotting the graphs produced.

4.3.3 Numerical Efficiencies

In addition to being user friendly, the `MPT-Calculator` benefits from:

- The implementation of the ROM, discussed in Section 3.3. When used, it was shown to reduce wall clock time by 86% [131], this reduction is affected by the number of full order solutions used in the generation of the ROM and the number of output frequencies desired. Further details of computational cost comparisons will be presented in Chapter 5.

- The parallelised option offered also greatly reduces the wall clock time. Due to the nature of the problem, many steps such as the **for** loops solving $\boldsymbol{\theta}_i^{(1)}$ have no bearing

on each other removing the need for them to be done sequentially. This therefore allows them to be trivially parallelised across many CPUs.

- Due to the 3-D nature of the problem, for each full order solution there are three directions which require a solution, this reduces to solving three linear systems which take the form of that presented in (3.9). Due to construction, all of these systems have the same stiffness matrix $\mathbf{A}(\boldsymbol{\omega})$ and differ only by their forcing function $\mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega})$. This allows the preconditioning of the matrix, which for the BDDC preconditioner is the most computationally expensive step, to be done only once, and then be reused for the different forcing functions reducing wall clock time for the preconditioning step.

- When producing an MPT spectral signature the forcing functions $\mathbf{r}(\boldsymbol{\theta}^{(0,hp)}, \boldsymbol{\omega})$ in the full order problem are proportional to frequency, this has been exploited in the implementation by initialising the forcing functions only once for the sweep then scaling by ω for each full order solution to be be obtained.

- Since the MPT is symmetric, when post-processing the solutions to calculate the MPT coefficients only half of the off diagonal coefficients are calculated reducing wall clock time for post-processing.

Exploiting all of the efficiencies above, as well as those of `NGSolve`, the reduction in wall clock time is substantial when compared to similar codes.

4.4 Github and YouTube

As discussed previously the `MPT-Calculator` is an open source software, accessible at the from a github repository created by the author [134]. Along with the code there is detailed documentation provided which helps guide the user in how best to interact with the code, with steps on how to install the code, detailed descriptions of the input and `.geo` files and multiple demonstrative examples. In addition to the code and documentation, there is a series of videos uploaded to the `MPT-Calculator` YouTube channel [130]. This series of videos provides a walk through guide on how to install and use the `MPT-Calculator`, covering topics such as constructing an object in the `.geo` file format, hp-refinement, obtaining additional outputs and post-processing existing outputs of the code, and how to export the eddy current fields and visualise these fields in paraview.

60
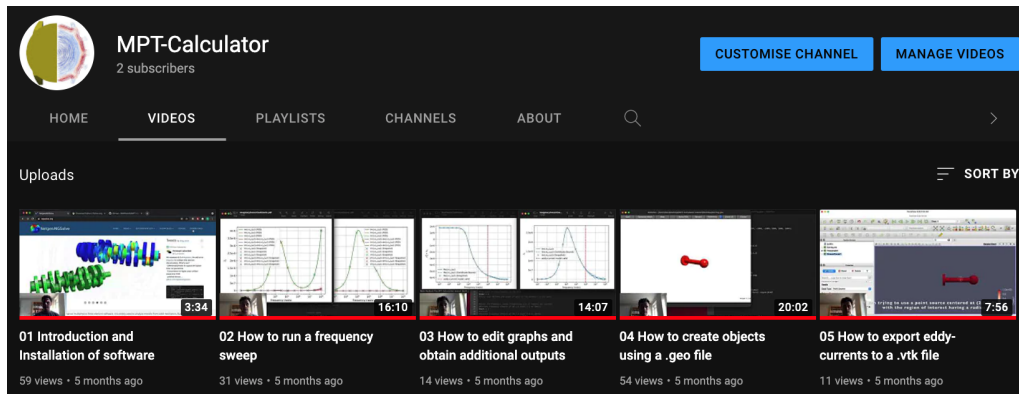
Figure 4.6: Screenshot of the `MPT-Calculator` YouTube channel.

Video 1: The first video of the series is an introductory video and the shortest of the five, this video gives a short overview of what is covered in the series. It discusses the software dependencies and explains where the relevant software (`NGSolve` and Python) can be downloaded from. It also explains how to download and setup the `MPT-Calculator` software. Finally, it points the user to the user to the documentation included when downloading the `MPT-Calculator`.
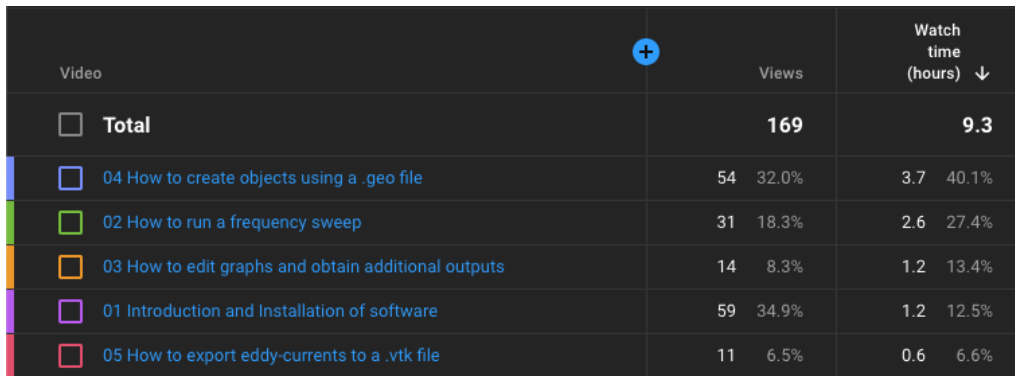
Video 2: The second video in the series is designed to teach the user how to interact with the software through a demonstrative example showing the user how to obtain an MPT spectral signature for one the pre-made `.geo` files. In the video the user is shown how to use the option to use the ROM for fast computation of an MPT spectral signature and how to perform hp-refinement to obtain converged results for the object. The video finishes at the point where a converged MPT spectral signature has been obtained

Video 3: The third video in the series focuses on the outputs of the `MPT-Calculator`, the video explains the structure of the output folder, data and graphs generated in the previous video. It explains how to obtain additional outputs, which may be of interest to the user, such as, the error certificates, and object dependant eddy-current limit. The video also covers how to use the basic visualisation tool included with the code, which allows the user to change which lines are plotted and the look of the graphs by editing the line styles used when plotting.

Video 4: The focus of video 4 is on how to create an object of interest by combining primitives using `NetGen`'s Constructive Solid Geometry (CSG), using the `.geo` file format, the example of a simple stud earring is used. The video shows how to set material properties for the different regions in the domain and explains how to save the mesh created so that it can be used to produce an MPT spectral signature. The video finishes with an MPT spectral signature being obtained for the earring.

61

Video 5: The final video in the series shows a simulation to produce an MPT at a single frequency and shows hows the resulting eddy-current field can be exported as a `.vtk` file so that the eddy-currents within the conductor associated with the 3 transmission problems solved can be visualised. This `.vtk` file is then opened in `paraview` and the user is shown how to apply a threshold filter to remove the outer domain of the simulation and finally apply a stream tracer to visualise the eddy-currents.

The addition of these supplementary videos to the software is important, as it increases the accessibility of the software significantly, with written instructions alone commonly being misconstrued. The videos, which have only recently been uploaded have been and are being utilised, this usage is summarised in Figure 4.7, as can be seen from the views metric the videos have been viewed 169 times, although by YouTube standards this is low, in an academic setting this is significant. This is also expected grow now that the paper [133] has appeared and as the approach becomes better known. From the watch time metric the content has been viewed for over 9 hrs in total, with the majority of the time spent viewing video 2 and 4, this also fits with what is expected since video 2 and 4 are the most technical and are therefore expected to be rewatched, this gives credence to the statistics and that the theory that viewers purposefully found the videos and rewatched elements of them.

Video		Views		Watch time (hours) ↓	
☐ **Total**		**169**		**9.3**	
☐ 04 How to create objects using a .geo file		54	32.0%	3.7	40.1%
☐ 02 How to run a frequency sweep		31	18.3%	2.6	27.4%
☐ 03 How to edit graphs and obtain additional outputs		14	8.3%	1.2	13.4%
☐ 01 Introduction and Installation of software		59	34.9%	1.2	12.5%
☐ 05 How to export eddy-currents to a .vtk file		11	6.5%	0.6	6.6%

Figure 4.7: Usage statistics of `MPT-Calculator` its dedicated YouTube channel as of 16th November 2021.

4.5 Chapter Summary

This chapter presented work related to the development, computational implementation and support offered for the `MPT-Calculator`. The chapter provided justification for the choice to use `NGSolve` as the FE framework with which to build the `MPT-Calculator` and provides an explicit example of the implementation used to obtain an approximate solution to the problem presented in (3.2). The chapter presents an overview of the `MPT-Calculator`, documenting key aims and objectives of the software providing insight into the structure, use of, and numerical efficiencies implemented

in the software. The chapter also discusses support offered for the code in the form of documentation and video tutorials provided on the `MPT-Calculator`'s dedicated YouTube channel. The main novelties of the chapter are: the implementation of novel weak forms in `NGSolve`; the efficient implementation of the ROM discussed in Section 3.3, along with associated a-posteriori error certificates derived in Section 3.3.3; and the production and support offered in the open source software the `MPT-Calculator`. Results pertaining to the computational savings, along with exploratory results obtained using the `MPT-Calculator` will follow in Chapter 5 and are also presented by the author in [133].

Chapter 5

Numerical Results for the Reduced Order Model

5.1 Introductory Remarks

This chapter presents a series of illustrative examples to demonstrate the implementation and performance of the ROM, with a-posteriori error estimates, proposed in Chapters 3 and 4. The chapter has been adapted and extended from the work presented by the author in [133].

The key aim of the chapter is to document the accuracy, efficiency and time savings offered by the ROM, with the main novelties of the chapter being as follows. The comparison of linearly vs logarithmically chosen full order snapshots, comparing the associated singular value from the TSVD for varying numbers of full order snapshots and the relative error produced in the eigenvalues of the MPT for the two choices. A demonstration of the accuracy of the ROM where the eigenvalues obtained using the ROM are compared with those of the full order and analytical solutions for a permeable conducting sphere are also compared. The application of a-posteriori error certificates to the ROM derived in Lemma 3.3.1. An investigation into the number of shapshots used in the ROM, along with the positioning of new snapshots. The application of the ROM and associated error certificates to a range of simple geometric object. As well as numerical examples of the scaling results derived in Section 2.7, where the possibility of producing MPT spectral signatures for objects which vary in size and or conductivity at negligible computational cost is shown.

The structure of the chapter is as follows; First Section 5.2 contains results for a permeable conducting sphere, showing results comparing number of snapshots along with linearly vs logarithmically chosen snapshots, the accuracy, efficiency and speed will be benchmarked. In Section 5.3 results for a permeable conducting torus are presented. Next, in Section 5.4 results for a permeable conducting tetrahedron are presented. Then, in Section 5.5 results for an inhomogeneous conducting bar are shown. In Section 5.6 contains numerical examples of the scaling results presented in Lemmas 2.7.1 and 2.7.2 for a permeable conducting sphere and a permeable conducting tetrahedron. The Chapter

is concluded in Section 5.7 with some closing remarks.

5.2 Conducting Permeable Sphere

Consider the case where $B_\alpha = \alpha B$ is a permeable conducting sphere of radius $\alpha = 0.01$ m and B is the unit sphere centred at the origin. The sphere is chosen to have a relative permeability $\mu_r = 1.5$ and conductivity $\sigma_* = 5.96 \times 10^6$ S/m. To produce the snapshots of the full order model, Ω is set to be a ball 100 times the radius of B [1], a mesh of $26\,385$ unstructured tetrahedra, refined towards the object is generated, and $p = 3$ elements are applied. This discretization is chosen since it has already been found to produce an accurate representation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for $10^2 < \omega < 10^8$ rad/s by comparing with exact solution of the MPT spectral signature for a sphere [128, 71]. Indeed, provided that the geometry discretisation error is under control, performing p-refinement of the full order model solution results in exponential convergence to the true solution [69]. For this mesh, p-refinement has been found to give exponential convergence of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ down to relative error of less than 0.01% and so further increases in the size of Ω were not deemed to be required.

Two different schemes for choosing frequencies ω for generating the solution vectors $\mathbf{q}(\omega)$ required for $\mathbf{D}$ in (3.11) are considered. Firstly, linearly spaced frequencies $\omega_{min} \leqslant \omega_n \leqslant \omega_{max}$, $n = 1, 2, \ldots, N$, are considered where, as in Section 3.3.1, N is the number of representative full order solution snapshots that follow from using each value of ω_n, in turn, and this choice of samples is denoted by "Lin" in the results. Secondly, logarithmically spaced frequencies $\omega_{min} \leqslant \omega_n \leqslant \omega_{max}$ are considered and this regime is denoted by "Log" in the results.

Considering both linearly and logarithmically spaced frequencies with $\omega_{min} = 1 \times 10^2$ rad/s, $\omega_{max} = 1 \times 10^8$ rad/s and $N = 9, 13, 17$, in turn, to generate the snapshots, the application of an SVD to $\mathbf{D}$ in (3.12) leads to the results shown in Figure 5.1 where the values have been scaled by σ_1 and are strictly decreasing. Observe that "Log" case produces singular values σ_i/σ_1, which tend to 0 with increasing i, while the "Lin" case produces σ_i/σ_1, which tend to a finite constant with increasing i. Also shown is the tolerance $TOL = 1 \times 10^{-3}$, i.e. defining M such that $\sigma_M/\sigma_1 \leqslant TOL < \sigma_{M+1}/\sigma_1$ where the matrices $\mathbf{U}^M$, $\mathbf{\Sigma}^M$ and $(\mathbf{V}^H)^M$ are created by taking the first M columns of $\mathbf{U}$, first M rows of $\mathbf{V}^H$ and first M rows and columns of $\mathbf{\Sigma}$.

---

[1]Loosely speaking, given the decay of the solutions to (2.6) and (2.7), this truncation is such that the difference between the continuous solutions on the bounded and unbounded domains is not more than 1%, but typically much smaller in practice.

(a) Linearly spaced ω_n

(b) Logarithmically spaced ω_n
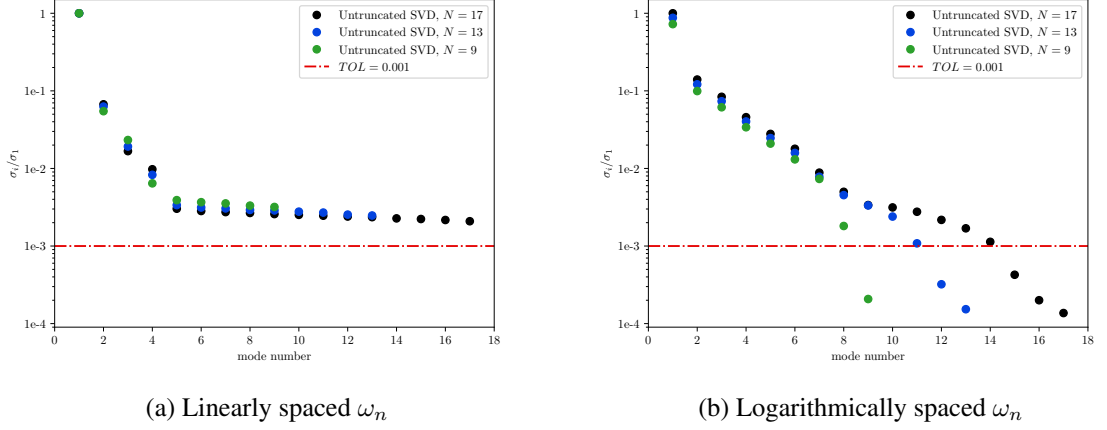
Figure 5.1: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ showing σ_i/σ_1 for (a) linearly spaced ω_n and (b) logarithmically spaced ω_n.

The superior performance of logarithmically spaced frequencies over those linearly spaced is illustrated in Figure 5.2 using the error measure $|e(\Lambda_i(\omega))| := |\Lambda_i^{exact}(\omega) - \Lambda_i^{PODP}(\omega)|/|\Lambda_i^{exact}(\omega)|$ with ω, where $\Lambda_i(\omega) = \lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]) + i\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, $\lambda_i(\cdot)$ indicates the ith eigenvalue and where the exact solution has been calculated from $(2.14)^2$. Notice that choosing ω_n to be logarithmically spaced for the representative full order model solution snapshots results in a smaller error compared to choosing ω_n to be linear spaced and also shows an algebraic increase with ω for $N > 5$.
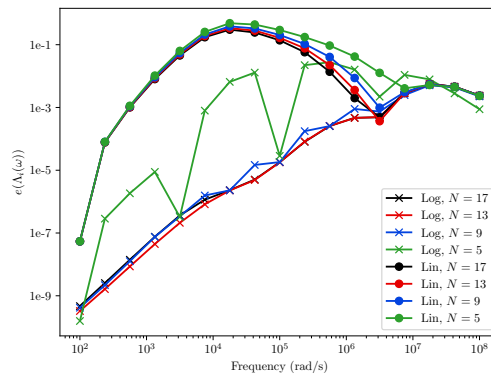
Figure 5.2: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ showing variation of $e(\Lambda_i(\omega))$ with ω for linearly and logarithmically spaced frequencies.

---

[2]Further details of the error associated with the PODP approximation for this example can be found in the documentation of the `MPT-Calculator` [134].

Further tests reveal that the accuracy of the PODP using $N = 9, 13, 17$ and logarithmically spaced ω_n remains similar to that shown in Figure 5.2 for $TOL \leqslant 1 \times 10^{-3}$ for this problem. Next, in Figure 5.3, a comparison of $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω, for the full order model, PODP using $N = 9$ and the exact solution. Again, the results for $i = 1, 2, 3$ are identical and, hence, only $i = 1$ is shown. In this figure, excellent agreement can be observed between PODP, the full order model solution and exact solution.

(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$

Figure 5.3: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $N = 9$ and $TOL = 1 \times 10^{-4}$ showing (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$ each with ω.

In Figure 5.4, the certification of the output $(\tilde{\mathcal{R}}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$ (summation of repeated indices is not implied) and $(\mathcal{I}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$, each with ω are shown. These are obtained using the a-posteriori error estimate in Lemma 3.3.1 and computed using the technique described in Section 3.3.3 for the case where $i = 1$ and with $N = 17, 21$ and $TOL = 1 \times 10^{-6}$. Similar certification can be obtained for the other tensor coefficients. The output certification is observed to be almost indistinguishable from the the MPT coefficients obtained with PODP for low frequencies in both cases and the error estimates rapidly tend to 0 for all ω as N is increased. Note that the a-posteriori error bounds always vanish when ω corresponds to an ω_n used for the representative full order solution snapshots. The larger error estimates for higher frequencies indicate that the MPT coefficients obtained by PODP for these frequencies is less reliable than those for smaller frequencies. However, smaller error bounds and increased reliability is achieved by increasing N as shown in Figure 5.4 (c) and (d) compared to Figure 5.4 (a) and (b). Note that $TOL = 1 \times 10^{-6}$ is chosen as larger tolerances lead to larger error estimates, however, this reduction in tolerance does not substantially affect the computational cost of the ROM. Although the effectivity indices $(\Delta[\omega])_{11}/|(\mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r] - \mathcal{R}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{11}|$ and $(\Delta[\omega])_{11}/|(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r] - \mathcal{I}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{11}|)$ of the PODP with respect to the full order model are clearly larger at higher frequencies, it

is emphasised that they are computed at negligible additional cost, they converge rapidly to the MPT coefficients obtained with PODP as N is increased and give credibility in the PODP solution without the need of performing additional full order model solutions to validate the ROM.

(a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}, N = 17$

(b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}, N = 17$

(c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}, N = 21$

(d) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}, N = 21$
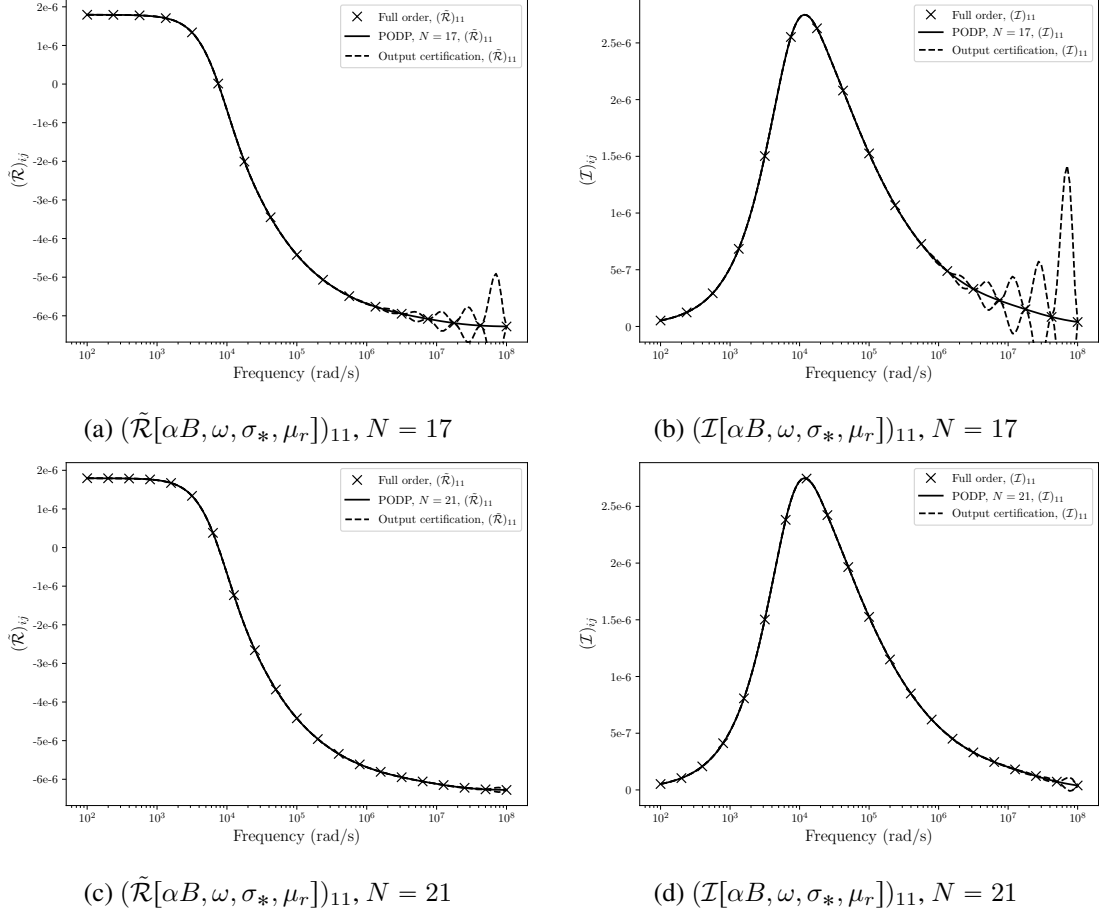
Figure 5.4: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $TOL = 1 \times 10^{-6}$ showing the PODP solution, full order model solutions and certification of the output $(\cdot) \pm (\Delta[\omega])_{11}$ for (a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 17$, (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 17$, (c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 21$ and (d) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 21$, each with ω.

Alternatively, smaller error certificates and increased fedility can be achieved by choosing the additional ω_n for the representative offline solution snapshots corresponding to where $(\Delta[\omega])_{ij}$ is largest and, if desired, this could be used as part of an adaptive process in a similar manner to [55]. In Figure 5.5 the effect of adding an additional, targeted, snapshot at the value of ω where $\sum_{i=1}^{3} \sum_{j=1}^{3} (\Delta[\omega])_{ij}$ is largest is shown. Initially, setting $N = 17$ and $TOL = 1 \times 10^{-6}$, the results in Figure 5.5 (a) and (b) are obtained, where the error certificates are found to be largest at $\omega = 7.08 \times 10^7$ rad/s. Adding an additional snapshot

$\omega_n = 7.08 \times 10^7$ rad/s, such that $N = 18$ and $TOL = 1 \times 10^{-6}$, reduces the size of the error certificates shown in Figure 5.5 (c) and (d). Comparing, Figure 5.4 (c), (d), and Figure 5.5 (c), (d), the reduction in $(\Delta[\omega])_{ij}$ using a single targeted snapshot is far greater than merely increasing the number of logarithmically spaced snapshots. Note that when additional snapshots are produced, the projection $\mathbf{U}^M$, must be recomputed. This means the adaptive scheme has additional computational overheads of recomputing the projection $\mathbf{U}^M$ for each additional snapshot (or set of snapshots).

(a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$, $N = 17$

(b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$, $N = 17$

(c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$, $N = 18$

(d) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$, $N = 18$
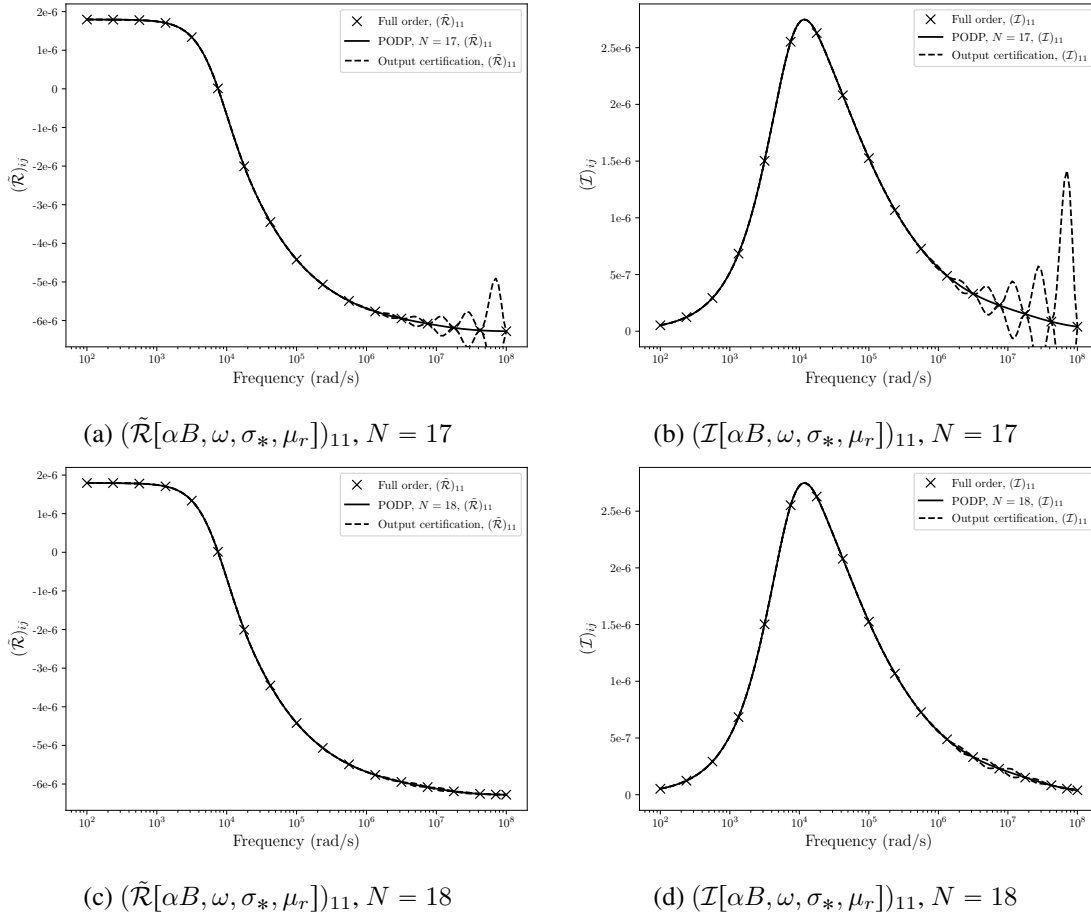
Figure 5.5: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $TOL = 1 \times 10^{-6}$ showing the PODP solution, full order model solutions and certification of the output $(\cdot) \pm (\Delta[\omega])_{11}$ for (a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 17$, (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 17$, (c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 18$ and (d) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ using $N = 18$, each with ω.

The computational speed-ups offered by using the PODP compared to a frequency sweep performed with the full order model are shown in Figure 5.6 (a) where $N = 9, 13, 17$ and logarithmically spaced ω_n are chosen with $\omega_{min} = 1 \times 10^2$ rad/s, $\omega_{max} = 1 \times 10^8$ rad/s, as before. For the comparison, the number of output points N_0 produced in a frequency

sweep is varied and the time taken to produce each of these frequency sweeps is measured when using a 2.9 GHz quad core Intel i5 processor[3] and also show the percentage speed up offered by each of these PODP sweeps. Figure 5.6 (b) shows the break down of the computational time for the offline and online stages of the PODP for the case where $N = 13$. Note, in particular, that the computational cost increases very slowly with N_0 and that the additional cost involved in computing the output certification is small. The breakdown of computational costs for other N is similar. The implementation in `NGSolve`, and in the `MPT-calculator` tool, is parallelised and further reductions in time can be achieved by increasing the number of cores used. In particular, in `NGSolve` parallelism is exploited in many aspects, which include meshing, matrix assembly, linear algebra and iterative solution of the linear systems, and is further exploited in `MPT-calculator` through the computation of the representative full order model solution snapshots, computation of the ROM solutions at different output frequencies, computation of the PODP MPT coefficients and calculation of the a-posteriori error estimate.

(a) Sweep Time (b) Break Down of PODP Timings
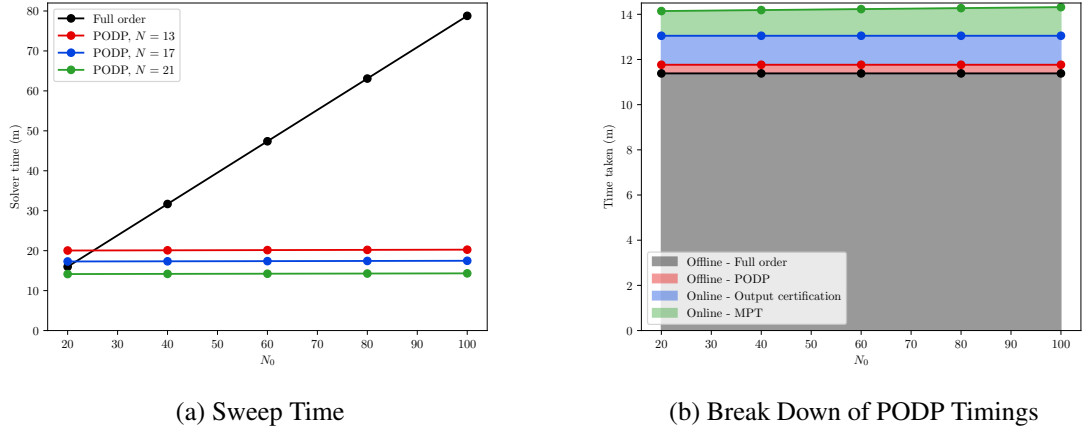
Figure 5.6: Sphere with $\mu_r = 1.5$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $TOL = 1 \times 10^{-6}$ showing, for different numbers of outputs N_0, (a) sweep computational time for $N = 13, 17, 21$ compared with full order and (b) a typical break down of the offline and online computational times for $N = 13$.

5.3   Conducting Permeable Torus

Next, consider $B_\alpha = \alpha B$ to be a torus where B has major and minor radii $a = 2$ and $b = 1$, respectively, $\alpha = 0.01$ m and the object is permeable and conducting with $\mu_r = 1.5$, $\sigma_* = 5 \times 10^5$ S/m. The object is centred at the origin so that it has rotational symmetry around the e_1 axis, therefore, $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ has independent coef-

---

[3]In subsequent chapters a more powerful machine will be used due to the large computational overhead for producing so many frequency sweeps.

ficients $(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$ and $(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{22} = (\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{33}$, and, thus, $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ each have 2 independent eigenvalues for each ω. Hence, there are only two independent curves for $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, as functions of ω. To compute the full order model, Ω is set to be a sphere of radius 100, centred at the origin and containing B, a mesh of 26 142 unstructured tetrahedra, refined towards the object is generated, and $p = 3$ elements are applied. This discretisation has already been found to produce an accurate representation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for the frequency range with $\omega_{min} = 1 \times 10^2$ rad/s and $\omega_{max} = 1 \times 10^8$ rad/s with the full order model.

The ROM is constructed using $N = 13$ representative full order solution snapshots that follow from using each value of the logarithmically spaced ω_n in turn and $TOL = 1 \times 10^{-4}$. Figure 5.7 shows the results for $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω, for both the full order model and the PODP. The agreement is excellent in both cases.

(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$
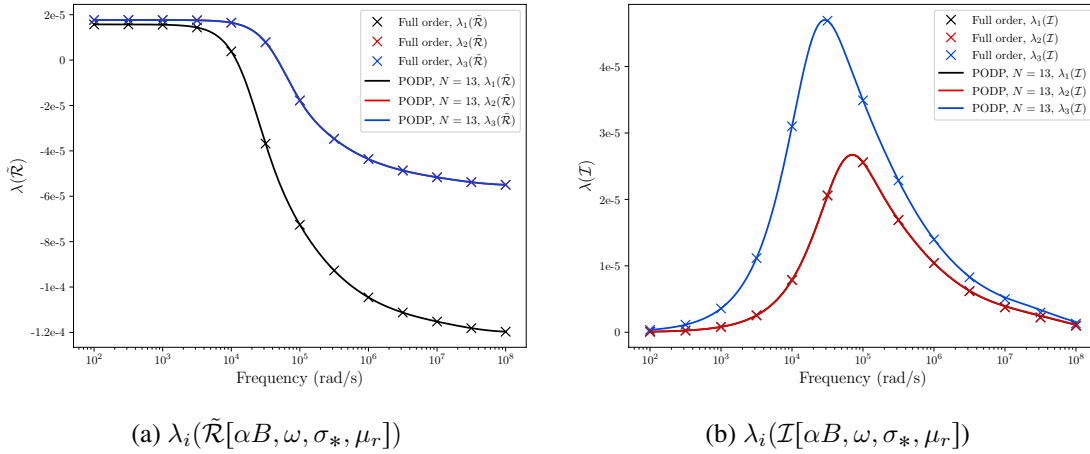
Figure 5.7: Torus with major and minor radii of $a = 2$ and $b = 1$, respectively, and $\mu_r = 1.5$, $\sigma_* = 5 \times 10^5$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ $N = 13$ and $TOL = 1 \times 10^{-4}$ showing (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω.

In Figure 5.8 the certification of the output $(\tilde{\mathcal{R}}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$ (no summation over repeated indices implied) and $(\mathcal{I}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$ are shown, each with ω. These are obtained using the a-posteriori error estimate in Lemma 3.3.1 and computed using the technique described in Section 3.3.3 for the case where $N = 17$ and $TOL = 1 \times 10^{-6}$. Note the increase in the number of snapshots from $N = 13$ to $N = 17$ and the reduction in tolerance to ensure a small error estimate.
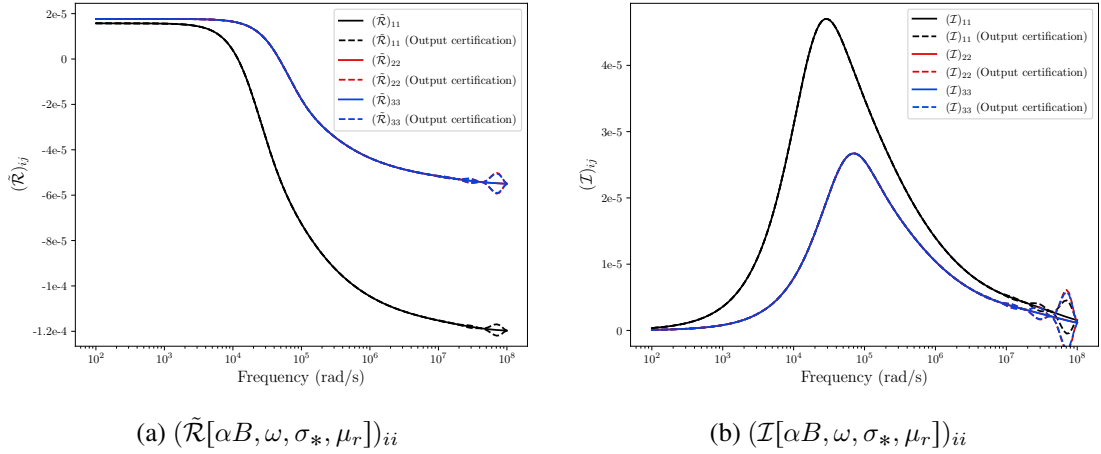
(a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$ (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$

Figure 5.8: Torus with $\mu_r = 1.5$, $\sigma_* = 5 \times 10^5$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $TOL = 1 \times 10^{-6}$ and $N = 17$ showing the PODP solution and certification of the output $(\cdot) \pm (\Delta[\omega])_{ii}$ for (a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \mu_r])_{ii}$, (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$, each with ω.

5.4 Conducting Permeable Tetrahedron

The third object considered is where $B_\alpha = \alpha B$ is a conducting permeable tetrahedron. The vertices of the tetrahedron B are chosen to be at the locations

$$v_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \ v_2 = \begin{pmatrix} 7 \\ 0 \\ 0 \end{pmatrix}, \ v_3 = \begin{pmatrix} 5.5 \\ 4.6 \\ 0 \end{pmatrix} \text{ and } v_4 = \begin{pmatrix} 3.3 \\ 2 \\ 5 \end{pmatrix}, \tag{5.1}$$

the object size is $\alpha = 0.01$ m and the tetrahedron is permeable and conducting with $\mu_r = 2$ and $\sigma_* = 5.96 \times 10^6$ S/m. The object does not have rotational or reflectional symmetries, therefore, $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ has 6 independent coefficients and, thus, $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ each have 3 independent eigenvalues for each ω. To compute the full order model, Ω is set to be a cube with sides of length 200 centred about the origin, a mesh of 21 427 unstructured tetrahedra, refined towards the object is generated, and $p = 3$ elements are applied. This discretisation has already been found to produce an accurate representation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for the frequency range with $\omega_{min} = 1 \times 10^2$ rad/s and $\omega_{max} = 1 \times 10^8$ rad/s.

The ROM is constructed using $N = 13$ representative full order solution snapshots that follow from using each value of the logarithmically spaced ω_n in turn and $TOL = 1 \times 10^{-4}$. Figure 5.9 shows the results for $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω, for both the full order model and the PODP. The agreement is excellent in both cases.

73
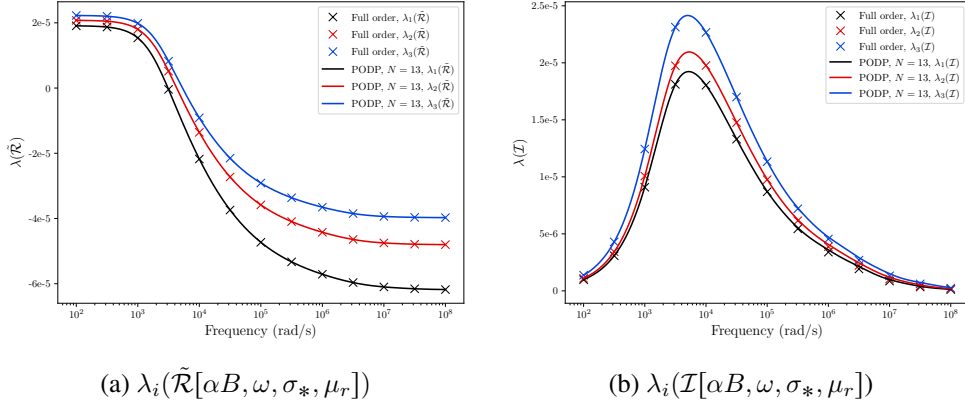
(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$

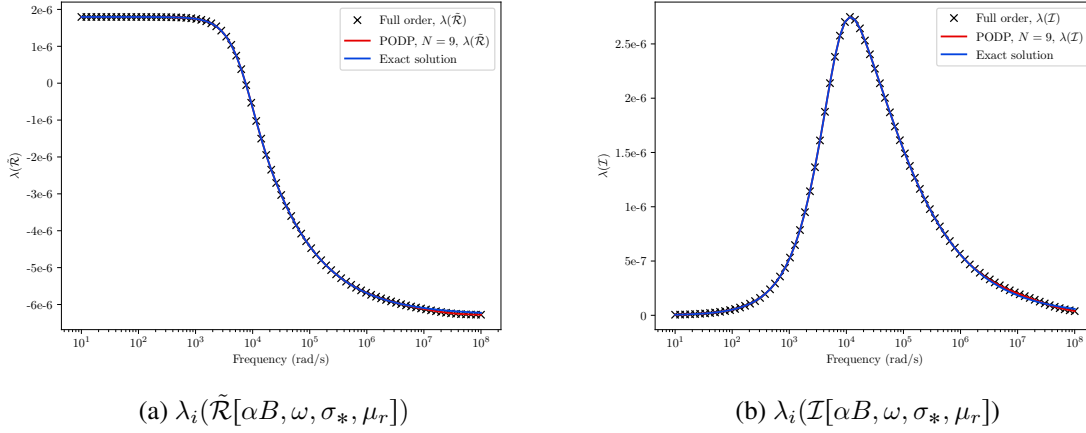(b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$

Figure 5.9: Irregular tetrahedron with $\mu_r = 2$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ $N = 13$ and $TOL = 1 \times 10^{-4}$ showing (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω.

(a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$

(b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$
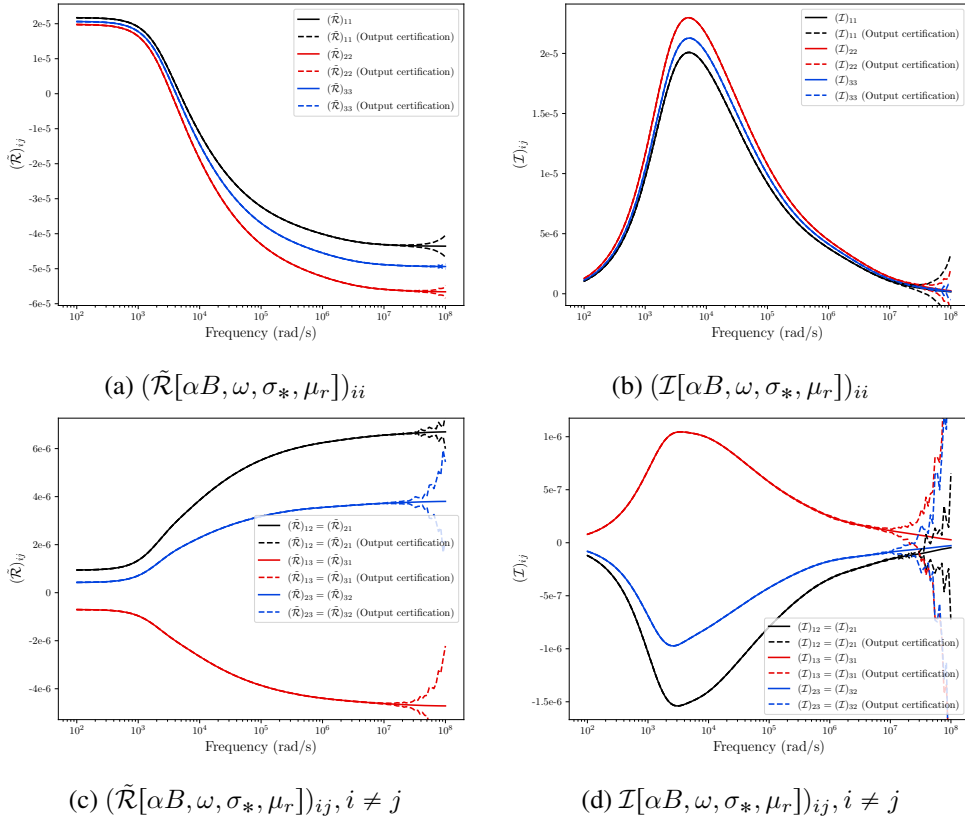
(c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}, i \neq j$

(d) $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}, i \neq j$

Figure 5.10: Irregular tetrahedron with $\mu_r = 2$, $\sigma_* = 5.96 \times 10^6$ S/m, $\alpha = 0.01$ m: PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $TOL = 1 \times 10^{-6}$ and $N = 21$ showing the PODP solution and certification of the output $(\cdot) \pm (\Delta[\omega])_{ij}$ for (a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$, with $i = j$ (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$, with $i = j$, (c) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$, with $i \neq j$, (d) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ with $i \neq j$, each with ω.

74

In Figure 5.10 the certification of the output $(\tilde{\mathcal{R}}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} \pm (\Delta[\omega])_{ij}$ and $(\mathcal{I}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} \pm (\Delta[\omega])_{ij}$ are shown, both with ω, for $i = j$ and $i \neq j$ obtained using the a-posteriori error estimate in Lemma 3.3.1 and computed using the technique described in Section 3.3.3 for the case where $N = 21$ and $TOL = 1 \times 10^{-6}$. Once again, increasing the number of snapshots from $N = 13$ to $N = 21$ and reducing the tolerance to ensure tight certificates bounds, except at large frequencies.

5.5   Inhomogeneous Conducting Bar

As a final example consider $B_\alpha = \alpha B$ to be the inhomogeneous conducting bar made up from two different conducting materials. The size, shape and materials of this object are the same as those presented in Section 6.1.3 of [74]. This object has rotational and reflectional symmetries such that $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ has independent coefficients $(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{11}$, $(\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{22} = (\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r])_{33}$ and, thus, $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ each have 2 independent eigenvalues for each ω. To compute the full order model, Ω is set to be a sphere of radius 100, centred about the origin, a mesh of 30 209 unstructured tetrahedra, refined towards the object is generated, and $p = 3$ elements are applied. This discretisation has already been found to produce an accurate representation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ for the frequency range with $\omega_{min} = 1 \times 10^2$ rad/s and $\omega_{max} = 1 \times 10^8$ rad/s.

The ROM is constructed using $N = 13$ representative full order solution snapshots that follow from using each value of the logarithmically spaced ω_n in turn and $TOL = 1 \times 10^{-4}$. Figure 5.11 shows the results for $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω, for both the full order model and the PODP. The agreement is excellent in both cases. The behaviour of $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ with ω for the inhomogeneous conducting bar is different to that for a homogeneous object, showing the presence of multiple non-local points of inflection rather being sigmoid with ω. Similarly, $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, for the inhomogeneous conducting bar, shows the presence of local maxima rather than a single maxima. Further details about the behaviour of MPT spectral signature of inhomogeneous objects can be found in [74].

In Figure 5.12 the output certificates $(\tilde{\mathcal{R}}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$ (no summation over repeated indices implied) and $(\mathcal{I}^{PODP}[\alpha B, \omega, \sigma_*, \mu_r])_{ii} \pm (\Delta[\omega])_{ii}$ are shown, both with ω. These are obtained using the a-posteriori error estimates in Lemma 3.3.1 and computed using the technique described in Section 3.3.3 for the case where $N = 23$ and $TOL = 1 \times 10^{-6}$. Note that increasing the number of snapshots from $N = 13$ to $N = 23$ and reducing the tolerance to ensure small error estimates, except at large frequencies.
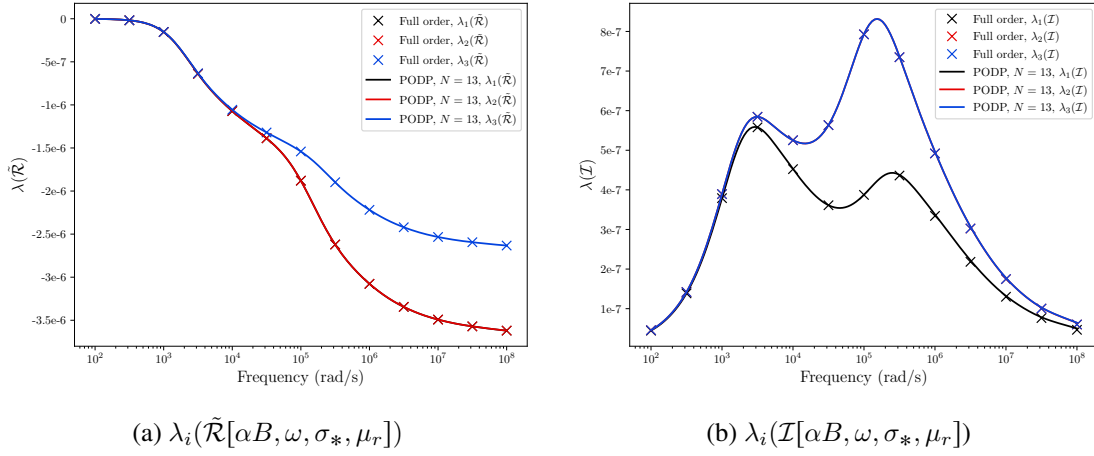
(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$

(b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$

Figure 5.11: Inhomogeneous bar with two distinct conductivities (see Section 6.1.3 of [74]): PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ $N = 13$ and $TOL = 1 \times 10^{-4}$ showing (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, each with ω.
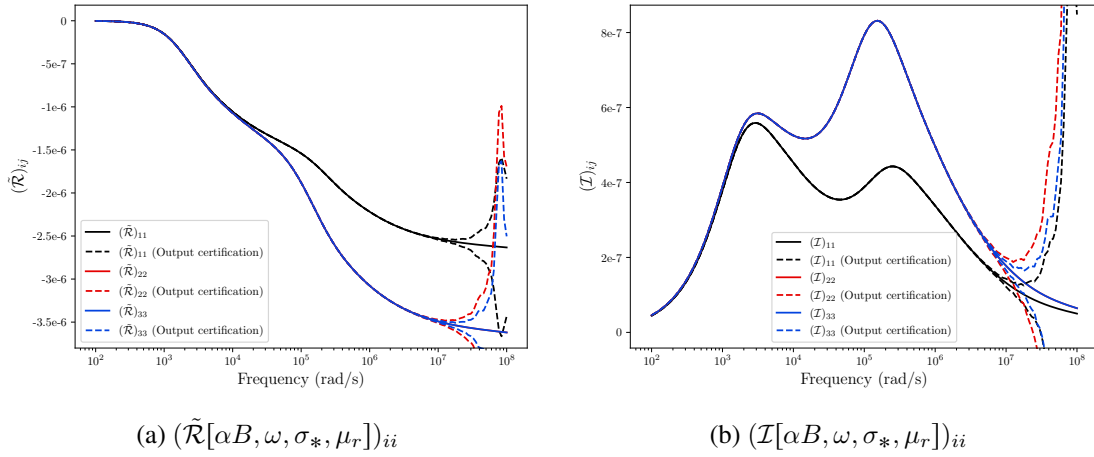
(a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$

(b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$

Figure 5.12: Inhomogeneous bar with two distinct conductivities (see Section 6.1.3 of [74]): PODP applied to the computation of $\mathcal{M}[\alpha B, \omega, \sigma_*, \mu_r]$ with $N = 23$ showing the PODP solution and certification of the output $(\cdot) \pm (\Delta[\omega])_{ii}$ for (a) $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$, (b) $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ii}$, each with ω.

76

5.6 Numerical Examples of Scaling

In this section the application of the results presented in Section 2.7 are illustrated.

5.6.1 Scaling of Conductivity

As an illustration of Lemma 2.7.1, consider a conducting permeable sphere $B_\alpha = \alpha B$ where $\alpha = 0.01$ m with materials properties $\mu_r = 1.5$ and $\sigma_*^{(1)} = 1 \times 10^7$ S/m and a second object, which has the same shape, size and permeability as the first except that $\sigma_*^{(2)} = s\sigma_*^{(1)} = 10\sigma_*^{(1)}$. In Figure 5.13, a comparison for the full order computations of $\mathcal{M}[\alpha B, \omega, \sigma_*^{(1)}, \mu_r]$ and $\mathcal{M}[\alpha B, \omega, \sigma_*^{(2)}, \mu_r]$ with that obtained from (2.30) is made. Observe that the translation predicted by (2.30) is in excellent agreement with the full order model solution for $\mathcal{M}[\alpha B, \omega, \sigma_*^{(2)}, \mu_r]$.

(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$
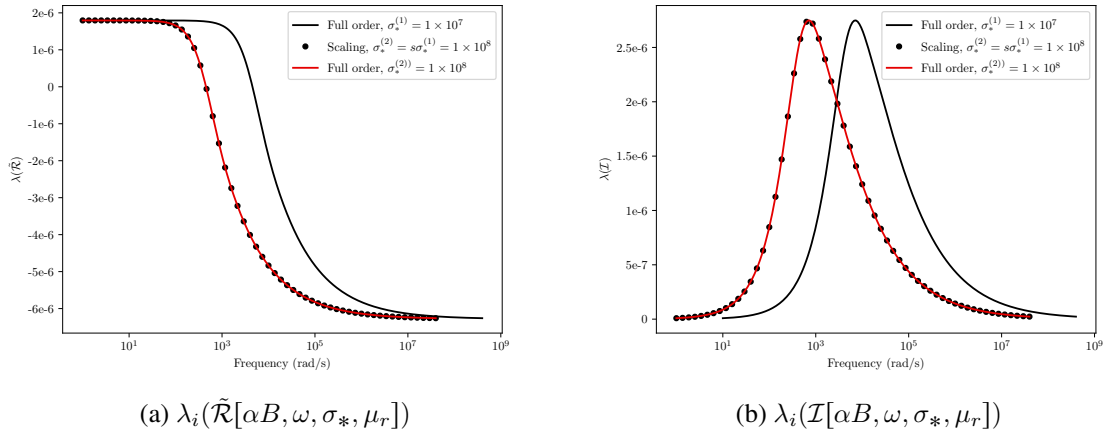
Figure 5.13: Sphere with $\mu_r = 1.5$, $\sigma_*^{(1)} = 1 \times 10^7$ S/m , $\alpha = 0.01$ m and second sphere, which is the same as the first except that $\sigma_*^{(2)} = s\sigma_*^{(1)} = 10\sigma_*^{(1)}$: showing the translation predicted by (2.30) compared with the full order model solutions for (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$.

5.6.2 Scaling of Object Size

To illustrate Lemma 2.7.2, consider a conducting permeable tetrahedron $B_\alpha^{(1)} = \alpha^{(1)} B = 0.01B$ with vertices as described in Section 5.4 and material properties $\mu_r = 1.5$ and $\sigma_* = 1 \times 10^6$ S/m. Then, consider a second object $B_\alpha^{(2)} = \alpha^{(2)} B = s\alpha^{(1)} B = 0.015B$, which, apart from its size, is otherwise the same as $B_\alpha^{(1)}$. In Figure 5.14, a comparison for the full order computations of $\mathcal{M}[\alpha^{(1)} B, \omega, \sigma_*, \mu_r]$ and $\mathcal{M}[\alpha^{(2)} B, \omega, \sigma_*, \mu_r]$ with that obtained from (2.31) is made. Observe that the translation and scaling predicted by (2.31) is in excellent agreement with the full order model solution for $\mathcal{M}[\alpha^{(2)} B, \omega, \sigma_*, \mu_r]$.

(a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$
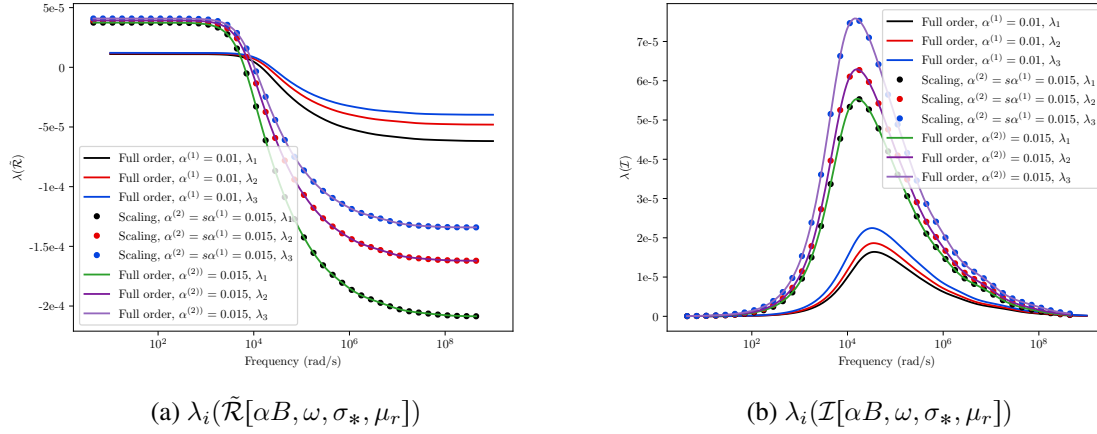
Figure 5.14: Tetrahedron $B_\alpha^{(1)} = \alpha^{(1)}B = 0.01B$ with $\mu_r = 1.5$ and $\sigma_* = 1 \times 10^6$ S/m, $\alpha = 0.01$ m and a second tetrahedron, which is the same as the first except that $B_\alpha^{(2)} = \alpha^{(2)}B = s\alpha^{(1)}B = 0.015B$: showing the translation and scaling predicted by (2.31) compared with the full order model solutions for (a) $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$.

5.7 Chapter Summary

In this chapter a series of illustrative examples for the implementation of the ROM, with a-posteriori error estimates, was presented, which demonstrated the accuracy and efficiency of the method. First, the choice of linearly vs logarithmically chosen full order snapshots was compared, where the associated singular values from the TSVD for varying numbers of full order snapshots and the relative error produced in the eigenvalues of the MPT for a conducting sphere was compared for the two choices. Second, the results of an implementation of the a-posteriori error certificates derived in Lemma 3.3.1 was presented, showing that error certificates can be reduced in size by increasing the number of snapshots and a brief description was included on how an adaptive scheme may be created with the error certificates being used to inform the choices of future snapshots. Third, the time savings offered with the implementation of the ROM was discussed. Fourth, a series of illustrative examples producing MPT spectral signatures, using the ROM, for a range of homogeneous geometric objects, along with presenting results for an inhomogenous conducting bar, showing error certificates for all cases were presented. Lastly, numerical examples of the scaling results derived in Section 2.7 were presented, showing the possibility of producing spectral signatures for objects which vary in size and or conductivity at negligible computational cost.

Part III

Object Characterisation

Chapter 6

Decomposing the Inverse Problem to one of Object Location and Object Classification

6.1 Introductory Remarks

This chapter describes the decomposition of the inverse problem, to one of object location and object classification, the latter of which will be based on object characterisations that are invariant under position and rotation, which can be obtained from the MPT coefficients.

The chapter has been adapted and extended from the work presented by the author in [76]. The key novelty of the chapter is the proposal of several candidate MPT spectral signature invariants, which would make ideal ML features including: tensor eigenvalues; tensor invariants, both principal and deviatoric; and eigenvalues of the commutator or the real and imaginary part of the MPT.

The structure of the chapter is as follows: In Section 6.2, the decomposition of the inverse problem into one of location and characterisation is considered and methods for determining object location via a MUSIC algorithm along with a method for determining MPT coefficients by solving an overdetermined linear system are presented. In Section 6.3, candidate MPT spectral signature invariants are presented and discussed, these include tensor eigenvalues, tensor invariants, both principal and deviatoric, and eigenvalues of the commutator. The chapter is concluded in Section 6.4 with some closing remarks.

6.2 Inverse Problem

Recall from Section 2.2.2 that the inverse problem is to determine the location, shape and material properties (σ_* and μ_*) of the conducting object B_α from measurements of $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ taken at a range of locations $\boldsymbol{x}$ in the air. A traditional approach to the solution of this inverse problem involves creating a discrete set of voxels, each with unknown σ and μ, and posing the solution to the inverse problem as an optimisation process in

which σ and μ are found through minimisation of an appropriate functional e.g. [122]. From the resulting images of σ and μ one then attempts to infer the shape and position of the object. However, this problem is highly ill-posed [24] and presents considerable challenges mathematically and computationally in the case of limited noisy measurement data. Instead, the problem may be simplified by first splitting the problem into two, one of location, and one of classification. In the following sections the combination of a MUSIC (multiple signal classification) algorithm for location and measurement of the MPT coefficients will be discussed.

6.2.1 Location of a Hidden Object

The task of determining the MPT of a hidden object separates in to first determining the position of the object z and secondly determining the MPT coefficients from the measurements. The former can be done using a range of methods, some common techniques include the MUSIC algorithm proposed by [13, 14], for which a method for multiple object location has been presented by Ledger *et al* in [74]; A voxelated approach such as commonly employed for imaging conductivity in MIT (previously discussed in Chapter 1) could be used to determine a targets location (only) rather than seeking an accurate conductivity distribution by using a coarse voxelated discretisation. Alternatively an extension of the approach described in Section 6.2.2, where both the coefficients of the MPT and the object's location are sought together by posing a functional to be minimized for both could be employed e.g. [86, 87].

6.2.2 Measurement of MPT Coefficients

In the electrical engineering community, the measurement of MPT coefficients as a function of frequency is sometimes called MPT spectroscopy and procedures have been developed for anti-personal landmine detection [94, 10, 33, 7] as well as for MPT measurement and real time classification for security screening [86, 87, 82, 83, 141] and other applications. The basic idea is that background fields $\boldsymbol{H}_0^{(m)}$, $m = 1, \ldots, M_e$, from M_e different exciters are generated, in turn, and measurements of the resulting magnetic field perturbation $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0^{(m)})(\boldsymbol{x})$ caused by the presence of a hidden conducting permeable object are made at sufficiently many positions $\boldsymbol{x}$ around the object for a range of exciting frequency. Considering a single frequency excitation, the perturbed magnetic field is usually measured as an induced voltage in the form

$$\mathrm{V}_{nm}^{\mathrm{ind}} = \int_{C^{(n)}} \boldsymbol{n} \cdot (\boldsymbol{H}_\alpha - \boldsymbol{H}_0^{(m)})(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{6.1}$$

where $n = 1, \ldots, M_r$ correspond to the different receiver coils and $C^{(n)}$ is an appropriate surface related to coil n [71]. In light of (2.4), and the MPT object characterisation, we see that the leading order term gives an approximation to $\mathrm{V}_{nm}^{\mathrm{ind}}$ and the accuracy of the approximation will depend on the uniformity of $\boldsymbol{H}_0(\boldsymbol{z})$ as well as the object size. The measurements $\mathrm{V}_{nm}^{\mathrm{ind, measured}}$ will also have unavoidable errors and noise from a range of

sources, accepting these errors and noise, determining the MPT's coefficients from an over determined linear system of the form

$$\mathbf{Au} = \mathbf{b}, \tag{6.2}$$

using least squares where Roman fonts are used for matrices and vectors in linear systems. In the above, $\mathbf{u}$, $\mathbf{A}$ and $\mathbf{b}$ have the entries

$$(\mathbf{u})_j = \mathrm{u}_j = (\mathcal{M})_{qr},$$

$$(\mathbf{A})_{ij} = \mathrm{A}_{ij} = (\boldsymbol{H}_0^{(m)}(\boldsymbol{z}))_r \int_{C^{(n)}} (\boldsymbol{D}_{\boldsymbol{x}}^2 G(\boldsymbol{x}, \boldsymbol{z})\boldsymbol{n})_q \mathrm{d}\boldsymbol{x},$$

$$(\mathbf{b})_i = \mathrm{b}_i = \int_{C^{(n)}} \boldsymbol{n} \cdot (\boldsymbol{H}_\alpha - \boldsymbol{H}_0^{(m)})(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx V_{nm}^{\text{ind, measured}},$$

where $i = (n-1)M_e + m$, $m = 1, \ldots, M_e$, $n = 1, \ldots, M_r$ $j = 3(q-1) + r$, $q, r = 1, 2, 3$. In addition, the known symmetry of $\mathcal{M}$ can be used to reduce the number of complex unknowns in (6.2) from 9 to 6. Clearly, we need the product $M_e M_r > 6$ in order for the system to be over determined, however, it is not only important that we have sufficiently many measurements, but also that the location of emitting and receiving coils are correctly chosen so that all of the coefficients of $\mathcal{M}$ can be determined. For further details, and an algorithm of how this can be automated, see [71]. The discrete, measured spectral signature of $\mathcal{M}$ follows by repeating the above process using $V_{mn}^{\text{ind, measured}}$ at different excitation frequencies ω_m, $m = 1, \ldots, M$[1] and, depending on the system and application, errors of around 1-5% can be expected with current systems [32, 82, 83]. Note that using a higher order expansion of (2.4), derived in [72], will lead to an improved approximation of V_{nm}^{ind} in (6.1), particularly if $\boldsymbol{H}_0(\boldsymbol{z})$ is non-uniform, and allow the coefficients of generalised MPTs to be found.

6.3 MPT Spectral Signature Invariants for Object Classification

Bishop [20] describes the process of classification as taking an input vector $\mathbf{x}$ and assigning it to one of K discrete classes C_k, $k = 1, \ldots, K$. For example, in security screening, the simplest form of classification with $K = 2$ involves only the classes *threat* (C_1) and *non-threat* (C_2), and one with a higher level of fidelity might include the classes of metallic objects such as key (C_1), coin (C_2), gun (C_3), knife (C_4) ... where the class numbers are assigned as desired. He recommends that it is convenient (in probabilistic methods of classification) to use a 1-of-K coding system in which the entries in a vector $\mathbf{t} \in \mathbb{R}^K$ take the form

$$\mathrm{t}_i := \begin{cases} 1 & \text{if } i = k \\ 0 & \text{otherwise} \end{cases},$$

---

[1]Note also, M is distinguished from the number of modes in the ROM and subsequently refers to the number of output frequencies, previously given the symbol F in Chapter 5. F will subsequently be used when describing feature for ML classification.

if the correct class is C_k. Requiring that we always have $\sum_{k=1}^{K} t_k = 1$, then this approach has the advantage that t_k can be interpreted as the probability that the correct class is C_k. In this section, we focus on alternative choices of the F features in the input vector $\mathbf{x} \in \mathbb{R}^F$ for the classifier. In subsequent chapters we will compare the performance of different ML classifiers based for these alternatives. The focus here is on suitable features that are invariant to rotation of the object. Note that the rank 2 MPT, and hence the invariants considered below, are invariant to the position of the object, which was previously discussed in Chapter 2.

6.3.1 Tensor Eigenvalues

Recall that the diagonal matrices $\mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]}$ and $\mathbf{\Lambda}^{\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]}$ contain the eigenvalues of $\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]$, respectively, and satisfy the object rotation invariant property

$$
\begin{aligned}
\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]) &= (\mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]})_{ii} \\
&= (\mathbf{\Lambda}^{\tilde{\mathcal{R}}[\alpha \mathbf{R}(B), \omega_m, \sigma_*, \mu_r]})_{ii} = \lambda_i(\tilde{\mathcal{R}}[\alpha \mathbf{R}(B), \omega_m, \sigma_*, \mu_r]), \\
\lambda_i(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]) &= (\mathbf{\Lambda}^{\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]})_{ii} \\
&= (\mathbf{\Lambda}^{\mathcal{I}[\alpha \mathbf{R}(B), \omega_m, \sigma_*, \mu_r]})_{ii} = \lambda_i(\mathcal{I}[\alpha \mathbf{R}(B), \omega_m, \sigma_*, \mu_r]),
\end{aligned}
$$

at each discrete frequencies ω_m, $m = 1, \ldots, M$ in the MPT spectral signature, where $\mathbf{R}$ is an orthogonal matrix describing the object rotation (see also Section 2.3.1). Thus, one option is to select the features for the classifier as

$$
(\mathbf{x})_i = \mathrm{x}_i = \left\{ \begin{array}{ll} \lambda_j(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m-1)M \\ \lambda_j(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m+2)M \end{array} \right. ,
$$

where $j = 1, 2, 3$ and $m = 1, \ldots, M$ so that $F = 6M$. This is particularly attractive, since any hidden object is likely to be in some unknown rotated configuration compared to canonical choice of the corresponding object in the library and, as the eigenvalues are invariant under object rotation, knowledge of the orthogonal rotation matrix $\mathbf{R}$ is not needed to perform the classification. Furthermore, in practice, measurements lead to noisy tensor coefficients in the form $\mathcal{M} + \mathcal{E}_r + \mathrm{i}\mathcal{E}_\mathrm{i}$ where $\mathcal{E}_r + \mathrm{i}\mathcal{E}_\mathrm{i}$ is a complex symmetric rank 2 tensor and represents the noise. To understand the effects of noise, consider for simplicity a symmetric real matrix $\mathbf{A}$ corrupted by a real symmetric $\mathbf{E}$, applying results on eigenvalue perturbations [42] it can be shown that

$$
\sum_{i=1}^{3} (\lambda_i(\mathbf{A} + \mathbf{E}) - \lambda_i(\mathbf{A}))^2 \leqslant \|\mathbf{E}\|_F^2,
$$

so that the eigenvalues λ_i of $\mathbf{A}$ are similar to those of $\mathbf{A} + \mathbf{E}$ provided $\mathbf{E}$ represents the low-moderate noise. However, for an eigenvalue-eigenvector pair $\lambda_1, \mathbf{q}_1$ [42]

$$
\mathrm{dist}(\mathbf{q}_1(\mathbf{A}), \mathbf{q}_1(\mathbf{A} + \mathbf{E})) \leqslant \frac{4}{d} C(\mathbf{E}),
$$

where $d = \min_{\mu \in \lambda_i(\mathbf{A}), \mu \neq \lambda_i} |\lambda_i - \mu| > 0$ and $C(\mathbf{E})$ is a constant depending on $\mathbf{E}$. In other words, if the eigenvalues are close (so d is small), the eigenvectors will be badly effected by the noise. The same applies to the real and imaginary parts of $\mathcal{M} + \mathcal{E}_r + \mathrm{i}\mathcal{E}_\mathrm{i}$ when the coefficients are arranged as matrices.

6.3.2 Tensor Invariants

While $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega_m, \mu_r, \sigma_*])$, $\lambda_i(\mathcal{I}[\alpha B, \omega_m, \mu_r, \sigma_*])$, $m = 1, \ldots, M$, are invariant under object rotation, their behaviour is well understood and they behave well for noisy measurements, classifying objects on the basis of these may still cause practical issues. Firstly, care is needed with the ordering of the eigenvalues since choosing a simple rule such as $\lambda_1 \geqslant \lambda_2 \geqslant \lambda_3$ may lead to confusing results. For example, if the object has rotational and/or reflectional symmetries, one might find there are only 2 independent eigenvalues at each frequency in the real and imaginary parts of the MPT, then, applying the aforementioned rule independently to $\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]$ could lead to $\lambda_2(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]) = \lambda_3(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r])$ and $\lambda_1(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]) = \lambda_2(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r])$. Secondly, there is a danger that different ordering rules are applied in the creation of the training library for the classifier compared to that used for testing some new candidate object. To overcome this, tensor invariants can be used, which are independent of how λ_1, λ_2 and λ_3 are assigned. One possibility are the principal tensor invariants, which, for a rank 2 tensor $\mathcal{A}$, are (e.g. [22])

$$I_1(\mathcal{A}) := \mathrm{tr}\,(\mathcal{A}) = \lambda_1(\mathcal{A}) + \lambda_2(\mathcal{A}) + \lambda_3(\mathcal{A}), \tag{6.3a}$$

$$I_2(\mathcal{A}) := \frac{1}{2}\left(\mathrm{tr}\,(\mathcal{A})^2 - \mathrm{tr}\,(\mathcal{A}^2)\right) = \lambda_1(\mathcal{A})\lambda_2(\mathcal{A}) + \lambda_1(\mathcal{A})\lambda_3(\mathcal{A}) + \lambda_2(\mathcal{A})\lambda_3(\mathcal{A}), \tag{6.3b}$$

$$I_3(\mathcal{A}) := \det(\mathcal{A}) = \lambda_1(\mathcal{A})\lambda_2(\mathcal{A})\lambda_3(\mathcal{A}), \tag{6.3c}$$

which contain the same information as the tensor's eigenvalues $\lambda_i(\mathcal{A})$ and can also be computed from (6.3). They satisfy

$$\lambda^3 - I_1(\mathcal{A})\lambda^2 + I_2(\mathcal{A})\lambda - I_3(\mathcal{A}) = 0, \tag{6.4}$$

are rotationally invariant and, like the eigenvalues, are less-susceptible to noise than the tensor's eigenvectors.

Borrowing notation from continuum mechanics (e.g. [22]), $I_1(\mathcal{A})$ is related to the *hydrostatic part* of $\mathcal{A}$ given by $\mathcal{H} = \frac{1}{3}\mathrm{tr}(\mathcal{A})\mathbb{I}$, where $\mathbb{I}$ is the identity matrix and is associated with the extent to which the operation $\mathcal{H}\boldsymbol{v}$ *stretches* or *shrinks* the magnitude of $\boldsymbol{v}$. The invariant $I_2(\mathcal{A})$ is often, but not exclusively, related to the *deviatoric part* of $\mathcal{A}$ given by $\mathcal{S} = \mathcal{A} - \mathcal{H}$ describing the extent to which $\mathcal{S}\boldsymbol{v}$ distorts the components of $\boldsymbol{v}$. The invariant $I_3(\mathcal{A})$ describes the extent of coupling of the two aforementioned cases and whether or not the tensor $\mathcal{A}$, when arranged as a 3×3 matrix, is singular or not. In addition, when applied to (limiting cases) of $\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]$, it has a further physical interpretation: Recall that the product αB implies that there are an infinite number of ways to choose $\alpha \ll 1$ and B, which still result in the same αB. For example,

if $|B|$ is chosen such that $\det(\mathcal{N}^0[\alpha B, \mu_r]) = I_3(\mathcal{N}^0[\alpha B, \mu_r]) = \alpha^9$ then this invariant provides object size information, while, in general $I_3(\mathcal{N}^0[\alpha B, \mu_r])$, will be a function of $|B|$, α and μ_r. Similarly, $I_3(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r])$ and $I_3(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r])$ will be functions of $|B|$, α, ω_m, σ_* and μ_r. Thus, $I_3(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r])$ and $I_3(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r])$, for fixed α, ω_m, σ_* and μ_r, will scale like $|B|^3$. This in turn will lead to larger maxima for $I_3(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$, when considering objects with a larger volume, this phenomena can be observed and is commented on in Chapter 7. Using principal invariants, we could then select the features as

$$
\mathrm{x}_i = \begin{cases} I_j(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m-1)M \\ I_j(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m+2)M \end{cases}, \tag{6.5}
$$

where $j = 1, 2, 3$ and $m = 1, \ldots, M$ so that $F = 6M$.

As an alternative to the principal invariants stated in (6.3), the alternative set of invariants (e.g. [22])

$$
I_1(\mathcal{A}) := \mathrm{tr}(\mathcal{A}) = \lambda_1(\mathcal{A}) + \lambda_2(\mathcal{A}) + \lambda_3(\mathcal{A}), \tag{6.6a}
$$

$$
J_2(\mathcal{A}) := \frac{1}{2}\mathrm{tr}(\mathcal{S}^2) = \frac{1}{3}I_1(\mathcal{A})^2 - I_2(\mathcal{A}) = \frac{1}{2}(s_1(\mathcal{S})^2 + s_2(\mathcal{S})^2 + s_3(\mathcal{S})^2)
$$
$$
= \frac{1}{2}((\lambda_1(\mathcal{A}) - I_1(\mathcal{A})/3)^2 + (\lambda_2(\mathcal{A}) - I_1(\mathcal{A})/3)^2 + (\lambda_3(\mathcal{A}) - I_1(\mathcal{A})/3)^2), \tag{6.6b}
$$

$$
J_3(\mathcal{A}) := \det(\mathcal{S}) = \frac{2}{27}I_1(\mathcal{A})^3 - \frac{1}{3}I_1(\mathcal{A})I_2(\mathcal{A}) + I_3(\mathcal{A}) = \frac{1}{3}s_1(\mathcal{S})s_2(\mathcal{S})s_3(\mathcal{S})
$$
$$
= (\lambda_1(\mathcal{A}) - I_1(\mathcal{A})/3)(\lambda_2(\mathcal{A}) - I_1(\mathcal{A})/3)(\lambda_3(\mathcal{A}) - I_1(\mathcal{A})/3), \tag{6.6c}
$$

where $\lambda_i(\mathcal{A}) = s_i(\mathcal{A}) + I_1(\mathcal{A})/3$ can be used. These invariants satisfy

$$
s^3 - J_2(\mathcal{A})s - J_3(\mathcal{A}) = 0,
$$

and the roots of this equation are the eigenvalues s_i of $\mathcal{S}$. The invariants $J_2(\mathcal{A})$ and $J_3(\mathcal{A})$ are both related to the extent to which $\mathcal{S}v$ distorts the components of v. In this case, we can select the features as

$$
\mathrm{x}_i = \begin{cases} I_1(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = 1 + (m-1)M \\ J_j(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m-1)M \\ I_1(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = 1 + (m+2)M \\ J_j(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m+2)M \end{cases}, \tag{6.7}
$$

where $j = 2, 3$ and $m = 1, \ldots, M$ and $F = 6M$. One potential advantage of using (6.6) as a set of features is that, for the case where $\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]$ are each a multiple of identity (such as for the MPT characterisation of a cube or sphere), J_2 and J_3 vanish.

$$(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} \qquad\qquad \lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$$
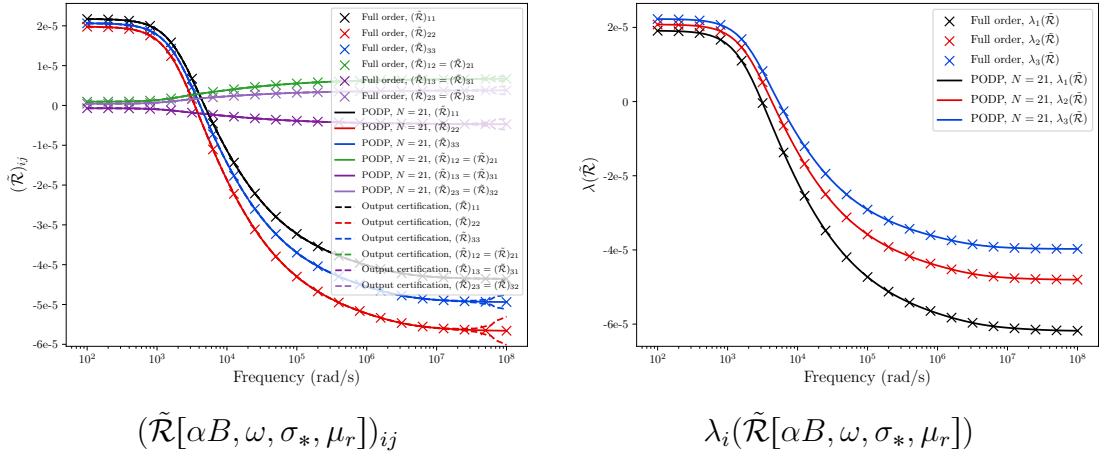
Figure 6.1: Irregular tetrahedron B with vertices (5.1), $\alpha = 0.01$ m $\mu_r = 2$ and $\sigma_* = 5.96 \times 10^6$ S/m: Comparison of $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ and $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$.

6.3.3 Eigenvalues of the Commutator of $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$

The off-diagonal entries $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ and $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ with $i \neq j$ are much smaller than those on the diagonal with $i = j$ as the results

$$|(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}| \leqslant |\mathrm{tr}(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])| = \left| \sum_{k=1}^{3} \lambda_k(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]) \right|, \quad (6.8\mathrm{a})$$

$$(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} \leqslant \mathrm{tr}(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]) = \sum_{k=1}^{3} \lambda_k(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]), \quad (6.8\mathrm{b})$$

obtained in Lemma 6.1 of [73] show. This implies that the tensor's eigenvalues, and hence its principal invariants (6.3) (as well as the alternative invariants (6.6)), are dominated by the behaviour of its diagonal coefficients. To illustrate this, Figure 6.1 shows the comparison between $\lambda_i(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ for the irregular tetrahedron discussed in Section 2.6.1. Observe that the behaviour of the eigenvalues is dominated by the diagonal coefficients of $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$, similar arguments also apply to $\lambda_i(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$ and $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$.

To improve the discrimination between objects whose tensors $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ have different eigenvectors, their commutator is considered, which has coefficients

$$(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])_{ij} := (\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ik}(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{kj}$$
$$- (\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ik}(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{kj} \quad (6.9)$$

where Einstein summation convention of the indices is implied. The commutator measures how different the eigenspaces of $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are. It vanishes when the $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are simultaneously diagonalisable

87

(i.e. the eigenvectors of $\mathbf{Q}^{\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}^{\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]} = \mathbf{Q}(B)$ are the same). In Lemma 8.11 of [73], $|(\mathcal{Z}[\alpha B,\omega,\sigma_*,\mu_r])_{ij}|$ has been shown to grow at most linearly with ω. In addition, the coefficients of $\mathcal{Z}$ transform as a rank 2 tensor and so the eigenvalues of $\mathcal{Z}[\alpha B]$ and $\mathcal{Z}[\alpha \mathbf{R}(B)]$ are the same.

It is easy to show that, since $\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r]$ and $\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r]$ are symmetric, $\mathcal{Z}[\alpha B,\omega,\sigma_*,\mu_r]$ has vanishing diagonal coefficients and is skew symmetric. Then, by arranging the coefficients of $\mathcal{Z}[\alpha B,\omega,\sigma_*,\mu_r]$ as a 3×3 matrix, we find that its eigenvalues are zero or purely imaginary

$$\lambda_i(\mathcal{Z}) \in \left\{ 0, \pm \mathrm{i}\sqrt{(\mathcal{Z})_{12}^2 + (\mathcal{Z})_{13}^2 + (\mathcal{Z})_{23}^2} \right\},$$

and, thus, $\sqrt{(\mathcal{Z})_{12}^2 + (\mathcal{Z})_{13}^2 + (\mathcal{Z})_{23}^2} = \sqrt{I_2(\mathcal{Z}[\alpha B,\omega,\sigma_*,\mu_r])}$ is useful as an additional classifier for situations where the off-diagonal coefficients of the tensors are amongst its independent coefficients [2]. For an object where the only independent coefficients $(\tilde{\mathcal{R}}[\alpha B,\omega,\sigma_*,\mu_r])_{ij}$ and $(\mathcal{I}[\alpha B,\omega,\sigma_*,\mu_r])_{ij}$ are associated with $i = j$ then $\sqrt{(\mathcal{Z})_{12}^2 + (\mathcal{Z})_{13}^2 + (\mathcal{Z})_{23}}$ vanishes. This invariant can easily be added to the list of features in (6.3), (8.1) or (6.7) as

$$\mathrm{x}_i = \sqrt{I_2(\mathcal{Z}[\alpha B,\omega_m,\sigma_*,\mu_r])}, \qquad i = 6M + m, \tag{6.10}$$

for $m = 1,\ldots,M$ bringing the total number of features to $F = 7M$.

6.4 Chapter Summary

This chapter presented material detailing the decomposition of the inverse problem, to one of location and classification, the latter of which was restricted to be based on characterisations that are invariant under position and rotation, which can be obtained from the MPT coefficients. First, the MUSIC algorithm for locating hidden targets is briefly reviewed and the method documented. Second, a method of determining the coefficients of the MPT by creation of an over determined linear system wass presented. Third, a systematic review of candidate MPT spectral signature invariants is given, this proposed, tensor eigenvalues, both principal and deviatoric tensor invariants, and eigenvalues of the commutator. The key novelty of the chapter is the proposal of these candidate MPT spectral signature invariants as features for a ML classifier, with the principal and deviatoric invariants not being considered previously. In Chapter 7, results will be presented to compare MPT spectral signature invariants for realistic threat and non-threat objects. Additionally the idea to use MPT spectral signature invariants will be further developed in Chapter 8 with numerical results being presented in Chapter 9.

---

[2] Note that $I_1(\mathcal{Z}[\alpha B,\omega,\sigma_*,\mu_r]) = I_3(\mathcal{Z}[\alpha B,\omega_m,\sigma_*,\mu_r]) = 0$

Chapter 7

Real World Object Characterisation Dataset

7.1 Introductory Remarks

This chapter provides a series of illustrative examples to demonstrate how the ROM approach described in Chapter 3 can be combined with an appropriate choice of eigenvalues or tensor invariants in Section 6.3 and sampling at M frequencies to form a realistic dataset for object classification.

The chapter has been adapted and extended from the work presented by the author in [76, 135] and the full open source dataset, `MPT-Library` [132].

The novelties of the chapter are as follows: Firstly, the application of the ROM approach described in Chapter 3 to a set of real world geometries generating a first of its kind dataset, which is openly available for other people to use. Secondly, constructing this dataset based upon the invariants described in Section 6.3. Lastly, the investigation into how small changes in a geometry affect the MPT spectral signature and the invariants derived from this signature.

This chapter is organised in the following distinct sections: In the first part (Sections 7.2-7.3) exemplar non-threat objects are presented, in the second part (Sections 7.4-7.5) threat objects are discussed and in the third part (Section 7.6) details about the open source dataset the `MPT-Library` are documented.

This chapter is organised as follows: In Section 7.2, results for house keys are presented including an investigation into how small changes in key geometry affect the MPT spectral signature and its invariants. In Section 7.3, results of MPT spectral signatures for British coins are presented. In Section 7.4, results of MPT spectral signatures for the TT-33 receiver are presented with a short investigation into how small changes in model geometry affect the MPT spectral signature. In Section 7.5, results of MPT spectral signatures for several kitchen knives are presented. In Section 7.6, a series of meshes for objects in the `MPT-Library` are presented. The chapter is concluded in Section 7.7 with some closing remarks.

7.2 Non-Threat Items: Keys for Pin-Tumbler Locks

Common materials of keys for pin-tumbler locks include brass, plated brass, nickel silver, and steel. Amongst these, brass is often chosen due to its low cost, ease of cutting and its self lubricating characteristics, which avoids the key getting stuck in a lock. Therefore, this study, is restricted to keys made of brass, with the material parameters to be $\mu_r = \mu_*/\mu_0 = 1$ and $\sigma_* = 1.5 \times 10^7$ S/m being selected. An illustration of the cross-section of a key for a pin-tumbler lock is included in Figure 7.1 where the dimensions are similar to a house key and the physical key B_α is indicated as well as the non-dimensional object B used in the computations.

(a) B_α (b) B
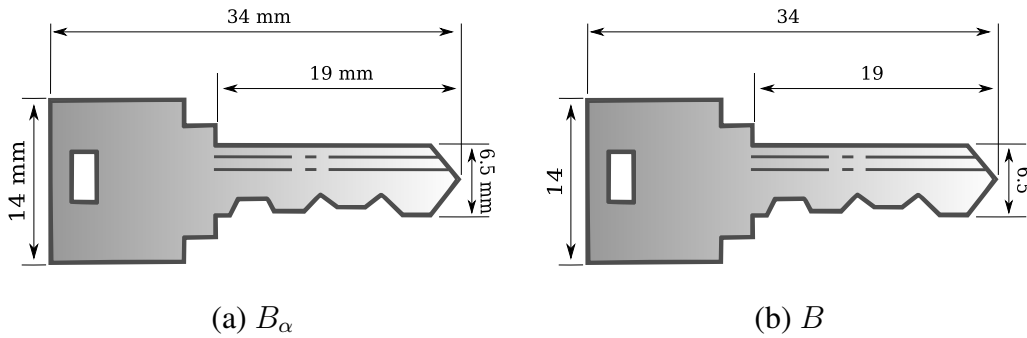
Figure 7.1: Typical key for a pin-tumbler lock: (a) Physical object B_α and (b) non-dimensional object B.

To understand how the effects of small changes in the key geometry affect the MPT frequency spectral signature for frequencies in the range $10^2 \leqslant \omega \leqslant 10^8$ rad/s, a sequence of 9 different key geometries were produced, showing incremental geometric refinement towards keys 8 and 9. In each case, setting $\alpha = 0.001$m and specifying the dimensions of the different cases for B to be non-dimensional. For example, in the case of key 1, B with length 34, has a width of 6.5 (min)-14 (max) and a thickness of 2.5 whereas B for key 4 has a maximum thickness of 2.5 and a deep blade cut of 0.75 and notches of maximum size 1.75. The meshes of the two sets of keys are shown in Figures 7.2 and 7.3. These meshes have local refinement towards the edges of the keys and each case the mesh extends out to a truncation boundary in the form of the $[-1000, 1000]^3$ rectangular box and comprise of between 51 726 and 108 523 unstructured tetrahedra. Importantly, note that the connectedness of the different key types. Of the different keys, keys 2, 4, 6, 7, 8 and 9 are multiply connected and have Betti numbers $\beta_0(B) = \beta_1(B) = 1$ and $\beta_2(B) = 0$ the remaining keys are simply connected with $\beta_0(B) = 1$ and $\beta_1(B) = \beta_2(B) = 0$.

7.2.1 Set 1 of Brass House Keys

Restricting consideration to the set 1 of house keys, it can be seen that under p-refinement of the mesh of 56 241 unstructured tetrahedra for key 1 using $p = 0, 1, 2, 3$ order elements leads to a rapid convergence of the MPT spectral signature presented in the form of the

(a) Key 1 (b) Key 2 (c) Key 3 (d) Key 4
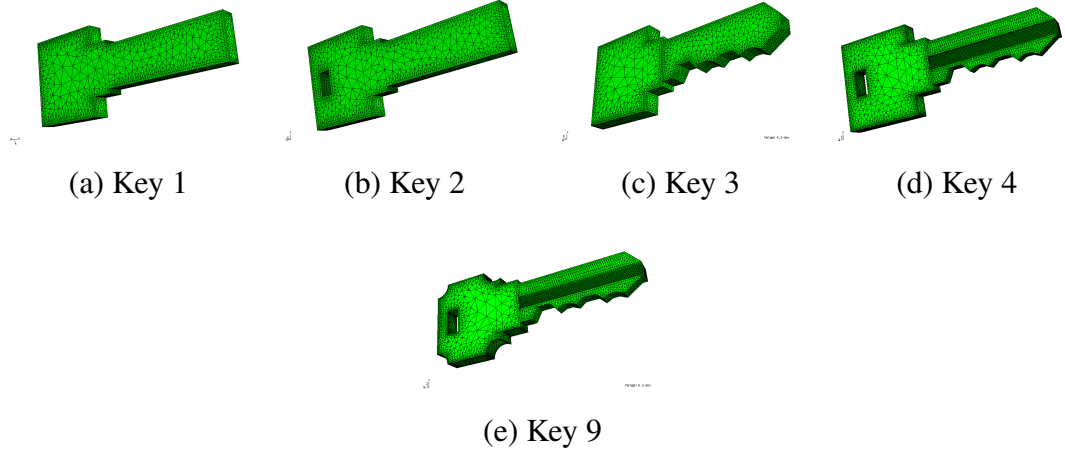
(e) Key 9

Figure 7.2: Set 1 of brass house keys : Surface distribution of elements of the keys in the meshes cases 1-4 and 9.

(a) Key 5 (b) Key 6 (c) Key 7 (d) Key 8
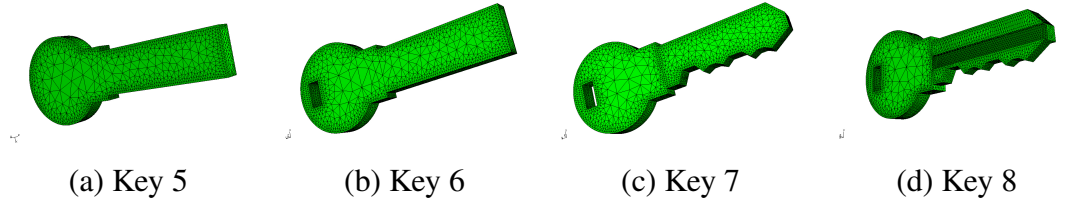
Figure 7.3: Set 2 of brass house keys : Surface distribution of elements of the keys in the meshes cases 5-8 (see Figure 7.2 for key 9 also in this set).

eigenvalues of $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ [1] and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$, namely $\lambda_i(\tilde{\mathcal{R}})$ and $\lambda_i(\mathcal{I})$, $i = 1, 2, 3$, as illustrated in Figures 7.4 and 7.5. Note that, due to the reflectional symmetries for key 1, there are only three independent coefficients each in $\tilde{\mathcal{R}}$ and $\mathcal{I}$.

The role played by a key's topology and its equivalent ellipsoid at a fixed frequency is now considered. Previously, in Lemma 2.5.5, the equivalence between $\lambda_i(\mathcal{M}^\infty[\alpha B])$ and $\lambda_i(\mathcal{T}[\alpha B, 0]) = \mathcal{T}[\alpha E(\infty), 0]$, for the situation where $\beta_1(B) = 0$ was established. In Figure 7.6, $\lambda_i(\tilde{\mathcal{R}})$ and $\lambda_i(\mathcal{T}[\alpha B, 0]) = \mathcal{T}[\alpha E(\infty), 0]$ for key 1 and key 2 are compared, the former having $\beta_1(B) = 0$ and the latter having $\beta_1(B) = 1$. As expected, since $\lim_{\omega \to \infty}(\mathcal{I})_{ij} = 0$, good agreement can be seen in the limiting case as $\omega \to \infty$ (up to the limit of the eddy current model) between $\lambda_i(\tilde{\mathcal{R}})$ and $\lambda_i(\mathcal{T}[\alpha B, 0])$ for key 1, but not for key 2 where $\lambda_i(\tilde{\mathcal{R}})$ is different to $\lambda_i(\mathcal{T}[\alpha B, 0]) = \mathcal{T}[\alpha E(\infty), 0]$ since the equivalent ellipsoid $E(\infty)$ can not describe the MPT spectral signature as $\omega \to \infty$ if $\beta_1(B) \neq 0$. Thus, highlighting the important role that an object's topology plays.

---

[1]Note that the coefficients of $\mathcal{N}^0$ vanish in this case as $\mu_r = 1$, but the notation of $\tilde{\mathcal{R}} = \mathcal{N}^0 + \mathcal{R}$ is retained for ease of comparison with later results.

(a) $\lambda_1(\tilde{\mathcal{R}})$

(b) $\lambda_2(\tilde{\mathcal{R}})$

(c) $\lambda_3(\tilde{\mathcal{R}})$
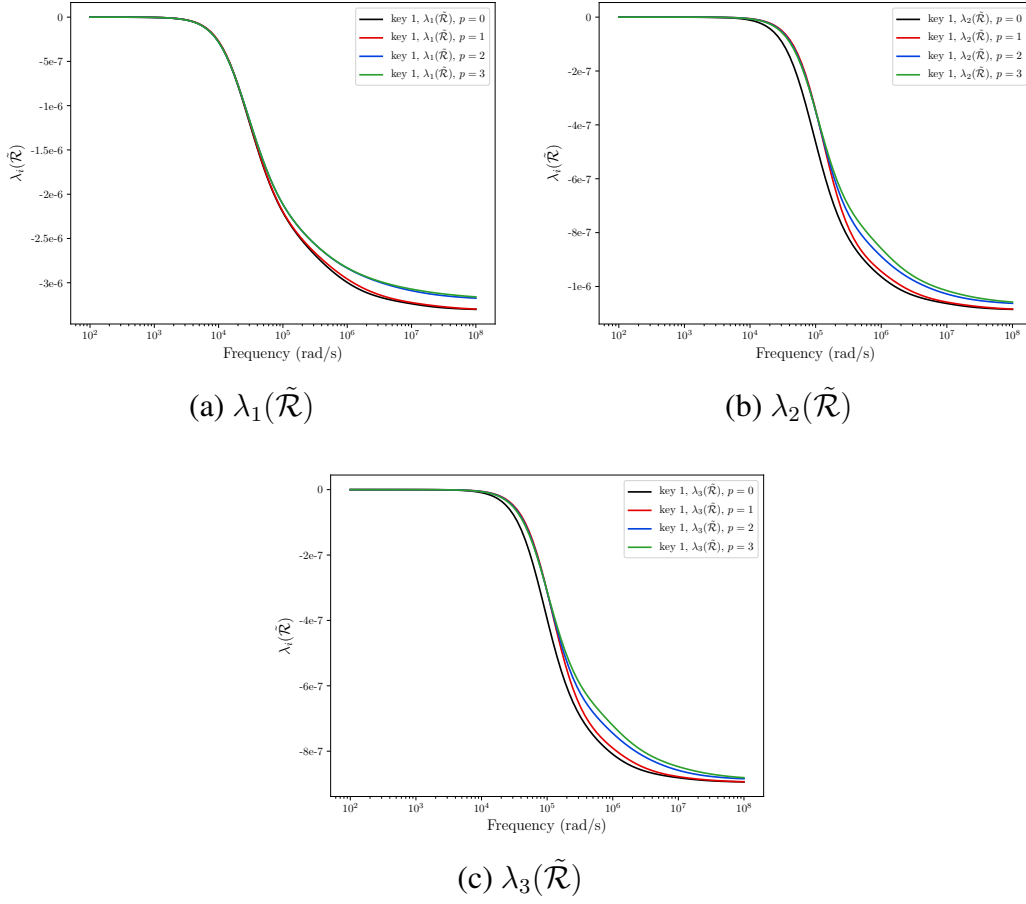
Figure 7.4: Key 1 from set 1 of brass house keys: p-refinement study using $p = 0, 1, 2, 3$ order elements for (a) $\lambda_1(\tilde{\mathcal{R}})$, (b) $\lambda_2(\tilde{\mathcal{R}})$, (c) $\lambda_3(\tilde{\mathcal{R}})$.

In a similar manner to the results shown in Figures 7.4 and 7.5, by performing p-refinement on the meshes for the other keys, and considering snapshot frequencies, the MPT coefficients were found to converge using $p = 3$ elements. However, to accelerate the computation of the MPT spectral signature for the keys, the approach described in Chapter 3, was followed. This involves computing solutions at N representative full order model solutions at logarithmically spaced frequencies and then extracting a basis using a tolerance of $TOL = 10^{-8}$ and solving reduced sized problems to approximate $\boldsymbol{\theta}_i^{(1)}(\omega)$ at other frequencies and, henceforth, predict the MPT coefficients at other frequencies. This process is illustrated in Figure 7.7 for key 1 using $N = 31$. The a-posteriori error estimates $(\Delta[\omega])_{ij}$ that are obtained at low-computational cost during the online stage of the ROM, are used to certify the ROM solutions that have been obtained, are also shown in this figure. These illustrate that, in this case, the ROM predictions are reliable with respect to the full order model prediction of the MPT. Note that the PODP solutions are also very acceptable using $N = 13$ representative full order model solution snapshots, but instead $N = 31$ have been used to ensure $(\Delta[\omega])_{ij}$ is small at all but the highest frequencies, a

(a) $\lambda_1(\mathcal{I})$

(b) $\lambda_2(\mathcal{I})$

(c) $\lambda_3(\mathcal{I})$
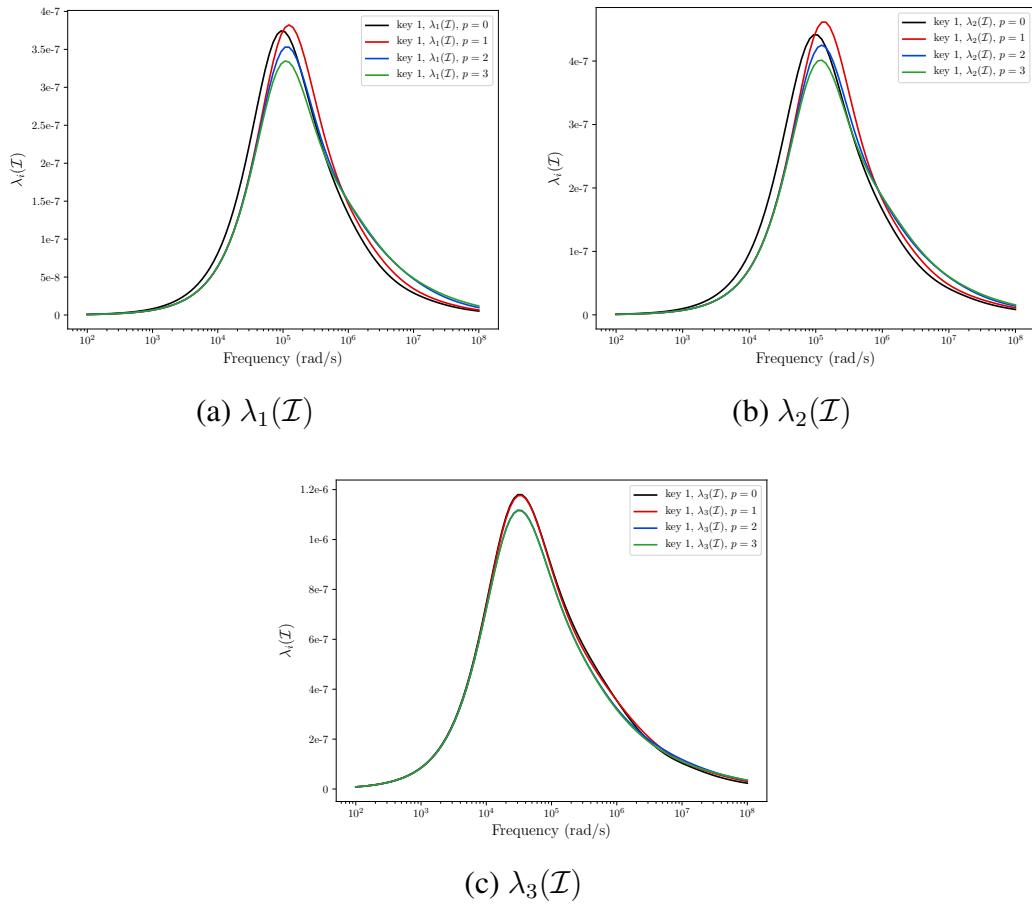
Figure 7.5: Key 1 from set 1 of brass house keys: p-refinement study using $p = 0, 1, 2, 3$ order elements for (a) $\lambda_1(\mathcal{I})$, (b) $\lambda_2(\mathcal{I})$ and (c) $\lambda_3(\mathcal{I})$.
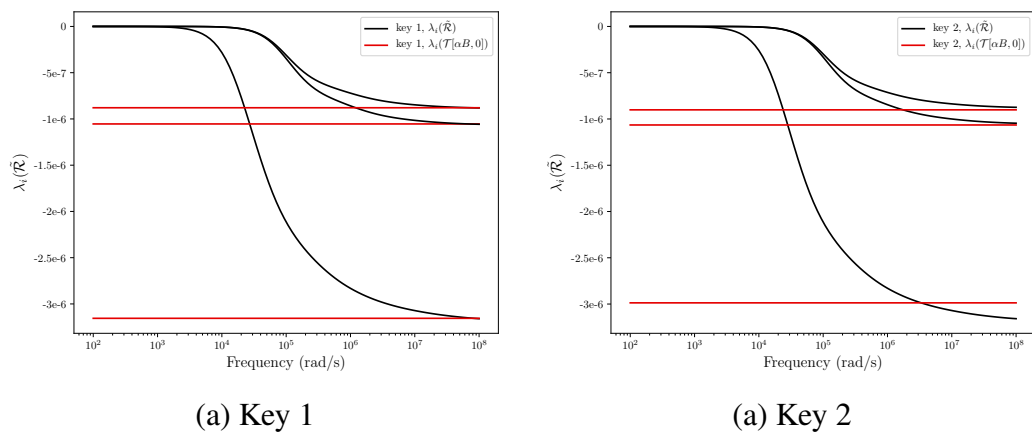
(a) Key 1

(a) Key 2

Figure 7.6: Set 1 of brass house keys: Comparison of $\lambda_i(\tilde{\mathcal{R}})$ and $\lambda_i(\mathcal{T}[\alpha B, 0])$ for (a) Key 1 and (b) Key 2

more detailed study on this is presented in Section 5.2. Still further, the frequency ω_{limit}, obtained using the method described in Section 2.4.3, at which the eddy current approximation for this object is predicted to break down is shown. Smaller $(\Delta[\omega])_{ij}$ could be obtained by increasing N, however, this was not deemed to be necessary as the bounds, which provide confidence that the PODP predictions are accurate, are already tight for $\omega \leqslant \omega_{limit}$.

(a) $(\tilde{\mathcal{R}})_{ij}$ (a) $(\mathcal{I})_{ij}$
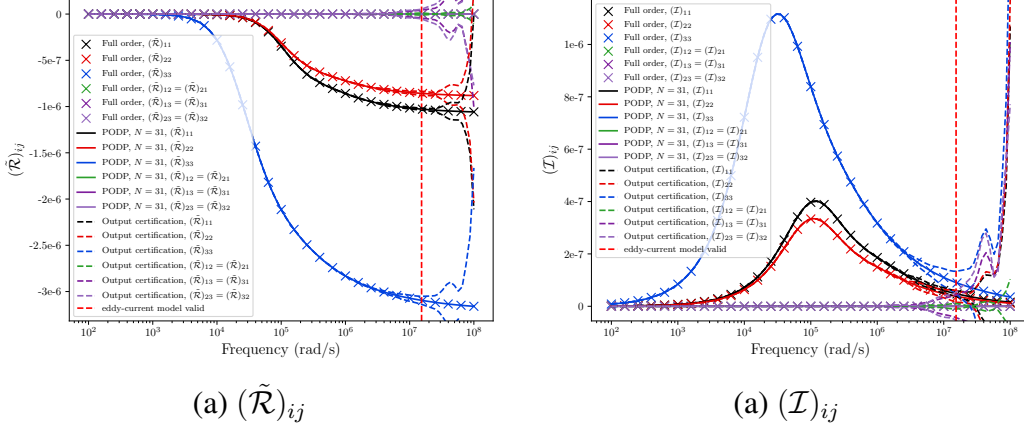
Figure 7.7: Key 1 from set 1 of brass house keys: PODP prediction of the spectral signature showing also the frequencies used for the representative full order solution snapshots and limiting frequency for (a) $(\tilde{\mathcal{R}})_{ij}$ and (b) $(\mathcal{I})_{ij}$.

To compare the results for different keys in set 1, the MPT spectral signature using the principal invariants I_i, $i = 1, 2, 3$, for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are presented, these have been obtained using the PODP approach in Figure 7.8. In this figure, a family of curves that each show a similar behaviour for all the keys in the set can be observed.

The invariant $I_1(\tilde{\mathcal{R}})$ is monotonically decreasing with $\log \omega$, implying the hydrostatic part of $\tilde{\mathcal{R}}$ is associated with a maximum response at high frequencies, while the invariant $I_2(\tilde{\mathcal{R}})$, which is monotonically increasing with $\log \omega$, implies the deviatoric part of $\tilde{\mathcal{R}}$ begin to plays a significant role for $\omega > 10^5$rad/s. The invariant $I_3(\tilde{\mathcal{R}})$, which is monotonically decreasing with $\log \omega$, implies the interaction between the hydrostatic and deviatoric part of $\tilde{\mathcal{R}}$ begin to plays a significant role for $\omega > 10^5$rad/s. The invariants $I_i(\mathcal{I})$, $i = 1, 2, 3$, each have a single local maximum and are greater or equal to 0 for all ω. The invariant $I_1(\mathcal{I})$ implies that hydrostatic part of $\mathcal{I}$ is associated with a maximum response at $\omega \approx 10^5$rad/s and has a broad response over the frequency range $10^2 \leqslant \omega \leqslant 10^8$rad/s while the invariant $I_2(\mathcal{I})$ implies the deviatoric part of $\mathcal{I}$ is associated with a maximum response at $\omega \approx 10^5$rad/s , but its effects are more limited to the narrower frequency band 10^4rad/s $\leqslant \omega \leqslant 10^7$rad/s. Finally, the invariant $I_3(\mathcal{I})$ has a maximum at $\omega \approx 10^5$rad/s, although interaction between hydrostatic and deviatoric parts are more limited to the 10^4rad/s $\leqslant \omega \leqslant 10^6$rad/s. Comparing the keys, the effects are diminished from keys 1,2,3,4 and 9, in turn and, for example, the peak value of $I_3(\mathcal{I})$ reduces in sequence of the volumes of the keys which reduce from 7.81×10^{-7} m^3 for

key 1 to 6.31×10^{-7} m^3 for key 9, as expected. Furthermore, the results for I_i, $i = 2, 3$ applied to $\tilde{\mathcal{R}}$ and $\mathcal{I}$ are similar when comparing the keys 1 and 2.
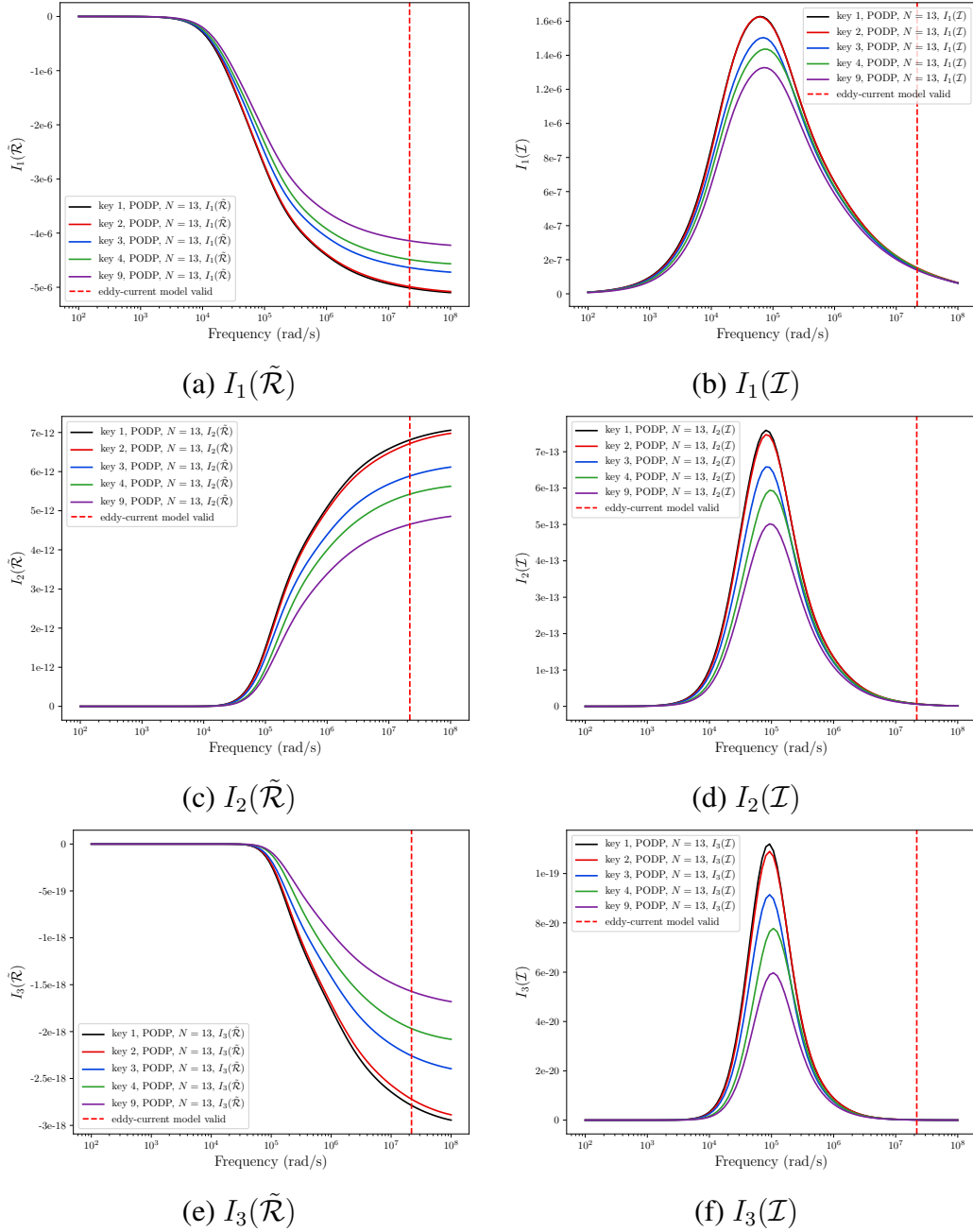
Figure 7.8: Set 1 of brass house keys: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$.

Next, in Figure 7.9, the MPT spectral signature using the alternative invariants J_i, $i = 2, 3$, for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are presented, these have been obtained using the PODP approach. Note that the invariant I_1 is not reproduced, as this forms part of both sets, and has already been shown in Figure 7.8. It can be seen that, for the different keys making up set 1, a family of similar curves is obtained and there are similarities to

the behaviour of the invariants I_i, $i = 2, 3$, with frequency for these tensors. However, the following differences are noteworthy, firstly, the monotonic increase and decrease of $J_2(\tilde{\mathcal{R}})$ and $J_3(\tilde{\mathcal{R}})$ with $\log \omega$, respectively, is much more rapid between $10^4 \leqslant \omega \leqslant 10^5$ than that of $I_2(\tilde{\mathcal{R}})$ and $I_3(\tilde{\mathcal{R}})$. Secondly, the curves for $J_i(\tilde{\mathcal{R}})$ and $J_i(\mathcal{I})$, $i = 2, 3$, are very similar when comparing the keys 3 and 4, whereas the corresponding curves for $I_i(\tilde{\mathcal{R}})$ and $I_i(\mathcal{I})$, $i = 2, 3$, for these keys are different. On the other hand, the similarities previously observed between the invariants I_i, $i = 2, 3$, for $\tilde{\mathcal{R}}$ and $\mathcal{I}$ keys 1 and 2 are also reflected by the invariants J_i, $i = 2, 3$, for these tensors and keys.
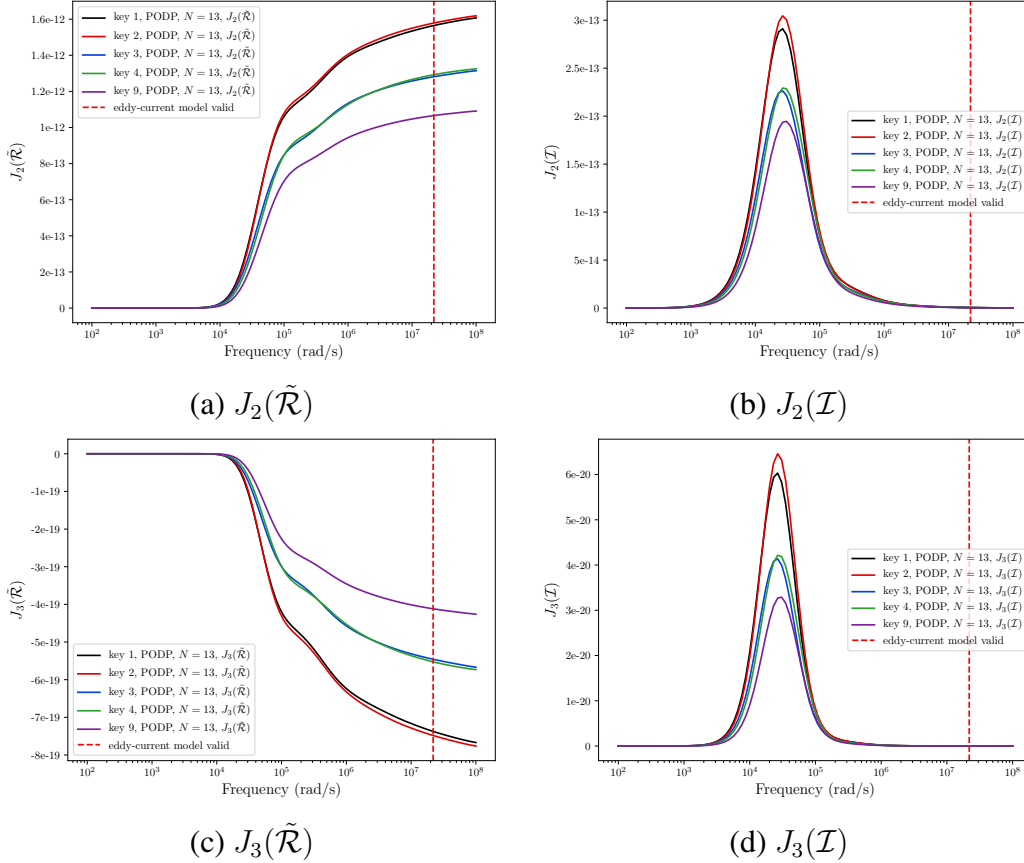
Figure 7.9: Set 1 of brass house keys: Comparison of tensor invariants. (a) $J_2(\tilde{\mathcal{R}})$, (b) $J_2(\mathcal{I})$, (c) $J_3(\tilde{\mathcal{R}})$ and (d) $J_3(\mathcal{I})$.

Finally, comparisons of the invariant $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])}$ are considered as a function of ω for keys in set 1, this can provide additional information about the object's characterisation if the independent coefficients of the MPT are not only associated with its diagonal coefficients. The convergence of $\sqrt{I_2(\mathcal{Z})}$ to 0 under p-refinement for key 1 is first shown in Figure 7.10. The mirror symmetries for this object imply that $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ each have only 3 independent coefficients (at each frequency) that lie on the diagonal of the tensors and so $\sqrt{I_2(\mathcal{Z})}$ is expected to vanish for exact computations. Alongside this, in the same figure, the convergence under p-refinement of key 4 is shown, this has 6 independent coefficients each in $\tilde{\mathcal{R}}$ and $\mathcal{I}$ (at each frequency) and exhibits rapid convergence of the invariant $\sqrt{I_2(\mathcal{Z})}$ to the shown curve as a function of ω. The be-

haviour of $\sqrt{I_2(\mathcal{Z})}$ for keys 4 and 9 is shown in Figure 7.11. For keys $1-3$, $\sqrt{I_2(\mathcal{Z})} = 0$ shall henceforth be set.

Key 1   Key 4
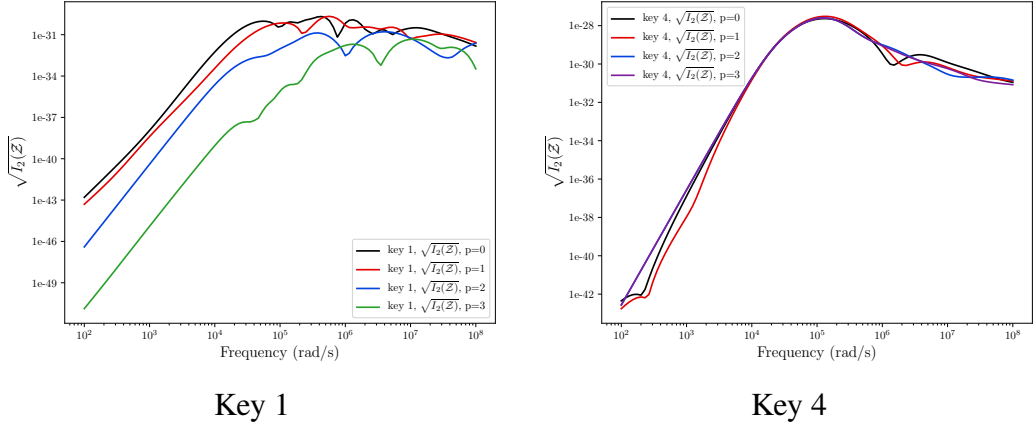
Figure 7.10: Set 1 of brass keys: p-refinement study for $\sqrt{I_2(\mathcal{Z})}$ for (a) Key 1 and (b) Key 4.
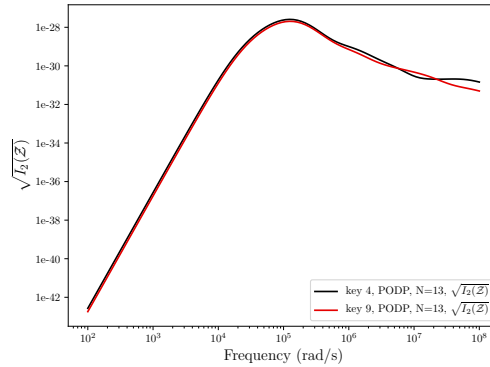
Figure 7.11: Set 1 of brass keys: Comparison of $\sqrt{I_2(\mathcal{Z})}$ for keys 4 and 9.

7.2.2 Set 2 of Brass House Keys

Now considering set 2 of the brass house keys, the results previously shown in Figure 7.8 are reproduced for the invariants I_i, $i = 1, 2, 3$, for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ [2] and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ for the second set of keys and show these in Figure 7.12. In a similar manner to Figure 7.8, the results included in Figure 7.12 form a family of similar curves and that their behaviour follows a similar pattern to that previous described for the keys in set 1. The results for I_i, $i = 1, 2, 3$ for the tensor characterisations of keys 5 and 6 are similar, which is not surprising given the similarities in these geometries. In addition, there are only small differences in I_i, $i = 1, 2, 3$ for the tensor characterisations of keys 8

---

[2]Note that the coefficients of $\mathcal{N}^0$ vanish as $\mu_r = 1$, but the notation of $\tilde{R} = \mathcal{N}^0 + \mathcal{R}$ is retained for ease of comparison with later results.

(a) $I_1(\tilde{\mathcal{R}})$

(b) $I_1(\mathcal{I})$
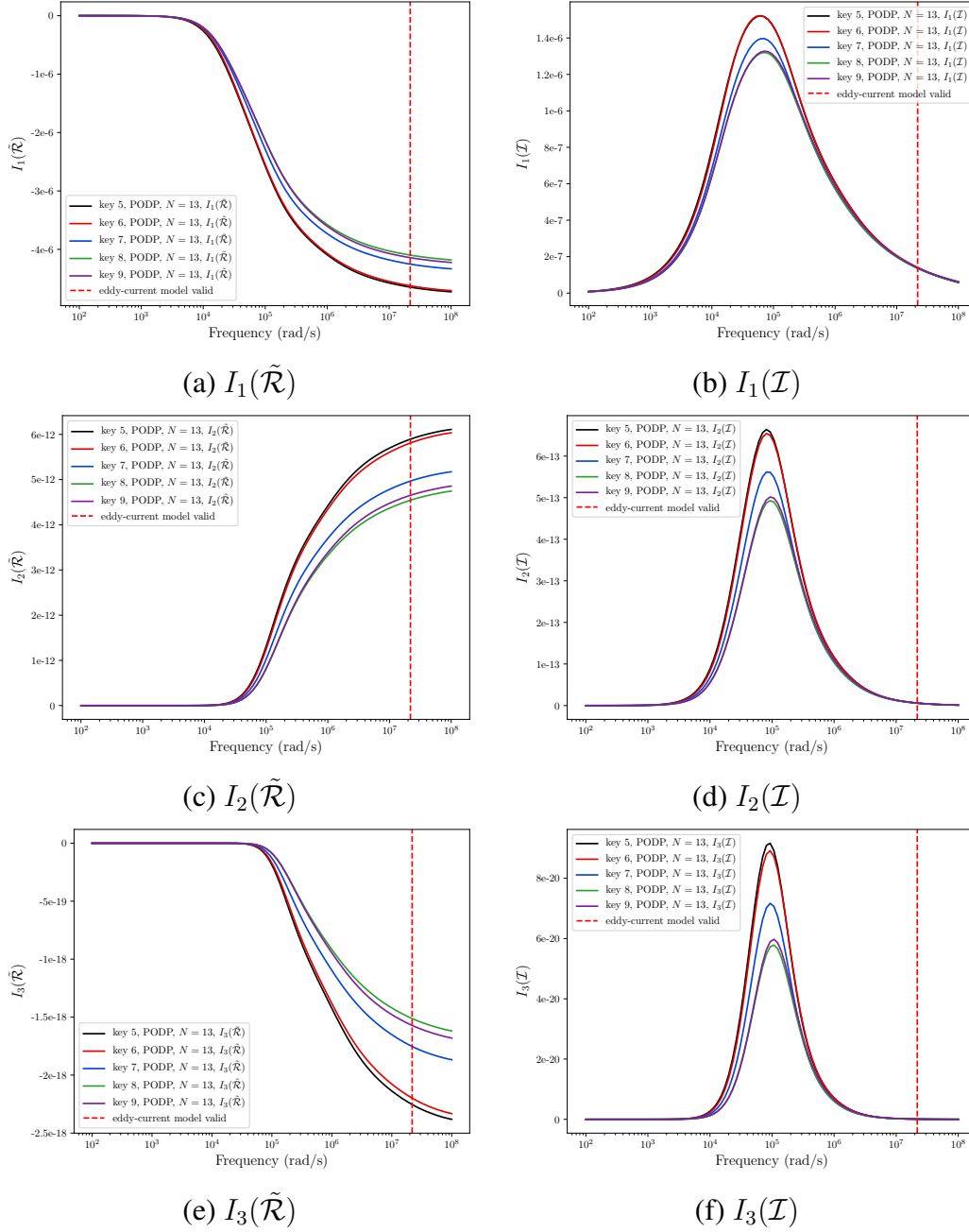
(c) $I_2(\tilde{\mathcal{R}})$

(d) $I_2(\mathcal{I})$

(e) $I_3(\tilde{\mathcal{R}})$

(f) $I_3(\mathcal{I})$

Figure 7.12: Set 2 of brass house keys: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$

and 9. Note that key 8 has a circular head and key 9 a polygonal head, but the volume of material is similar and the symmetries of the objects and the number of independent coefficients in $\tilde{\mathcal{R}}$ and $\mathcal{I}$ (for each frequency) are otherwise the same for these two keys. Keys $5-8$ are associated with a gradual reduction in the volume of the material for the key and thus the magnitude of the associated $I_3(\mathcal{I})$ and $I_3(\tilde{\mathcal{R}})$, curves for these cases reduces as expected.

Similarly, in Figure 7.13, the results for the invariants J_i, $i = 2, 3$ form a family of curves with the behaviour of the invariants similar to that described for the keys in set 1. Note

(a) $J_2(\tilde{\mathcal{R}})$ (b) $J_3(\mathcal{I})$

(c) $J_3(\tilde{\mathcal{R}})$ (d) $J_3(\mathcal{I})$
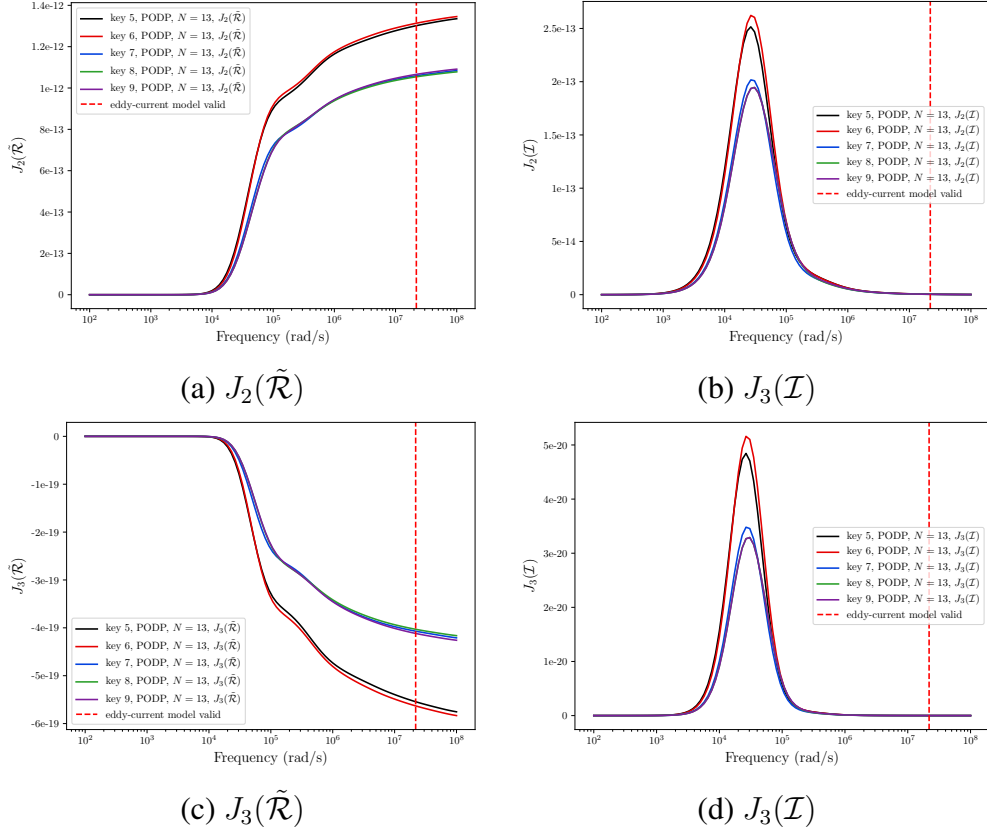
Figure 7.13: Set 2 of brass house keys: Comparison of tensor invariants. (a) $J_2(\tilde{\mathcal{R}})$, (b) $J_2(\mathcal{I})$, (c) $J_3(\tilde{\mathcal{R}})$ and (d) $J_3(\mathcal{I})$

that the results for the keys in set 2 for J_i, $i = 2, 3$ can be grouped into keys 5, 6 and keys 7, 8 and 9 where the results for keys 8 and 9 for $J_2(\mathcal{I})$ (and $J_3(\mathcal{I})$) are indistinguishable on this scale. The former group does not contain the notches or the blade cut while the latter set all have the same notches, keys 8 and 9 have the deep blade cut and key 9 differs from the others by having a polygonal head rather than a circular head (although has a similar volume to keys 7 and 8).
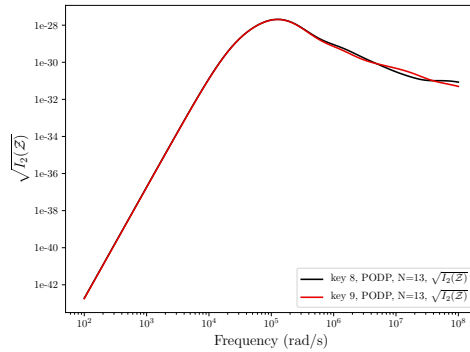
Figure 7.14: Set 2 of brass house keys: Comparison of the tensor invariant $\sqrt{I_2(\mathcal{Z})}$.

Of the keys in set 2, only keys 8 and 9 have independent coefficients in $\mathcal{R}[\alpha B, \omega, \sigma_*, \mu_r]$

and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ that are not only associated with the diagonal entries of the tensor. The behaviour of $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])}$ for these keys is shown in Figure 7.14. For the other keys in set 2 $\sqrt{I_2(\mathcal{Z})} = 0$ is set.

7.3 Non-threat items: British Coins

In this section, inspired by the previous article on MPT characterisations of US coins [32], MPT spectral signature characterisations for British coins in the denominations 1 penny (p), 2p, 5p, 10p, 20p, 50p, £1, £2 are presented. Examples of the 1982 (20p), 1992 (1p, 2p), 1997 (50p), 1998 (£2), 2012 (5p,10p), 2017 (£1) issues of these denominations are listed in Table 7.1, which also summarises the shape, diameter, thickness, composition based on the information available from the Royal Mint [91]. The table also sets out the electrical properties, where the conductivity values for the different material compositions have been obtained from [58] at room temperature. For the quoted compositions, a value of $\mu_r = 1$ has been assumed, however, in practice, some Copper-Nickel mixtures with a high iron content can have a μ_r slightly above 1 (e.g [47]). The later issues of the 1p, 2p, 5p and 10p coins have a significantly different composition to that presented in Table 7.1 and, instead of a high copper content, they are instead copper plated steel. Note that each of the coins considered are simply connected.

With the exception of the £1 and £2 denominations, the coins are modelled as homogeneous conductors while the former are each modelled as an annulus with two different materials. The majority of the coins have a circular face and only the 20p and 50p differ, being Reuleaux heptagonal discs. The coins with a circular face are modelled so that their circular region lies in the plane spanned by e_1 and e_2 and, hence, they have rotational symmetry about the e_3 axis (for any angle). Consequently, the independent coefficients of $\tilde{\mathcal{R}}[\alpha B, \omega, \mu_r, \sigma_*]$ [3] and $\mathcal{I}[\alpha B, \omega, \mu_r, \sigma_*]$ for such coins are $(\tilde{\mathcal{R}})_{11} = (\tilde{\mathcal{R}})_{22}$, $(\tilde{\mathcal{R}})_{33}$ and $(\mathcal{I})_{11} = (\mathcal{I})_{22}$ and $(\mathcal{I})_{33}$ (for each frequency). The Reuleaux heptagonal discs are modelled in a similar way, with a 51.428 (4dp) degree rotational symmetry about the e_3 and, consequently, it also follows that their independent coefficients of the MPT for such coins are associated with the same entries.

To model the 1p coin, B is considered to be a circular disc of diameter 20.3 and thickness 1.52 and set $\alpha = 0.001$ m. An unstructured mesh of $33\,351$ unstructured tetrahedra was generated to model the object and the region surrounding it out to a truncation boundary in the form of the rectangular box $[-1000, 1000]^3$. In a similar way, unstructured meshes of between $24\,963$ and $36\,957$ tetrahedra were generated to model the other coins. On these meshes, $p = 4$ elements were found to be satisfactory for accurately computing the representative full order model solution snapshots. In order to produce the MPT spectral signature for the coins, $N = 13$ representative full order solution snapshots were obtained at logarithmically spaced frequencies over the range $10^1 \leqslant \omega \leqslant 10^{10}$rad/s were used in combination with the PODP approach and a tolerance of $TOL = 10^{-6}$.

---

[3]Note that the coefficients of $\mathcal{N}^0$ vanish as $\mu_r = 1$, but the notation of $\tilde{R} = \mathcal{N}^0 + \mathcal{R}$ is kept for ease of comparison with later results

Coin	Shape	Diameter in mm	Thickness in mm	Composition	Relative Perme-ability (μ_r)	Conductivity (σ_*) in S/m
1p (1971-Date)	Circular Disc	20.3	1.52	97% Copper, 2.5% Zinc and 0.5% Tin	1	4.03×10^7
2p (1971-Date)	Circular Disc	25.9	2.03	97% Copper, 2.5% Zinc and 0.5% Tin	1	4.03×10^7
5p (1990-Date)	Circular Disc	18	1.7	75% Copper and 25% Nickel	1	2.91×10^6
10p (1990-Date)	Circular Disc	24.5	1.85	75% Copper and 25% Nickel	1	2.91×10^6
20p (1982-Date)	Reuleaux Hep-tagonal Disc	21.4	1.7	84% Copper and 16% Nickel	1	5.26×10^6
50p (1997-Date)	Reuleaux Hep-tagonal Disc	27.3	1.78	75% Copper and 25% Nickel	1	2.91×10^6
£1 (2017-Date)	Annulus	15.2 /23.45 (in/out)	2.8 / 2.8 (in/out)	Nickel Plated Brass / 70% Copper, 24.5% Zinc and 5.5% Nickel	1 /1 (in/out)	1.63×10^7 / 5.26×10^6 (in/out)
£2 (1998-Date)	Annulus	21 /28.4 (in/out)	2.5 /2.5 (in/out)	75% Copper and 25% Nickel/ 97% Copper, 2.5% Zinc and 0.5% Tin	1 /1 (in/out)	2.91×10^6 / 1.93×10^7 (in/out)

Table 7.1: Set of British Coins 1p, 2p, 5p,10p, 20p, 50p, £1 and £2 : Coin shape, dimensions and electrical properties.

Although the PODP solutions are very acceptable using $N = 13$ representative full order model solution snapshots, in order to achieve smaller a-posteriori error estimates, results obtained with $N = 21$ and $TOL = 10^{-8}$ are considered and shown in Figure 7.15. Also included in this figure is the limiting frequency ω_{limit} predicted by following the approach in Section 2.4.3. The rotational symmetry of the object implies that the object has just two independent coefficients each in $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$, which lie on the diagonal of the tensors. Of these $(\tilde{\mathcal{R}})_{33}$ and $(\mathcal{I})_{33}$ have the largest magnitude in a direction that is perpendicular to the plane of the disc, which is as expected for a non-magnetic disc [32]. Note that the 1p coin issued after 1992, which has a high μ_r value, would have dominant components $(\tilde{\mathcal{R}})_{11} = (\tilde{\mathcal{R}})_{22}$ and $(\mathcal{I})_{11} = (\mathcal{I})_{22}$ in the plane of the disc, as expected for a magnetic disc [32].

$(\tilde{\mathcal{R}})_{ii}$

$(\mathcal{I})_{ii}$
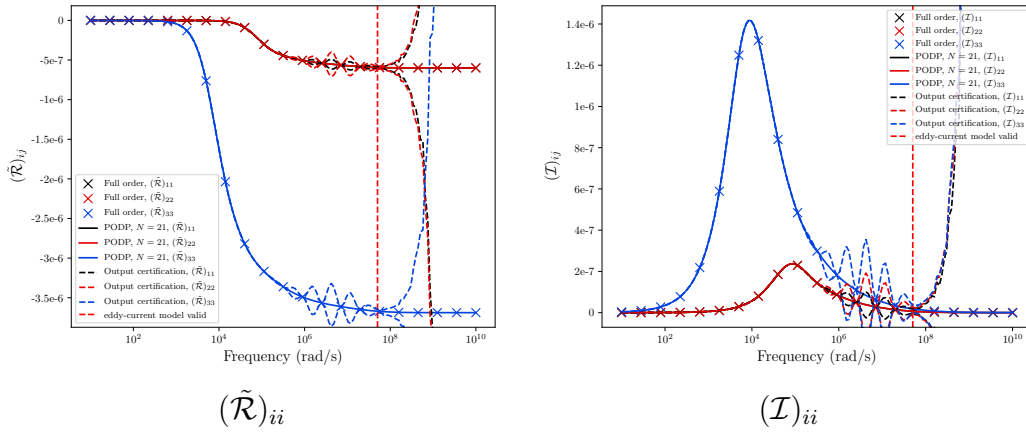
Figure 7.15: 1p Coin from set of British coins: PODP prediction of the spectral signature showing also the frequencies used for the representative full order solution snapshots and limiting frequency for (a) $(\tilde{\mathcal{R}})_{ii}$ and (b) $(\mathcal{I})_{ii}$.

To compare the results for the different coins, the MPT spectral signatures using the principal invariants I_i, $i = 1, 2, 3$, for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are presented these have been obtained using the PODP approach in Figure 7.16. In this figure, consideration has been restricted to frequencies such that $10^2 \leqslant \omega \leqslant 10^8$ rad/s in order to allow comparisons with the earlier key results. In practice, the eddy current model breaks down at a frequency of $\omega_{limit} < 10^8$ rad/s (or greater) for all the coins considered and so higher frequencies are physically invalid in any case. Unlike the corresponding results for the house keys shown in Figures 7.8 and 7.12, the results obtained for the coins shown in Figure 7.16 do not form a family of similar curves since both the volumes and materials of the coins vary significantly motivating the ability to discriminate between different coins, however, some of the trends previously observed cary over to this case also. The curves for $I_1(\tilde{\mathcal{R}})$ and $I_3(\tilde{\mathcal{R}})$ are monotonically decreasing with $\log \omega$, while $I_2(\tilde{\mathcal{R}})$ is monotonically increasing with $\log \omega$. The curves for $I_i(\mathcal{I})$, $i = 1, 2, 3$, each have a single local maximum, although the peaks appear at different frequencies for different coins and the different invariants, however, the width of the frequency band reduces for all cases, when

considering $I_2(\mathcal{I})$ and $I_3(\mathcal{I})$ compared to $I_1(\mathcal{I})$. On considering the different coins, similarities can be seen between the MPT spectral signatures of the 1p, 2p coins, the 5p, 10p coins and the 20p and 50p coins. This can be explained as follows: the composition of the coins in these groups is the same and their dimensions can be approximately obtained by a simple scaling, hence, the scaling results in Lemma 5.2 of [133] predict that the tensor coefficients of the larger sized coin can be obtained from the smaller object by a translation and scaling, which is also observed in the invariants.
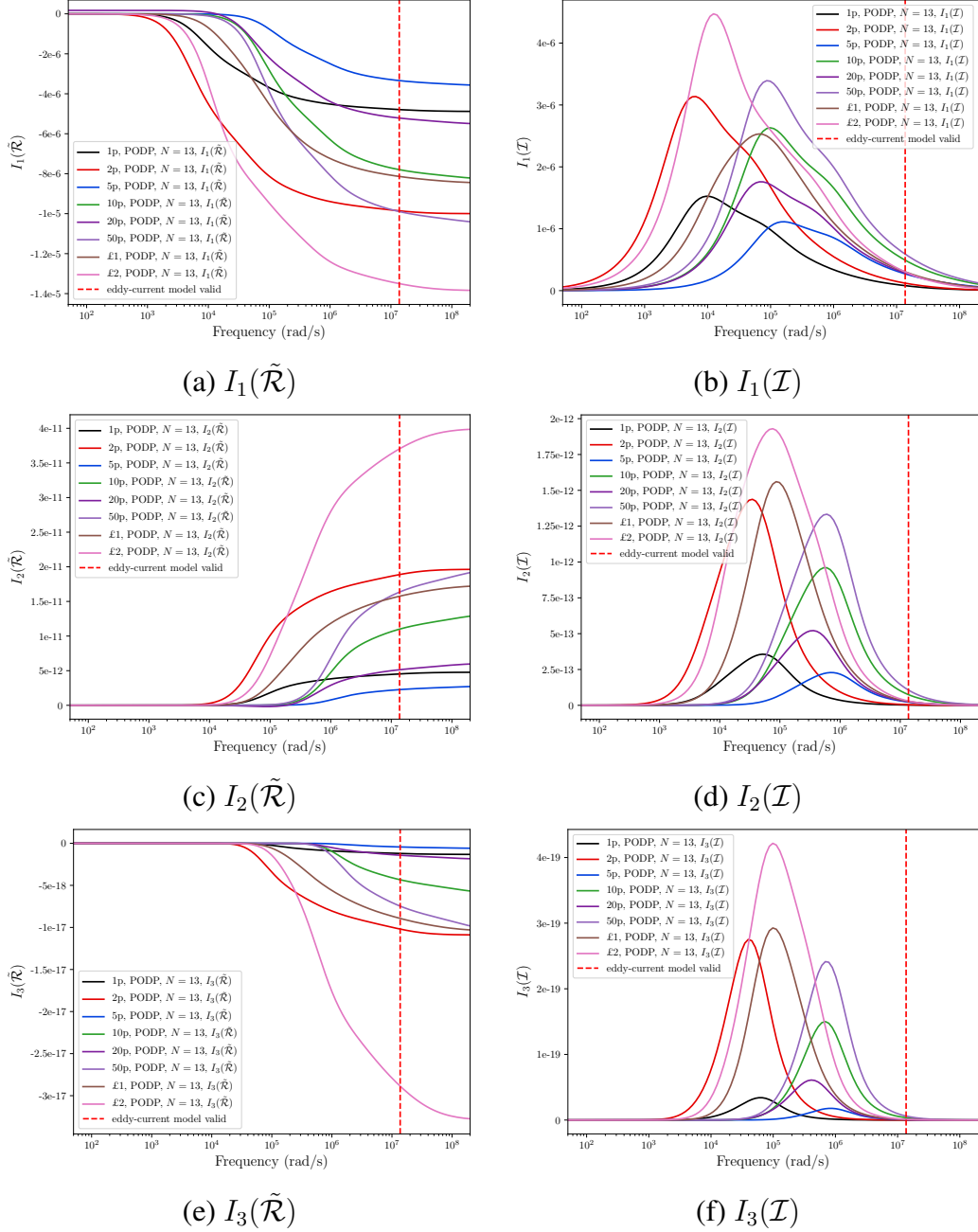

Figure 7.16: Set of British coins: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$.

The £2 coin has the largest volume and also the highest peak value in $I_3(\mathcal{I})$, the magnitude of the peaks reduce in sequence of the volumes of the coins, as expected. The multiple local maxima in the coefficients of $\mathcal{I}$ and the multiple points of inflection in the coefficients of $\tilde{\mathcal{R}}$, which are known to be associated with objects with inhomogeneous conductivity [74], are not easily distinguished on the invariants for the £1 and £2 coins, probably due to the difference in conductivities being approximately 1 order of magnitude or less.
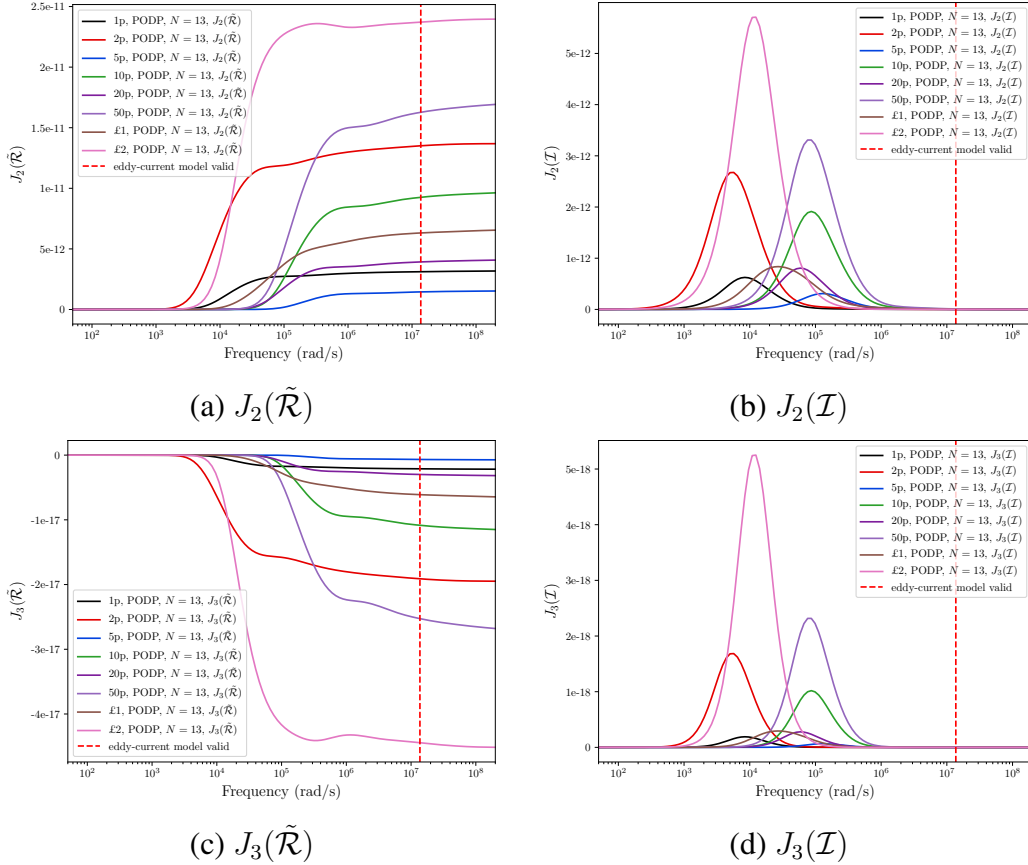
Figure 7.17: Set of British coins: Comparison of tensor invariants. (a) $J_2(\tilde{\mathcal{R}})$, (b) $J_2(\mathcal{I})$, (c) $J_3(\tilde{\mathcal{R}})$ and (d) $J_3(\mathcal{I})$.

The corresponding results obtained for the alternative invariants J_i, $i = 2, 3$, for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ are presented in Figure 7.17. Again, unlike the keys, the plots of these invariants do not form a family of similar curves as both the volumes and conductivities of the different coins are different. While for most coins $J_2(\tilde{\mathcal{R}})$ is monotonically increasing with $\log \omega$ and $J_3(\tilde{\mathcal{R}})$ is monotonically decreasing with $\log \omega$, however, there are exceptions, most notably with the £2 coin, which can be explained by its inhomogeneous materials. One might expect a similar behaviour with the inhomogeneous £1 coin, but it is difficult to observe on this scale. The behaviour of $J_2(\mathcal{I})$ and $J_3(\mathcal{I})$ with $\log \omega$ shows a single local maximum for each coin where the presence of multiple local maxima for the £1 and £2 can't be observed on this scale. Of the coins considered, the curves associated with the 2p, 50p and £2 cases have the largest magnitude, indicating

that they have the largest deviatoric component, which is expected. Due to the coefficients of the MPT having units of volume, they are a function of the geometry, frequency and materials, any invariants are computed from the coefficients, so an increase in volume will lead to a larger magnitude of the coefficients and hence also a larger magnitude of the invariants. It is observed that the deviatoric invariants highlight this dependence to a greater extent as this scales with α^9, $|B|^3$ (i.e. volume cubed).

(a) $|\text{Re}(\boldsymbol{J}^e)|$ for $\omega = 10^3$ rad/s.

(b) $|\text{Re}(\boldsymbol{J}^e)|$ for $\omega = 10^5$ rad/s.

(c) $|\text{Re}(\boldsymbol{J}^e)|$ for $\omega = 10^7$ rad/s.
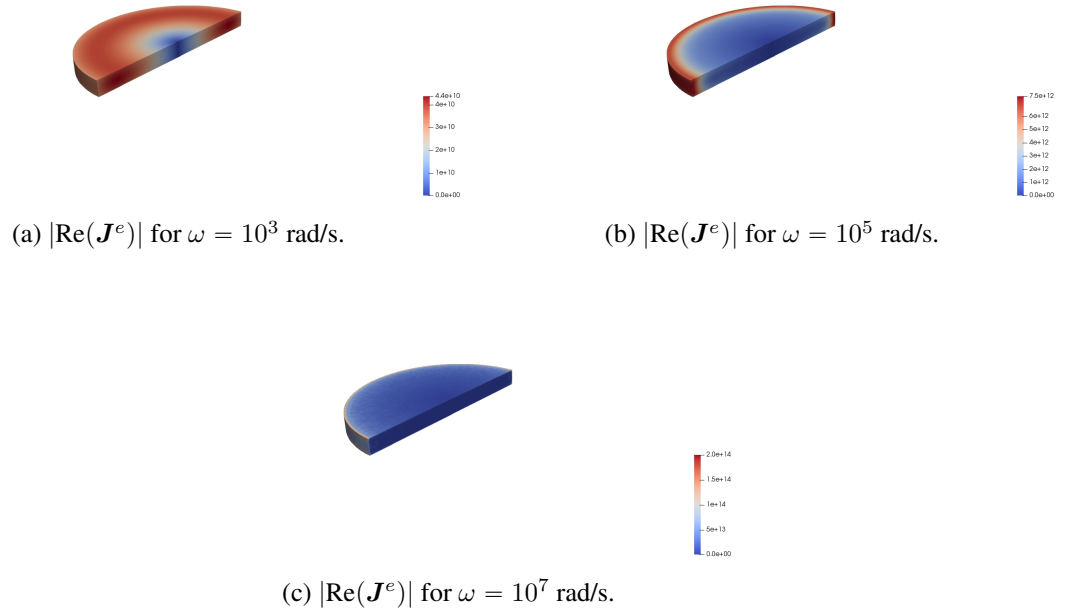
Figure 7.18: 1p coin from the set of British coins: Contours of the eddy-currents $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_3^{(1)}$ for different values of ω, (a) $\omega = 10^3$ rad/s, (b) $\omega = 10^5$ rad/s, (c) $\omega = 10^7$ rad/s in a cut through the coin, on the plane spanned by $\boldsymbol{e}_1$ and $\boldsymbol{e}_3$.
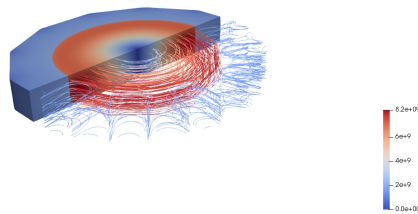
Figure 7.19: £1 coin from set of British coins: Contours of $|\text{Re}(\boldsymbol{J}^e)|$ in a cut through the coin, on the plane spanned by $\boldsymbol{e}_1$ and $\boldsymbol{e}_3$, where $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_3^{(1)}$ are the eddy currents and showing the field lines corresponding to $\text{Im}(\boldsymbol{J}^e)$ with $\omega = 10^3$ rad/s.

In order to compute the MPT spectral signature, the solution of an ROM is obtained

at each frequency of interest. As the frequency increases, the skin depth reduces and the associated eddy currents become confined to a thin layer close to the surface of the conductor. In Figure 7.18, a cut through the 1p coin is shown, on the plane spanned by e_1 and e_3, in order to illustrate the eddy currents $J^e = i\omega\sigma_*\theta_3^{(1)}$ obtained at the frequencies $\omega = 10^3$ rad/s, $\omega = 10^5$ rad/s and $\omega = 10^7$ rad/s.

In Figure 7.19, a contour plot is shown for the eddy-currents $J^e = i\omega\sigma_*\theta_3^{(1)}$ in cut through the £1 coin, on a plane spanned by e_1 and e_3 for $\omega = 10^3$rad/s. This figure also includes the field lines for $\text{Re}(J^e)$. For the coin models, each of the associated MPT frequency spectra have independent coefficients that are only associated with diagonal entries of the tensor. Thus, $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])} = 0$ is set in each case.

7.4 Threat Items: TT-33 Semi-Automatic Pistol

In this section, MPT spectral signature characterisations for components of an exemplar semi automatic pistol are presented. The Tokarev TT-33 has been chosen, this can be seen in Figure 7.20, and was originally designed in the Soviet Union in the late 1920's, with production in the USSR between 1930-1954 [123]. It has also been produced in other countries including China, Hungary, North Korea, Pakistan, Romania, Vietnam and Yugoslavia and exported to other nations around world. It is still used by the Bangladeshi and North Korean armed forces and the police in Pakistan often carry the pistol as a side arm. Under a different name, it is occasionally supplied to the police and armed forces in China [60]. This gun is chosen due to both it's simplicity and prevalence in conflict zones and less economically developed countries with about 1.7 million being produced in total [123].

Figure 7.20: Image of Tokarev TT-33 from [16] reproduced with permission under Creative Commons License CC-BY

The starting point of the characterisation are the blueprints for the larger parts of a non-firing replica [5], which is sufficient for the purpose. The focus of the investigation is on

the receiver (with the magazine and ammunition removed), the reason being that some-one wishing to disguise a semi-automatic pistol might disassemble the main pieces and carry them separately through a security control check. As the receiver is one of the larger components, it should be easiest to identify by a metal detector. The exact ma-terials of the receiver are not known although it is likely to be made of a carbon steel alloy such as 1020 or 4140, which has a conductivity of around $\sigma_* = 4.5 \times 10^6$ S/m to $\sigma_* = 6.25 \times 10^6$ S/m [92] and, for the simulations, a conductivity of $\sigma_* = 6.2 \times 10^6$ S/m has been chosen. Such steels are ferrous and exhibit a non-linear constitutive relationship between the magnetic flux density $\boldsymbol{B}$ and the magnetic field $\boldsymbol{H}$, but, if restricted to low field strengths, where the relationship is linear, $\mu_* = \mu_r \mu_0 = |\boldsymbol{B}|/|\boldsymbol{H}|$ and the mathe-matical model developed in [13, 69] still applies. Values of μ_r obtained experimentally for different steels vary enormously (eg from $\mu_r = 100$ to $\mu_r = 600$ or larger) as of-ten $\mu_* = \mu_r \mu_0 = |\boldsymbol{B}|/|\boldsymbol{H}|$ is applied when the curve is no longer straight. Numerical simulations using high values of μ_r become increasingly challenging and so a relative permeability of $\mu_r = 5$ has been chosen.

An extreme simplification of the receiver for TT-33 is to model it as simple L-shape made up of two rectangular regions glued together (the overall dimensions of the physical L-shape are 148 mm × 17.5 mm × 10.1 mm). A mesh discretising the L-shape, with overall dimensions 148 × 17.5 × 101, and the surrounding region out to a truncation boundary, in the form of a box of dimensions $[-1000, 1000]^3$, was generated with h-refinement towards the edges, containing 62 656 unstructured tetrahedra with $\alpha = 0.001$ m. A se-quence of geometric improvements on the basic L-shape model was then considered, as shown in Figure 7.21, which shall be referred to as TT-33 with a trigger hole, TT-33 with no internals, TT-33 without chamfers and TT-33 with chamfers, having discretisations comprising of between 94 092 and 175 217 tetrahedral elements, respectively. With the exception of the L-shape, all contain a model of the trigger guard (a loop of steel where the trigger would be placed) and assume that the magazine is removed, as the top-view of TT-33 with trigger hole shown in Figure 7.22 illustrates. TT-33 with no internals and TT-33 without chamfers offer further geometric improvements with TT-33 with chamfers being the closet to the actual blueprint and includes small holes in the receiver used to fix the other components of the pistol in place. Only the L-shape is simply connected with $\beta_1(B) = 0$, the TT-33 with trigger hole $\beta_1(B) = 2$, TT-33 with no internals and TT-33 without chamfers each have $\beta_1(B) = 4$ while TT-33 without chamfers has $\beta_1(B) = 13$ each object has $\beta_0(B) = 1$ and $\beta_2(B) = 0$.

In each case, $N = 13$ representative solution snapshots to full order problem at logarith-mically spaced frequencies in the range $8 \times 10^{-1} \leqslant \omega \leqslant 8 \times 10^8$ rad/s were found to converge by using $p = 4$ elements. Then, by applying the PODP approach described in Chapter 3 with a tolerance of $TOL = 10^{-6}$, the MPT spectral signature for each of the receiver models was obtained. With the exception of the model with chamfers each of the models of the receiver has an axis of symmetry in the $\boldsymbol{e}_2$ direction and so there are 4 independent coefficients each in $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ corresponding to $(\tilde{\mathcal{R}})_{11}$, $(\tilde{\mathcal{R}})_{22}$, $(\tilde{\mathcal{R}})_{33}$, $(\tilde{\mathcal{R}})_{13} = (\tilde{\mathcal{R}})_{31}$ at each frequency with similar for $\mathcal{I}$. The model with chamfers, which lacks this symmetry, has 6 independent coefficients each in $\tilde{\mathcal{R}}$ and
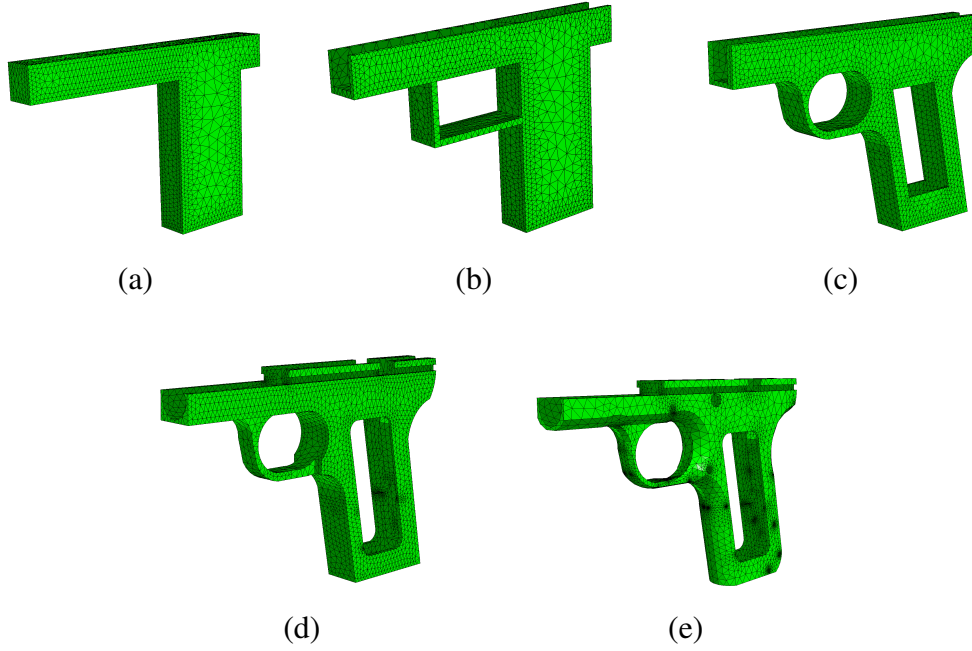
Figure 7.21: Set of receiver models for TT-33 pistol: surface distribution of elements for (a) L-shape domain, (b) TT-33 with a trigger hole, (c) TT-33 with no internals, (d) TT-33 without chamfers and (e) TT-33 with chamfers
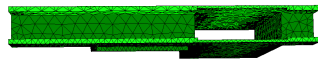
Figure 7.22: Set of receiver models for TT-33 pistol: top view of TT-33 with a trigger hole

$\mathcal{I}$ (at each frequency), which means that all coefficients of the tensor are independent of each other.

A comparison of the MPT spectral signatures using the principal invariants I_i, $i = 1, 2, 3$ for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ is presented, these have been obtained using the PODP approach in Figure 7.23. In this figure, consideration has been restricted to frequencies such that $10^2 \leqslant \omega \leqslant 10^8$ rad/s in order to allow comparisons with the earlier key and coin results. In practice, the eddy current model breaks down at a frequency $\omega_{limit} < 2 \times 10^6$ rad/s for all the TT-33 models considered and so, in practice, higher frequencies are not relevant. While the $I_i(\mathcal{I})$, $i = 1, 2, 3$, invariants for the TT-33 models are similar to that for the keys and coins, the behaviour of $I_i(\tilde{\mathcal{R}})$, $i = 1, 2, 3$ are quite different due to $\mu_r \neq 1$. For $I_1(\tilde{\mathcal{R}})$, the curves are monotonically decreasing with $\log \omega$, but do not asymptote to 0 for small ω. For $I_2(\tilde{\mathcal{R}})$ and $I_3(\tilde{\mathcal{R}})$ the curves are neither monoton-

ically increasing or decreasing with $\log \omega$ and the curves do not asymptote to 0 for small ω motivating that discrimination between the object is possible. The L-shape exhibits significant differences to the TT-33 models with a different location of resonant peak in $I_i(\mathcal{I})$, $i = 1, 2, 3$ and a significantly different behaviour for $I_i(\tilde{\mathcal{R}})$, $i = 1, 2, 3$. The results for the TT–33 with a trigger hole and TT-33 with no internals models are similar with further differences for the TT-33 without chamfers and TT-33 with chamfers. However, all these latter four cases exhibit a resonance peak of around $\omega = 10^4$rad/s for $I_i(\mathcal{I})$, $i = 1, 2, 3$. The magnitude of the resonance peak for $I_3(\mathcal{I})$ decreases in sequence of the associated volume of the different TT-33 models.

Figure 7.23: Set of receiver models for TT-33 pistol: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$.

The corresponding results obtained for the invariants J_i, $i = 2,3$ for $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ that have been obtained using the PODP approach are shown in Figure 7.24. These results again illustrate the significant difference between the L-shape and the other models. With the exception of the *L-shape*, the results for $J_2(\tilde{\mathcal{R}})$ show a monotonic increase with $\log \omega$, those for $J_3(\tilde{\mathcal{R}})$ show a monotonic decrease, $J_2(\mathcal{I})$ and $J_3(\mathcal{I})$ show a single local maximum. The results for TT-33 with and without chamfers are similar with greater differences exhibited between the TT-33 with no internals and TT-33 with trigger hole. The results shown in Figures 7.23 and 7.24 indicate the significant

difference between the spectral signatures of the L-shape and the more realistic models, which adds the credibility that the MPT spectral signature makes it possible to distinguish between a carpenters metallic set-square (which closely resembles an L-shape) and the receiver of a pistol, for example.

Figure 7.24: Set of receiver models for TT-33 pistol: Comparison of tensor invariants. (a) $J_2(\tilde{\mathcal{R}})$, (b) $J_2(\mathcal{I})$, (c) $J_3(\tilde{\mathcal{R}})$ and (d) $J_3(\mathcal{I})$.

For the TT-33 models, each of the associated MPT frequency spectra have independent coefficients that are associated with both on and off diagonal entries of the tensor. The behaviour of $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])}$ for the different models is shown in Figure 7.25.

Finally, in Figure 7.26 the contours of $|\mathrm{Re}(\boldsymbol{J}^e)|$ and field lines for $\mathrm{Re}(\boldsymbol{J}^e)$ are shown on the plane spanned by $\boldsymbol{e}_1$ and $\boldsymbol{e}_3$ with $\xi_2 = 0$ and the TT-33 with chamfers model for the situations where $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_1^{(1)}$, $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_2^{(1)}$, $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_3^{(1)}$ are the eddy currents corresponding to $\omega = 10^3$ rad/s.
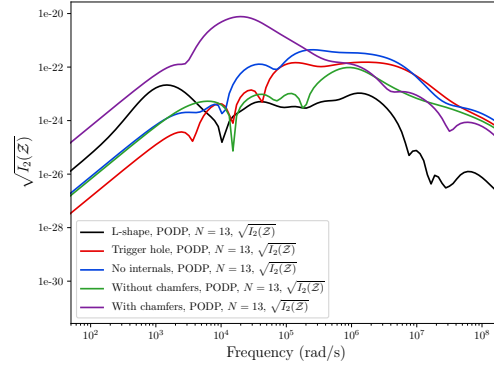
Figure 7.25: Set of receiver models for TT-33 pistol: Comparison of the tensor invariant $\sqrt{I_2(\mathcal{Z})}$.

(a) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_1^{(1)}$
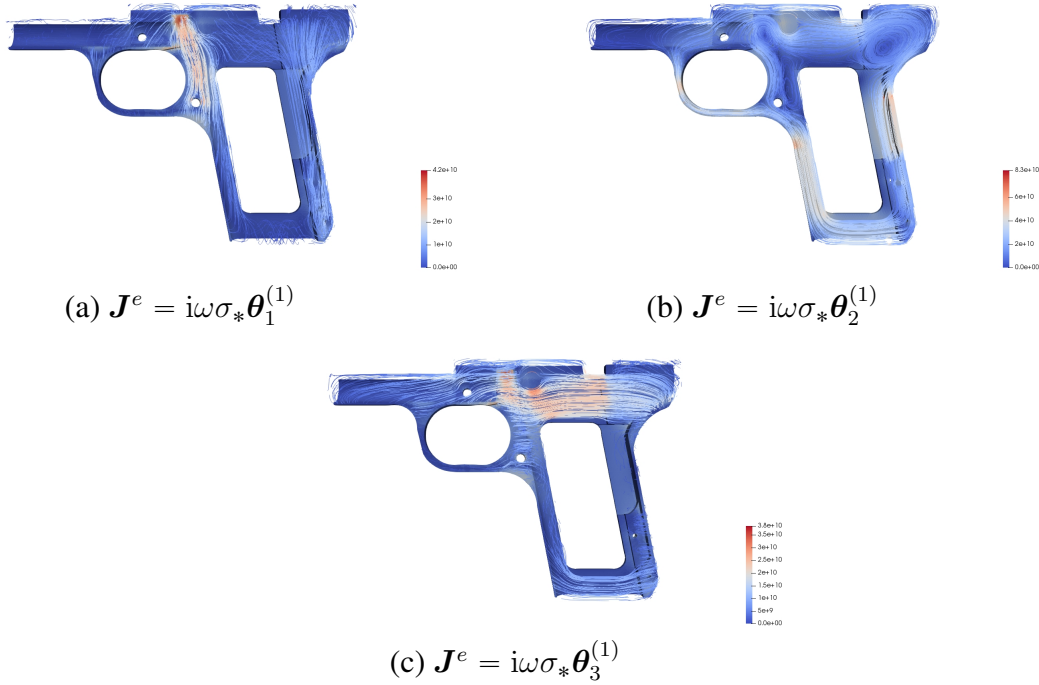
(b) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_2^{(1)}$

(c) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_3^{(1)}$

Figure 7.26: TT-33 with chamfers from the set of receiver models for TT-33 pistol: Contours of $|\mathrm{Re}(\boldsymbol{J}^e)|$ and streamlines for $\mathrm{Re}(\boldsymbol{J}^e)$ on the plane spanned by $\boldsymbol{e}_1$ and $\boldsymbol{e}_3$ with $\xi_2 = 0$. (a) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_1^{(1)}$, (b) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_2^{(1)}$ and (c) $\boldsymbol{J}^e = \mathrm{i}\omega\sigma_*\boldsymbol{\theta}_3^{(1)}$

7.5 Threat Items: Knives

Knife crime in the U.K. is a persistent issue with 47 000 offences involving a knife or sharp instrument in England and Wales from April 2018 - March 2019, 285 of which currently recorded as homicide [38] with trend being a significant increase in the last 8-9 years. The early recognition of threat objects through metal detection may help to reduce the number of offences involving a sharp instrument. In this section, MPT spectral signature characterisations for exemplar knife models are presented.

A set of 5 different knife models is considered, which are named as chef, cutlet, meat cleaver, Santoku and Wusthof. The chef knife is a model of a cheap chef knife, featuring a 20cm long, 5cm tall blade with a constant thickness of 1.5mm and a partial tang [4]. This model is an example of a stamped knife would normally be constructed with a plastic handle and only the cutting edge of the knife would be sharpened to a point. Obviously, only the metallic part of the knife has been modelled . The cutlet knife is a model of a cheap cutlet, featuring a 11cm long, 2cm tall blade with a constant thickness of 1.25mm and a partial tang. This model is an example of a stamped knife and would normally be constructed with a plastic handle and only the cutting edge of the knife would be sharpened to a point. The cleaver is a model of a meat cleaver featuring a 20cm long, 9cm tall blade with a thickness of 3mm over the majority of the blade with a double bevel 6.4 cm from the spin and a full tang with 3 rivets, which are each 2cm long, have radius 3mm and are spaced 45mm apart centre to centre. This model could be made using either the method of stamping or forging. The Santoku features a 17cm long, 4.5cm tall blade with a thickness of 1.275mm at its spine, which tapers over the height of the blade to a point at the cutting edge. It features a full tang with 3 rivets, which are each 2cm long, have radius 3.5mm and are spaced 42.5mm apart centre to centre. This model would normally be constructed from a single piece of steel with the two sides of the handle being made with either wood or a plastic material which are then both stuck and riveted to the steel. Finally, the Wusthof has a 20cm long, 5cm tall blade with a thickness of 2mm at its spine, which tapers over the height of the blade to a point at the cutting edge. It features a full tang with 3 rivets, which are each 2cm long, have radius 3.5mm and are spaced 42.5mm apart centre to centre. In each case, the measurements quoted have been obtained by approximately measuring the dimensions of common household knives. The blade of the knives have been assumed to be made of 440 grade stainless steel, which has a relative permeability $\mu_r = 62$ [125] and conductivity $\sigma_* = 1.6 \times 10^6$ S/m[92], but modelled instead with a lower relative permeability $\mu_r = 5$, and the rivets to be made of copper, which is non-magnetic having a relative permeability $\mu_r = 1$ and a conductivity $\sigma_* = 5.8 \times 10^7$ S/m [92]. Note that each of the knives are simply connected.

A mesh of each of the geometries was generated assuming dimensionless units, the size parameter $\alpha = 0.001$ m and by placing the knife configuration centrally in a box of dimensions $[-1000, 1000]^3$. The resulting meshes contain $25\,742$, $14\,935$, $55\,226$, $55\,226$ and $79\,945$ unstructured tetrahedra for the chef, cutlet, meat cleaver, Santoku and Wusthof knives, respectively, and images of the distribution of elements on the surface of the object are reproduced in Figure 7.27. Each of the knives has been orientated so that the blade is parallel to the e_1 direction and lies in the plane spanned by e_1 and e_2 with the knife configuration being symmetrical in the e_3 direction. Thus, there are 4 independent coefficients each in $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ corresponding to $(\tilde{\mathcal{R}})_{11}$, $(\tilde{\mathcal{R}})_{22}$, $(\tilde{\mathcal{R}})_{33}$, $(\tilde{\mathcal{R}})_{13} = (\tilde{\mathcal{R}})_{31}$ at each frequency with similar for $\mathcal{I}$.

The results obtained by using $N = 13$ representative full order solution snapshots at

---

[4] The tang is the back portion of the blade, which extends or connects to a handle, a full tang extends the full length of the handle while a partial tang only extends partially in to the handle [129]
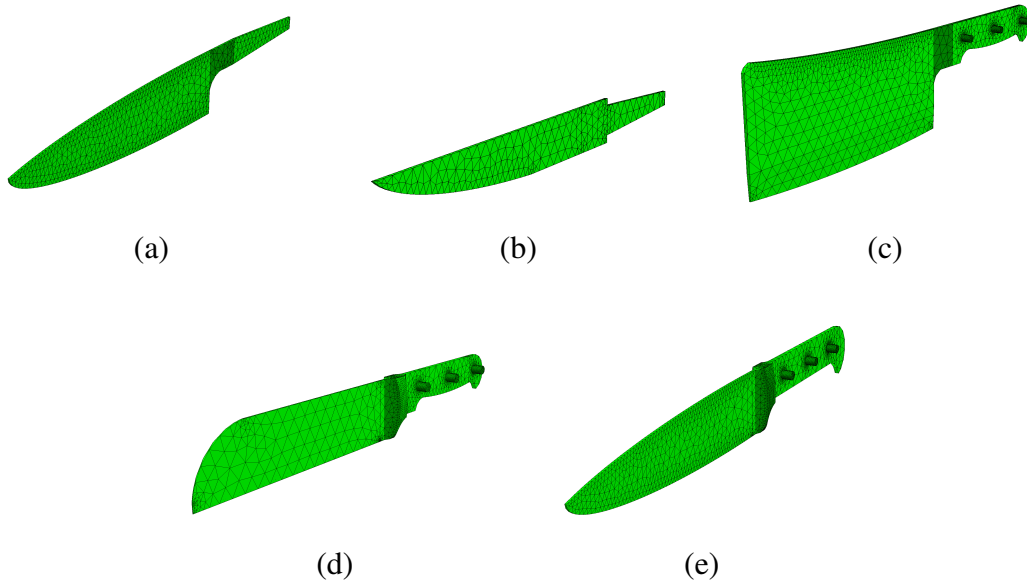
Figure 7.27: Set of knives: surface distribution of elements for (a) chef, (b) cutlet, (c) meat cleaver, (d) Santoku and (e) Wusthof

logarithmically spaced frequencies in the range $10^1 \leqslant \omega \leqslant 10^{10}$ rad/s were found to converge with $p = 4$ elements. Then, by applying the PODP algorithm described in Chapter 3 with a tolerance of $TOL = 10^{-6}$, the MPT spectral signature for each of the receiver models was obtained. A comparison of the MPT spectral signatures using the principal invariants I_i, $i = 1, 2, 3$ for $\tilde{\mathcal{R}}[\alpha B, \omega, \mu_r, \sigma_*]$ and $\mathcal{I}[\alpha B, \omega, \mu_r, \sigma_*]$ that have been obtained using the PODP approach in Figure 7.28. In this figure, consideration has been restricted to frequencies such that $10^2 \leqslant \omega \leqslant 10^8$rad/s in order to allow comparisons with the earlier results. In practice, the eddy current model breaks down at a frequency $\omega_{limit} < 5 \times 10^6$rad/s for all the knives considered and so higher frequencies are not relevant.

The results obtained for the different models shown in Figure 7.28 have some similarities to the TT-33 models in that $I_1(\tilde{\mathcal{R}})$ is monotonically increasing with $\log \omega$ and $I_2(\tilde{\mathcal{R}})$, $I_3(\tilde{\mathcal{R}})$ are not monotonically increasing or decreasing with $\log \omega$ and the curves $I_i(\mathcal{I})$, $i = 1, 2, 3$, each show a single local maximum with $\log \omega$. However, the characteristics of the curves is otherwise quite different, again motivating that discrimination between objects is possible. Comparing the different knife models, different behaviour is observed for the invariants in each case. On closer inspection of the eigenvalues of $\tilde{\mathcal{R}}$ and $\mathcal{I}$ it is possible to observe multiple non-stationary points of inflection and multiple local maxima, respectively, particularly when considering the Santoku and Wusthof knives, which are inhomogeneous. Although the cleaver also has inhomogeneous materials, the larger extent of material in the blade largely disguises these effects. The corresponding results for the alternative invariants are shown in 7.29, where again observe a significant difference between the clever and the other models.

114

(a) $I_1(\tilde{\mathcal{R}})$

(b) $I_1(\mathcal{I})$
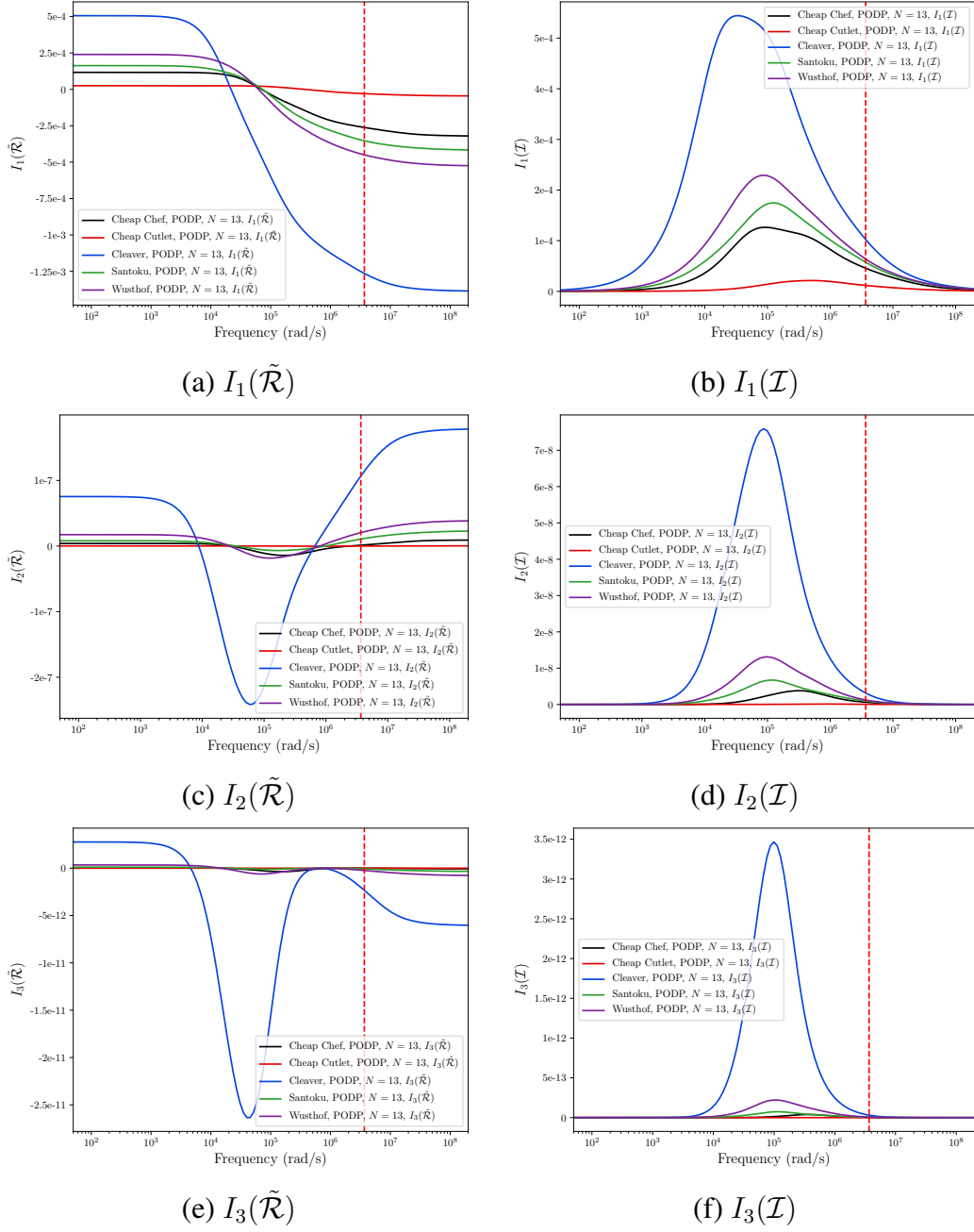
(c) $I_2(\tilde{\mathcal{R}})$

(d) $I_2(\mathcal{I})$

(e) $I_3(\tilde{\mathcal{R}})$

(f) $I_3(\mathcal{I})$
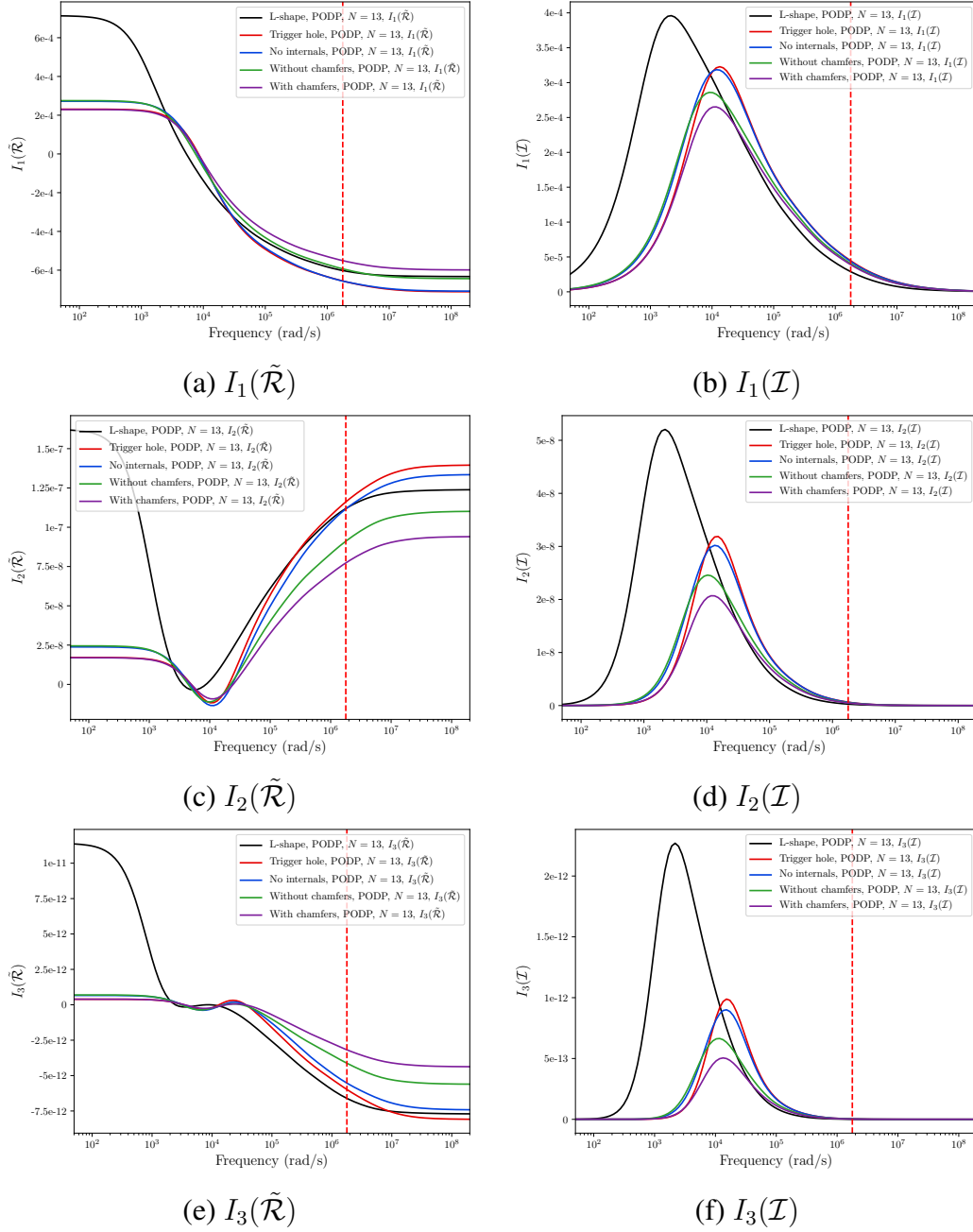
Figure 7.28: Set of knives: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$.

115

(a) $J_2(\tilde{\mathcal{R}})$ (b) $J_2(\mathcal{I})$
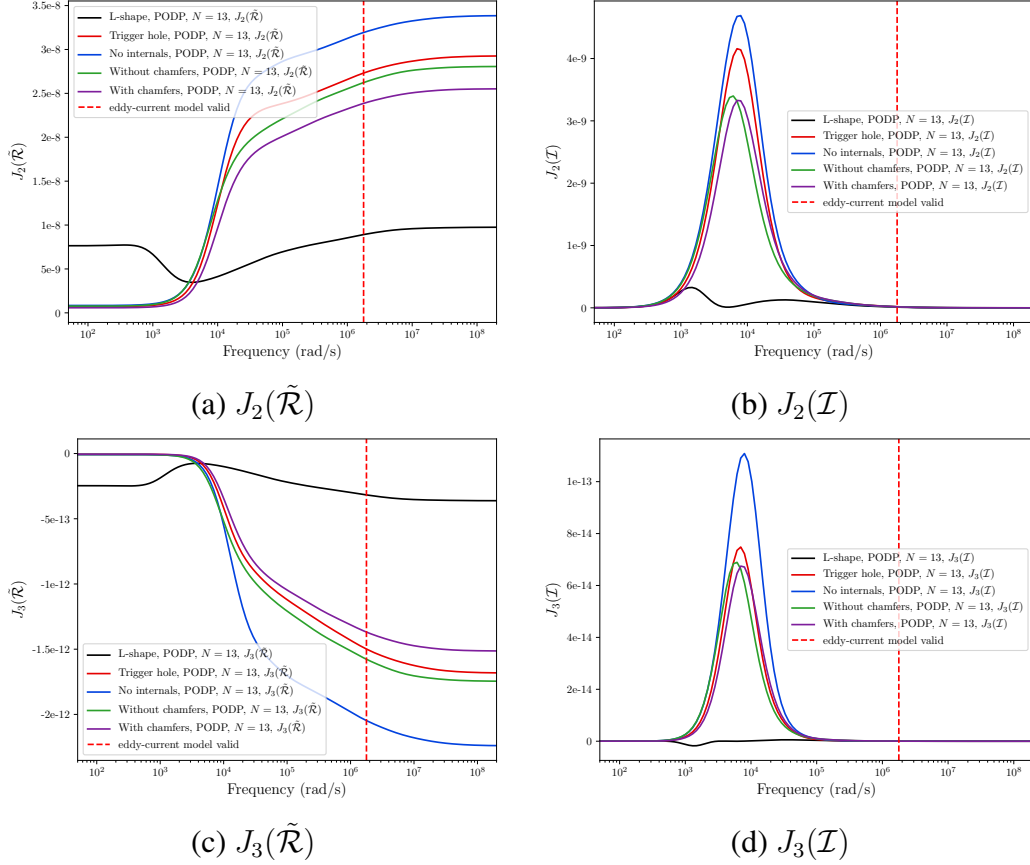
(c) $J_3(\tilde{\mathcal{R}})$ (d) $J_3(\mathcal{I})$
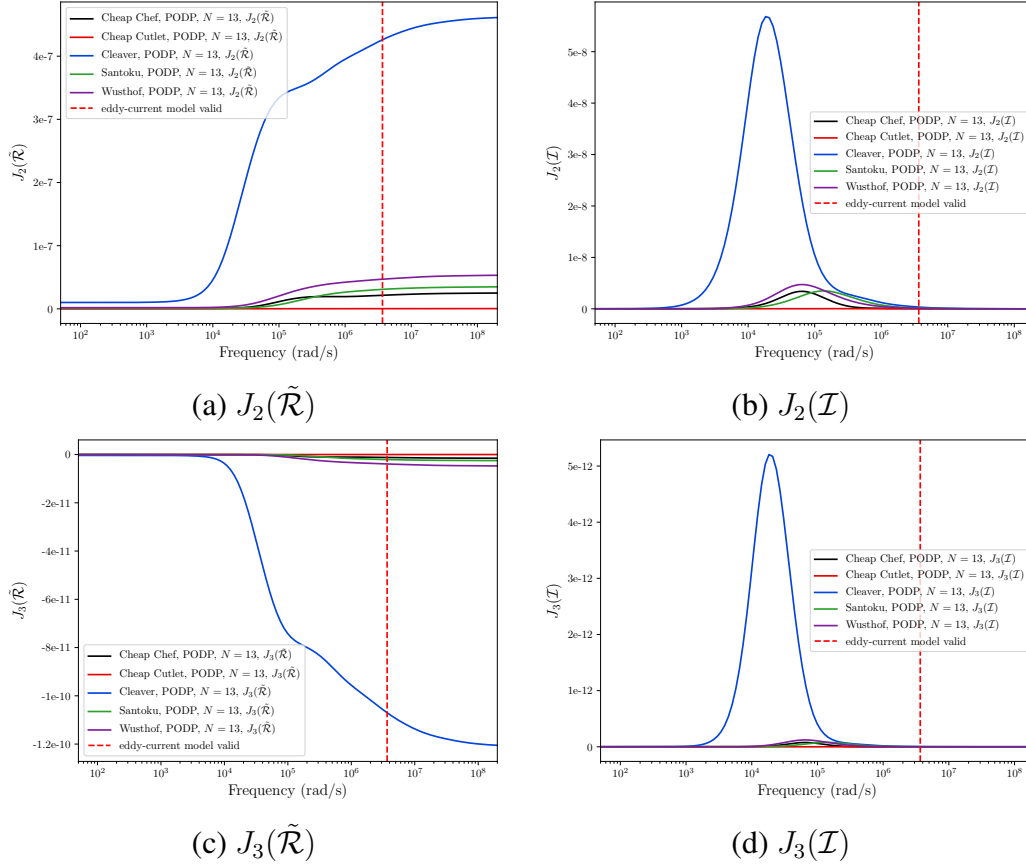
Figure 7.29: Set of knives: Comparison of tensor invariants. (a) $J_2(\tilde{\mathcal{R}})$, (b) $J_2(\mathcal{I})$, (c) $J_3(\tilde{\mathcal{R}})$ and (d) $J_3(\mathcal{I})$.

For the knife models, each of the associated MPT frequency spectra have independent coefficients that are associated with both on and off diagonal entries of the tensor. The behaviour of $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])}$ for the different models is shown in Figure 7.30.
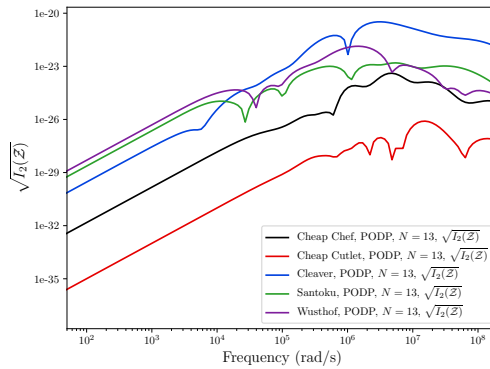
Figure 7.30: Set of knives: Comparison of the tensor invariant $\sqrt{I_2(\mathcal{Z}[\alpha B, \omega, \sigma_*, \mu_r])}$.

7.6 **MPT-Library**

In addition to the examples of real world object considered previously, a larger, more extensive list of objects has been produced and documented in the form of an open source dataset, `MPT-Library` [132]. This dataset has been constructed using the ROM method described in Section 3.3 using the open source `MPT-Calculator` software [134], which was discussed in Chapter 4. A full list of the of the objects can be found in Table 7.2, this includes the number of different material versions of each object there are in the library. Some exemplar objects can be found in Figures 7.31 and 7.32, these figures show threat and non-threat objects respectively, for which MPT spectral signatures have been obtained and are contained in [132].

Hammer heads

Knuckle dusters

Pairs of scissors

Screw drivers
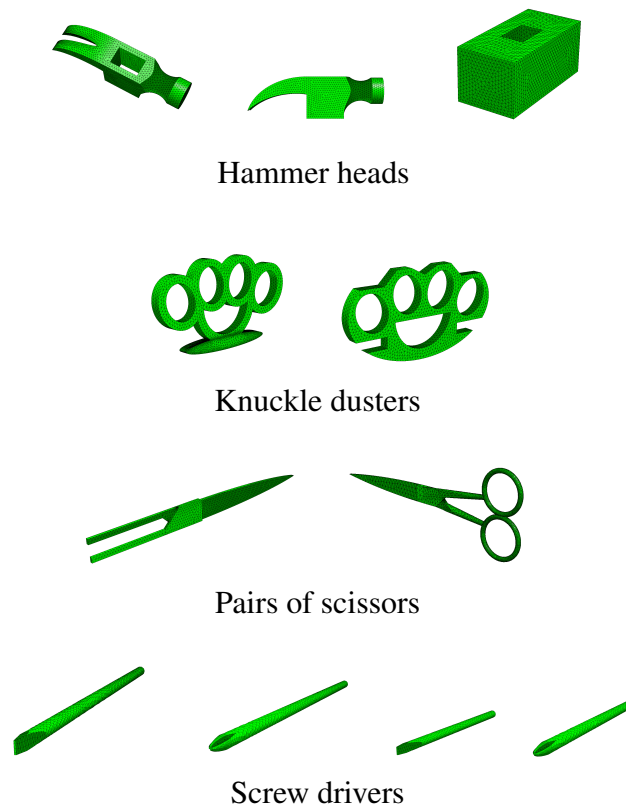
Figure 7.31: Set of multiple threat and non-threat objects: Sample illustrations of some of the different threat object geometries considered (not to scale).

Bracelets

Belt buckles

Earrings / Piercings

Pendents

Rings

Shoe shanks

Watch/ watch with metallic strap
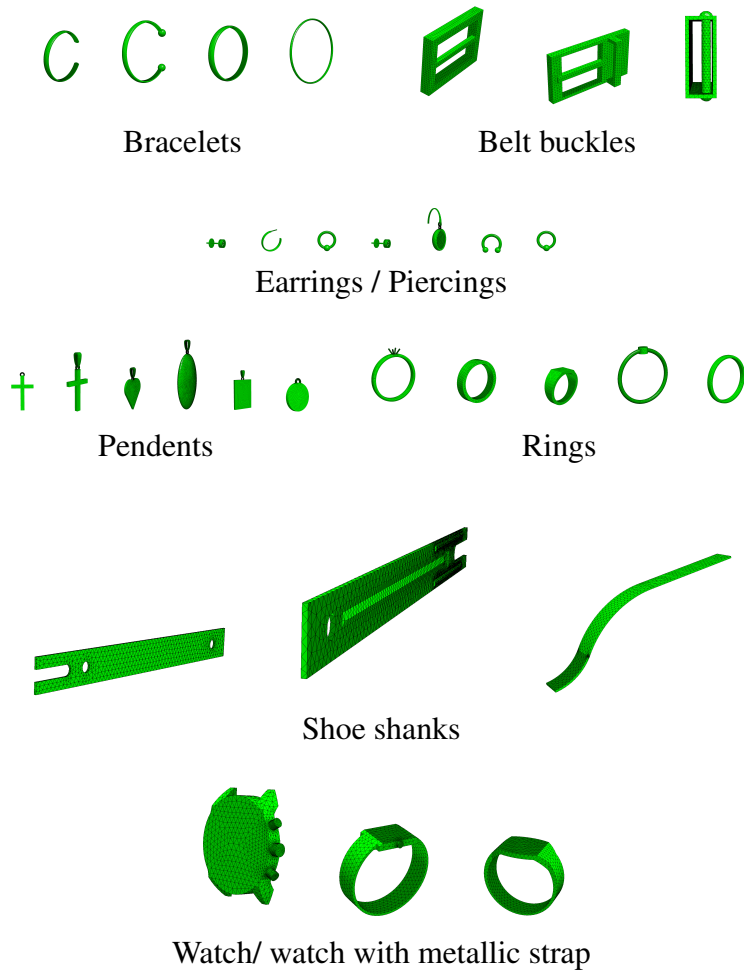
Figure 7.32: Set of multiple threat and non-threat objects: Sample illustrations of some of the different non-threat object geometries considered (not to scale).

7.7   Chapter Summary

This chapter provided a series of illustrative examples which demonstrated how the ROM approach described in Chapter 3 can be combined with an appropriate choice of eigenvalues or tensor invariants in Section 6.3 and sampling at M frequencies to form a realistic dataset of object characterisations.

The key novelties of the chapter were, the application of the ROM approach described in Chapter 3 and 4 to a set of real world geometries creating a first of its kind dataset of MPT spectral signature characterisations for real world objects. Constructing this dataset based upon the invariants described in Section 6.3 is highly novel and previously not considered. The investigation into how small changes in a geometry affect the MPT spectral signature and the invariants derived from this signature, this also gives insight to the level of detail required for objects in such a library. The `MPT-Library` discussed in this chapter will be used in Chapter 9 for classification based on the library.

Table 7.2: List of objects contained in the `MPT-Library` along with number of different material versions.

Object	Versions	Object	Versions	Object	Versions
Bangle cutout	3	Flat shank indent	1	Regular bar	3
Bangle cutout spheres	3	Flat signet ring	3	Regular circular flathead	3
Box design buckle	4	Hair dressing scissors	3	Regular circular plushead	3
Buckle with loop	4	Heart pendant	3	Ring with ball	3
Cheap Chef	1	High heel shank	1	Round Groove	1
Cheap Cutlet	1	Knuckle dusters	1	Round No Groove	1
Cheap partial tang scissors	3	Knuckle dusters rounded	1	Santoku	1
Claw wooden handle	3	Large bar	3	Short circular flathead	3
Cleaver	1	Large gen earings single	3	Short circular plushead	3
Cross	3	Leather band watch man	3	Signet ring	3
Cross with bail	3	Link band rectangle watch female	3	Soverign pendant	3
Curved bar	3	Link band watch female	3	Soverign pendant with bail	3
Curved bar 270	3	Link band watch man	3	Square Chamfer Groove	1
Engagement ring claw	3	Long circular flathead	3	Square Groove	1
Engagement ring hope	3	Long circular plushead	3	Square hammer	3
Fifty p	1	One p pre	1	Stud earings single	3
Five p pre	1	One Pound	1	Ten p pre	1
Flat buckle	4	Oval pendant	2	Thick 270 hoop single	3
Flat end bar	3	Plain bangle	3	Twenty p	1
Flat ring	3	Plain bangle thin	3	Two p pre	1
Flat shank	1	Receiver	1	Two Pound	1
Wedding band male	3	Wusthof	1		

Part IV

Object Classification

Chapter 8

Probabilistic and Non-Probabilistic Machine Learning Classifiers

8.1   Introductory Remarks

This chapter documents and compares probabilistic and non-probabilistic ML classifiers that are appropriate for classifying objects when the features are with MPT invariants, with the goal of developing an ML classifier that could be deployed in a walk through metal detector.

The chapter has been adapted from the work presented by the author in [135].

The novelties of the chapter are as follows: Firstly, a methodology is documented to create a dictionary of threat/ non-threat objects based on principal tensor invariants. Then, consideration of appropriate noise that could be added to simulate real world measurements is considered. Lastly, ML classifiers are reviewed and collated in a form suitable for being used in conjunction with principal invariants (discussed in Section 6.3.2) as object features.

The chapter is organised as follows: First in Section 8.2, a recap of arguments for the advocated use of principal tensor invariants as ML features, the methodology for the creation of a dictionary based on these invariants and the method used to add noise to simulate a real world measurement system are presented. In Section 8.3, a series of probabilistic and non-probabilistic ML classifiers are documented and compared. In Section 8.4, metrics to measure performance of the classifiers along with methods to further validate these classifiers are presented. The chapter is concluded in Section 8.5 with some closing remarks.

8.2   MPT Spectral Signature Invariants for Object Classification

As discussed in Section 6.3, the process of classification, as described by Bishop [20], involves asigning an input vector $\mathbf{x}$ and assigning it to one of K discrete classes C_k, $k = 1, \ldots, K$. Following the 1-of-K coding system recommended by Bishop [20], the

entries in a vector $\mathbf{t} \in \mathbb{R}^K$ take the form

$$
\mathrm{t}_i := \left\{ \begin{array}{ll} 1 & \text{if } i = k \\ 0 & \text{otherwise} \end{array} \right. ,
$$

if the correct class is C_k. In Section 6.3, different choices for the F features in the input vector $\mathbf{x} \in \mathbb{R}^F$ are considered, which are associated with either the eigenvalues, principal invariants or deviatoric invariants of $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$, respectively, evaluated at different frequencies $\omega = \omega_m$, $m = 1, \ldots, M$. The focus of this chapter is the situation where

$$
\mathrm{x}_i = \left\{ \begin{array}{ll} I_j(\tilde{\mathcal{R}}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m-1)M \\ I_j(\mathcal{I}[\alpha B, \omega_m, \sigma_*, \mu_r]), & i = j + (m+2)M \end{array} \right. , \tag{8.1}
$$

with $j = 1, 2, 3$, $m = 1, \ldots, M$. For exact arithmetic, and a rank 2 tensor $\mathcal{A}$,

$$
I_1(\mathcal{A}) := \mathrm{tr}(\mathcal{A}), \tag{8.2a}
$$

$$
I_2(\mathcal{A}) := \frac{1}{2} \left(\mathrm{tr}(\mathcal{A})^2 - \mathrm{tr}(\mathcal{A}^2) \right), \tag{8.2b}
$$

$$
I_3(\mathcal{A}) := \det(\mathcal{A}), \tag{8.2c}
$$

are the principal invariants. In the above, $\mathrm{tr}(\cdot)$ denotes the trace and $\det(\cdot)$ the determinate and it is assumed the entries of $\mathcal{A}$ are arranged as a symmetric 3×3 matrix, which is also celled $\mathcal{A}$. The input vector $\mathbf{x}$ is defined in this way for the following reasons:

1. Using features that are invariant to object rotation is important as both a hidden object's shape and its orientation are unknown. Using either eigenvalues $\lambda_i(\tilde{\mathcal{R}})$, $\lambda_i(\mathcal{I})$, $i = 1, 2, 3$ or the principal tensor invariants overcomes this issue as both are invariant to an object's unknown orientation and, hence, simplifies the classification problem.

2. Invariants overcome the ordering issue that is associated with assigning the eigenvalues as the invariants are independent of how the eigenvalues are assigned.

3. The invariants can be computed as either products or sums of the entries of $\mathcal{A}$ without first calculating $\lambda_i(\mathcal{A})$. Hence, they are smooth functions of the tensor coefficients. Rather than a sub-determinant method, an alternative approach for finding I_3 follows by first converting $\mathcal{A}$ to (upper) triangular form and the determinant follows by the product of its diagonal entries. However, while the eigenvalues of a triangular matrix are its diagonal entries, the eigenvalues of $\mathcal{A}$ are not preserved when it is converted and further computation is needed. There are many iterative computational alternatives for finding $\lambda_i(\mathcal{A})$ (see e.g. [42]), which are preferred to directly finding the roots of $\det(\mathcal{A} - \lambda \mathbb{I}) = 0$, especially for large matrices. Although, even a simple low-cost approach for directly determining the eigenvalues of a symmetric 3×3 matrix still requires determining an inverse cosine of a non-linear function of the matrix?s trace and determinant [116] and, hence, involves non-smooth operations. Thus, the finding $\lambda_i(\mathcal{A})$ may result in a loss of accuracy in practical numerics compared to using the simple sums or products to find $I_i(\mathcal{A})$.

4. The (probabilistic) ML classification algorithms considered are better at capturing an underlying relationship between features and the likelihood of class that is smooth, albeit, with noisy data. The process involved in finding eigenvalues may lead to a greater entanglement between class and features that a classifier might need to unravel compared to using invariants.

As an example, Figure 8.1 shows a comparison of the principal tensor invariants for a selection of 4 different metallic watch styles computed using the method presented in Chapter 3 and are objects included in the `MPT-Library`. The object dimensions are in mm so $\alpha = 0.001$m and the results shown are for the case where the material is gold, so that $\sigma_* = 4.25 \times 10^7$ S/m and $\mu_r = 1$ (`MPT-Library` also includes MPT spectral signatures for watches made of platinum and silver). An unstructured mesh of tetrahedra is used to discretise each object and the truncated unbounded region which surrounds it, resulting in meshes ranging from $14\,935$ to $17\,5217$ elements. In each case, the truncated boundary for the non-dimensional transmission problem is $[-1000, 1000]^3$. Order $p = 4$ elements were applied on the meshes and snapshot solutions obtained at 13 logarithmically spaced frequencies over the range 1 rad/s $\leqslant \omega \leqslant 1 \times 10^{10}$ rad/s. The MPT spectral signature for each object was produced using the PODP method discussed in Section 3.3 using a relative singular value truncation of 10^{-4}. Also shown is a vertical line, which indicates the value of ω that the eddy current model assumption is likely to become inaccurate for this geometry [76, 107]. Finally, a grey window is included corresponding to the frequency range 5.02×10^4 rad/s $\leqslant \omega \leqslant 8.67 \times 10^4$ rad/s, where measurements taken by a commercial walk through metal detector [82], and the greater range 7.53×10^2 rad/s $\leqslant \omega \leqslant 5.99 \times 10^5$ rad/s, where measurements are taken using recent MPT measurement system [96], the latter being able to capture more information from the signature. These spectral signatures will form part of the dictionary for object classification, which will be discussed later in Section 9.3.

8.2.1 Construction of the Dictionary

Each class C_k may be comprised of $G^{(k)}$ geometries and, in addition, $V^{(k)}$ variations in object size and object materials are considered so that each class is comprised of $P^{(k)}$ different samples. In total, over all the classes, there are $P = \sum_{k=1}^{K} P^{(k)}$ samples.

Given the information α, B, σ_*, μ_r the MPT spectral signature described by $\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r]$ and $\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r]$ can be obtained, as described in Section 3.3, and then invariants then follow from (8.1). This process is then repeated for each of the geometries $B^{(g_k)}$, $g_k = 1, \dots, G^{(k)}$ that makes up the class. To take account of the $V^{(k)}$ different object sizes and materials, physically motivated samples $\alpha \sim N(m_\alpha, s_\alpha)$ and $\sigma_* \sim N(m_{\sigma_*}, s_{\sigma_*})$ are drawn, where m_α and m_{σ_*} denotes means and s_α and s_{σ_*} standard deviation, respectively, and $N(m, s)$ denotes a normal distribution with mean m and standard deviation s. While it would be possible to also obtain the MPT spectral signature using the method described in Section 3.3 for each sample, instead, the computational cost of obtaining these spectral signatures is reduced by using the scaling results derived
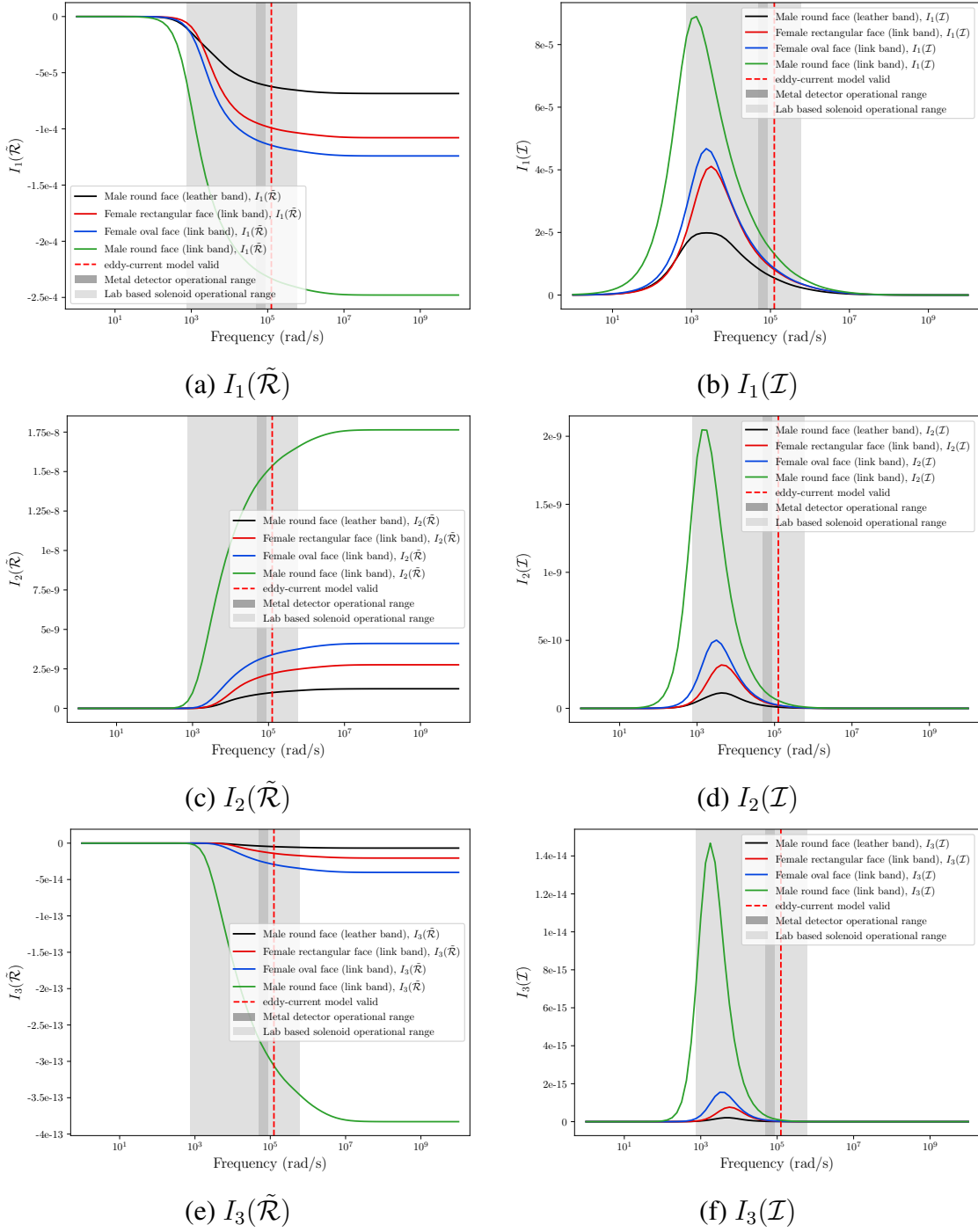
Figure 8.1: Set of watches: Comparison of tensor invariants. (a) $I_1(\tilde{\mathcal{R}})$, (b) $I_1(\mathcal{I})$ (c) $I_2(\tilde{\mathcal{R}})$, (d) $I_2(\mathcal{I})$, (e) $I_3(\tilde{\mathcal{R}})$ and (f) $I_3(\mathcal{I})$.

in Lemmas 2.7.1 and 2.7.2. Given an MPT spectral signature of an object for a given α, B, μ_r and σ_*, these results predict the MPT spectral signature of another object with the same B and μ_r, but different σ_* and α, at negligible computational cost. The results hold for objects with homogeneous materials and there are no restrictions on ω (upto the limit of the eddy current model), but assume that a broader band MPT spectral signature is

126

available for the original object compared to the grey windows highlighted in Figure 8.1. The invariants then again follow from (6.3). Finally, by setting the class labels, the pairs $(\mathbf{x}_p \in \mathbb{R}^F, \mathbf{t}_p \in \mathbb{R}^K)$, $p = 1 \ldots, P^{(k)}$ [1] for all the samples that make up the class C_k are obtained. Repeating this process for each of the classes gives rise to the the general dictionary

$$D = ((\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), \ldots, (\mathbf{x}_P, \mathbf{t}_P)), \tag{8.3}$$

or alternatively,

$$D = (D^{(1)}, D^{(2)}, ..., D^{(K)}), \tag{8.4}$$

where,

$$D^{(k)} = ((\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), ..., (\mathbf{x}_{P^{(k)}}, \mathbf{t}_{P^{(k)}})), \tag{8.5}$$

is the dictionary associated with class C_k and consists of $P^{(k)}$ observations.

In practice, the dictionary stated in (8.3) is split as $D = (D^{(\text{train})}, D^{(\text{test})})$ where $D^{(\text{train})}$ is the training and $D^{(\text{test})}$ is the testing dataset, respectively. The purpose of this splitting is to enable the classifier to be trained on given data $D^{(\text{train})}$ and then tested on previously unseen data $D^{(\text{test})}$. Although there is no optimal choice for the ratio of training to testing, the choice is made to employ a ratio of 3:1 throughout, which is commonly used in ML classification and is in the range of 3:1 to 4:1 that Kuhn and Johnson suggest [66][pg.71].

8.2.2 Noise

When MPT spectral signatures for hidden objects are measured by a metal detector they will contain un-avoidable errors, as pointed out in [133]. For example, if an object is placed in a non-uniform background magnetic field that varies significantly over the object there is a modelling error since the background field in the rank 2 MPT model assumes the field over the object is uniform. There are other errors and noise associated with capacitive coupling with other low-conducting objects or soil, if the object is buried, as well as other generals noise (e.g. from amplifiers, parasitic voltages and filtering) [83]. The accuracy of the signature can be improved by repeating the measurements and applying averaging filters, at the cost of spending more time to take the measurements. However, in a practical setting, there is trade-off to be made in terms of improving the accuracy against the meaurement time and, consequently, the accuracy of the measured MPT coefficients is about 1% to 5% [32, 82, 83], depending on the application.

The MPT spectral signature coefficients employed have been produced numerically using the method described in Section 3.3. This means that the MPT coefficients are obtained with higher accuracy than can currently be achieved from practical measurements, since the spectral signature is accurately computed for a large frequency range (up to the limit of the eddy current model) rather than noisy measurements being taken at a small number of discrete frequencies. The advantage of this is it allows a much larger library of objects and variations of materials to be considered, which is all highly desirable for achieving greater fidelity and accuracy when training an ML classifier. But, for practical classification, noise appropriate to the system must be added.

---

[1] Note that the entries of $\mathbf{t}_p$ are all 0 except for $(\mathbf{t}_p)_{C_k} = 1$ corresponding to the kth class

After filtering and averaging, the noise remaining in an MPT spectral signature measured by a metal detector can be well approximated by Gaussian additive noise, with a level dependent on the factors identified above. Hence, in this work, noise is added to the simulated MPT spectral signatures in the following way: A signal noise ratio (SNR) is specified in decibels and used to determine the amount of noise to add to each of the complex tensor coefficients $(\mathcal{M}[\alpha B^{(p)}, \omega, \sigma_*, \mu_r])_{ij}$ as a function of frequency for each object $\alpha B^{(p)}$ in the dictionary. Considering each of the i,jth MPT coefficients individually,

$$\mathrm{v} := (\mathcal{M}[\alpha B^{(p)}, \omega_m, \sigma_*, \mu_r])_{ij},$$

is introduced to calculate a noise-power measure as

$$\text{noise} = \frac{\overline{\mathrm{v}}\mathrm{v}}{10^{\text{SNR}/10}}.$$

The noisy coefficients are then specified as

$$(\mathcal{M}[\alpha B^{(p)}, \omega_m, \sigma_*, \mu_r])_{ij} + e_{ij} \qquad \text{where} \qquad e_{ij} = \sqrt{\frac{\text{noise}}{2}}(u + \mathrm{i}v)$$

with $u, v \sim N(0, 1)$. The above process is repeated for the 6 independent coefficients of the complex symmetric MPT and for each frequency in the spectral signature. SNR values of 40, 20 and 10dB lead to values of $|\frac{e_{ij}}{\mathcal{M}_{ij}}| = 0.01, 0.10$ and 0.32 on average, which is equivalent to 1%, 10% and 32% noise, respectively. One realisation of the effect of the added noise on $I_1(\tilde{\mathcal{R}})$ and $I_1(\mathcal{I})$ for a British one penny coin can be seen, in the next chapter, in Figure 9.1. Note, the model for noise results in a noise power measure that varies over the MPT spectral signature according to $\overline{\mathrm{v}}\mathrm{v}$. If the physical system behaves differently, this can be taken in to account by applying an appropriate model for the noise at this stage. Once the noisy $\mathcal{M}$ at each ω_m are found, the principal invariants of the real and imaginary parts of $\mathcal{M}$ at each ω_m easily follow. Hence, an entry $(\mathbf{x}, \mathbf{t}) \in D$ is replaced with $(\mathbf{x}^{\text{noise}}, \mathbf{t})$. By repeating this for all objects leads to the updated dictionary D.

8.3 Classification

In this section, a quick hands-on review of ML classification is provided. Readers who are familiar with this subject should skip this section as this material can be found in the references cited below.

8.3.1 Probabilistic versus Non-Probabilistic Classification

Applied to classification problems, Bayes' theorem can be expressed in the form [20]

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}, \qquad k = 1, \cdots, K, \tag{8.6}$$

which relates the posterior $p(C_k|\mathbf{x})$ to the likelihood $p(\mathbf{x}|C_k)$ and the prior $p(C_k)$ where, for classification,

$$p(\mathbf{x}) = \sum_{n=1}^{K} p(\mathbf{x}|C_n)p(C_n),$$

is easily explicitly obtained as the normalising constant.

In the inference stage of probabilistic classification, one seeks to design a classifier $\gamma_k(\mathbf{x})$ that provides a probabilistic output, which approximates $p(C_k|\mathbf{x})$. On the other hand, non-probabilistic classifiers either predict a class C_k with certainty or, more commonly, have a statistical interpretation that provides a frequentist approximation $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to $p(C_k|\mathbf{x})$. One measure of accuracy of classification is the mean squared error (MSE)

$$\text{MSE}(\gamma_k) = E_{\mathbf{x}}[\gamma_k(\mathbf{x}) - p(C_k|\mathbf{x})]^2, \tag{8.7}$$

where $E_{\mathbf{x}}$ is the expectation with respect to $p(\mathbf{x})$ [85][pg 309.]. If desired, this can be summed over the classes $k = 1, \ldots, K$ or considered for each class. Other metrics are considered in Section 8.4.

Given approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$, $k = 1, \ldots, K$, the class decision is typically achieved using the maximum a-posterior (MAP) estimate $\arg\max_{k \in K}(p(C_k|\mathbf{x})) \approx \arg\max_{k \in K}(\gamma_k(\mathbf{x}))$ i.e. the MAP estimate corresponds to the class C_n with n such that

$$n = \arg\max_{k \in K}(\gamma_k(\mathbf{x})). \tag{8.8}$$

However, the MAP may have drawbacks if there are several similar probabilities and/or if the data is noisy. Hence, understanding the uncertainty in the approximations of $p(C_k|\mathbf{x})$ are also important. This is considered in Section 8.3.5.

8.3.2 Bias and Variance

The classifiers considered are based on ML algorithms. For an ML method Γ, which takes $\mathbb{D} = D^{\text{train}}$ as the input and returns a learned classifier γ_k, $k = 1, \ldots, K$, Manning, Raghavan and Schütze [85][pg. 309-312] define the Learning-error$(\Gamma) = E_{\mathbb{D}}[\text{MSE}(\Gamma_{\mathbb{D}})]$ as a measure of accuracy of the classifier, which is to be minimised, and they show that it can be expressed as

$$\begin{aligned}
\text{Learning-error}(\Gamma) :=& E_{\mathbf{x}}[\text{bias}(\Gamma, \mathbf{x}) + \text{variance}(\Gamma, \mathbf{x})], \\
\text{bias}(\Gamma, \mathbf{x}) :=& [p(C_k|\mathbf{x}) - E_{\mathbb{D}}\Gamma_{\mathbb{D}}(\mathbf{x})]^2, \\
\text{variance}(\Gamma, \mathbf{x}) :=& E_{\mathbb{D}}[\Gamma_{\mathbb{D}}(\mathbf{x}) - E_{\mathbb{D}}\Gamma_{\mathbb{D}}(\mathbf{x})]^2,
\end{aligned}$$

where $\Gamma_{\mathbb{D}}(\mathbf{x})$ implies that the ML method is applied to data set $\mathbb{D}$ and outputs approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, for data $\mathbf{x}$.

Bias measures the difference between the true $p(C_k|\mathbf{x})$ and the prediction $\Gamma_{\mathbb{D}}(\mathbf{x})$ averaged over the training sets [85][pg 311]. Bias is large if the classifier is consistently wrong, which may stem from erroneous assumptions. A small bias may indicate several things and not just that the classifier is consistently correct, for further details see [85]. Related

to high bias is underfitting, which refers to a classifier that is unable to capture the relationship between the input and output variables correctly and produces large errors on both the training and testing data sets.

Variance is the variation of the prediction of learned classifiers and is the average difference between $\Gamma_{\mathbb{D}}(\mathbf{x})$ and its average $E_{\mathbb{D}}\Gamma_{\mathbb{D}}(\mathbf{x})$ [85][pg. 311]. Variance is large if different training sets give rise to different classifiers and is small if the choice of training set has only a small influence on the classification decisions, for further details see [85]. Related to high variance is overfitting, which is the opposite of underfitting, and refers to a classifier that has too much complexity and also learns from the noise resulting in high errors on the test data.

The ideal classifier would be a classifier having low bias and low variance, however, the two are inextricably linked and there is therefore a trade off between two [20, 85]. While the performance of classifiers is very application dependent, linear classifiers tend to have a high bias and low variance and non-linear classifiers tend to have a low bias and high variance [85]. The ML methods Γ that are considered here are commonly found in established ML libraries such as `scikit-learn`, which is the implementation that is used, due to its good range of probabilistic and non-probabilistic classifiers within a single environment and ease of use. The finer details of the different methods can be found in [20, 54, 40] among many others, although a brief summary of the methodologies is given, for those less familiar with the approaches and to set notation. First non-probabilistic methods are discussed and then probabilistic classifiers.

8.3.3 Non-Probabilistic Classifiers

8.3.3.1 Decision Trees

Tree based algorithms can simply be interpreted as making a series of (binary) decisions that ultimately lead to the prediction of a class. For $F = 2$, this results in the partition of the feature space into a series of K rectangular regions corresponding to the different classes. To establish the regions, and, hence the classes, a tree is constructed where, at each node, a binary decisions about a component of $\mathbf{x}$, and the process is terminated with a decision at the leaf-node (for example, if $x_1 \leqslant t_1$ and $x_2 \leqslant t_2$ then the class is C_1, whereas if $x_1 \leqslant t_1$ and $x_2 > t_2$ the class is C_2 etc). In order to grow a tree, a greedy algorithm is applied to decide how to split the variables, the split points and the topology of the tree [54][pg. 308]. Growing a tree that is too large may overfit the data, while a small tree may not capture the structure. To overcome this, a larger than needed tree is usually grown and is then pruned. To understand how this works, consider a tree, then by applying the rules of the tree, a subset of the training data for its m node is obtained. It then follows that $\hat{p}_{mk}$ is the proportion of this data that has class C_k and the associated class is determined as $k(m) = \arg\max_k \hat{p}_{mk}$. The pruning is then achieved by applying a cost-complexity optimisation based on the node impurity measures *misclassification error*, *Gini index* or *cross-entropy*, which can each be written in terms of ((non-linear) functions and/or sums of) $\hat{p}_{mk}$. For further details see [54][pg. 309] [20][pg. 666]. Decision trees are often

used as a base classifier for more advanced ensemble methods such as gradient boost and random forests discussed below.

8.3.3.2   Random Forests

The random forests classifier is an example of what is known as "a bootstrap aggregating (bagging) algorithm" that attempts to reduce the variance of the typically high-variance low-bias decision tree algorithm [54][pg. 587]. As Hastie, Tibshirani and Friedman [54][pg. 587] continue to describe, the idea behind bagging is to average many noisy, but unbiased models to reduce their variance. Trees are notoriously noisy, and hence benefit from averaging. Since each tree generated by bagging is identically distributed then expectation of an average of such trees is the same as the expectation of any of them. This means that bias unchanged, but random forests offer the hope of improvement by variance reduction. For details of their implementation see [54][Chapt. 15].

8.3.3.3   Support Vector Machine

A support vector machine (SVM) classifier generalises simple linear classifiers (e.g. Fisher's linear discriminant analysis) by producing non-linear, rather than linear, classification boundaries. These are obtained by constructing a linear boundary in a transformed version of the feature space, which becomes non-linear in the feature space [20][pg. 325], [54][Chapt. 12], [30]. So, after making a transformation $\psi(\mathbf{x}) : \mathbb{R}^F \to \mathbb{R}^{\tilde{F}}$ with $\tilde{F} \geqslant F$ being possibly infinite dimensional, the goal of the classifier is to learn how to determine $\mathbf{w}$ and w_0 in

$$G(\mathbf{x}) = \mathbf{w}^T \psi(\mathbf{x}) + \mathrm{w}_0, \tag{8.9}$$

with $\|\mathbf{w}\| = 1$ such that it predicts $C_1(say)$ if $G(\mathbf{x}) < 0$ and C_2 if $G(\mathbf{x}) > 0$. For the separable case, the idea behind SVM is to find the hyperplane that creates the biggest margin, defined by $2M$ between the training data describing the two classes. Given N training points $(\mathbf{x}_1, \mathrm{y}_1), (\mathbf{x}_2, \mathrm{y}_2), \ldots, (\mathbf{x}_N, \mathrm{y}_N)$ with $\mathrm{y}_i \in \{-1, 1\}$ indicating the class label, then this problem can be framed as the optimisation problem

$$\max_{\mathbf{w}, \mathrm{w}_0, \|\mathbf{w}\|=1} M$$
$$\text{subject to } \mathrm{y}_i(\mathbf{w}^T \psi(\mathbf{x}_i) + \mathrm{w}_0) \geqslant M, \qquad i = 1, \ldots, N \tag{8.10}$$

which can also be rephrased as a convex optimisation problem (quadratic criterion and linear constraints). In the non-separable case, slack variables $s_1, \ldots, s_N$ are introduced to deal with points that lie on the wrong side of the margin and the linear constraint is replaced by $\mathrm{y}_i(\mathbf{w}^T \psi(\mathbf{x}_i) + \mathrm{w}_0) \geqslant M(1 - s_i)$. For further details, and its computational implementation using Lagrange multipliers, see [54][pg. 420]. This practical implementation involves the introduction of symmetric positive definite or symmetric positive semi definite kernel functions $k(\mathbf{x}, \mathbf{x}') = \psi(\mathbf{x}')^T \psi(\mathbf{x})$ [54][pg. 424], which avoids the introduction of $\psi(\mathbf{x})$ itself, but imposes limitations on their choice in order that optimisation problem remains convex. Typically kernel types include polynomial, Gaussian, and radial basis function kernels, however, the investigation is limited to the latter.

To apply SVM to multi-class problems, the problem is reduced into a series of binary classification problems. This is done by employing either an *ovo* (one versus one) or an *ovr* (one versus rest) strategy, with the former being chosen in this work, where an SVM is trained for each possible binary classification. This leads to $\frac{K(K-1)}{2}$ models being trained, for example in a 3-class problem ($K = 3$) a classifier would be trained to separate the pairs (C_1, C_2), (C_1, C_3) and (C_2, C_3), then a voting scheme based on these classifiers would be applied.

8.3.4   Probabilistic Classification

Considering problems with $K > 2$, it is beneficial for the practical probabilistic classifiers described below to write (8.6) in the form of the softmax function (also known as the normalised exponential)

$$p(C_k|\mathbf{x}) = \sigma(\mathbf{x}) := \frac{\exp a_k}{\sum_{k=1}^{K} \exp a_k}, \tag{8.11}$$

where $a_k = \ln(p(\mathbf{x}|C_k)p(C_k))$.

If $p(\mathbf{x}|C_k)$ has a simple Gaussian form, the evaluation of $p(C_k|\mathbf{x})$ for given $\mathbf{x}$ becomes explicit. However, in many practical cases, $p(\mathbf{x}|C_k)$ will have a complicated form and this will dictate the use of classifier $\gamma_k(\mathbf{x})$ that approximates $p(C_k|\mathbf{x})$ instead. Nonetheless, all probabilistic classifiers benefit from establishing (approximations of) the likelihood of each of the classes C_k, $k = 1, \ldots, K$ rather than just a single output.

Some alternative ML methods Γ, which provide probabilistic classifiers, will be explored below.

8.3.4.1   Logistic Regression

In the case that $p(\mathbf{x}|C_k)$ has a simple Gaussian form, a suitable linear classifier is logistic regression [20][Chapt. 4]. This is based on the following assumptions

1. The likelihood probability distribution is Gaussian:

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{F/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{m}_k)^T \mathbf{\Sigma}^{-1}(\mathbf{x} - \mathbf{m}_k)\right),$$

    where $\mathbf{m}_k$ is the mean of all $\mathbf{x}_i$, $(\mathbf{x}_i, \mathbf{t}_i) \in D^{(k)}$, that are associated with class C_k and $\mathbf{\Sigma}$ is a covariance matrix.

2. The covariance matrix $\mathbf{\Sigma}$ is common to all the classes.

In this case, the evaluation of $p(C_k|\mathbf{x})$ is explicit with a_k in (8.11) replaced with the rescaled $\tilde{a}_k$ for $K > 2$ [20]

$$\tilde{a}_k = \tilde{\mathbf{w}}_k^T \mathbf{x} + \tilde{w}_{k0},$$

where

$$\tilde{\mathbf{w}}_k = \mathbf{\Sigma}^{-1}\mathbf{m}_k,$$

$$\tilde{w}_{k0} = -\frac{1}{2}\mathbf{m}_k\mathbf{\Sigma}^{-1}\mathbf{m}_k + \ln p(C_k).$$

In the generative approach, the learning involves first computing $\mathbf{m}_k$ and $\boldsymbol{\Sigma}$ directly from the training data $D^{(\text{train})}$ while in discriminative approach, the $(K-1)(F+1)$ coefficients of $\tilde{\mathbf{w}}_k$ and $\mathbf{w}_{k0}$ compared to the $KF + F^2/2$ coefficients needed otherwise are found by numerical optimisation from $D^{(\text{train})2}$. When applied to other data sets, this results in an approximate $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to $p(C_k|\mathbf{x})$. Note, that only $\tilde{\mathbf{w}}_k$ and $\tilde{\mathbf{w}}_{k0}$ for $k = 1, \cdots, K-1$ need to be determined since $\sum_{k=1}^{K} p(C_k|\mathbf{x}) = 1$, which allows approximation $p(C_K|\mathbf{x}) \approx \gamma_K(\mathbf{x})$ to be found from $\gamma_k(\mathbf{x})$, $k = 1, \ldots, K-1$.

8.3.4.2 Multi-Layer Perceptron

If $p(\mathbf{x}|C_k)$ does not have a simple form, and $p(C_k|\mathbf{x})$ is not explicit, then the multi-layer perceptron (MLP) neural network can be applied in an attempt to approximate $p(C_k|\mathbf{x})$ [20][Chapt. 5]. For example, for a $K > 2$ class problem using 3-layers (with 1 input, 1 hidden and 1 output) and J internal variables (neurons) in the hidden layer, the approximation to $p(C_K|\mathbf{x})$ takes the form.

$$p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x}, \mathbf{w}) = \sigma\left(\sum_{j=1}^{J} \mathbf{w}_{kj}^{(2)} \sigma\left(\sum_{i=1}^{F} \mathbf{w}_{ji}^{(1)}(\mathbf{x})_i + \mathbf{w}_{j0}^{(1)}\right) + \mathbf{w}_{k0}^{(2)}\right) \qquad (8.12)$$

where $\mathbf{w}_{ji}^{(1)}$, $\mathbf{w}_{kj}^{(2)}$ are the $J(F+1+K) + K$ coefficients of $\mathbf{w}$ to be found from network training and $\sigma(\cdot)$ is the soft-max activation function defined in (8.11). In this case, the input layer comprises of the features $\mathbf{x} \in \mathbb{R}^F$ and output layer are the approximate of the posterior probabilities $\gamma_k(\mathbf{x}, \mathbf{w})$, $k = 1 \ldots, K$. If the number of internal variables in each hidden layer is fixed at J, and there are L hidden layers, then the total number of parameters, $\mathbf{w}_{ji}^{(1)}$, $\mathbf{w}_{ji}^{(2)}$, ..., which describe the network, that need to be found are $J^2(L-1) + J(F+L+K) + K$. Given N training points $(\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), \ldots, (\mathbf{x}_N, \mathbf{t}_N)$, and following a maximum likelihood [20][pg. 232] approach, the parameters are found by optimising a logloss error function evaluated over the training data set

$$E(\mathbf{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} (\mathbf{t}_n)_k \ln \gamma_k(\mathbf{x}_n, \mathbf{w}), \qquad (8.13)$$

or, alternatively, they can be found by a Bayesian approach [20][pg. 277].

As remarked by Richard and Lippmann [102], MLP can provide good estimates of $p(C_k|\mathbf{x})$ if sufficient training data is available, if the network is complex enough and if the classes are sampled with the correct a-prori class probabilities in the training data. Nonetheless, designing appropriate networks, with the correct number of hidden layers and neurons, can be challenging. Furthermore, a complex network with a large number of neurons can require a large amount of training data to avoid overfitting.

8.3.4.3 Gradient Boost

Gradient boost is an example of what is known as a "boosting algorithm" [37, 35]. Boosting attempts to build a stronger classifier by combining the results of weaker base clas-

---

[2]The implementation employed in the numerical examples is based on the discriminative approach.

sifiers through a weighted majority vote [54][Chapt. 10] and, in the case of gradient boost, this is achieved through optimisation using steepest descent. As described by Friedman [37], gradient boost can be applied to approximating $p(C_k|\mathbf{x})$ in probabilistic classification. By again considering (8.13), with $y_k(\mathbf{x}_n, \mathbf{w})$ replaced by (8.11) with $\mathbf{x} = \mathbf{x}_n$, and choosing the parameters $\mathbf{w} = \mathbf{w}(\mathbf{x})$ as $(\mathbf{w}(\mathbf{x}))_k = \mathrm{w}_k(\mathbf{x}) = a_k(\mathbf{x})$, then the kth component of the negative gradient of the loss function is

$$r_k = -\frac{\partial E(\mathbf{w})}{\partial w_k} = \sum_{n=1}^{N} (\mathbf{t}_n)_k - \gamma_k(\mathbf{x}_n, \mathbf{w}).$$

Starting with an initial guess $a_k^{[0]}(\mathbf{x}) = 0$, $k = 1, \ldots, K$, then, for a given $\mathbf{x}$, an iterative procedure is used to improve the estimate $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x}, \mathbf{w}^{[m]})$ of $p(C_k|\mathbf{x})$ at iteration m. In this procedure, K decision trees are trained at each iteration to predict r_k, $k = 1, \ldots, K$, and the leaf nodes of the tree are then used to update $a_k^{[m]}(\mathbf{x})$ until a convergence criteria is reached. For details of the practical implementation see [37].

8.3.5 Understanding Uncertainty in Classification

For the majority of the classifiers considered, an ML algorithm $\Gamma_\mathbb{D}$ trained on dictionary $\mathbb{D}$ produces a classifier $\Gamma_\mathbb{D}(\mathbf{x}) = \gamma_k(\mathbf{x}) \approx p(C_k|\mathbf{x})$, which provides an indication of the likelihood of the class C_k being correct. The decision, as to the correct class, is then based on the MAP estimate. When this process is repeated for different pairs $(\mathbf{x}_i, \mathbf{t}_i) \in \mathbb{T} = D^{(\text{test})}$, $\gamma_k(\mathbf{x}_i)$ may be different for each $\mathbf{x}_i$. It is useful to explore how sensitive $\gamma_k(\mathbf{x}_i)$ is to changes in $\mathbf{x}_i$ when it is evaluated for different $(\mathbf{x}_i, \mathbf{t}_i) \in \mathbb{T}_\ell = D^{(\text{test},(\ell))}$ associated with the test data for one class C_ℓ. To do this, confidence intervals for the average $E_{\mathbb{T}_\ell} \gamma_k(\mathbf{x})$ are considered.

A first approach might be to use the sample mean and sample variance

$$\overline{\gamma_k} = \frac{1}{P^{(\text{test},(\ell))}} \sum_{i=1}^{P^{(\text{test},(\ell))}} \gamma_k(\mathbf{x}_i) \qquad \text{and} \qquad S_k = \sqrt{\frac{\sum_{i=1}^{P^{(\text{test},(\ell))}} (\gamma_k(\mathbf{x}_i) - \overline{\gamma_k})^2}{P^{(\text{test},(\ell))}}},$$

to construct an interval in the form

$$\overline{\gamma_k} - CV \frac{S_k}{\sqrt{P^{(\text{test},(\ell))}}} \leqslant E_{\mathbb{T}_\ell} \gamma_k(\mathbf{x}) \leqslant \overline{\gamma_k} + CV \frac{S_k}{\sqrt{P^{(\text{test},(\ell))}}}. \tag{8.14}$$

In the above, CV is a critical value based on a t-test and the confidence level chosen. However, in practice, if $\overline{\gamma_k} \to 0.5$ as the sample size $P^{(\text{test},(\ell))} \to \infty$ and, small confidence bounds have been obtained, it might wrongly be concluded that $\gamma_k(\mathbf{x}) = 0.5$ with a high degree of confidence. Instead, this may also indicate that half the observations are predicting $\gamma_k(\mathbf{x}_i) \approx 0$ and the half are predicting $\gamma_k(\mathbf{x}_i) \approx 1$, which has the same sample mean. This can occur, since at most $S_k = 0.5$, and, for large $P^{(\text{test},(\ell))}$, the confidence bounds produced by (8.14) are narrow due to division by this quality in computation of the bounds. Hence, $\overline{\gamma_k}$ and (8.14) do not give any insight into the variation within the different observations $\gamma_k(\mathbf{x}_i)$.

Instead, ordering $\gamma_k(\mathbf{x}_i)$ as

$$O(C_k) = (\gamma_k \mathbf{x}_1), \gamma_k(\mathbf{x}_2), ..., \gamma_k(\mathbf{x}_{P(\text{test},(\ell))})), \quad \text{such that} \quad \gamma_k(\mathbf{x}_i) \leqslant \gamma_k(\mathbf{x}_{i+1}), \quad (8.15)$$

the yth-percentile is defined as

$$\gamma_{k,y} = (O(C_k))_{\frac{y}{100} P(\text{test},(\ell))}, \quad (8.16)$$

and use interpolation between neighbouring values if $\frac{y}{100}$ is not an integer. The median value of $\gamma_k(\mathbf{x})$ is then considered, given by $\gamma_{k,50}$, as the average and use the percentiles corresponding to $Q_1 \equiv \gamma_{k,25}$, $Q_3 \equiv \gamma_{k,75}$ and $\gamma_{k,5}$, $\gamma_{k,95}$ to understand uncertainty in the predictions.

8.4 Evaluating the Performance of Classifiers

Metrics for assessing performance that are applicable to both probabilistic and non-probabilistic classifiers given $D^{(\text{train})}$ and $D^{(\text{test})}$ data sets are now described.

8.4.1 Metrics

8.4.1.1 Confusion Matrices, Precision, Sensitivity and Specificity

First the definitions of true positive, false positive, true negative and false negative for a given class C_k are recalled (see e.g. [98]).

- **True positive (TP)**, the case where the classifier predicts $\mathbf{x}$ <u>belongs</u> to C_k and is <u>correct</u> in its prediction.

- **False positive (FP)** (type 1 error), the case where the classifier predicts $\mathbf{x}$ <u>belongs</u> to C_k and is <u>incorrect</u> in its prediction.

- **True negative (TN)**, the case where the classifier predicts $\mathbf{x}$ <u>does not belong</u> to C_k and is <u>correct</u> in its prediction.

- **False negative (FN)** (type 2 error), the case where the classifier predicts $\mathbf{x}$ <u>does not belong</u> to C_k and is <u>incorrect</u> in its prediction.

Following the training of a classifier, its performance can be evaluated on the test data set $D^{(\text{test})}$. Applying the classifier to each sample $(\mathbf{x}_n, \mathbf{t}_n) \in D^{(\text{test},(i))}$, where the true class label is C_i, the number of predictions of each class C_j, $j = 1, \ldots, K$ can be counted and the result recorded in the $(\mathbf{C})_{ij}$th element of a confusion matrix $\mathbf{C} \in \mathbb{R}^{K \times K}$. Repeating this process for $i = 1, \ldots, K$ leads to the complete matrix [3]. The 4 cases (TP, FP, TN,

---

[3]Note the convention used by `scikit-learn` is used for $\mathbf{C}$, other references use a different convention where the rows and columns are swapped.

FN) for each class C_k can be defined in terms of $(\mathbf{C})_{ij}$ as [98, 40]

$$\mathrm{TP}(C_k) := (\mathbf{C})_{kk}, \qquad\qquad \mathrm{FN}(C_k) := \sum_{\substack{j=1 \\ j \neq k}}^{K} (\mathbf{C})_{kj},$$

$$\mathrm{FP}(C_k) := \sum_{\substack{i=1 \\ i \neq k}}^{K} (\mathbf{C})_{ik}, \qquad\qquad \mathrm{TN}(C_k) := \sum_{\substack{i=1 \\ i \neq k}}^{K} \sum_{\substack{j=1 \\ j \neq k}}^{K} (\mathbf{C})_{ij},$$

and the precision, sensitivity and specificity for each of the classes C_k using [98]

$$\mathrm{precision}(C_k) := \frac{\mathrm{TP}(C_k)}{\mathrm{TP}(C_k) + \mathrm{FP}(C_k)} := \frac{\mathrm{TP}(C_k)}{\#\text{predicted positives for } C_k},$$

$$\mathrm{sensitivity}(C_k) := \frac{\mathrm{TP}(C_k)}{\mathrm{TP}(C_k) + \mathrm{FN}(C_k)} := \frac{\mathrm{TP}(C_k)}{\#\text{actual positives for } C_k},$$

$$\mathrm{specificity}(C_k) := \frac{\mathrm{TN}(C_k)}{\mathrm{TN}(C_k) + \mathrm{FP}(C_k)} = \frac{\mathrm{TN}(C_k)}{\#\text{predicted negatives for } C_k}.$$

The precision and sensitivity (also known as the true positive rate or recall) are measures of the proportion of positives that are correctly identified and specificity (also called the true negative rate) measures the proportion of negatives that are correctly identified.

The entries in confusion matrices are often presented as frequentist probabilities (i.e. $(\mathbf{C})_{ij}$ is normalised by $\sum_{p=1}^{K} \sum_{q=1}^{K} (\mathbf{C})_{pq}$), which, as the sample size $P^{(k)}$ becomes large, provides an approximation to $p(C_j|\mathbf{x})$ with $(\mathbf{x}, \mathbf{t}) \in D^{(\text{test},(i))}$. Confusion matrices will also be presented in this way.

8.4.1.2 κ Score

Possible choices for a metric which provides an overall score of the performance of the classifier include accuracy, the F_1 score and Cohen's κ score [113, 114, 98, 28, 105]. Vairants of the commonly used F_1 score include the macro-averaged F_1 score (or macro F_1 score), the weighted-average F_1 score (or weighted F_1 score) and the micro-averaged F_1 score (micro F_1 score). However, the F_1 score can sometimes lead to an incorrect comparison of classifiers [99, 113]. As Powers' [99] notes, the macro F_1 score is not normalised, which is overcome by the weighed F_1 score and the F_1 score is not symmetric with respect to positive and negative cases. Some of these drawbacks are taken in to account by using the micro F_1 score, however, the κ score also takes into account chance agreement [28, 114]. This is useful when comparing problems with both, differing numbers of instances per class and differing numbers of classes as it takes the chance a naive classifier has into account with $\kappa \in (-\infty, 1)$, with a naive classifier obtaining a value of 0. For these reasons, the κ score will be used and which is defined as

$$\kappa := \frac{\text{accuracy} - \text{random accuracy}}{1 - \text{random accuracy}} \tag{8.17}$$

136

for comparing classifiers where

$$\text{accuracy} := \frac{\sum_{k=1}^{K} \text{TP}(C_k)}{\sum_{k=1}^{K} \text{TP}(C_k) + \text{FN}(C_k)},$$

$$\text{random accuracy} := \sum_{k=1}^{K} \frac{(\text{TP}(C_k) + \text{FN}(C_k)) \cdot (\text{TP}(C_k) + \text{FP}(C_k))}{(\text{TP}(C_k) + \text{FP}(C_k) + \text{TN}(C_k) + \text{FN}(C_k))^2}.$$

8.4.2 Validation Methods

Evaluating the performance of different classifiers can be considerably enhanced by employing cross validation [66]. This is particularly important if $D^{(\text{test})}$ is small and, otherwise, may lead to inaccurate predictions of a classifier's performance. The Monte Carlo cross validation (MCCV) technique (also known as "Leave group out cross validation") is employed [66][pg. 71]. This involves performing ℓ iterations where, for each iteration, the dataset D is split into training and testing $D = (D^{(\text{test})}, D^{(\text{train})})$ with $D^{(\text{train})}$ and $D^{(\text{test})}$ being drawn differently from D each time, irrespective of the splittings in previous iterations. Other variants of cross validation include k-fold cross validation, repeated k-fold cross validation and bootstrapping, for further details see Kuhn and Johnson [66]. Kuhn and Johnson explain that no resampling method is uniformly better than another and that the differences between the different methodologies is small for larger samples sizes, which further motivates that actually performing cross validation is more important than the method chosen for doing so.

8.5 Chapter Summary

This chapter has documented and compared a selection of appropriate probabilistic and non-probabilistic ML classifiers which are appropriate for use with principal invariants of the MPT as features, other approaches could also be considered, but the selection does provide a good range of different methods.

The key novelties of the chapter were: Firstly, a methodology to create a dictionary of threat/ non-threat objects based on principal tensor invariants has been documented. Secondly, application of noise to simulate real world measurements has been considered. Lastly, the collation of ML classifiers that are appropriate for use in conjunction with principal tensor invariants as ML features were presented.

In Chapter 9 the ML classifiers discussed in this chapter will be applied to a series of dictionaries created using principal tensor invariants of the MPT.

Chapter 9

Machine Learning Classification Results

9.1 Introductory Remarks

This chapter documents results for the application of probabilistic and non-probabilistic classifiers that were discussed in Section 8.3 to the dictionary created using method described in Section 8.2.1.

This chapter has been adapted from the work presented by the author in [135].

The novelties of the chapter are as follows: Firstly, the construction of a large dictionary of object characterisations based on the results of Section 8.2.1 along with applying Lemmas 2.7.1 and 2.7.2 and a sampling to method to provide considerable enhancements in terms of variations of object conductivity and size at negligible computational cost. An investigation of classifier performance as a function of number of excitation frequencies and applied noise. Note that this is the first time that a computational dictionary of realistic MPT spectral signatures has been employed for object classification. Finally, a novel investigation into the classification of unseen objects is presented.

The chapter is organised as follows: First, in Section 9.2 construction and classification of British coins using logistic regression on the coin dictionary is documented. In Section 9.3.1 a methodology for the construction of a multi-class dictionary is discussed. In Sections 9.3.2 and 9.3.3, classification results for an 8 class and 15 class problems are presented, where multiple probabilistic and non-probabilistic ML classifiers are applied to the respective dictionaries. Then, in Section 9.3.4, classification results of unseen objects using the 8 class dictionary are presented. Limitations and potential improvements about the methods are then reviewed in Section 9.4. The chapter is concluded in Section 9.5 with closing remark.

9.2 Classification of British Coins

Building on the MPT spectral signature characterisations of British coins presented in Section 7.3, this section deals with their classification using logistic regression.

9.2.1 Construction of the Coin Dictionary

To create the coin dictionary, the approach described in Section 8.2.1 is followed. The kth class C_k, $k = 1, ..., K = 8$ is chosen, to correspond to the kth British coin denomination one penny (1p), two pence (2p), five pence (5p), ten pence (10p), twenty pence (20p), fifty pence (50p), one pound (£1), two pounds (£2), respectively, in order of increasing value. The coins have different geometries, and, in some cases different materials [76], although within each class the restriction of a single geometry $G^{(k)} = 1$ is chosen, then considering a fixed $P^{(k)} = P/K$ number of samples for each class so that $P^{(k)} = V^{(k)}$ in this case. Note, $P^{(k)} = 2000$ is chosen for coin classification unless otherwise stated. The shape $B^{(k)}$ of the kth coin class is as described in the specification of Table 7.1 and in the MPT-Library [132] the MPT spectral signatures for each coin geometry have previously been obtained. From these, signatures evaluated at M equally spaced frequencies were chosen, with ω_m such that 5.02×10^4 rad/s $\leqslant \omega_m \leqslant 8.67 \times 10^4$ rad/s, although the larger frequency range of 7.53×10^2 rad/s $\leqslant \omega_m \leqslant 5.99 \times 10^5$ rad/s has been considered. To account for the fact that the measured MPT coefficients will be noisy, Figure 9.1 shows realisations of noise being added to the spectral signatures of $I_1(\tilde{\mathcal{R}})$ and $I_1(\mathcal{I})$ for the 1p coin. The curves in this figure correspond to the cases of no noise and noise with SNR values of 40dB, 20dB and 10dB.
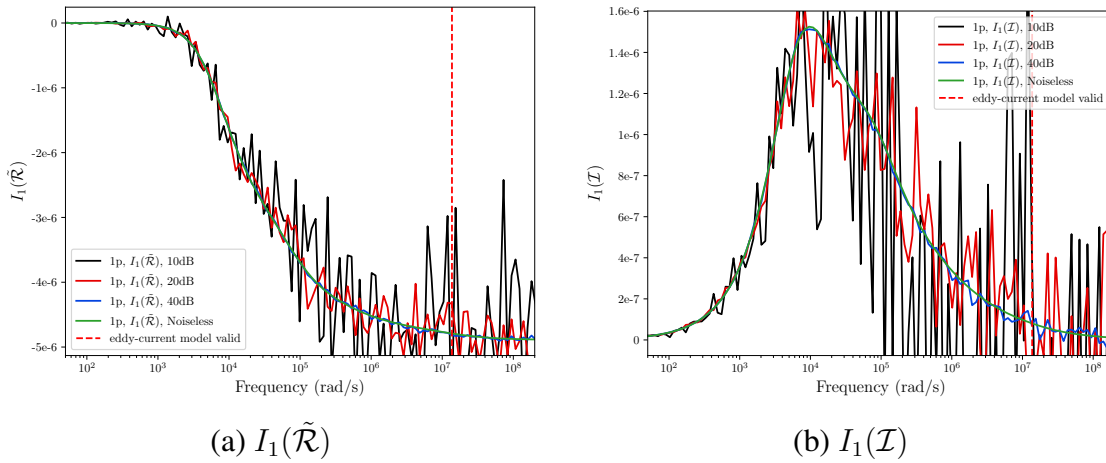
(a) $I_1(\tilde{\mathcal{R}})$ (b) $I_1(\mathcal{I})$

Figure 9.1: Set of British coins: 1p coin αB, $\alpha = 0.001$m, $\mu_r = 1$ and $\sigma = 4.03 \times 10^7$ S/m showing the spectral signatures for noiseless and realisations of noise with SNRs of 40dB, 20dB and 10dB (a) $I_1(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])$ and (b) $I_1(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])$.

The $V^{(k)}$ variations account for the fact that within each class there can be:

1. Variation in the object size α, such that the volume in the different observations of coins of a certain denomination can change. While it is expected that a coins size may be fairly uniform when they leave the mint, they are likely to become increasingly bashed and dented once they enter circulation, hence their object size may change over time. the object size is chosen to be $\pm 2.52\%$ of the coins specification,

140

by setting $m_\alpha = 0.001$m for each coin and $s_\alpha = 0.001(1.0252-1)/3 = 8.4 \times 10^{-6}$m i.e. to be $1/3$ the difference of the upper limit and the respective mean.

2. Variation in the object conductivity/conductivities $\sigma_*(x)$ to account for variations in the manufacturing process, such that the conductivity in the different observations of coins of a certain denomination can change. In most cases the coins are assumed to be homogeneous conductors, but for £1 and £2 coin denominations the objects are each an annulus. As the coins dominant composition material is copper, using [6, 58] an upper limit for conductivity is found to be $\pm 7.09\%$ of the coins specification. Thus, m_{σ_*} is set according to the conductivities of each coin denomination, so for a 1p coin, for example, $m_{\sigma_*} = 4.307 \times 10^7$S/m and s_{σ_*} in a similar way to s_α, so that $s_{\sigma_*} = 4.307 \times 10^7(1.0709-1)/3 = 9.52 \times 10^5$S/m.

3. Note that the object's permeability will be fixed as $\mu_r = 1$ as all the coins considered are assumed to be non-magnetic [76].

Given that $p\left(-3 \leqslant \frac{\alpha - m_\alpha}{s_\alpha} \leqslant 3\right) = 0.9974$, 99.74% of the object sizes generated are expected to fall within the prescribed variation in object sizes due to being bashed and dented in circulation. Similarly, 99.74% of the σ_* values generated are expected to fall within our prescribed variation in σ_*. Overall, this means that $0.9974 \cdot 0.9974 = 0.9948$ or 99.48% of the values generated are representative of genuine currency.

The effect of these samples on the MPT spectral signature is illustrated in Figure 9.2 for the 1p coin class (C_1) and noiseless data. This will now be explored further: Given
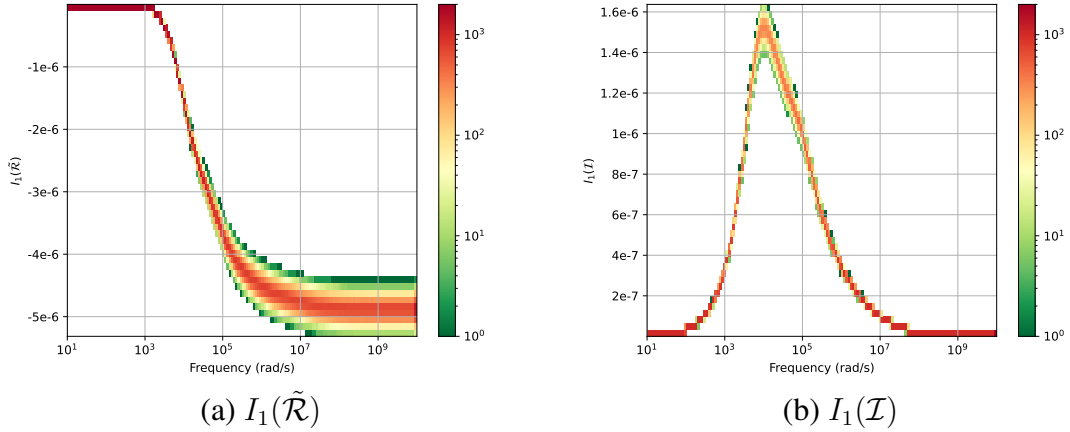
(a) $I_1(\tilde{\mathcal{R}})$ (b) $I_1(\mathcal{I})$

Figure 9.2: Set of British coins: 1p coin (class C_1) with $P^{(k)} = P/K = 2000$, with $\alpha \sim N(0.001, 8.4 \times 10^{-6})$ m and $\sigma_* \sim N(4.03 \times 10^7, 9.52 \times 10^5)$ S/m showing the histograms for the distribution of the spectral signatures (a) $I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega, \sigma_*, \mu_r])$ and (b) $I_1(\mathcal{I}[\alpha B^{(1)}, \omega, \sigma_*, \mu_r])$.

$\alpha \sim N(0.001, 8.4 \times 10^{-6})$ m, $\sigma_* \sim N(4.03 \times 10^7, 9.52 \times 10^5)$ S/m, drawing $P^{(k)}$ samples of α and σ_* and applying the scaling results in Lemmas 2.7.1 and 2.7.2, the histograms of the random variables $X = I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r) \sim p(\mathrm{x}_{1+(m-1)M}|C_1)$

(a) $I_1(\tilde{\mathcal{R}})$ (b) $I_1(\mathcal{I})$

(c) $\lambda_1(\tilde{\mathcal{R}})$ (d) $\lambda_1(\mathcal{I})$
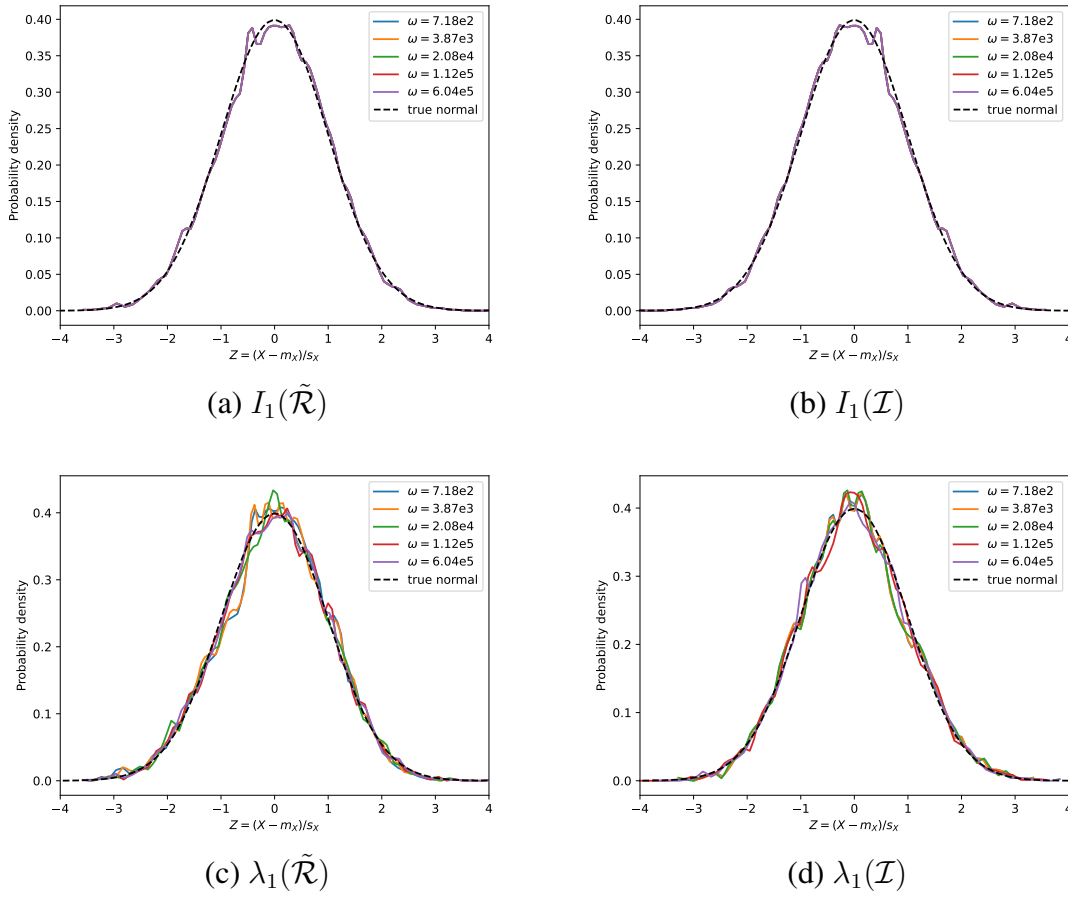
Figure 9.3: Set of British coins: 1p coin (class C_1) with $P^{(k)} = P/K = 2000$, with $\alpha \sim N(0.001, 8.4 \times 10^{-6})$ m and $\sigma_* \sim N(4.03 \times 10^7, 9.52 \times 10^5)$ S/m showing the density of the histograms of $(Z - m_X)/s_X$, presented in the form of probability densities, where X is instances of the following (a) $I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$, (b) $I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$, (c) $\lambda_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ and (d) $\lambda_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ at distinct frequencies ω_m.

and $X = I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r]) \sim p(\mathrm{x}_{1+(m+2)M}|C_1)$ are obtained and shown in Figure 9.2. Then, by taking cross section sections at selected frequencies ω_m the histograms shown in Figure 9.3, which are presented in the form of probability densities. The corresponding distributions obtained by sampling $X = \lambda_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ and $X = \lambda_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ are also included in this figure. In each case, these distributions have been normalised using the transformation $Z = (X - m_X)/s_X$, where m_X and s_X indicates the mean and standard deviation of X [1] and a curve of best fit made through the density of the histogram. As a comparison, the standard normal distribution is included. Notice that the normalised sample distributions of $X = I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ for different ω_m are identical, as are those for $X = I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ for different ω_m, and have a close fit to the standard normal distribution. The conclusion is

---

[1]The sample mean and sample standard deviation are used as an approximation to m_X and s_X, respectively.

similar for the other invariants and other coins. This outcome can be explained by the central limit theorem, which implies, given a large enough sample size, the samples of $I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ and $I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ are expected to follow a normal distribution even if the parent distribution is not normal [2]. The normalised sample distributions of $X = \lambda_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ and $X = \lambda_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ also follow a normal distribution, but the fit is not as good as for the invariants. The results are similar for other eigenvalues and other coins. For our chosen $\alpha \sim N(0.001, 8.4 \times 10^{-6})$ m, $\sigma_* \sim N(4.03 \times 10^7, 9.52 \times 10^5)$ S/m and smaller sample sizes, the distributions of $X = I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$, $X = I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$, $X = \lambda_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ and $X = \lambda_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ still approximately follow a normal distribution with the fit being superior for the invariants. By considering different instances of noise, similar histograms to those shown in Figure 9.3 can be obtained and again similar conclusions about the resulting distributions of the eigenvalues and invariants at each ω_m apply.

9.2.2 Classification Results

For the coin classification problem, consideration is only given to the logistic regression classifier using default settings of `scikit-learn`, as Figure 9.3 indicates that a normal distribution is a good approximation for the sample distributions of $I_i(\tilde{\mathcal{R}}[\alpha B^{(k)}, \omega_m, \sigma_*, \mu_r])$ and $I_i(\mathcal{I}[\alpha B^{(k)}, \omega_m, \sigma_*, \mu_r])$, $i = 1, 2, 3$, for a sufficiently large sample size. It is also observed that the feature space can be separated linearly. To illustrate this, the simplest case of just $F = 2$ features is examined, $I_1(\tilde{\mathcal{R}}[\alpha B^{(k)}, \omega_1, \sigma_*, \mu_r])$ and $I_1(\mathcal{I}[\alpha B^{(k)}, \omega_1, \sigma_*, \mu_r])$, and $M = 1$ with $\omega_1 = 6.85 \times 10^4$ rad/s. Figure 9.4 shows the class boundaries when the MAP estimate (8.8) is applied for different levels of noise, the crosses indicate the locations of the means $\mathbf{m}_k$ for each class obtained from $D^{(\text{train})}$ and the circles indicate the samples from $D^{(\text{test})}$ assuming a 3:1 training-testing $D = (D^{(\text{train})}, D^{(\text{test})})$ splitting and MCCV with $\ell = 100$ (as described in Section 8.4.2), which is employed throughout. From this figure, the class boundaries are observed to change only slightly if noise with SNR of 40dB is added and, with greater noise, the changes to the boundaries are only moderate. It is also possible to see that the number of misclassifications is very small for SNR= 40dB and 20dB and still modest for 10dB, which has a $\kappa = 0.66$ score using (8.17). Furthermore, and importantly, the locations of the means $\mathbf{m}_k$ for each class do not change significantly for $P^{(k)} = P/K = 2000$ since using this large number of instance per class has the effect of largely averaging out the effects of noise and the object variations that were previously illustrated in Figures 9.1 and 9.2. While this figure indicates that the samples form a $p(\mathbf{x}|C_k)$ that is normally

---

[2]Although $\alpha \sim N(m_\alpha, s_\alpha)$ and $\sigma_* \sim N(m_{\sigma_*}, s_{\sigma_*})$ are chosen the parent distributions $p(\mathrm{x}_{1+(m-1)M}|C_1)$ and $p(\mathrm{x}_{1+(m+2)M}|C_1)$, for $(\tilde{\mathcal{R}}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$ or $(\mathcal{I}[\alpha B, \omega, \sigma_*, \mu_r])_{ij}$, respectively, should not be expected to be normally distributed, due to the powering operation involved in the scaling in Lemma 2.7.2, which, for large s_α, can have significant effect. However, the invariant I_1 only involves summation and will not change the distribution further for independent variables. The invariants I_2 and I_3 do involve products and so are likely to further change the parent distribution, but, compared to the root finding in eigenvalues, these are much smoother operations and so their effects are expected to be smaller.

distributed, especially for noiseless and noisy data with SNR of 40dB, 20dB, which is consistent with the assumptions of this classifier, described in Section 8.3.4.1, the observation that the assumption of a common covariance matrix between the classes does not hold for coin data set is clear. The variance between the features is anisotropic for each cluster, as indicated by different sized and different orientated ellipses, which also becomes increasingly apparent for increased noise levels. While logistic regression typically has a high-bias and low-variance, its bias is expected to be lower for this problem than others given the above.

(a) Noiseless

(b) SNR =40dB

(c) SNR=20dB

(d) SNR=10dB
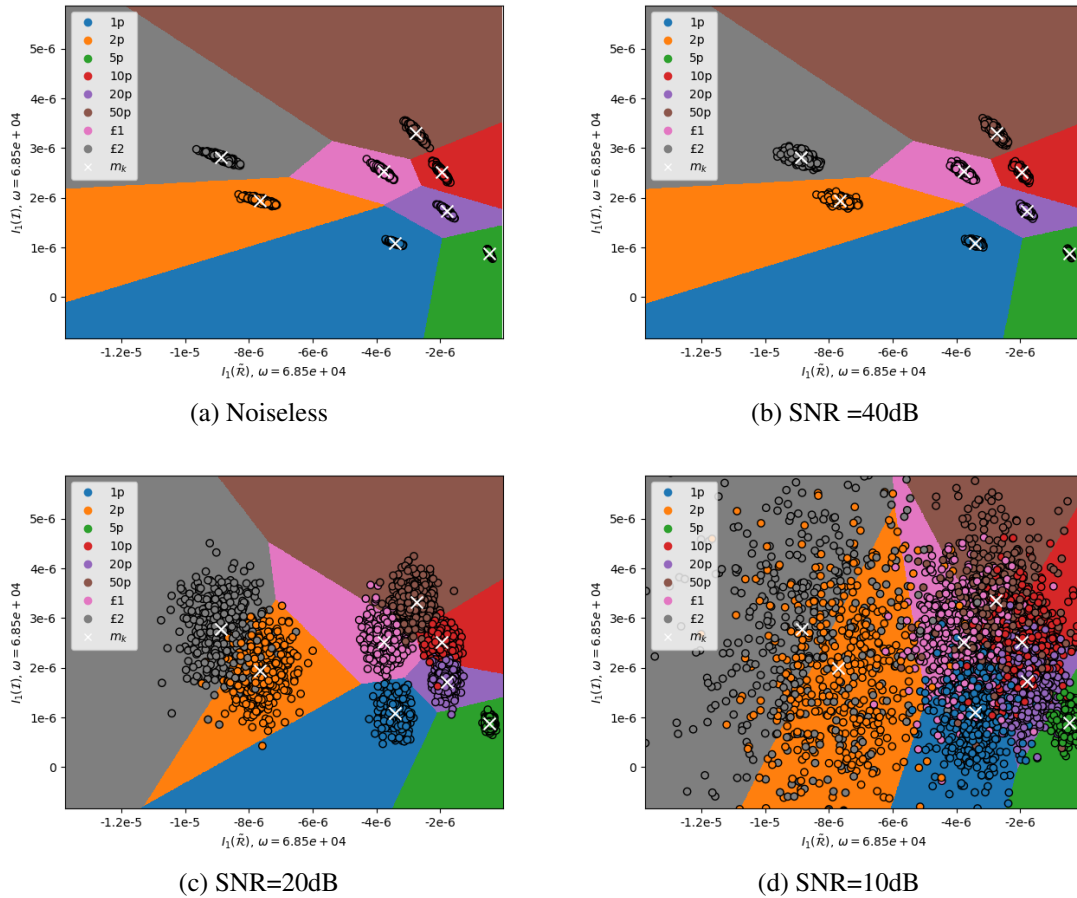
Figure 9.4: Set of British coins: linear feature space splitting for the classes C_k, $k = 1, \ldots, K$, for a simplified case of $F = 2$ features based on $\max_k \gamma_k(\mathbf{x})$ for logistic regression and $P^{(k)} = P/K = 2000$ for (a) noiseless and SNR of (b) 40dB, (c) 20dB, (d) 10dB.

This behaviour also carries over when $F = 6M$ and greater M is used. In Figure 9.5 the overall performance of the classifier is illustrated, using the κ score (8.17) as a function of $P^{(k))}$ for test data with SNR=10dB noise and (a) for noiseless training data and (b) noisy training data with SNR=10dB. The different curves correspond to $M = 1, 2, 3, 5, 10, 20$ frequencies. The curve for $M = 1$ corresponds to the same frequency considered in

Figure 9.4, but has $F = 6$ features instead of $F = 2$. Increasing M also increases F and, for either noiseless or noisy training data, the classifier's performance is improved for fixed $P^{(k)}$ as more feature information is available in $\mathbf{x} \in \mathbb{R}^F$ for each $(\mathbf{x}, \mathbf{t}) \in D^{\text{(train)}}$ and, hence, it becomes easier for the classifier to find relationship between the features and classes and, in the decision stage, partitioning according to (8.8) becomes easier for larger F. On the other hand, increasing $P^{(k)} = P/K$, for a fixed M and noiseless training data, reduces the κ score and increases the variability as the classifier becomes increasingly overfitted to the training data and experiences more misclassifications as $P^{(k)}$ is increased. For noisy training data, the classifier is exposed to more noisy data in $D^{\text{(train)}}$ as $P^{(k)}$ is increased and, hence, its performance improves and its variability decreases. The relatively high accuracy of logistic regression for the coin classification problem, even with an SNR of 10dB, can in part be attributed to how well the assumptions of logistic regression hold in practice for this problem and the normalisation of the data that is performed prior to training.

(a) Noiseless training data   (b) SNR=10dB training data
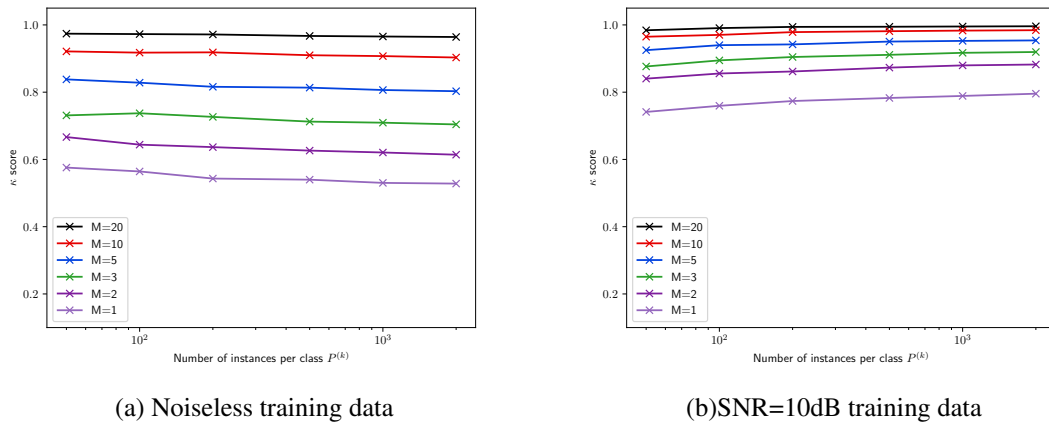
Figure 9.5: Set of British coins: Overall performance of logistic regression classifier as a function of $P^{(k)}$ and different numbers of frequencies M using the κ score (8.17) for a testing noise SNR=10dB, for (a) noiseless training data and (b) training data with SNR=10dB.

The relationship between noise level, number of frequencies and classifier performance is further considered in Figure 9.6. This figure shows the noise level against M, with the contours indicating the resulting κ score for fixed $P^{(k)}$. The concentric curves correspond to all the systems with a M and a noise level that achieve the same accuracy. As is to be expected, results for the classifier can be improved by increase both/ either M or SNR. This figure is of practical value as it allows practitioners to choose M, given an SNR, in order to achieve a desired level of accuracy.

Some illustrative approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1\ldots, K$, that are obtained for SNR=10dB are illustrated in Figure 9.7. For each
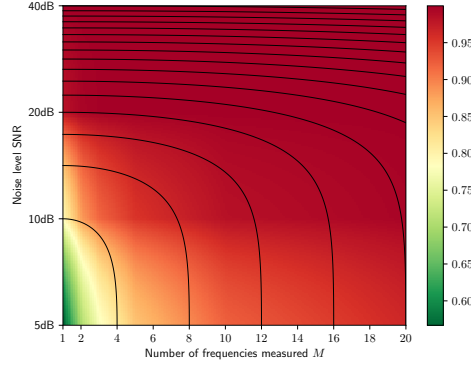
Figure 9.6: Set of British coins: Overall performance of logistic regression classifier with $P^{(k)} = 2000$, comparing M and SNR using the κ score (8.17).

$(\mathbf{x}, \mathbf{t}) \in D^{\text{(test)}}$, a potentially different distribution can be expected and, in the cases shown, $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(3))}}$ and $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(4))}}$ have been chosen so that correct classifications should be C_3 (a 5p coin) and C_4 (a 10p coin), respectively. Additionally, the bars shown are for the median value $\gamma_{k,50}$, obtained by considering all the samples $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(3))}}$ and $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(4))}}$, respectively, and indicate the Q_1, Q_3 quartiles as well as $\gamma_{k,5}$ and $\gamma_{k,95}$ percentiles, which have been obtained using (8.16). The cases shown correspond to the best and worst cases among all $D^{\text{(test,(k))}}$ for this level of noise. A common trait of logistic regression is that it gives a strong $\gamma_k(\mathbf{x})$ for one class and low values for the other classes and the results obtained also exhibit this. Comparing $\gamma_k(\mathbf{x})$, $k = 1, \ldots, K$, for $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(3))}}$ and $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(4))}}$, the most likely classes are found to correspond to the 5p and 10p coins, respectively. For the 5p coin, the inter quartile and inter percentile ranges are small and so there is high confidence in this prediction and a low variability. For the 10p coin, they are larger indicating less confidence in the prediction and a higher variability.

Next, frequentist approximations to $p(C_j|\mathbf{x})$ for $(\mathbf{x}, \mathbf{t}) \in D^{\text{(test,(i))}}$ are presented in the form of a confusion matrix with entries $(\mathbf{C})_{ij}$, $i, j = 1, \ldots, K$, for the cases of SNR=20dB and SNR=10dB in Figure 9.8, using the approach described in Section 8.4.1.1. Considering the case with SNR=10dB, the performance of the classifier is compared using $P^{(k)} = 50$ and $P^{(k)} = 2000$ instances per class. There are only a small number of misclassifications for the $P^{(k)} = 50$ case and these are further reduced by using $P^{(k)} = 2000$.

9.3 Multi-Class Problem

Building on the MPT spectral signature characterisation of threat and non-threat objects presented in Chapter 7, this section deals with the construction of dictionaries for the 8-class and 15-class problems and classification using a range of ML classifiers.

(a) 5p

(b) 10p
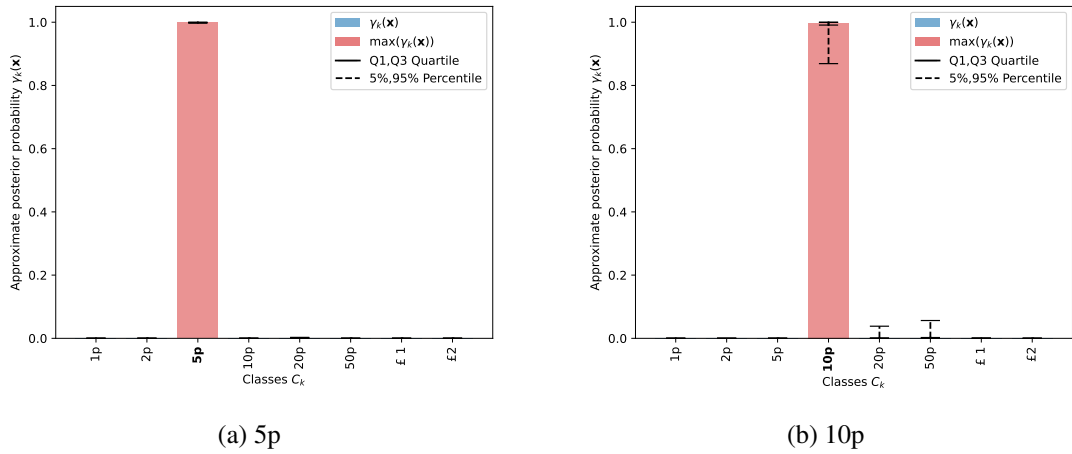
Figure 9.7: Set of British coins: Approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, using the logistic regression classifier using $P^{(k)} = 2000$ where (a) $(\mathbf{x}, \mathbf{t}) \in D^{(\text{test},(3))}$ and (b) $(\mathbf{x}, \mathbf{t}) \in D^{(\text{test},(4))}$.
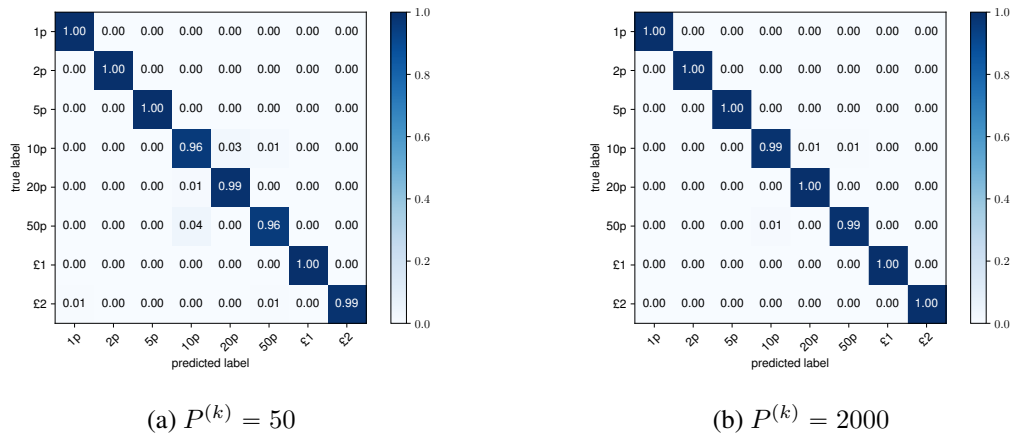
(a) $P^{(k)} = 50$

(b) $P^{(k)} = 2000$

Figure 9.8: Set of British coins: Confusion matrices for noise corresponding to SNR=10dB showing showing the effect of different numbers of instance per class (a) $P^{(k)} = 50$ and (b) $P^{(k)} = 2000$.

9.3.1 Construction of the Multi-Class Dictionary

To create the multi-class dictionary, the approach described in Section 8.2.1 is followed where, in the most general setting, the classes C_k, $k = 1, ..., K$, are chosen to correspond to the different threat and non-threat type objects listed in Table 9.1. Unlike the coins, each class is comprised of objects of different geometries, as well as different sizes and materials, so that $G^{(k)} \neq 1$ in general. However, when creating the classes, geometries that have (physical) similarities have been grouped together. For example, the coins class C_1 includes the $G^{(1)} = 8$ different denomination of British coins described in the previous section. Furthermore, in Figures 7.31 and 7.32, the surface finite element meshes are

147

shown corresponding to exemplar threat and non-threat object geometries, respectively, within each of these different classes and Table 9.2 gives an overall summary of the materials and object sizes. In the case of the coins, guns, keys and knives, the simulated MPT spectral signatures are those presented in Chapter 7, with the complete set of MPT spectral signatures for all objects in the `MPT-Library` dataset [132]. These simulated spectral signatures were generated in a similar way to those described in Chapter 7. In total, there are $\sum_{k=1}^{K} G^{(k)} = 67$ different distinct geometries and 158 including different material variations. In Table 9.1 the relationship between $P^{(k)}$, $V^{(k)}$ and $G^{(k)}$ is given for each class C_k and $V^{(k)}$ is chosen times such that an approximately equal number of samples $P^{(k)} \approx P/K$ for each class of object, with $P^{(k)} = 5000$ in the following unless otherwise stated. While $m_{\sigma*}$ is object specific, $m_\alpha = 0.001$m is set, to fix the object size, and $s_\alpha = 0.0084 m_\alpha$ and $s_{\sigma*} = 0.024 m_{\sigma*}$ have been chosen to account for manufacturing imperfections. A fixed number of $M = 28$ linearly spaced frequencies is set, such that 7.53×10^2 rad/s $\leqslant \omega_m \leqslant 5.99 \times 10^5$ rad/s, although some comments are made about the performance using 5.02×10^4 rad/s $\leqslant \omega_m \leqslant 8.67 \times 10^4$ rad/s. In a similar manner to the coin classification problem, noise corresponding to SNR values of 40dB, 20dB is added. Consideration of an SNR of 10dB is dropped in this section as this represents a very high level of 32% noise, which, of course, performs worse than 20dB noise. Using the information above, two different types of dictionaries were formed. Firstly, D_{15} for the complete set of $K = 15$ classes and, secondly, D_8 comprising of $K = 8$ different classes. The grouped classes for the D_8 are described in Table 9.3.

For these dictionaries, in the majority of cases $G^{(k)} \neq 1$. Considering $\alpha \sim N(m_\alpha, s_\alpha)$, $\sigma_* \sim N(m_{\sigma*}, s_{\sigma*})$ and $g_k = 1, \ldots, G^{(k)}$, the parent distributions of the variables $X = I_i(\tilde{\mathcal{R}}[\alpha B^{(g_k)}, \omega_m, \sigma_*, \mu_r) \sim p(x_{i+(m-1)M}|C_k)$ and $X = I_i(\mathcal{I}[\alpha B^{(g_k)}, \omega_m, \sigma_*, \mu_r]) \sim p(x_{i+(m-1)M}|C_k)$ will be far from normal. For $P^{(k)} = P/K = 5000$ samples and the class C_1, comprised of the $G^{(1)} = 8$ different denominations of UK coins, the probability density distributions are shown in Figure 9.9. Even with a sample size of $P^{(k)} = P/K = 5000$ the sample distributions are also far from normal and a very large sample is expected to be needed in order for the central limit theorem to apply in this case.

In the following, classification using the dictionary D_8 is presented and then proceed to present results for D_{15}.

9.3.2 Classification Results Using D_8

From the observations in Figure 9.9, logistic regression is not expected to perform well using the D_8 dictionary and for it to have a high bias. Instead, consideration is given to the full range of probabilistic and non-probabilistic classifiers described in Sections 8.3.3 and 8.3.4 and retain logistic regression for comparison. An optimised set of hyperparameters for each classifier were obtained as follows: A grid based search was performed to maximise the κ score (8.17) for the relevant hyperparameters for each classifier for the dictionary corresponding to $P^{(k)} = 5000$ and SNR=40dB and these were then adopted for the simulations presented in this section. To illustrate the hyperparameter grid-based

Class	# Geometries $(G^{(k)})$	#Materials per geometry	# Additional variations $(V^{(k)})$	Total $(P^{(k)})$
Guns (C_1)	1	1	$V^{(1)}$	$V^{(1)}$
Hammers (C_2)	2	3	$V^{(2)}$	$6V^{(2)}$
Knives (C_3)	5	1	$V^{(3)}$	$5V^{(3)}$
Knuckle dusters (C_4)	2	1	$V^{(4)}$	$2V^{(4)}$
Screw drivers (C_5)	6	3	$V^{(5)}$	$18V^{(5)}$
Scissors (C_6)	2	3	$V^{(6)}$	$6V^{(6)}$
Bracelets (C_7)	4	3	$V^{(7)}$	$12V^{(7)}$
Belt buckles (C_8)	3	4	$V^{(8)}$	$12V^{(8)}$
Coins (C_9)	8	1	$V^{(9)}$	$8V^{(9)}$
Earrings (C_{10})	9	3	$V^{(10)}$	$18V^{(10)}$
Keys (C_{11})	4	1	$V^{(11)}$	$4V^{(11)}$
Pendents (C_{12})	7	3	$V^{(12)}$	$21V^{(12)}$
Rings (C_{13})	7	3	$V^{(13)}$	$21V^{(13)}$
Shoe shanks (C_{14})	3	1	$V^{(14)}$	$3V^{(14)}$
Watches (C_{15})	4	3	$V^{(15)}$	$12V^{(15)}$

Table 9.1: Set of multiple threat and non-threat objects: Full list of 15 threat and non-threat object classes detailing the number of geometries in each class, $G^{(k)}$, the number of materials per geometry, the number of additional variations to account for geometrical and material variations and the total number in each class, $P^{(k)}$.

(a) $I_1(\tilde{\mathcal{R}})$ (b) $I_1(\mathcal{I})$
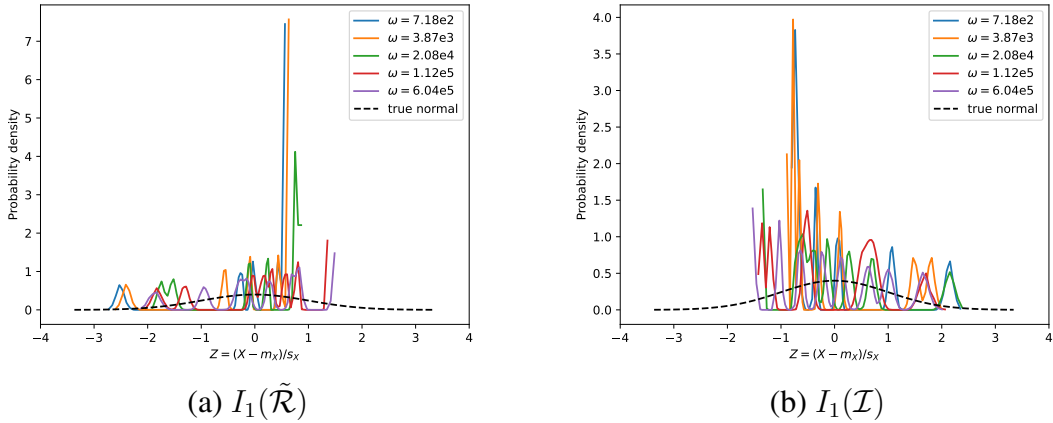
Figure 9.9: Set of multiple threat and non-threat objects: British coins (class C_1) with $P^{(k)} = P/K = 5000$, with $\alpha \sim N(0.001, 8.4 \times 10^{-6})$ m and $\sigma_* \sim N(m_{\sigma_*}, 0.024 m_{\sigma_*})$, where m_{σ_*} is determined by the material $B^{(k)}$, showing the densities for normalised histograms of $(Z - m_X)/s_X$, presented in the form of probability densities, where X is instances of the following (a) $I_1(\tilde{\mathcal{R}}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r)$ and (b) $I_1(\mathcal{I}[\alpha B^{(1)}, \omega_m, \sigma_*, \mu_r])$ at distinct frequencies ω_m.

Class	$\min(\sigma_*)$ S/m	$\max(\sigma_*)$ S/m	$\min(\mu_r)$	$\max(\mu_r)$	$\min(\alpha^3|B|)$ m^3	$\max(\alpha^3|B|)$ m^3
Guns (C_1)	6.2×10^6	6.2×10^6	5	5	3.3×10^{-5}	3.3×10^{-5}
Hammers (C_2)	1.3×10^6	1.6×10^6	1.02	5	4.3×10^{-5}	2.0×10^{-4}
Knives (C_3)	1.5×10^5	5.8×10^7	1	5	3.3×10^{-6}	6.5×10^{-5}
Knuckle dusters (C_4)	1.5×10^7	1.5×10^6	1	1	1.7×10^{-5}	1.8×10^{-5}
Screw drivers (C_5)	1.3×10^6	1.6×10^6	1.02	5	1.1×10^{-6}	3.4×10^{-6}
Scissors (C_6)	1.3×10^6	1.6×10^6	1.02	5	2.4×10^{-6}	9.5×10^{-6}
Bracelets (C_7)	9.4×10^6	6.3×10^7	1	1	5.5×10^{-7}	2.1×10^{-6}
Belt buckles (C_8)	5.6×10^5	1.5×10^7	1	5	1.5×10^{-5}	2.1×10^{-5}
Coins (C_9)	2.9×10^6	4.0×10^7	1	1	4.3×10^{-7}	1.9×10^{-6}
Earrings (C_{10})	9.4×10^6	6.3×10^7	1	1	1.0×10^{-8}	2.3×10^{-7}
Keys (C_{11})	2×10^7	1.5×10^7	1	1	6.3×10^{-7}	6.7×10^{-7}
Pendents (C_{12})	9.4×10^6	6.3×10^7	1	1	8.0×10^{-8}	1.6×10^{-6}
Rings (C_{13})	9.4×10^6	6.3×10^7	1	1	7.0×10^{-8}	9.2×10^{-7}
Shoe shanks (C_{14})	6.2×10^6	6.2×10^6	5	5	8.9×10^{-7}	1.4×10^{-6}
Watches (C_{15})	9.4×10^6	6.3×10^7	1	1	4.7×10^{-6}	3.3×10^{-5}

Table 9.2: Set of multiple threat and non-threat objects: Full list of 15 threat and non-threat object classes detailing composition of different materials and different object sizes in each class.

optimisation, the κ score is investigated for different choices of L and J for MLP in Figure 9.10. For this result, it has been assumed the same number of neurons in each layer and used `max_iter=300` rather than `max_iter=200` to allow an increased number of iterations to be performed to ensure convergence. From this figure, it can be observed there are a range of different L and J that lead to a network with a similar level of accuracy. As remarked in Section 8.3.4.2, for the type of network being considered, the number of variables grow quadratically with J and linearly with L. Hence, from a computational cost perspective, choosing a network with a small J and a large L is generally preferable to a network with a large J and small L, if the cost of computing each variable is assumed the same. For this reason, a network with $L = 3$ and $J = 50$ is adopted as MLP architectures in this range result in high κ score, while minimising computational cost. Also, if desired, J could be further reduced without compromising accuracy. For SVM, rather than the default *ovr* strategy, `decision_function_shape='ovo'` is employed, this is due to the performance of kernel based methods not scaling in proportion with the size of the training dataset. The grid-based optimisation led to a significant variation in performance for SVM, with the optimum values being a regularisation parameter C =

Class	Composition	Total $P^{(k)}$
Tools (C_1)	Hammers	$6V^{(2)} + 6V^{(6)} + 18V^{(5)}$
	Scissors	
	Screwdrivers	
Weapons (C_2)	Guns	$V^{(1)} + 2V^{(4)} + 5V^{(3)}$
	Knuckle dusters	
	Knives	
Clothing (C_3)	Belt buckles	$12V^{(8)} + 3V^{(14)}$
	Shoe shanks	
Earrings (C_4)	Earrings	$18V^{(10)}$
Pendants (C_5)	Pendants	$21V^{(12)}$
Pocket items (C_6)	Coins	$8V^{(9)} + 4V^{(11)}$
	Keys	
Rings (C_7)	Rings	$21V^{(13)}$
Wrist items (C_8)	Bracelets	$21V^{(7)} + 12V^{(15)}$
	Watches	

Table 9.3: Set of multiple threat and non-threat objects: Amalgamated list of $K = 8$ threat and non-threat object classes detailing their composition and total number in each class $P^{(k)}$.
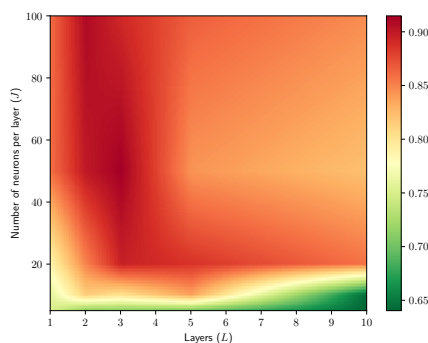
Figure 9.10: Set of multiple threat and non-threat objects: Overall performance of MLP for different uniform network architectures, with $P^{(k)} = 5000$ when $K = 8$ and SNR=40dB, showing κ score for different numbers of hidden layers L and numbers of neurons per layer J.

`10e6` and a Kernel coefficient `gamma = 1`. For the random forest classifier, the grid based optimisation resulted in the choice of `max_depth = 100` and `n_estimators = 100`. For gradient boost, the grid-based optimisation led to choice of `n_estimators = 50` and `max_dpeth = 2` and later, in the case of an unseen object, it is shown that the effects of varying the number of trees within the ensemble and the maximum depth

of each tree. Finally, for the decision tree classifier, the grid-based optimisation led to the choice `max_depth = 100`. Of course, the hyperparameter choices for each classifier have been optimised for $P^{(k)} = 5000$, SNR = 40dB and this $K = 8$ class problem, for different $P^{(k)}$ and SNR levels, as well as other classification problems, this choice may no longer be optimum. More sophisticated alternatives to our simple grid-based optimisation include using a Bayesian optimisation [117], a hyperband optimisation [36] or simulated annealing [139].

For MLP, rather than the default settings of `hidden_layer_sizes=(100)`, which means a network with $L = 1$ hidden layer and $J = 100$ neurons, $L = 3$ hidden layers are used with $J = 50$ neurons in each layer, a choice which will be justified shortly. In addition, `max_iter=300` is set rather than `max_iter=200` to allow an increased number of iterations to be performed to ensure convergence. For gradient boost `n_estimators=100` is chosen with `max_depth=3` and later the effects of varying the number of trees within the ensemble and the maximum depth of each tree is shown.

In Figure 9.11, the overall performance of the classifiers is shown with different levels of noise. The κ score (8.17) is used to assess the performance of the classification due to the variations within the classes. In each case, it is observed that increasing $P^{(k)} \approx P/K$ generally leads to an improved performance of the classification in all cases, since the classifier is exposed to more noisy data in $D_8^{(\text{train})}$ and its variability decreases. The figure shows that, in both noise cases, the best performing classifier is random forests, although, for large $P^{(k)}$, the performance of random forest, gradient boost, decision trees and SVM (particularly for SNR = 40dB) are all very similar with $\kappa \approx 1$ indicating a low bias and low variance. As random forest is a bagging algorithm and gradient boost is a boosting algorithm the expectation is for them to perform well. However, the good performance of decision trees is surprising. While the performance of SVM is good, it is less robust as (small) changes in hyperparameters can have a significant affect on its performance. The second best probabilistic classifier is MLP, which shows a significant benefit for large $P^{(k)}$. Comparing SNR=40dB and SNR=20dB a slight reduction in accuracy is observed for a given $P^{(k)}$ using SNR=20dB, although, by increasing $P^{(k)}$, the effects of noise can be overcome. In particular, SVM sufferes noticeably more with SNR = 20dB compared to random forest, gradient boost and decision trees, but its performance for large $P^{(k)}$ is still good and may be improved further by additional hyperparameter optimisation. Also, although not included, the corresponding results for 5.02×10^4 rad/s $\leqslant \omega_m \leqslant 8.67 \times 10^4$ rad/s using $M = 20$ are not as good as those for 7.53×10^2 rad/s $\leqslant \omega_m \leqslant 5.99 \times 10^5$ rad/s using $M = 28$, with those shown offering at least a 5% improvement for the best performing classifiers, small $P^{(k)}$ and SNR=20dB. Interestingly, logistic regression improves by 25% when the larger frequency range is used. Attention is given to the two best performing probabilistic classifiers, gradient boost and MLP, in the following.

The approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, obtained for gradient boost and MLP are shown in Figure 9.12, with $(\mathbf{x}, \mathbf{t}) \in D_8^{(\text{test},(6))}$ chosen so that the correct classification should be C_6 (ie a pocket item: a coin or a key). Additionally, the bars for the median value $\gamma_{k,50}$ of $\gamma_k(\mathbf{x})$, obtained by considering all the samples $(\mathbf{x}, \mathbf{t}) \in D_8^{(\text{test},(6))}$, are shown together with the Q_1, Q_3 quartiles as well as
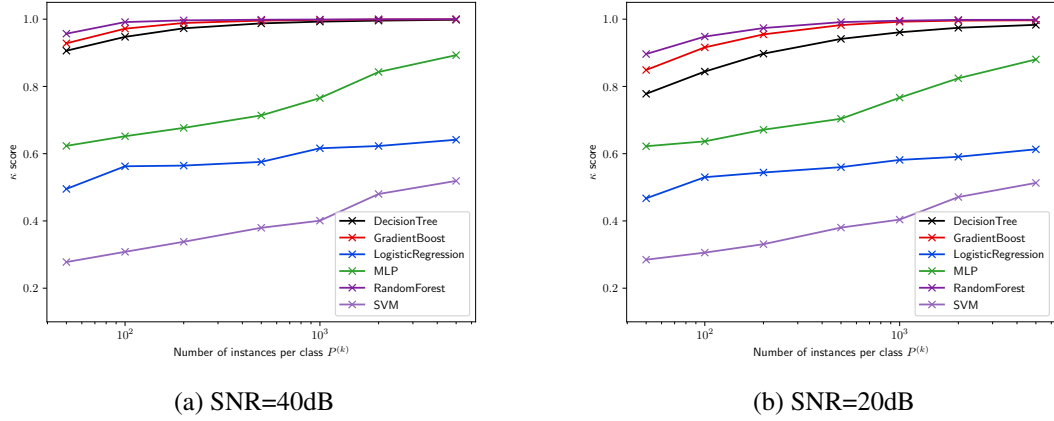
(a) SNR=40dB	(b) SNR=20dB

Figure 9.11: Set of multiple threat and non-threat objects: Overall performance of different classifiers as a function of $P^{(k)}$ when $K = 8$ using the κ score (8.17) showing (a) SNR=40dB and (b) SNR=20dB.

$\gamma_{k,5}$ and $\gamma_{k,95}$, for different SNR, which have been obtained using (8.16). The results for SNR=40dB strongly indicate that the most likely class is a pocket item for both classifiers, since $\gamma_{6,50} \approx 1$. For the gradient boost classifier, the inter quartile and inter percentile ranges are small and, so, there is high confidence in this prediction. However, there is less confidence in the corresponding prediction for the MLP as both the inter quartile and inter percentile ranges are larger. For SNR=20dB, the median value $\gamma_{6,50}$ falls for both classifiers and, so, there is much greater uncertainty in the classification over the samples, as illustrated by the larger inter percentile ranges for the different object classes. Comparing the two classifiers, there is less confidence in the prediction with MLP than for gradient boost.

Next, the frequentist approximations to $p(C_j|\mathbf{x})$ for $(\mathbf{x}, \mathbf{t}) \in D^{(\text{test},(i))}$ presented in the form of a confusion matrix with entries $(\mathbf{C})_{ij}$, $i, j = 1, \ldots, K$, for the cases of SNR=40dB and SNR=20dB and the gradient boost and MLP classifiers in Figure 9.13. As expected, for SNR=20dB, an increased misclassification is observed amongst the classes compared to SNR=40dB with situation being worse for the MLP classifier compared to the gradient boost. Looking at the row corresponding to the true label for the C_6 (pocket items) class, for both SNR=40dB and SNR=20dB, it can be observed that the frequentist probability in column j is approximately similar to the median approximate posterior probability $\gamma_j(\mathbf{x})$ shown in Figure 9.12. Also, while the gradient boost exhibits near perfect classification for SNR=40dB (and SNR=20dB), MLP does not perform as well, particularly among the earrings and pendents.

In Table 9.4, the precision, sensitivity and specificity are shown for each of the different object classes C_k, $k = 1, \ldots, K$, for the case of SNR=20dB and the MLP classifier. In general, the proportion of negatives that are correctly identified is very high (as indicated by the specificity) and is close to 1 in all cases, whereas the proportions of positives

153
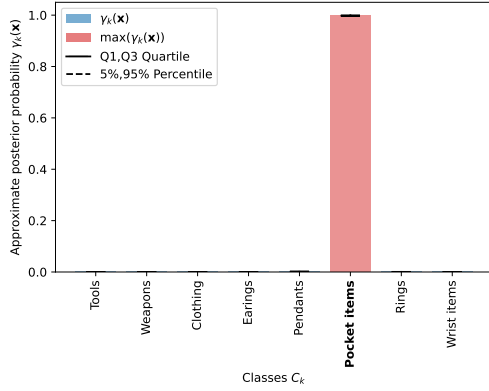
(a) Gradient boost SNR=40dB
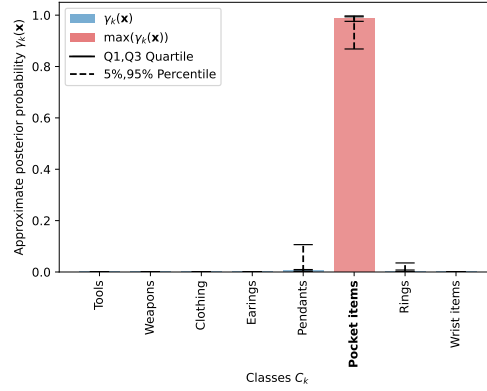
(b) Gradient boost SNR=20dB
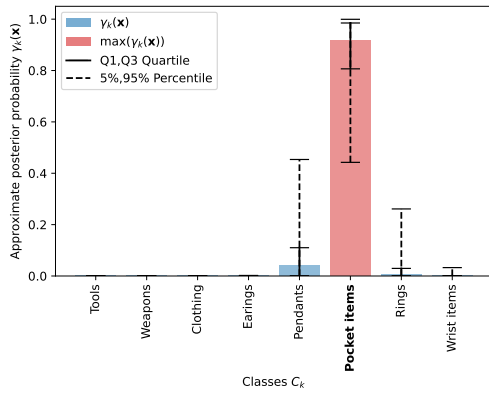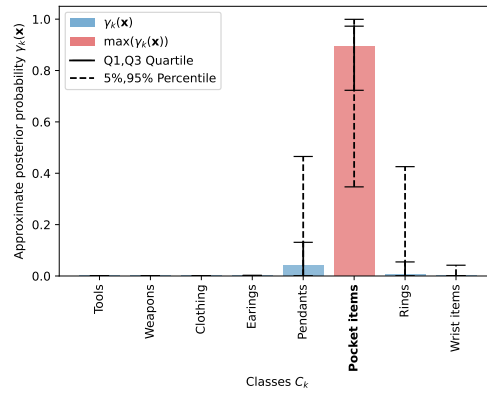
(c) MLP SNR=40dB

(d) MLP SNR=20dB

Figure 9.12: Set of multiple threat and non-threat objects: Approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, where $(\mathbf{x}, \mathbf{t}) \in D_8^{(\text{test},(2))}$ for $P^{(k)} = 5000$ when $K = 8$ showing the classifiers (a) gradient boost SNR=40dB, (b) gradient boost SNR=20dB, (c) MLP SNR=40dB and (d) MLP SNR=20dB.

correctly identified (indicated by the precision and sensitivity) varies amongst the different object classes, the best case being C_1 (weapons) and worst case C_5 (pendents). The corresponding results for gradient boost are all close to 1.

To justify the choice of $L = 3$ and $J = 50$ for MLP, the κ score is shown for different choices of L and J in Figure 9.10 for the case where SNR=40dB. For this result, it is assumed that there are the same number of neurons in each layer. From this figure, it is observed there are a range of different L and J that lead to a network with a similar level of accuracy. As remarked in Section 8.3.4.2, for the type of network being considered, the number of variables grows quadratically with J and linearly with L. Hence, from a computational cost perspective, choosing a network with a small J and a large L is generally preferable to a network with a large J and a small L, if the cost of computing each variable is assumed the same. For this, reason, a network with $L = 3$ and $J = 50$ is adopted as MLP architectures in this range result in high κ score, while minimising com-

(a) Gradient boost SNR=40dB

(b) Gradient boost SNR=20dB

(c) MLP SNR=40dB
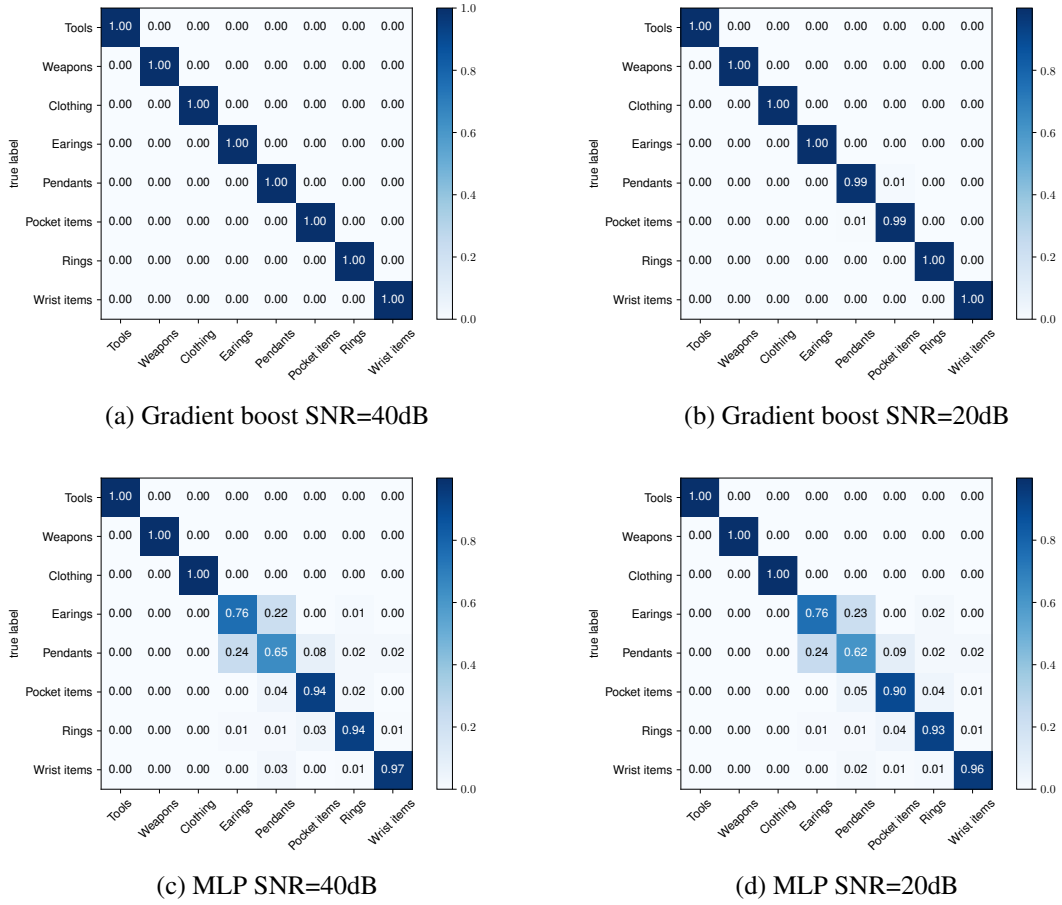
(d) MLP SNR=20dB

Figure 9.13: Set of multiple threat and non-threat objects: Comparison of confusion matrices for $P^{(k)} = 5000$ when $K = 8$ showing the classifiers (a) gradient boost SNR=40dB, (b) gradient boost SNR=20dB, (c) MLP SNR=40dB and (d) MLP SNR=20dB.

C_k	Precision	Sensitivity	Specificity
Tools (C_1)	0.97	0.98	1.00
Weapons (C_2)	0.99	0.99	1.00
Clothing (C_3)	0.98	0.98	1.00
Earrings (C_4)	0.64	0.84	0.93
Pendants (C_5)	0.60	0.34	0.97
Pocket items (C_6)	0.75	0.73	0.96
Rings (C_7)	0.70	0.82	0.95
Wrist items (C_8)	0.90	0.87	0.99

Table 9.4: Set of multiple threat and non-threat objects: Precision, sensitivity and specificity measures (to 2d.p.) for each of the classes C_k when SNR=20dB and $P^{(k)} = P/K = 5000$ for the MLP classifier.

putational cost. Also, if desired, J could be further reduced without comprising accuracy. Of course, this choice has been optimised for $P^{(k)} = 5000$, SNR=40dB and this $K = 8$ class problem, for different $P^{(k)}$ and SNR levels, as well as other classification problems, this choice may no longer be optimum.

9.3.3 Classification Results Using D_{15}

Figure 9.14 repeats the investigation shown in Figure 9.11 for D_{15}, instead of D_8, and, given the relationship between the multi-class dictionaries, using the same classifier hyperparameters. The trends described previously again apply, except, with a further significant gain in the performance for all classifiers for the increased fidelity $K = 15$ class problem compared to the previous $K = 8$ class problem. This is because each class, for $K = 15$, is comprised of objects that have increased similarity between their volumes, shapes and materials, and, hence, their MPT spectral signatures, compared to the $K = 8$ problem. This, in turn, reduces each classifier's bias as it becomes easier to establish the relationship between the features and class. Nonetheless, $X = I_i(\tilde{\mathcal{R}}[\alpha B^{(g_k)}, \omega_m, \sigma_*, \mu_r]) \sim p(\mathbf{x}_{i+(m-1)M}|C_k)$ and $X = I_i(\mathcal{I}[\alpha B^{(g_k)}, \omega_m, \sigma_*, \mu_r]) \sim p(\mathbf{x}_{i+(m-1)M}|C_k)$ are still far from normal and, so, logistic regression does not perform well. The best performance being again given by random forests, gradient boost, decision trees and SVM (particularly for SNR $= 40$dB). Again, attention is focused on the gradient boost and MLP, which are the best two performing probabilistic classifiers in the following.
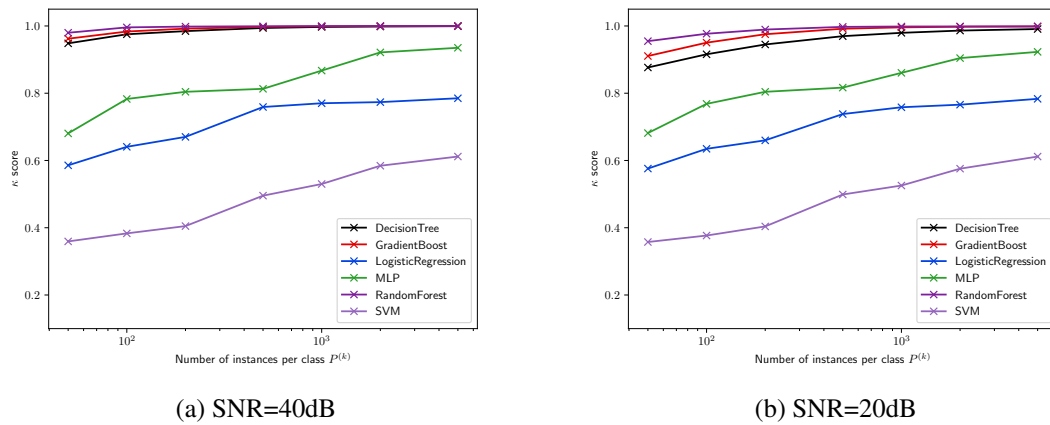
(a) SNR=40dB   (b) SNR=20dB

Figure 9.14: Set of multiple threat and non-threat objects: Overall performance of different classifiers as a function of $P^{(k)}$ when $K = 15$ using the κ score (8.17) showing (a) SNR=40dB and (b) SNR=20dB.

The approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, obtained for gradient boost and MLP are shown in Figure 9.15, with $(\mathbf{x}, \mathbf{t}) \in D_{15}^{(\text{test},(9))}$ chosen so that the correct classification should be C_9. The bars are for $\gamma_{k,50}$, obtained

by considering all the samples $(\mathbf{x}, \mathbf{t}) \in D_{15}^{(\text{test},(9))}$, with the Q_1, Q_3 quartiles indicated as well as $\gamma_{k,5}$ and $\gamma_{k,95}$, for different SNR, which have been obtained using (8.16). The results for SNR=40dB strongly indicate that the most likely class is a pocket item for both classifiers, since $\gamma_{9,50} \approx 1$. For the gradient boost classifier, the inter quartile and inter percentile ranges are very small and so there is very high confidence in this prediction; the MLP classifier has larger ranges and, therefore, less confidence. For SNR=20dB, $\gamma_{9,50}$ falls slightly for gradient boost and by a larger amount for MLP. The gradient boost still shows a high degree of confidence in the prediction, but the MLP is more uncertain. Compared to the results shown in Figure 9.12 for D_8, the performance in Figure 9.15 for D_{15} is improved for MLP and remains excellent for gradient boost (when considering the amalgamated pocket item class and the split coin and keys classes).

(a) Gradient boost SNR=40dB   (b) Gradient boost SNR=20dB

(c) MLP SNR=40dB   (d) MLP SNR=20dB
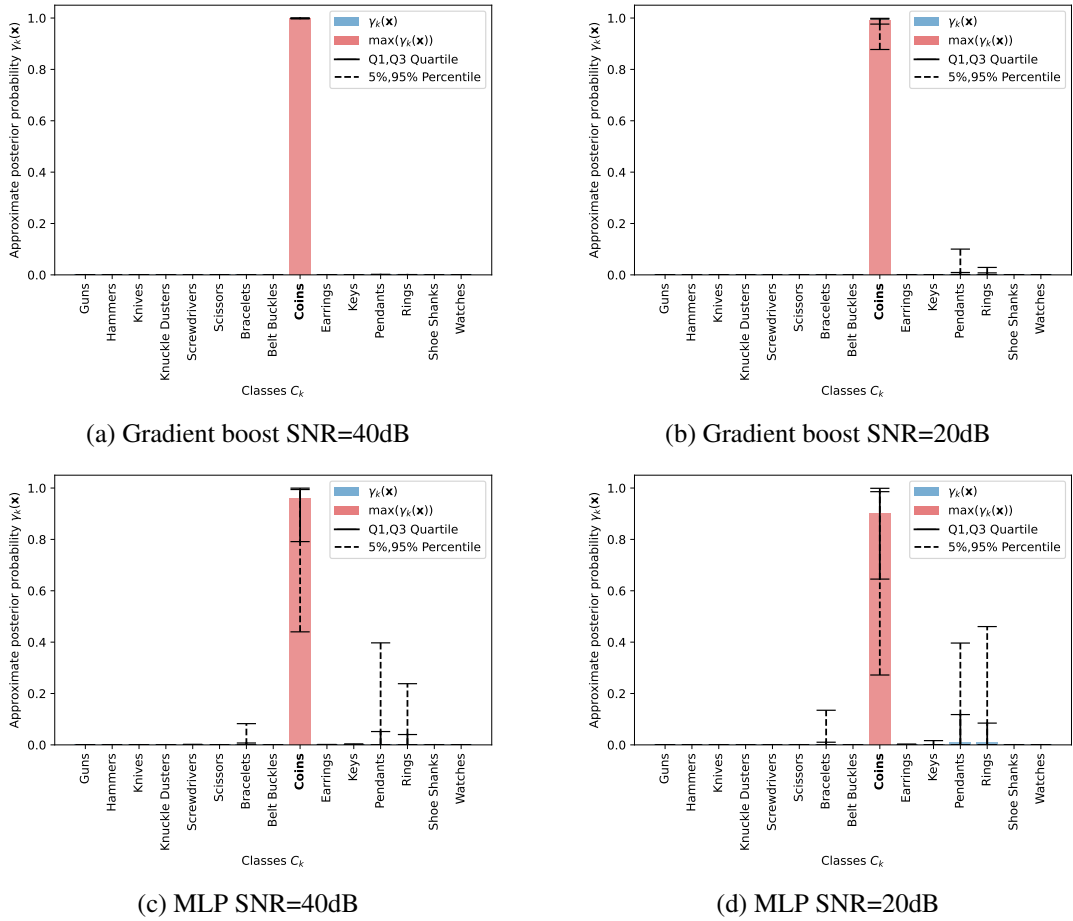
Figure 9.15: Set of multiple threat and non-threat objects: Approximations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$ to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, where $(\mathbf{x}, \mathbf{t}) \in D_{15}^{(\text{test},(9))}$ (Coins) for $P^{(k)} = 5000$ when $K = 15$ showing (a) gradient boost SNR=40dB, (b) gradient boost SNR=20dB, (c) MLP SNR=40dB and (d) MLP SNR=20dB.

Next, the frequentist approximations to $p(C_j|\mathbf{x})$ for $(\mathbf{x}, \mathbf{t}) \in D^{(\text{test},(i))}$ are presented in the form of a confusion matrix with entries $(\mathbf{C})_{ij}$, $i, j = 1, \ldots, K$, for the cases of SNR=40dB

and SNR=20db and the MLP classifier, in Figure 9.16. The results for the gradient boost are not shown as it has a near perfect identity confusion matrix on this scale for these noise levels. Compared to the corresponding results shown in Figure 9.13 for D_8, the results for D_{15} show the ability of the classifier to better discriminate between different objects. However, MLP still shows significant misclassifications for pendents whereas gradient boost does not.

(a) MLP SNR=40dB   (b) MLP SNR=20dB
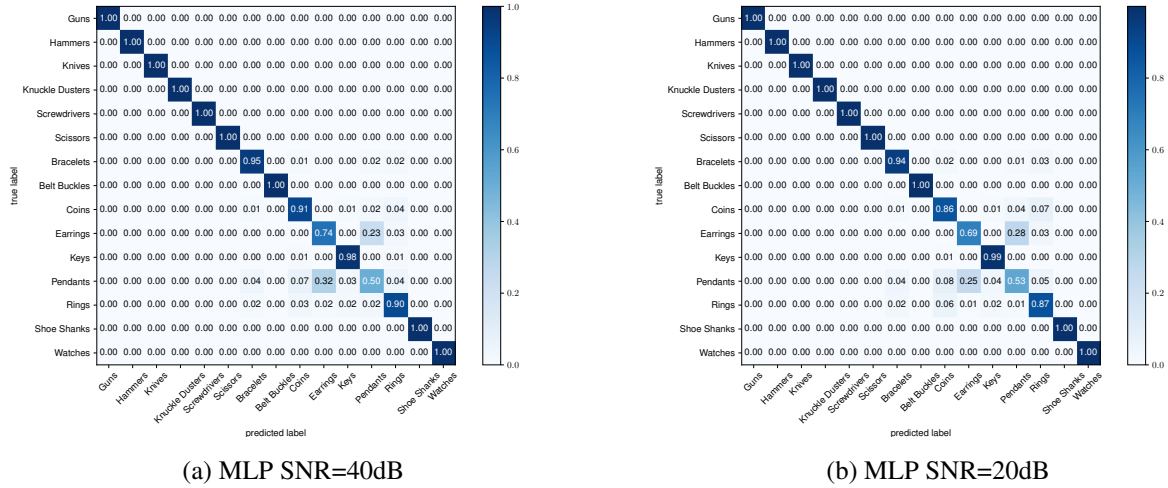
Figure 9.16: Set of multiple threat and non-threat objects: Comparison of confusion matrices for $P^{(k)} = 5000$ when $K = 15$ showing (a) MLP SNR=40dB and (b) MLP SNR=20dB.

In Table 9.5, the precision, sensitivity and specificity are shown for each of the different object classes C_k, $k = 1, \ldots, K$, for the case of SNR=20dB and the MLP classifier. In general, the proportion of negatives that are correctly identified (as indicated by the specificity) is very high and is close to 1 in all cases. The proportions of positives correctly identified (indicated by the precision and sensitivity) varies amongst the different object classes, but is generally much closer to 1 than shown in Table 9.4 for D_8. The classes C_1, C_3 and C_4 (guns, knives and knuckle dusters in $D^{(15)}$), which make up the amalgamated weapons class C_2 in D_8, all perform very well, but the worst case still remains C_{12} (the pendents). The corresponding results for precision, sensitivity and specificity for the gradient boost classifier are all close to 1.

9.3.4 Classification of Unseen Objects using D_8

When testing the performance of classifiers in the previous sections, the construction of the dictionary, described in Section 9.3.1, means that $D^{(\text{train})}$ and $D^{(\text{test})}$ are both comprised of samples that have MPT spectral signatures associated with objects that share the same geometry and have similar object sizes and material parameters. To illustrate the ability of a classifier to recognise an unseen threat object, $D_8^{(\text{train})}$ is constructed, as in Section 9.3.1,

C_k	Precision	Sensitivity	Specificity
Guns (C_1)	1.00	1.00	1.00
Hammers (C_2)	1.00	1.00	1.00
Knives (C_3)	1.00	1.00	1.00
Knuckle dusters (C_4)	1.00	1.00	1.00
Screwdrivers (C_5)	0.95	0.96	1.00
Scissors (C_6)	1.00	1.00	1.00
Bracelets (C_7)	0.83	0.85	0.99
Belt buckles (C_8)	0.99	0.98	1.00
Coins (C_9)	0.70	0.72	0.98
Earrings (C_{10})	0.75	0.75	0.98
Keys (C_{11})	0.90	0.95	0.99
Pendants (C_{12})	0.67	0.53	0.98
Rings (C_{13})	0.71	0.77	0.98
Shoe shanks (C_{14})	0.99	1.00	1.00
Watches (C_{15})	0.99	0.99	1.00

Table 9.5: Set of multiple threat and non-threat objects: Precision, sensitivity and specificity measures (to 2d.p.) for each of the classes C_k when SNR=20db and $P^{(k)} = P/K = 5000$ for the MLP classifier.

except, for one class C_k, where $D_8^{(\text{train},(k))}$ is replaced with data that is obtained from $G^{(k)} - 1$ (instead of $G^{(k)}$) geometries and $V^{(k)}$ samples. Also, $P^{(k)} = 2000$, instead of $P^{(k)} = 5000$ is chosen, due to the higher computational cost of the investigation presented in the following. The classifier is tested using a sample that is constructed only from $V^{(k)}$ samples of the unseen $G^{(k)}$th geometry.

A geometry is removed from the C_2 class of weapons, which originally has $G^{(2)} = 8$ geometries, and the unseen geometry is varied to be one of the chef, cutlet, meat cleaver, Santoku, and Wusthof knives, shown in Figure 7.31, where the naming convention from Section 6.4 of [76] is adopted. The gradient boost classifier is applied to this problem, as it was seen to perform best for both the $K = 8$ and $K = 15$ class problems. Previously, the optimised hyperparameters `n_estimators=50` and `max_depth=2` have been shown to lead to accurate results. However, this problem is more challenging, as it involves attempting to classify data from the samples in $(\mathbf{x}, \mathbf{t}) \in D_8^{(\text{test},(2))}$ that are only constructed from samples of the unseen $G^{(2)}$th geometry, and, therefore, the previous hyperparameters are no longer optimal. This is illustrated in Figure 9.17 for the case where SNR=40dB and, here, the average κ score obtained from considering the situations when instances of the chef, cutlet, meat cleaver, Santoku, and Wusthof knife geometries as being unseen is presented. This suggests the optimal performance will be for a very limited region where `n_estimators` ≈ 30 and `max_depth=1` and, away from this, the performance of the classifier will be poor.

The poor performance of the gradient boost classifier for this problem for a large range

Figure 9.17: Set of multiple threat and non-threat objects: Overall performance of the gradient boost classifier for different values in the hyperparameter space, with $P^{(k)} = 2000$ when $K = 8$ and SNR=40dB, showing average κ score for different values of numbers of trees (`n_estimators`) and tree depth (`max_depth`).

of hyperparameters is due to its inability to correctly classify the cutlet knife geometry, with the classifier instead predicting this as a tool rather than a weapon in the majority of cases (indicating bias against this geometry) and, additionally, for other geometries, the relatively high degree of uncertainty that is associated with $\gamma_2(\mathbf{x})$ despite $\gamma_{2,50}$ being high (indicating a high variance). This can further be explained by the comparison of the knife volumes using a fixed $\alpha = 0.001$m shown in Table 9.6, where it can be seen that the cutlet knife geometry has a volume that is an order of magnitude smaller than that of the knives. The MPT spectral signature depends on the object's volume as well as its materials and geometry and, as each cutlet knife tends to be associated smaller volumes to those considered in $D_8^{(\text{train})}$ this has contributed to the classifier not being able to recognise it.

Knife	Volume (m^3)
Chef	1.46×10^{-5}
Cutlet	3.28×10^{-6}
Meat cleaver	6.50×10^{-5}
Santoku	2.51×10^{-5}
Wusthof	3.48×10^{-5}

Table 9.6: Set of multiple threat and non-threat objects: Comparison of volumes for different knife models.

The situation can be improved by increasing the standard deviations s_α and s_{σ_*}, so that $D_8^{(\text{train},(2))}$ includes MPT spectral signatures that are closer to that of the omitted $G^{(2)}$th geometry. In Table 9.7, three alternatives A, B and C are considered to the previous control choice. Then, in Figure 9.18, the investigation shown in Figure 9.17 for cases A, B and C is repeated. In this figure the classifier is observed to have less variability

160

and performs increasingly well over a wide range of hyperparameters, as s_α and $s_{\sigma*}$ are increased. In the limiting case of C, the overall performance of the classifier is uniform with $\kappa = 0.75$ over the complete space of hyperparameters considered.

scaling regime	s_α	$s_{\sigma*}$
Control	$0.0084m_\alpha$	$0.0236333m_{\sigma*}$
A	$0.02m_\alpha$	$0.05m_{\sigma*}$
B	$0.05m_\alpha$	$0.1m_{\sigma*}$
C	$0.1m_\alpha$	$0.2m_{\sigma*}$

Table 9.7: Set of multiple threat and non-threat objects: List of the parameters for the sampling distributions considered.

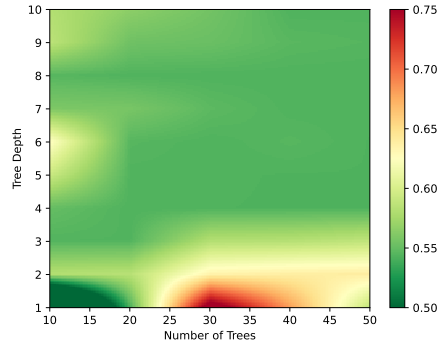(a) Control

(b) A

(c) B

(d) C
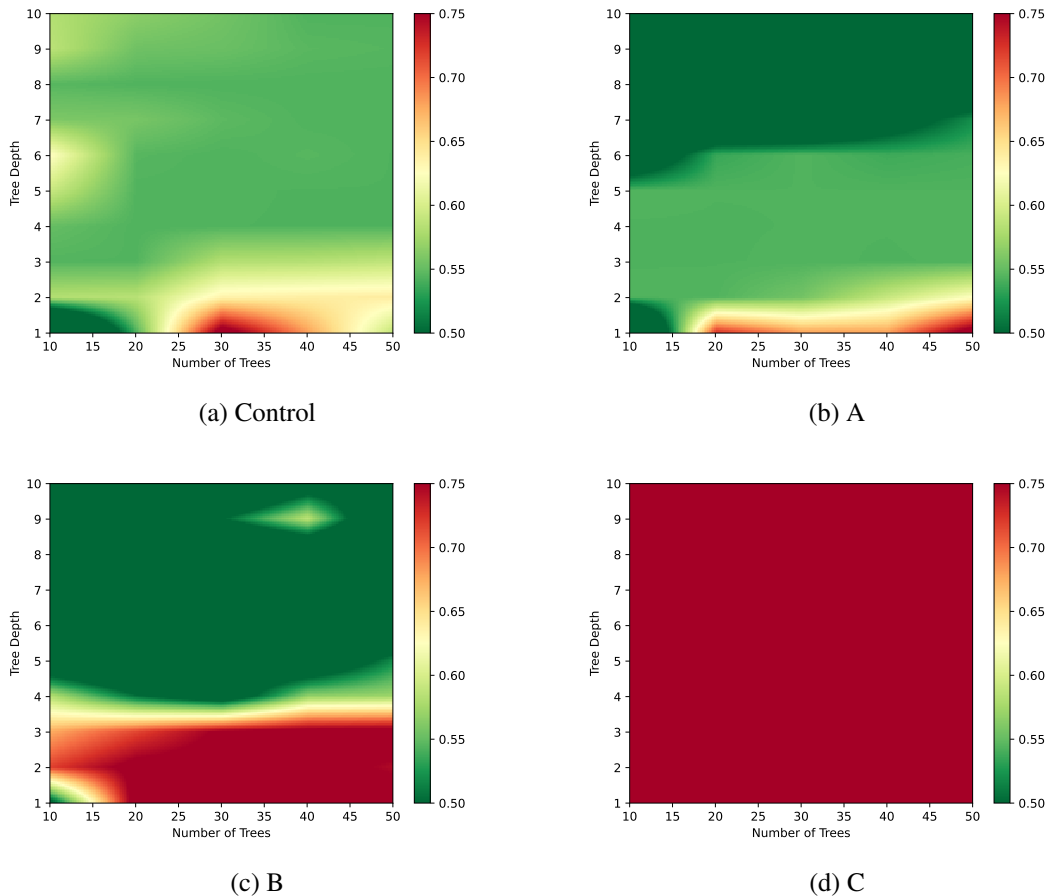
Figure 9.18: Set of multiple threat and non-threat objects: Overall performance of the gradient boost classifier for different values in the hyperparameter sapce, with $P^{(k)} = 2000$ with $K = 8$ and SNR=40dB, showing average κ score for different values n_estimators and max_depth, for different scaling regimes (a) Control, (b) A, (c) B and (d) C.
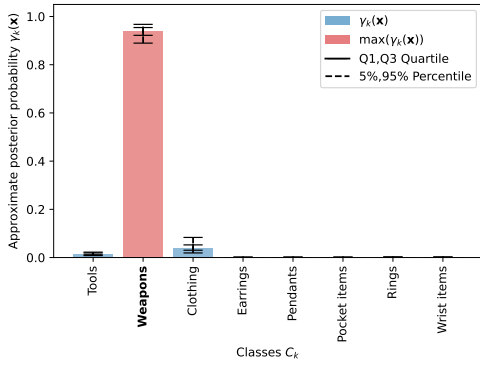
(a) $D^{(\text{test})}$ = Chef
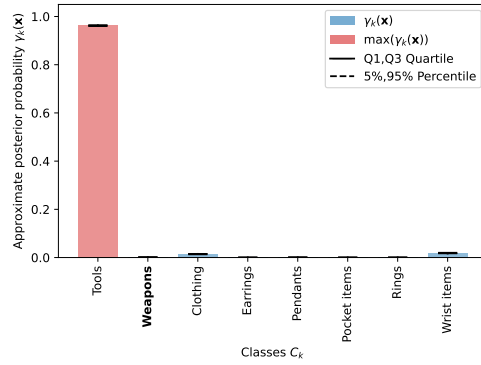
(b) $D^{(\text{test})}$ = Cutlet
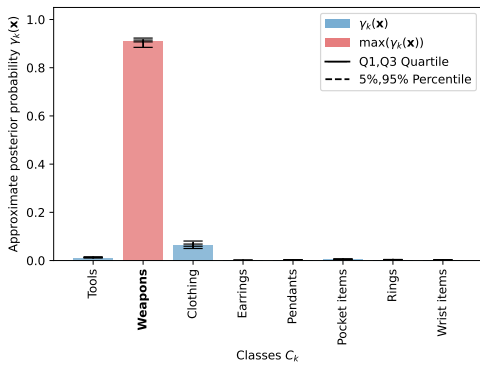
(c) $D^{(\text{test})}$ = meat cleaver
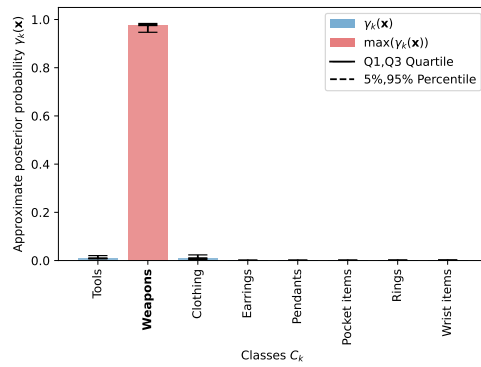
(d) $D^{(\text{test})}$ = Santoku
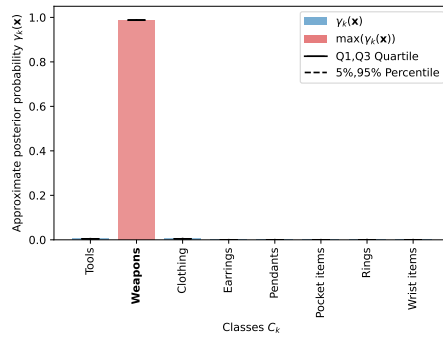
(e) $D^{(\text{test})}$ = Wusthof

Figure 9.19: Set of multiple threat and non-threat objects: Approximate posterior probabilities $\gamma_k(\mathbf{x}) \approx p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, using the gradient boost classifier for $P^{(k)} = 2000$ when $K = 8$ and SNR=40dB showing the case when $D^{(\text{train})}$ is constructed using the scaling regime C and cases where $D^{(\text{test})}$ is constructed of instances (a) chef, (b) cutlet, (c) meat cleaver, (d) Santoku and (e) Wusthof.

Furthermore, in Figure 9.19, for case C, approximatations $p(C_k|\mathbf{x}) \approx \gamma_k(\mathbf{x})$, to posterior probabilities $p(C_k|\mathbf{x})$, $k = 1, \ldots, K$, obtained for the case where the training samples are taken as $(\mathbf{x}, \mathbf{t}) \in D_8^{(\text{train},(2))}$, with either the chef, cutlet, meat cleaver, Santoku or

Wusthof geometry being treated as unseen, in turn. These results were obtained with `n_estimators=50` and `max_depth=2` with SNR=40dB. From this figure it can be observed that $\gamma_{2,50} \approx 1$ for the unseen chef, meat cleaver, Santoku or Wusthof knives suggesting the most likely class is C_2 (a weapon) with small interpercentile and interquartile ranges, which indicates a high degree of certainty associated with the prediction and also a low variability. However, when the unseen object is a cutlet knife, $\gamma_{1,50} \approx 1$ with small interpercentile and interquartile ranges, which indicates that the classifier is still consistently misclassifying this object as a tool, instead of a weapon, despite the classifier being trained over a wider range of object sizes and conductivities. Hence, the classifier remains biased against this geometry. It is conjectured that this is due to the significant difference in the shape of the MPT spectral signature for the cutlet knife geometry shown in Figure 33 of [76], compared to the other knives and gun geometry on which the classifier is trained.

9.4 Limitations and Potential Improvements

One important as yet unanswered question from the previous sections is the limitations of the proposed approach. This section discusses the limitations of the method and potential ways these could be mitigated. Limiting factors of the method are as follows:

1. As was investigated in Section 9.3.4, the extensiveness of the dictionary plays a key role in determining the accuracy of the generated classifier. The ability to vary size and conductivity greatly improves the extent to which realistic objects are characterised along with the extensiveness of the dictionary, however this could be further improved by varying permeability which also hasn't been previously considered. The increase in the number of distinct geometries, which are relevant for the metal detection application being considered, will produce more accurate classifiers.

2. The present classifier has used simulated measurements throughout as access to laboratory measurements was not available due to the covid pandemic between 2020-2022. In work after this PhD, measurements of the MPT spectral signature have been shown to be in good agreement with those simulated by MPT calculator for a range of objects and testing of the classifier is underway. As was studied extensively in this chapter, the accuracy of the measurement system plays a vital role in the accuracy of the predictions being made by the classifier. The considered method for adding noise in this thesis is certainly a good approximation of laboratory measurements, however, in other systems the measurement noise may vary from system to system and from application to applications. There is also the potential for missing information, for example, in the case of a buried object, measurements cannot be taken all around the object that prevents all the MPT coefficients being accurately being recovered and the impact of this is not considered. There are also mathematical modelling assumptions that are not taken account. Again for buried objects, the soil's conductivity is not taken account and is instead assumed

to be non-conducting. Improving the mathematical model in this case would also improve the object characterisation.

3. The current method considers the characterisation of a single object at a time, this could lead to potential issues if multiple small objects are close to each other, leading to the potential misclassification of a of a single larger object. The perturbed magnetic field in the presence of multiple objects has previously been considered by Ledger *et al* [74], which found that as long as the objects are separated by at least the size of the largest object the objects can be independently characterised by distinct MPT object characterisation and, in such cases, the procedure described in this chapter can be applied to each object in turn. Otherwise the group of objects cannot be individually characterised.

4. The propensity for machine learning classifiers to overfit to small datasets is another concern, although an attempt has been made to combat this with variations of object based on scalings and simulated noise being applied, a more thorough study is required on the subject. Other methods such as early stopping could also be employed where loss on a validation set is monitored to gain insight into whether the classifier is overfitting to the dataset. This validation set could also be constructed from real world measurements this would also be beneficial as currently the training and testing sets are constructed using the same method.

Work on each of the points raised above is now ongoing in several future projects.

9.5 Chapter Summary

This chapter has documented results for the application of probabilistic and non-probabilistic classifiers that were discussed in Section 8.3 and were applied to dictionaries of threat and non-threat objects, which were created using method described in Section 8.2.1.

The key novelties of the chapter were as follows: The construction of a large dictionary of object characterisations based on the results of Section 8.2.1 along with applying Lemmas 2.7.1 and 2.7.2 and a sampling to method to provide considerable enhancements in terms of variations of object conductivity and size at negligible computational cost. The investigation of classifier performance as a function of number of excitation frequencies and applied noise, with this being the first time that a computational dictionary of realistic MPT spectral signatures has been employed for object classification. The novel investigation into the classification of unseen objects is presented. Finally, a review of limitations and potential improvements was then presented.

This chapter will be followed by conclusions and possible further work presented in Chapter 10.

Part V

Conclusions

Chapter 10

Conclusions and Future Work

10.1   Concluding Remarks

This thesis has harnessed the state-of-the-art mathematical developments in MPT object characterisation described in the introduction and reviewed in Chapter 2, which have been applied in conjunction with recent developments in machine learning classification to the metal detection problem. First, a review of recent developments of MPT object characterisations was provided. A problem specific, hp-finite element discretisation was presented, which paved the way for the development of a PODP based ROM that benefits from a-posteriori error estimates. Thus, allowing for the rapid production of MPT spectral signatures the accuracy of which was guaranteed. This methodology was then implemented in Python, using the `NGSolve` finite element library, where other problem specific efficiencies were also included, this software was then packaged and released as the open source `MPT-Calculator`. Using this software, MPT spectral signatures were then produced for a series of realistic threat and non-threat objects, creating the first of its kind synthetic dataset. This was also released as the open source `MPT-Library` dataset. Lastly, a series of ML classifiers were documented and applied to several supervised classification problems using this synthetic dataset. Therefore, it can be concluded that the aim and objectives stated in Chapter 1 have been achieved. In the following a more detailed description of the conclusions that can be extracted from each of the chapters of the thesis are provided.

10.2   Conclusions

The recent work and developments in the theory surrounding the MPT have been reviewed, with important results being presented.

- **Chapter 2: (Eddy-Current Model and Object Characterisation)** presented an introduction to important topics such as the formulation of the time harmonic eddy current model and an asymptotic expansion of the perturbed magnetic field. An overview of the MPT was presented, documenting recent key results and discussing

167

limiting cases, which provided a fundamental insight into the behaviour of the MPT and highlighted the important role object topology plays along with providing motivation for the use of MPT spectral signatures as opposed to MPTs at a single frequency. This chapter addresses the objective "Provide a review of the recent work and developments of theory surrounding MPTs".

A robust, efficient and accurate computational methodology for the solution of eddy-current problems was devised, implemented and packaged.

- **Chapter 3: (Finite Element Discretisation and the Reduced Order Model)** proposed a PODP based ROM for the acceleration of the solution of vectorial transmission problems by finite elements and the efficient computation of MPT spectral signatures. This included the presentation of a problem specific weak form, where, a $\boldsymbol{H}(\text{curl})$ based, hp-finite element discretisation was advocated, to obtain the solutions of the full order problem. The development of a PODP based ROM was proposed, this benefitted from a-posteriori error estimates, which provided an upper bound on the error in the predicted MPT coefficients with respect to the full order model solution.

- **Chapter 4: (Implementation Details of `MPT-Calculator` Software)** built on the work presented in Chapter 3, focusing on the associated implementation and the development of the open source software `MPT-Calculator`. The chapter justifies the choice of FE framework, documents the computational implementation of the methodology presented in Chapter 3 and discusses the support provided with, and usage of, the packaged open source `MPT-Calculator` software. Chapters 3 and 4 address the objective "Develop a robust and accurate computational methodology for the solution of eddy-current problems using finite elements software".

The accuracy and efficiency of the proposed ROM was shown for several demonstrative problems with a comparison being made to the use of finite elements alone.

- **Chapter 5: (Numerical Results for the Reduced Order Model)** presented a series of illustrative examples to demonstrate the implementation and performance of the reduced order model, with a-posteriori error estimates, proposed in Chapters 3 and 4. Discussing important topics such as snapshot placement, accuracy of the method, additional snapshot placement and time savings offered by the method. This chapter addresses the objective "Demonstrate the accuracy and efficiency offered by this computational methodology".

The methodology for the decomposition of the inverse problem to one of object location and object classification was then described, as part of this, several candidate tensor invariants were considered.

- **Chapter 6: (Decomposing the Inverse Problem to one of Object Location and Object Classification)** describes the decomposition of the inverse problem, to one

of object location and object classification, which follows since the MPT is independent of object position. In this chapter, a brief review of methods to obtain the location of a hidden target is presented, with the methodology of the MUSIC algorithm documented. A series of applicable tensor invariants that could be obtained from the MPT coefficients were then presented, with each being invariant of object rotation. This chapter contributes to the objective "Apply the method to a series of geometrically realistic objects to create a database of threat and non-threat object characterisations".

A synthetic dataset of realistic threat and non-threat object characterisations were produced using the methodology presented in chapters 3 and 4.

- **Chapter 7: (Real World Object Characterisation Dataset)** focused on the development of an open source synthetic dataset of realistic object characterisations that could be used for object identification in a possible security screening environment. The dataset, called the `MPT-Library`, consists of object characterisations of 67 unique geometries, with multiple materials considered for each geometry producing characterisation of 158 distinct objects. The number of basic object descriptions were then significantly enhanced using scaling results for object size and object conductivity. Chapters 6 and 7 address the objective "Apply the method to a series of geometrically realistic objects to create a database of threat and non-threat object characterisations".

A review of supervised ML classifiers applicable for deployment in a security screening setting was presented, this also included a recap of basic ML terminology, algorithms and evaluation methods.

- **Chapter 8: (Probabilistic and Non-probabilistic Machine Learning Classifiers)** built on the work presented in Chapter 7, documenting and comparing probabilistic and non-probabilistic Machine Learning (ML) classifiers that are appropriate for classifying objects when the features are MPT invariants, with the goal of developing an ML classifier trained on the `MPT-Library` that could be deployed in a walk through metal detector. First, an appropriate method for adding noise to the MPT coefficients that simulates the noise associated with measuring MPT coefficients in real world set ups was described. Then a review of candidate ML classifiers were documented with references provided for those less familiar with the methods. Finally metrics that were later used in Chapter 9 to evaluate the performance were discussed. This chapter contributes to the objective "Document and implement a series of supervised ML classifiers to the database of object characterisations".

A series of supervised ML classifiers were applied to several problems created using the open source synthetic dataset developed in Chapter 7.

- **Chapter 9: (Machine Learning Classification Results)** presented results for the application of the ML classifiers discussed in Chapter 8, applied to four problems, a

coin classification problem, two multi-class problems (using 8 and 15 classes) and classification of unseen objects, all created using the dataset developed in Chapter 7. The first was used to highlight how statistical properties of the dataset can be used to motivate the choice of a simple logistic regression classifier. The 8-class and 15-class problems were used to illustrate the need to use more advanced forms of classification and to motivate the need to optimise the hyperparameters. The 15-class problem highlighted the benefits of using increased object class fidelity. Chapters 8 and 9 address the objective "Document and implement a series of supervised ML classifiers to the database of object characterisations".

In summary, a problem specific weak form was presented for the solution of a transmission problem needed to compute MPT coefficients. The transmission problem was then discretised using FE. An ROM methodology based on the PODP was developed to accelerate solutions to the transmission problem for new frequencies. This benefitted from a-posteriori error estimates, which provided an upper bound on the error associated with the MPT coefficients. The ROM methodology was implemented in Python using the `NGSolve` FE library and then tested on a series of illustrative examples where the performance, accuracy and time savings associated with the ROM were documented showing time savings in the order of 86% were possible when using the ROM compared with FE alone for a frequency sweep using 161 output values. Investigations were also conducted where snapshot and additional snapshot placement was considered showing the advantages of using logarithmically spaced snapshots as well as showing the possibility to develop an adaptive scheme using the a-posteriori error estimates to inform the placement of new snapshots, which reduced the size of the error estimates increasing confidence in the prediction. A decomposition of the inverse problem into object location and object identification was presented. Object classification was the main focus and candidate tensor invariants that are invariant of object rotation were reviewed. These tensor invariants were then used in conjunction with the ROM methodology developed in the previous chapters to develop a first of its kind synthetic dataset of realistic object characterisations that could be used for object detection in a security screening environment. A review of applicable ML classifiers was presented along with validation methods and metrics that were later used to evaluate the performance of the classifiers. Finally, a series of four challenging classification problems were developed using the synthetic dataset previously developed, with the ML classifiers being used to solve these problems. These were used to motivate how statistical properties of the dataset can be used to choose an appropriate classifier, also considering the trade off between noise and number of frequencies sampled at, along with demonstrating the benefit of increased dataset size. Lastly an investigation was then conducted into the effect that removing specific knife geometries out of the training set had on the classifier's ability to detect these threat objects, the ability of the classifier to generalise was shown with 80% of the knife models still being correctly identified.

10.3 Recommendations for Future Work

The accomplishments of this thesis open up several possible lines of research that could be considered to improve and extend the capabilities of the current methodology. Several of these lines are being followed up as part of the grants EP/V049496/1 and EP/V009109/1 funded by the Engineering and Physical Research Council and led by Prof. P.D. Ledger. Future work can be split into two key areas, the improvement of methodologies already presented and the extension to new methodologies that have not been previously considered. Firstly, a list for the former is as follows:

- **Object discretisation:** In this thesis, the open source software `NetGen` from `NGSolve` [4] was used to produce and mesh objects of interest. However, for producing larger datasets of more complex geometries, improvements to the usability of the method could be made if integration with a commercial tool such as `Solidworks` or `Ansys` was developed. Further to this, with recent developments in 3D object reconstruction [89], a procedure could be developed within which a digital twin of a real world object could be created and simulated.

- **Adaptive snapshot placement:** In Chapter 3, the methodology for an adaptive snapshot procedure based on the a-posteriori error estimates, which built on the procedure described in [55], was presented. As was shown in Chapter 3, with the comparison of using a single targeted additional snapshot vs four additional predetermined snapshots, there is the possibility for a reduction in time and computational cost if such an automatic adaptive procedure was implemented to generate additional snapshots to further improve the accuracy.

- **Application specific datasets:** The dataset produced in relation to the work presented in this thesis was security screening motivated. However, for tasks other than security screening applications, such as humanitarian demining [7] and archeological digs [29] alternative, application specific datasets could be created.

The possible extensions of the work are summarised as follows:

- **Comparisons with real world measurements:** There is considerable scope in making comparisons from the `MPT-Library` to real world measurements, with work in this direction already being undertaken [97]. Further to this, consideration could be given to the deployment of an ML classifier that is trained using synthetic data and used to classify real world data rather than simulated data with noise.

- **Use of synthetic data for calibration and noise reduction:** Further to the previous point, due to the superior accuracy the MPT coefficients can be calculated to using the methodology presented in this thesis when compared with real world measurements. There is potential to use the synthetic measurements produced in a scheme to calibrate or reduce noise in measurements of real world systems. The procedure could be deployed in a measurement device and used as a preprocessing

171

step, this would involve training an ML algorithm such as an autoencoder using the real world measured MPT spectral signature as inputs and the synthetic data as the desired outputs. Work in related fields has been conducted on the topic of noise reduction such as in the case of medical imaging [43] where autoencoders were used.

- **Production of new geometries using a generative model:** An extension to the production of new geometries would be the automation of this process, where a generative model could be used to produce new geometries from existing ones. Generative models have been growing in popularity with recent developments such as generative adversarial networks (GANs) [44] being studied extensively in applications such as image-to-image translation [61] and more recently there has been increasing interest for extending dataset sizes in other fields using GANs such as datasets of MRI images [49]. An application of the 3D-GAN developed by Wu *et al.* [137] and related work could be used to extend a synthetic dataset such as the one developed as part of this thesis. This could also be used in conjunction with the idea of 3D object reconstruction [89] discussed earlier to further automate the process.

- **ML based ROM:** There is the potential to replace the methodology presented on ROMs in this thesis by replacing the ROM with an ML regression algorithm. This potentially has the ability to rapidly predict MPTs for objects with new permeabilities (for which scaling results do not exist) and, potentially, new geometries. ML based ROMs have been used before in related fields such as computational fluid dynamic where an ROM based on Gaussian process regression was used to model air flows in urban environments [138].

- **Improved fidelity of `MPT-Calculator`:** Improve the capabilities of the `MPT-Calculator`, in particular investigating cases for high values of μ_r where fidelity could be improved and increase the limits on numbers of degrees of freedom. Work is already being conducted on this.

- **GMPTs:** The MPT provides the object characterisation in the leading order term in an asymptotic expansion of $(\boldsymbol{H}_\alpha - \boldsymbol{H}_0)(\boldsymbol{x})$ as $\alpha \to 0$, as such there is scope for both calculating and measuring higher order object characterisations known as general magnetic polarizability tensors (GMPTs) [72], which provide improved object characterisations. Work has already begun on this [97].

- **Library of GMPTs:** There is also the potential to develop improved ML classifiers based on GMPT object characterisations. The extension is not trivial though, GMPTs are more complicated than MPTs and further theoretical developments are needed to better understand their properties. Work in this direction is underway in the aforementioned grants. Still further, unlike MPTs where either eigenvalues or principal invariants are invariant under rotation and provide appropriate ML features, choosing appropriate invariants for GMPTs is still open.

Part VI

Appendices

Appendix A

`MPT-Calculator` File Details

Input files: The four input files of the `MPT-Calculator` allow the user to dictate several aspects of the procedure[1]. The file `main.py` is designed to be the most interacted with allowing the user to define, firstly `.geo` file to be used, the geometric and polynomial refinement to be used, the frequencies the MPT should be produced at and how the simulation should be run, i.e. using a reduced order model and/ or in parallel. The file `Settings.py` contains several sections. The first of which `DefaultSettings` allows the user to define aspects of how the procedure should be carried out defining such things as how many CPUs are to be used when simulating in parallel, how many full order solutions should be used when producing the reduced order model, etc. The middle two sections `AdditionalOutputs` and `SaverSettings` allow the user to define which additional outputs they would like to produce and where to save the outputs, the additional outputs include the error certificates described in Section 3.3.3, the object dependant eddy current limit described in Section 2.4.3 and the vector field of eddy currents exported as a `.vtk` file. The final section `SolverParameters` allows the user to change aspects of how the problems are solved defining how much regularisation to add to the problem, the tolerance the solution should be obtained to etc. The `PlotterSettings.py` file defines how the graphical representation of the results will appear, this file allows the user to change which coefficients are shown, the line styles to be used, and whether the object dependent eddy current limit should be indicated on the graph. The `.geo` file is the file in which the geometry and material properties of the object are defined, the geometry of the object is built using Constructive Solid Geometry (CSG), where simple geometric objects such as spheres, blocks are used along with logical operators unions and complements to create larger and more interesting and intricate objects. The material properties of different regions can then be tagged using the API created by the author in the process described in Section 5 of the documentation for the `MPT-Calculator` [134].

Solution files: The solution files of the `MPT-Calculator` are associated with simulating/ obtaining a solution for the finite element problems. The first of these files is

---

[1]A more thorough explanation of these files is provided in the `MPT-Calculator` documentation [134].

`MeshCreation.py`, this file contains two functions, the first, `Meshmaker`, runs a subprocess calling `NGSolve` and produces a mesh for the object defined in the chosen `.geo` file, the second, `VolMatUpdater`, reads the `.geo` file and interprets the tagged material properties, updating the `.vol` file which will be used by `NGSolve` later in the process. The next file `ResultsFunctions.py` contains several functions and is used twice throughout the procedure, the first time, the function `FolderMaker` is called, it creates the folder structure which will be used when saving the outputs, it is at this point the input files used are copied to the results folder, this stops the user from changing these before the files are copied. The second time this function is used, it calls one of three saver functions `SingleSave`, `PODSave` or `FullSave`, this call is dependant on the inputs of `main.py`, these functions in turn call the relevant plotter function defined in `Plotters.py`, then save the relevant outputs in the results folder produced earlier. The final solution files, `FullSolvers.py`, `PODSolvers.py` and `SingleSove.py` are associated with the 'problem solving' part of the procedure, these interact directly with `NGSolve` and is tasked with creating finite element spaces within which the problems can be solved, creating and solving linear systems of equations (although this is done by calling functions which reside in `MPTFunctions.py`) and in the case of the parallelised version of code distributing the work between CPUs in an efficient manner.

Output files: The output files of the `MPT-Calculator` produce results which are to be saved. The first of these files, `Checkvalid.py`, is responsible for the production of the object specific eddy current limit, this is the additional output which was discussed previously in Section 2.4.3. The next file, `PODFunctions.py`, contains a function which is designed to produce the MPT coefficients directly from the reduced order model without the need to project the solutions back to the full order space, this comes with significant errors when calculating the imaginary coefficients and is therefore not used in the current version of the `MPT-Calculator`. The file `MPTFunctions.py` contains several functions, all of which can be used in parallel, which are tasked with the calculation of the MPT coefficients, the first two functions `Theta0` and `Theta1`, do not give an output and are sub-functions called by `SingleSove.py`, the functions `Theta1_Sweep` and `Theta1_Lower_Sweep` are sub-functions which are called by `FullSolvers.py` and `PODSolvers.py` they both solve the finite element problems and post-process the solutions calculating the MPT coefficients. The final function `MPTCalculator` post-processes the solutions calculating the MPT coefficients. The final file is `Plotters.py`, this file contains several functions all of which are tasked with producing the graphical outputs of the code, these are also affected by the inputs of `PlotterSettings.py` which determine the look of the graphs.

Bibliography

[1] Comsol. `https://uk.comsol.com/`.

[2] deal.ii. `https://www.dealii.org/`.

[3] Fenics. `https://fenicsproject.org/`.

[4] Ngsolve. `https://ngsolve.org`.

[5] TT33 blueprints. `https://pdfslide.net/documents/tt-33-model-blueprints-drawings-of-main-parts.html`. Date accessed 20th October 2020.

[6] Copper-nickel 90/10 and 70/30 alloys technical data, 1982.

[7] O. A. Abdel-Rehim, J. L. Davidson, L. A. Marsh, M. D. O'Toole, and A. J. Peyton. Magnetic polarizability tensor spectroscopy for low metal anti-personnel mine surrogates. *IEEE Sensors*, 16:3775–3783, 2016.

[8] H. Abdi and L. J Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[9] A. Agurto, Y. Li, G. Y. Tian, N. Bowring, and S. Lockwood. A review of concealed weapon detection and research in perspective. In *2007 IEEE International Conference on Networking, Sensing and Control*, pages 443–448. IEEE, 2007.

[10] D. Ambruš, D. Vasić, and V. Bilas. Robust estimation of metal target shape using time-domain electromagnetic induction data. *IEEE Transactions Instrumentation and Measurement*, 65:795–807, 2016.

[11] H. Ammari, T. Boulier, J. Garnier, and H. Wang. Shape recognition and classification in electro-sensing. *Proceedings of the National Academy of Sciences*, 111:11652–11657, 2014.

[12] H. Ammari, A. Buffa, and J.-C. Nédélec. A justification of eddy currents model for the Maxwell equations. *SIAM Journal on Applied Mathematics*, 60(5):1805–1823, 2000.

[13] H. Ammari, J. Chen, Z. Chen, J. Garnier, and D. Volkov. Target detection and characterization from electromagnetic induction data. *Journal de Mathématiques Pures et Appliquées*, 101(1):54–75, 2014.

[14] H. Ammari, J. Chen, Z. Chen, D. Volkov, and H. Wang. Detection and classification from electromagnetic induction data. *Journal of Computational Physics*, 301:201–217, 2015.

[15] H. Ammari and H. Kang. *Polarization and Moment Tensors with Applications to Inverse Problems and Effective Medium Theory*. Springer-Verlag New York, 2007.

[16] A. Antonosen. `https://www.flickr.com/photos/handvapensamlingen/6825679152/in/photolist-bpaoBj-eXwMWW-a664rz-2fhqR43-dQ7ipQ`, 2012. Date Accessed 28th October 2020.

[17] C. O. Ao, H. Braunisch, K. O'Neill, and J. A. Kong. Quasi-magnetostatic solution for a conducting and permeable spheroid with arbitrary excitation. *IEEE Transactions on Geoscience and Remote Sensing*, 40:887–897, 2002.

[18] I. Babuška and T. Strouboulis. *The finite element method and its reliability*. Oxford university press, 2001.

[19] B. E. Barrowes, K. O'Neill, T.M. Gregorcyzk, and J. A. Kong. Broadband analytical magnetoquasistatic electromagnetic induction solution for a conducting and permeable spheroid. *IEEE Transactions on Geoscience and Remote Sensing*, 42:2479–2489, 2004.

[20] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[21] A. Bjorck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, USA, 1996.

[22] J. Bonet and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 2010.

[23] B. H. Brown. Medical impedance tomography and process impedance tomography: a brief review. *Measurement Science and Technology*, 12(8):991, 2001.

[24] B.M. Brown, M. Marletta, and J. Reyes Gonzales. Uniqueness for an inverse problem in electromagnetism with partial data. *Journal of Differential Equations*, 260(8):6525–6547, 2016.

[25] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari. Tikhonov regularization and the l-curve for large discrete ill-posed problems. *Journal of computational and applied mathematics*, 123(1-2):423–446, 2000.

[26] A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7), 2000.

[27] M. S. Choucino. 3D simulation of magneto-mechanical coupling in MRI scanners using high order FEM and POD. 2021.

[28] J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

[29] M. Connor and D. D. Scott. Metal detector use in archaeology: An introduction. *Historical Archaeology*, 32(4):76–85, 1998.

[30] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other Kernel-Based Learning Methods*. Cambridge University Press (Cambridge), 2000.

[31] B. Dacorogna. *Introduction to the Calculus of Variations*. World Scientific Publishing Company, 2014.

[32] J. L. Davidson, O. A. Abdel-Rehim, P. Hu, L. A. Marsh, M. D. O'Toole, and A. J. Peyton. On the magnetic polarizability tensor of us coinage. *Measurement Science and Technology*, 29:035501, 2018.

[33] B. Dekdouk, C. Ktistis, L A Marsh, D. W. Armitage, and A. J. Peyton. Towards metal detection and identification for humanitarian demining using magnetic polarizability tensor spectroscopy. *Measurement Science and Technology*, 26:115501, 2015.

[34] G. Dingley and M. Soleimani. Multi-frequency magnetic induction tomography system and algorithm for imaging metallic objects. *Sensors*, 21(11):3671, 2021.

[35] J. Elith, J. R. Leathwick, and T. Hastie. A working guide to boosted regression trees. *Journal of Animal Ecology*, 77(4):802–813, 2008.

[36] Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *ICML*, 2018.

[37] J. H. Friedman. Greedy function approximation: A gradient boost machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

[38] L. Audickas G. Allen, P. Loft, and A. Bellis. Knife crime in england and wales. Technical Report SN4304, 2019. Date Accessed 9th October 2020.

[39] F. Galton. *Natural inheritance*. Macmillan and Company, 1889.

[40] A. Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2019.

[41] G. H. Golub, P. C. Hansen, and D. P. O'Leary. Tikhonov regularization and total least squares. *SIAM journal on matrix analysis and applications*, 21(1):185–194, 1999.

[42] G. H. Golub and C. F. Van Loan. *Matrix Computations*. JHU Press, 1996.

[43] L. Gondara. Medical image denoising using convolutional denoising autoencoders. In *2016 IEEE 16th international conference on data mining workshops (ICDMW)*, pages 241–246. IEEE, 2016.

[44] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[45] T.M. Gregorcyzk, B. Zhang, J.A. Kong, B.E.Barrowes, and K. O'Neill. Electromagnetic induction from highly permeable and conductive ellipsoids under arbitrary excitation: application to the detection of unexploded ordances. *IEEE Transactions on Geoscience and Remote Sensing*, 46:1164–1176, 2008.

[46] H. Griffiths. Magnetic induction tomography. *Measurement science and technology*, 12(8):1126, 2001.

[47] M. R. Gross. Magnetic characteristics of non-magnetic metallic materials comparison of properties in strong and weak fields. Technical Report E.E.S. Report 4E(2)66904, U.S. Naval Engineering Experiment Station, Annapolis, Maryland, USA, 1951.

[48] M. Hagenlocher, D. Hölbling, S. Kienberger, S. Vanhuysse, and P. Zeil. Spatial assessment of social vulnerability in the context of landmines and explosive remnants of war in battambang province, cambodia. *International Journal of Disaster Risk Reduction*, 15:148–161, 2016.

[49] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama. GAN-based synthetic brain MR image generation. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 734–738. IEEE, 2018.

[50] M. Hanke. *A Taste of Inverse Problems: Basic Theory and Examples*. SIAM, 2017.

[51] M. Hanke and C. W. Groetsch. Nonstationary iterated tikhonov regularization. *Journal of Optimization Theory and Applications*, 98(1):37–53, 1998.

[52] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. SIAM, Philadelphia, USA, 2005.

[53] P. C. Hansen and D. P. O'Leary. The use of the l-curve in the regularization of discrete ill-posed problems. *SIAM journal on scientific computing*, 14(6):1487–1503, 1993.

[54] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer Series in Statistics (New York), 2009.

[55] J. S. Hesthaven, G. Rozza, and B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations.* Springer, 2016.

[56] R. Hiptmair. Boundary element methods for eddy current computation. In *Boundary Element Analysis (Lecture Notes in Applied and Computational Mechanics Book Series Vol. 29)*, pages 213–248. Springer, 2007.

[57] R. Hiptmair and J. Ostrowski. Coupled boundary-element scheme for eddy-current computation. *Journal of Engineering Mathematics*, 51:231–250, 2005.

[58] C. Y. Ho, M. W. Ackerman, K. Y. Wu, T. N. Havill, R. H. Bogaard, R. A. Matula, S. G. Oh, and H. M. James. Electrical resistivity of ten selected binary alloy systems. *Journal of Physical and Chemical Reference Data*, 12:183–322, 1983.

[59] D. S. Holder. *Electrical impedance tomography: methods, history and applications.* CRC Press, 2004.

[60] TT33 Pistol Image. `https://en.wikipedia.org/wiki/TT_pistol`. Date accessed 20th October 2020.

[61] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[62] N. Karimian, M. D. O'Toole, and A. J. Peyton. Electromagnetic tensor spectroscopy for sorting of shredded metallic scrap. In *IEEE SENSORS 2017 - Conference Proceedings*. IEEE, 2017.

[63] T. A. K. Khairuddin and W. R. B. Lionheart. Do electro-sensing fish use the first order polarization tensor for object characterization? In *100 Years of Electrical Imaging*, page 149.

[64] T. A. K. Khairuddin and W. R. B. Lionheart. Fitting ellipsoids to objects by the first order polarization tensor. *Malaya Journal of Matematik*, 4(1):44–53, 2013.

[65] T. A. K. Khairuddin and W. R. B. Lionheart. Characterization of objects by electrosensing fish based on the first order polarization tensor. *Bioinspiration and Biomimetics*, 11:055004, 2016.

[66] M. Kuhn and K. Johnson. *Applied Predictive Modeling.* Springer (New York), 2013.

[67] W. W. Lai, X. Derobert, and P. Annan. A review of ground penetrating radar application in civil engineering: A 30-year journey from locating and testing to imaging and diagnosis. *Ndt & E International*, 96:58–78, 2018.

[68] P. D. Ledger. Personal communications, 2018-2021.

[69] P. D. Ledger and W. R. B. Lionheart. Characterising the shape and material properties of hidden targets from magnetic induction data. *IMA Journal of Applied Mathematics*, 80(6):1776–1798, 2015.

[70] P. D. Ledger and W. R. B. Lionheart. Understanding the magnetic polarizability tensor. *IEEE Transactions on Magnetics*, 52(5):6201216, 2016.

[71] P. D. Ledger and W. R. B. Lionheart. An explicit formula for the magnetic polarizability tensor for object characterization. *IEEE Transactions on Geoscience and Remote Sensing*, 56(6):3520–3533, 2018.

[72] P. D. Ledger and W. R. B. Lionheart. Generalised magnetic polarizability tensors. *Mathematical Methods in the Applied Sciences*, 41:3175–3196, 2018.

[73] P. D. Ledger and W. R. B. Lionheart. The spectral properties of the magnetic polarizability tensor for metallic object characterisation. *Mathematical Methods in the Applied Sciences*, 43:78–113, 2020.

[74] P. D. Ledger, W. R. B. Lionheart, and A. A. S. Amad. Characterisation of multiple conducting permeable objects in metal detection by polarizability tensors. *Mathematical Methods Applied Sciences*, 42(3):830–860, 2019.

[75] P. D. Ledger and K. Morgan. The application of the hp-finite element method to electromagnetic problems. *Archives of Computational Methods in Engineering*, 12(3):235–302, 2005.

[76] P. D. Ledger, B. A. Wilson, A. A. S. Amad, and W. R. B. Lionheart. Identification of metallic objects using spectral magnetic polarizability tensor signatures: Object characterisation and invariants. *International Journal for Numerical Methods in Engineering*, 2021.

[77] P. D. Ledger and S. Zaglmayr. *hp*-finite element simulation of three-dimensional eddy current problems on multiply connected domains. *Computer Methods in Applied Mechanics and Engineering*, 199:3386–3401, 2010.

[78] R. J. LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM, 2007.

[79] Z. Luo, J. Du, Z. Xie, and Y. Guo. A reduced stabilized mixed finite element formulation based on proper orthogonal decomposition for the non-stationary Navier–Stokes equations. *International Journal for Numerical Methods in Engineering*, 88(1):31–46, 2011.

[80] L. Ma and M. Soleimani. Magnetic induction tomography methods and applications: A review. *Measurement Science and Technology*, 28(7):072001, 2017.

[81] X. Ma, A. J. Peyton, S. R. Higson, and P. Drake. Development of multiple frequency electromagnetic induction systems for steel flow visualization. *Measurement Science and Technology*, 19(9):094008, 2008.

[82] J. Makkonen, L. A. Marsh, J. Vihonen, A. Järvi, D. W. Armitage, A. Visa, and A. J. Peyton. KNN classification of metallic targets using the magnetic polarizability tensor. *Measurement Science and Technology*, 25:055105, 2014.

[83] J. Makkonen, L. A. Marsh, J. Vihonen, A. Järvi, D. W. Armitage, A. Visa, and A. J. Peyton. Improving reliability for classification of metallic objects using a WTMD portal. *Measurement Science and Technology*, 26:105103, 2015.

[84] J. Mandel and C. R. Dohrmann. Convergence of a balancing domain decomposition by constraints and energy minimization. *Numerical linear algebra with applications*, 10(7):639–659, 2003.

[85] C.D. Manning, P. Raghavan, and H. Schütze. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, 2009.

[86] L. A. Marsh, C. Ktisis, A. Järvi, D. W. Armitage, and A. J. Peyton. Three-dimensional object location and inversion of the magnetic polarisability tensor at a single frequency using a walk-through metal detector. *Measurement Science and Technology*, 24:045102, 2013.

[87] L. A Marsh, C. Ktistis, A. Järvi, D. W .Armitage, and A. J. Peyton. Determination of the magnetic polarizability tensor and three dimensional object location for multiple objects using a walk-through metal detector. *Measurement Science and Technology*, 25:055107, 2014.

[88] L. A. Marsh, W. Van Verre, J. L. Davidson, X. Gao, F. J. W. Podd, D. J. Daniels, and A. J. Peyton. Combining electromagnetic spectroscopy and ground-penetrating radar for the detection of anti-personnel landmines. *Sensors*, 19(15):3390, 2019.

[89] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[90] Minesweepers. Facts about landmines.

[91] Royal Mint. `https://www.royalmint.com/discover/uk-coins/coin-design-and-specifications/`. Date accessed 20th October 2020.

[92] B. S. Mitchell. *An Introduction to Materials Engineering and Science: For Chemical and Materials Engineers*. John Wiley & Sons, 2004.

[93] P. Monk. *Finite element methods for Maxwell's equations*. Oxford University Press, 2003.

[94] S. J. Norton and I. J. Won. Identification of buried unexploded ordnance from broadband induction data. *IEEE Transactions Geoscience Remote Sensing*, 39:2253–2261, 2001.

[95] J. A. Osborn. Demagnetizing factors of the general ellipsoid. *Physical Review*, 67:351–357, 1945.

[96] T. Özdeğer, J. L. Davidson, W. Van Verre, L. A. Marsh, W. R. B. Lionheart, and A. J. Peyton. Measuring the magnetic polarizability tensor using an axial multi-coil geometry. *IEEE Sensors Journal*, 21:19322–19333, 2021.

[97] T. Özdeğer, P. D. Ledger, W. R. B. Lionheart, J. L. Davidson, and A. J. Peyton. Measurement of gmpt coefficents for improved object characterisation in metal detection. *IEEE Sensors Journal*, 2021.

[98] D. M. W. Powers. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *International Journal of Machine Learning Technology*, 2:37–63, 2011.

[99] D. M. W. Powers. What the f-measure doesn't measure: Features, flaws, fallacies and fixes. *arXiv preprint arXiv:1503.06410*, 2015.

[100] O. A. Abdel Rehim, J. L. Davidson, L. A. Marsh, M. D. O'Toole, D. Armitage, and A. J. Peyton. Measurement system for determining the magnetic polarizability tensor of small metallic targets. In *IEEE Sensor Application Symposium*, 2015.

[101] O. A. Abdel Rehim, J. L. Davidson, L.A. Marsh, M. D. O'Toole, and A. J. Peyton. Magnetic polarizability spectroscopy for low metal anti-personnel mine surrogates. *IEEE Sensors Journal*, 16:3775 – 3783, 2016.

[102] M. D. Richard and R. P. Lippmann. Neural network classifiers estimate Bayesian a-posteriori probabilities. *Neural Computation*, 3:461–483, 1991.

[103] T. Rylander, P. Ingelström, and A. Bondeson. *Computational electromagnetics*. Springer Science & Business Media, 2012.

[104] McCulloch W. S. and Pitts W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biophysics*, 52(1-2):115–133, 1944.

[105] C. Sammut and G. I. Webb. *Encyclopaedia of Machine Learning*. Springer Science & Business Media, 2011.

[106] O. Scherzer. *Handbook of mathematical methods in imaging*. Springer Science & Business Media, 2010.

[107] K. Schmidt, O. Sterz, and R. Hiptmair. Estimating the eddy-current modeling error. *IEEE Transactions on Magnetics*, 44(6):686–689, 2008.

[108] J. Schöberl. Netgen - an advancing front 2d/3d-mesh generator based on abstract rules. *Computing and Visualization in Science*, 1(1):41–52, 1997.

[109] J. Schöberl. C++11 implementation of finite elements in ngsolve. Technical report, ASC Report 30/2014, Institute for Analysis and Scientific Computing, Vienna University of Technology, 2014.

[110] J. Schöberl and S. Zaglmayr. High order Nédélec elements with local complete sequence properties. *COMPEL-The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 24(2):374–384, 2005.

[111] M. Seoane, P. D. Ledger, A. J. Gil, and M. Mallett. An accurate and efficient three-dimensional high-order finite element methodology for the simulation of magneto-mechanical coupling in mri scanners. *International Journal for Numerical Methods in Engineering*, 119(12):1185–1215, 2019.

[112] M. Seoane, P. D. Ledger, A. J. Gil, S. Zlotnik, and M. Mallett. A combined reduced order-full order methodology for the solution of 3D magneto-mechanical problems with application to MRI scanners. *International Journal for Numerical Methods in Engineering*, 121:3529–3559, 2020.

[113] B. Shmueli. Multi-class metrics made simple, part II. `https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca` Accessed: 2020-01-31.

[114] B. Shmueli. Multi-class metrics made simple, part III. `https://towardsdatascience.com/multi-class-metrics-made-simple-the-kappa-score-aka-cohens-kapp` Accessed: 2020-01-31.

[115] D. E. Smith. *A source book in mathematics*. Courier Corporation, 1929.

[116] O. K. Smith. Eigenvalues of a symmetric 3×3 matrix. *Communications of the ACM*, 4:168, 1961.

[117] J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25, 2012.

[118] M. Soleimani. Level-set method applied to magnetic induction tomography using experimental data. *Research in Nondestructive Evaluation*, 18(1):1–12, 2007.

[119] M. Soleimani. Improving the temporal resolution of magnetic induction tomography for molten metal flow visualization. *IEEE Transactions on Instrumentation and Measurement*, 59(3):553–557, 2009.

[120] M. Soleimani. Simultaneous reconstruction of permeability and conductivity in magnetic induction tomography. *Journal of Electromagnetic Waves and Applications*, 23(5-6):785–798, 2009.

[121] M. Soleimani and W. R. B. Lionheart. Absolute conductivity reconstruction in magnetic induction tomography using a nonlinear method. *IEEE Transactions on medical imaging*, 25(12):1521–1530, 2006.

[122] M. Soleimani, W. R. B. Lionheart, A. J. Peyton, X. Ma, and S. R. Higson. A three-dimensional inverse finite-element method applied to experimental eddy-current imaging data. *IEEE Transactions on Magnetics*, 42:1560–1567, 2006.

[123] Weapon Systems. Tokarev tt-33. `https://weaponsystems.net/system/653-Tokarev+TT-33`. Date accessed 20th October 2020.

[124] A. Taflove, S. C. Hagness, and M. Piket-May. Computational electromagnetics: the finite-difference time-domain method. *The Electrical Engineering Handbook*, 3, 2005.

[125] Carpenter Technology. Magnetic properties of stainless steels. `https://www.carpentertechnology.com/en/alloy-techzone/technical-information/technical-articles/magnetic-properties-of-stainless-steels`. Date Accessed 9th October 2020.

[126] International Campaign to Ban Landmines. *Landmine Monitor 2021*. 23rd edition, 2021.

[127] W. van Verre, T. Özdeğer, A. Gupta, F. J. W. Podd, and A. J. Peyton. Threat identification in humanitarian demining using machine learning and spectroscopic metal detection. In *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, pages 542–549. Springer, 2019.

[128] J. R. Wait. A conducting sphere in a time varying magnetic field. *Geophsics*, 16(4):666–672, 1951.

[129] Wikipedia. `https://en.wikipedia.org/wiki/Tang_(tools)`. Date Accessed 9th October 2020.

[130] B. A. Wilson. `MPT-Calculator` youtube channel. `https://www.youtube.com/channel/UCMIKA2RhBqBmEdcRVgKxLeA/featured`, 2021.

[131] B. A. Wilson and P. D. Ledger. Hidden security threat identification: A reduced order model for the rapid computation of object characterisations.

[132] B. A. Wilson and P. D. Ledger. Bawilson94/mpt-library: Version 1.0.0 mpt-library, May 2021.

[133] B. A. Wilson and P. D. Ledger. Efficient computation of the magnetic polarizability tensor spectral signature using pod. *International Journal for Numerical Methods in Engineering*, 2021. Accepted, DOI:10.1002/nme.6606.

[134] B. A. Wilson and P. D. Ledger. `MPT-Calculator`. `https://github.com/BAWilson94/MPT-Calculator`, 2021.

[135] B. A. Wilson, P. D. Ledger, and W. R. B. Lionheart. Identification of metallic objects using spectral magnetic polarizability tensor signatures: Object classification. *arXiv preprint arXiv:2110.06624*, (accepted) 2022.

[136] C. Woodfor. Metal detectors.

[137] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 82–90, 2016.

[138] D. Xiao, C. E. Heaney, L. Mottet, F. Fang, W. Lin, I. M. Navon, Y. Guo, O. K. Matar, A. G. Robins, and C. C. Pain. A reduced order model for turbulent flows in the urban environment using machine learning. *Building and Environment*, 148:323–337, 2019.

[139] Y. Yoo. Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches. *Knowledge-Based Systems*, 178:74–83, 2019.

[140] S. Zaglmayr. *High Order Finite Elements for Electromagnetic Field Computation*. PhD thesis, Johannes Kepler University Linz, 2006.

[141] Y. Zhao, W. Yin, C. Ktistis, D. Butterworth, and A. J. Peyton. Determining the electromagnetic polarizability tensors of metal objects during in-line scanning. *IEEE Transactions on Instrumentation and Measurement*, 65:1172–1181, 2016.

[142] M. Zolgharni, P. D. Ledger, D. W. Armitage, D. S. Holder, and H. Griffiths. Imaging cerebral haemorrhage with magnetic induction tomography: numerical modelling. *Physiological measurement*, 30(6):S187, 2009.