University of Glasgow

Liu, Yushi (2024) *Bayesian neural networks-based surrogate model-assisted evolutionary algorithms and their applications to microwave antenna design.* PhD thesis.

https://theses.gla.ac.uk/84608/

Copyright and moral rights for this work are retained by the author

A copy can be downloaded for personal non-commercial research or study, without prior permission or charge

This work cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author

The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given

Enlighten: Theses
https://theses.gla.ac.uk/
research-enlighten@glasgow.ac.uk

Bayesian Neural Networks-Based Surrogate Model-Assisted Evolutionary Algorithms and Their Applications to Microwave Antenna Design

Yushi Liu

SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

JAMES WATT SCHOOL OF ENGINEERING

COLLEGE OF SCIENCE AND ENGINEERING
UNIVERSITY OF GLASGOW

JUNE 2024

To my parents

Abstract

Gaussian process (GP) is a widely used machine learning model for optimising online surrogate model-assisted antenna design. Despite many successes, two improvements are essential for the GP-based antenna global optimisation methods. First, the GP model training costs when there are many design variables and specifications. Second is the convergence speed (i.e., the number of necessary electromagnetic (EM) simulations to obtain high-performance designs). In both aspects, the state-of-the-art GP-based methods show practical but undesirable performance, particularly for optimising modern antennas, which often have many design variables and specifications.

Therefore, a behavioural study of a potential surrogate model alternative, Bayesian neural network (BNN), which has yet to be paid attention to, is presented in this thesis. Through empirical studies, the properties of the BNNs, their co-work with prescreening methods, and their comparison with other machine learning model alternatives are investigated with a typical surrogate model-assisted evolutionary algorithm (SAEA) model management framework. The behaviour of BNNs regarding surrogate model prediction accuracy, the availability of prediction uncertainty estimation, and the training cost are demonstrated in the experiments, showing the potential of BNNs to be a competitive alternative for online surrogate model-assisted antenna design optimisation.

Thus, this thesis presents an upgraded antenna design optimisation method called self-adaptive Bayesian neural networks surrogate model-assisted differential evolution for antenna design exploration (SB-SADEA). The key innovations include (1) the introduction of the BNNs-based antenna surrogate modelling method into the research area, replacing widely used GP modelling, and (2) a bespoke self-adaptive lower confidence bound (LCB) method for antenna design landscape making use of the BNNs-based antenna surrogate model. A slotted monopole antenna (ultra-wideband, 40% area reduced), a 5G mm-wave antenna (20 design variables, 12 design specifications, four operating bands), a sub-6 GHz outdoor base station antenna (23 design variables, 18 design specifications including S-parameters, front-to-back ratio and half-power beam-

width) and a microstrip patch antenna (quasi-digitally coded, 62 design variables) are used to test the performance of SB-SADEA in the thesis. The results show considerable improvement in convergence speed and machine learning cost compared with the state-of-the-art GP-based antenna global optimisation methods.

Furthermore, the proposed BNN-based SAEA has been tested against global optimisation applications on a broader scope. A supercontinuum generation waveguide and a holistic radar signal processing and classification system (12 design variables, including data pre-processing and feature extraction parameters in binary, continuous and discrete forms) are used to test the performance of the proposed algorithm. The experiments show that the proposed algorithm is efficient in optimising not only antenna structures but also components, structures and systems in other domains. Moreover, the proposed algorithm is compatible with discrete and categorical design variables.

Contents

Abstract iii

Acknowledgements xi

Declaration xiii

Abbreviations xiv

1 Introduction 1

1.1 Introduction to electromagnetism and antennas 1

1.2 Classical antenna design methodologies: pros and cons 3

1.3 Antenna design exploration using optimisation techniques 4

1.3.1 Antenna design optimisation using EAs 5

1.3.2 Antenna design local optimisation 6

1.3.3 Multi-fidelity antenna optimisation 6

1.3.4 Domain knowledge-assisted antenna optimisation 7

1.4 Surrogate model assisted antenna design optimisation 8

1.4.1 Challenges and difficulties of the traditional antenna design optimisation methods 8

1.4.2 Opportunities of surrogate model-assisted design optimisation framework 9

1.5 The SADEA series 10

1.6 Challenges and opportunities in surrogate model assisted antenna design optimisation 11

1.6.1 Challenges and difficulties of the existing methods 11

1.6.2 Novelties and contributions 14

1.7 List of publications 16

2 Background knowledge 17

2.1 Global optimisation, local optimisation and antenna design optimisation 17

2.1.1 Optimisation fundamentals 18

2.1.2 Constrained optimisation 18

2.1.3 Antenna design optimisation 19

2.2 Evolutionary algorithms 19

2.2.1 The heuristics of evolutionary algorithms . . . 20
2.2.2 Differential evolution . . . 21
2.2.3 Genetic algorithm . . . 23
2.2.4 Particle swarm optimisation . . . 24
2.3 Surrogate modelling . . . 25
2.3.1 Gaussian process regression . . . 26
2.3.2 Radial basis function . . . 28
2.3.3 Artificial neural networks . . . 28
2.3.4 Multivariate polynomial regression . . . 30
2.4 Prescreening methods . . . 31
2.4.1 Expected improvement . . . 31
2.4.2 Probability of improvement . . . 32
2.4.3 Lower confidence bound . . . 33
2.5 Bayesian Optimisation . . . 34
2.6 Surrogate model-assisted evolutionary algorithms . . . 35
2.7 Surrogate model management . . . 36

3 Bayesian neural networks' behaviour in SAEAs 38
3.1 Motivation . . . 38
3.2 Bayesian neural networks . . . 40
3.3 Selection of the test problem set . . . 43
3.4 Characteristics of BNN prediction and its co-work with prescreening methods . . . 47
3.4.1 Characteristics of BNN prediction . . . 47
3.4.2 The characteristics of the BNN model working with prescreening methods . . . 49
3.5 Comparison of BNN, GP and the NNDO method in SAEA . . . 52
3.5.1 Comparison between BNN and GP models . . . 56
3.5.2 Comparison between BNN and the NNDO method . . . 57
3.5.3 Comparison discussion . . . 61

4 Bayesian neural network based self-adaptive surrogate model-assisted evolutionary algorithm for antenna design exploration 63
4.1 The algorithm framework . . . 64
4.2 Bayesian neural network as the surrogate model . . . 66
4.3 The self-adaptive lower confidence bound mechanism . . . 68
4.4 Parameter Settings . . . 70

5 Experimental results and verification 71
5.1 Case study 1: Compact UWB slotted monopole antenna optimisation . 73
5.1.1 Engineering background . . . 73

5.1.2 Optimisation results and discussion 75
5.2 Case study 2: Quadruple-band 5G mm-wave antenna optimisation . . . 77
5.2.1 Engineering background 77
5.2.2 Optimisation results and discussion 79
5.3 Case study 3: 5-G outdoor base station antenna optimisation 82
5.3.1 Engineering background 82
5.3.2 Optimisation results and discussion 86
5.4 Case study 4: Quasi-digitally coded microstrip patch antenna optimisation 87
5.4.1 Engineering background 87
5.4.2 Optimisation results and discussion 91

6 Applications of SB-SADEA in a wider scope **93**
6.1 Supercontinuum generation waveguide optimisation 94
6.1.1 Engineering background 94
6.1.2 Optimisation results and discussion 95
6.2 Radar signal processing and human activity recognition system optimisation 97
6.2.1 Engineering background 97
6.2.2 SVM as the classifier 99
6.2.3 Optimisation results and discussion: SVM as the classifier . . . 100
6.2.4 AlexNet as the classifier 100
6.2.5 Optimisation results and discussion: AlexNet as the classifier . . 101

7 Conclusions and future work **103**

Appendices **105**
A Test problems 105
A.1 F1: 10-D Ackley problem 105
A.2 F2: 10-D Griewank problem 105
A.3 F3: Circular antenna array optimization 106
A.4 F4: 12-resonator diplexer coupling matrix optimization 106

Bibliography **108**

List of Tables

3.1 The selected test problems 44
3.2 Performance of EI, PI and LCB when co-working with BNN 50
3.3 Comparison between GP and BNN as surrogate models 53
3.4 Total training and simulation time of GP and BNN for the test problems . 55
3.5 Comparison between ANN with drop-out and BNN as surrogate models . . 57
3.6 Total training and simulation time of NNDO and BNN for the test problems 59

5.1 Search ranges of the design variables and the optimal design by SB-SADEA (All sizes in mm) 74
5.2 Design specifications and the performance of a typical optimal design obtained by SB-SADEA 74
5.3 Number of EM simulations (average number) used to satisfy the specifications for different methods 74
5.4 Search ranges of the design variables and a typical optimal design obtained by SB-SADEA (All sizes in mm) 79
5.5 Design specifications and the performance of a typical optimal design obtained by SB-SADEA 80
5.6 Comparison between SB-SADEA and TR-SADEA (average values) 81
5.7 Search ranges of the design variables and the optimal design by SB-SADEA (all sizes in mm) 84
5.8 Design specifications and the performance of an optimal design 85
5.9 Search ranges of the design variables and a known optimal design obtained by SB-SADEA (All sizes in mm) 89

6.1 Optimised parameters: SVM as the classifier 101
6.2 Optimised parameters: AlexNet as the classifier 102

A.1 Search ranges and known optimum of the coupling coefficients 107

List of Figures

1.1 An illustration of a simple patch antenna design. 3
1.2 A flowchart of antenna design by evolutionary algorithms. 5

2.1 Flow diagram of differential evolution. 21
2.2 Illustration of an artificial neural network. 29

3.1 An illustrative figure of a basic BNN. 40
3.2 Limitations and usefulness of the Ackley function (20 runs, only successful runs are drawn). 44
3.3 GP and BNN predicted values and prediction uncertainty comparison using the F1 test problem. 48
3.4 Performance comparisons among LCB, EI and PI working with BNN (average of 20 runs, and only successful runs are drawn. All success rates are higher than 90%.) 51
3.5 Performance comparison between BNN and GP (Only the average of successful runs are drawn. All success rates are higher than 90%.) 54
3.6 Performance comparison between BNN and NNDO (Only the average of successful runs are drawn. All success rates are higher than 90%.) 58
3.7 NNDO and BNN predicted values and prediction uncertainty comparison using the F1 test problem. 60

4.1 Flow diagram of the SMAS framework 64
4.2 GP and BNN predicted values and prediction uncertainty during early, middle and late stage of the optimisation (ground truth is from EM simulations). 67

5.1 Layout of the compact UWB slotted monopole antenna. 73
5.2 Convergence trends of slotted monopole antenna optimisation using SB-SADEA. 75
5.3 Response of the optimal design obtained by SB-SADEA 76
5.4 The layout of the quadruple-band 5G mm-wave antenna. 78
5.5 Convergence trends of SB-SADEA (5 runs). 81
5.6 Responses of the optimal design obtained by SB-SADEA 82

5.7 5-G outdoor base station antenna illustration. This figure is modified from (Liu et al. 2021). 83
5.8 Convergence trends of SB-SADEA (six runs). 87
5.9 Responses of the optimal design obtained by SB-SADEA. 88
5.10 An illustrative layout of the quasi-digitally coded microstrip patch antenna. This figure is a modified screenshot from CST-MWS. 90
5.11 Convergence trends of qDMPA optimisation using SB-SADEA. 91
5.12 S-parameter response of the optimal design obtained by SB-SADEA. . . . 92

6.1 The structure of the waveguide. Screenshot of Ansys Lumerical MODE. . . 95
6.2 Supercontinuum generation waveguides optimisation using parametric sweep. This figure is modified from (McKay 2024) 96
6.3 The convergence trend of the supercontinuum generation waveguides optimisation. 96
6.4 Holistic HAR optimisation from signal processing to classification using SADEA. This figure is modified from (Li et al. 2024). 97
6.5 Convergence trend of the holistic HAR (with SVM as the classifier) optimisation using SADEA with a referenced result using manual design. 100
6.6 Convergence trend of the holistic HAR (with AlexNet as the classifier) optimisation using SADEA with a referenced result using manual design. . 101

Acknowledgements

This dissertation is the culmination of several years of research, hard work, and dedication. It would not have been possible without the support and guidance of many individuals, to whom I am deeply grateful.

First and foremost, I would like to express my deepest gratitude to my supervisors, Dr Bo Liu, Dr Masood Ur-Rehman and Dr Muhammad Imran. Their unwavering support, insightful feedback, and constant encouragement have been instrumental in the completion of this work. I am profoundly thankful for their patience and for constantly pushing me to achieve my best.

A special thanks to the funding bodies that made this research possible: MathWorks Inc. and the University of Glasgow. Your financial support provided the resources needed to pursue this project and allowed me to focus entirely on my research.

I also extend my sincere thanks to my collaborators, whose expertise and contributions have significantly enriched this dissertation. Special thanks to Dr Julien Le Kernec, Dr Zhenghui Li, Dr Marc Sorel and Elissa McKay for their invaluable input and for the productive discussions that greatly influenced my research. I am also grateful to my project managers and colleagues, Dr Rami Ghannam, Liyuan Xue, Tao Wu, Yijia Hao, Grant Quinn, Huihua Cheng, Dr Hans Senn and Dr Huabing Yin at the University of Glasgow, Dr Mobayode Akinsolu at Wrexham University, Dr Qiang Hua at the University of Huddersfield, Dr Timothy Storer, Talha Enes Ayranci and other colleagues at Glasgow University Software Service, Dr Vishwanath Iyer, Vinod Dwarapudi, AJ Ibraheem, Dr George Amarantidis and other colleagues at MathWorks and Zhengqi Gao at MIT. Your valuable feedback, rigorous discussions, collaborative spirit, and shared passion for our field have significantly enriched this journey.

My friends have been an essential source of support and motivation throughout this journey. I am particularly grateful to them for their camaraderie and understanding and for always being there to offer a listening ear or a word of encouragement when needed.

To my family, I owe a debt of gratitude for their unwavering belief in me. My parents have been my pillars of strength, offering endless love and encouragement.

Lastly, I would like to thank my girlfriend, Yiqi Zhuang. Your love, patience, and unwavering belief in me have been a constant source of inspiration. Your support has been invaluable, and I am deeply grateful for your understanding and encouragement through the highs and lows of this journey.

This dissertation is a testament to the collective support of all these remarkable individuals. I am truly appreciative of having each and every one of you in my life. Thank you.

Declaration

I declare that, except where explicit reference is made to the contribution of others, that this dissertation is the result of my own work and has not been submitted for any other degree at the University of Glasgow or any other institution.

Yushi Liu

Abbreviations

- 5G - Fifth-generation (technology standard for cellular networks)
- ABE - Active base element
- ANN - Artificial neural network
- BNN - Bayesian neural network
- BW - Bandwidth
- CMA-ES - Covariance matrix adaptation evolution strategy
- CST-MWS - CST Microwave Studio
- DE - Differential evolution
- EA - Evolutionary algorithm
- EC - Evolutionary computation
- EI - Expected improvement
- ELBO - Evidence lower bound
- EM - Electromagnetic
- FEA - Finite element analysis
- GA - Genetic algorithm
- GAN - Generative adversarial network
- GC - Generation control
- GNSS - Global navigation satellite system
- GP - Gaussian process
- GVD - Group velocity dispersion
- HAR - Human activity recognition
- IC - Integrated circuit
- KL divergence - Kullback–Leibler divergence
- KNN - K-Nearest neighbour
- LCB - Lower confidence bound
- MCMC - Markov chain Monte Carlo
- MIMO - Multiple input, multiple output
- MLP - Multi-layer perceptron
- mm-wave - Millimetre wave
- NN - Neural network

- NNDO - Neural network with dropout
- PDE - Partial differential equation
- PI - Probability of improvement
- P-SADEA - Parallel surrogate model assisted differential evolution for antenna design exploration (the third generation)
- PSO - Particle swarm optimisation
- qDMPA - Quasi-digitally coded microstrip patch antenna
- RBF - Radial basis function
- RF - Radio frequency
- SA - Simulated annealing
- SADEA - Surrogate model assisted differential evolution for antenna design exploration
- SADEA-II - Surrogate model assisted differential evolution for antenna design exploration, the second generation
- SAEA - Surrogate model assisted evolutionary algorithm
- SBO - Surrogate model-based optimisation
- SB-SADEA - Self-adaptive Bayesian neural networks surrogate model assisted differential evolution for antenna design exploration (the fifth generation)
- SMAS - Surrogate model-aware evolutionary search
- SQP - Sequential quadratic programming
- SVC - Support vector classifier
- SVM - Support vector machine
- TRS - Trust-region search
- TR-SADEA - Training cost reduced surrogate model assisted differential evolution for antenna design exploration (the fourth generation)
- UWB - Ultra-wide band

Chapter 1

Introduction

In this chapter, introductions to antenna optimisation are broken into several sections. Section 1.1 explains what an antenna is, what it does and how it does it. Section 1.2 introduces traditional antenna design methodologies, some off-the-shelf global optimisers that can be used in antenna design optimisation, and their advantages and disadvantages. Section 1.3 reviews antenna design exploration techniques in the past. Section 1.4 introduces surrogate model-assisted antenna design explorations methods. Section 1.5 reviews the SADEA series of algorithms, upon which the major contribution of this dissertation is built. At last, Section 1.6 introduces challenges and opportunities provided by the current development of the antenna design optimisation algorithms.

1.1 Introduction to electromagnetism and antennas

An antenna is a fundamental component of radio frequency (RF) and telecommunications engineering. It serves as a pivotal physical interface between electromagnetic waves and electronic systems. Its main functionality is transmitting and receiving electromagnetic (EM) signals, converting them into electrical current or vice versa (Wolff 1966). Typically, the performance of EM systems can be analysed by their EM field characteristics. EM fields are generated by time-varying electric and magnetic fields produced by charged particles in motion. When these fields change over time, they create electromagnetic waves propagating through space (Maxwell 1864, 1865; Hammond 1954). The classical equations of EM can be mathematically expressed in integral and differential forms and thus can be solved in the time domain and frequency domain (Hammond 1954).

For a given electric charge, the EM fields can be described in four laws, namely, Gauss's Law, which describes the relationship between electric fields and electric charges; Gauss's Law for Magnetism, which states the absence of magnetic charge; Faraday's Law of Induction, which describes how changing magnetic fields induce electromotive force and electric fields, and Ampere's Circuital Law (with the addition of the displacement current), which relates magnetic fields and electric currents and the rate of change of electric fields, and are expressed mathematically in equations as (Maxwell 1864, 1865)

$$\begin{cases} \overrightarrow{\nabla} \cdot \overrightarrow{E} = \dfrac{\rho}{\epsilon_0} & \text{Gauss's Law} \\ \overrightarrow{\nabla} \cdot \overrightarrow{B} = 0 & \text{Gauss's Law for Magnetism} \\ \overrightarrow{\nabla} \times \overrightarrow{E} = -\dfrac{\partial \overrightarrow{B}}{\partial t} & \text{Faraday's Law of Induction} \\ \overrightarrow{\nabla} \times \overrightarrow{B} = \mu_0(\epsilon_0 \dfrac{\partial \overrightarrow{E}}{\partial t} + \overrightarrow{J}) & \text{Ampere's Circuital Law} \end{cases} \tag{1.1}$$

where $\overrightarrow{E}$ is the electric field, $\overrightarrow{B}$ is the magnetic field, ρ is the electric charge density and $\overrightarrow{J}$ the current density. ϵ_0 is the vacuum permittivity and μ_0 the vacuum permeability.

(1.1) is referred to as Maxwell's equations, which play a fundamental role in the analysis and design of antennas. Antennas are physical devices that radiate or receive EM waves, and their behaviours are governed by the four laws outlined in Maxwell's equations. In essence, Maxwell's equations provide theoretical support for the underlying operations of antennas.

Besides RF engineering and telecommunication engineering, antennas also play an essential role in wireless communications, such as satellite communication, radar systems, broadcasting and more. The design and characteristics of antennas vary based on their applications. Antennas can have different sizes, geometries, materials and configurations and thus have different directional properties, radiation patterns, operational frequency bands, etc. (King et al. 1981). With modern technological advancements, such as 5G and the internet of things, antennas continue to evolve to the more complex, allowing improved signal propagation efficiency, reliability, connectivity, versatility, adaptability and compatibility in modern communication networks (Gao et al. 2009; Balanis 2011; Lim and Leung 2012; Fujimoto and Morishita 2013).

1.2 Classical antenna design methodologies: pros and cons

Traditionally, antenna design methods rely on parameter studies and manual design experience, such as rules of thumb (Blank 1990; Lytle and Laine 1978), for practical and intuitive antenna behaviours as for their well-established theoretical foundation. The physical dimensions are tuned to meet desired design specifications and often take real-world factors into account (Blank and Hutt 2005). Such methodologies are available for obtaining initial designs for simple antennas and can be helpful for quick prototyping. For example, to model a simple microwave patch antenna with a waveguide port by rules of thumb, the width of the waveguide should be empirically from 6 to 10 times the width of the micro-strip feed line as illustrated in Figure 1.1 (Weiland et al. 2008)

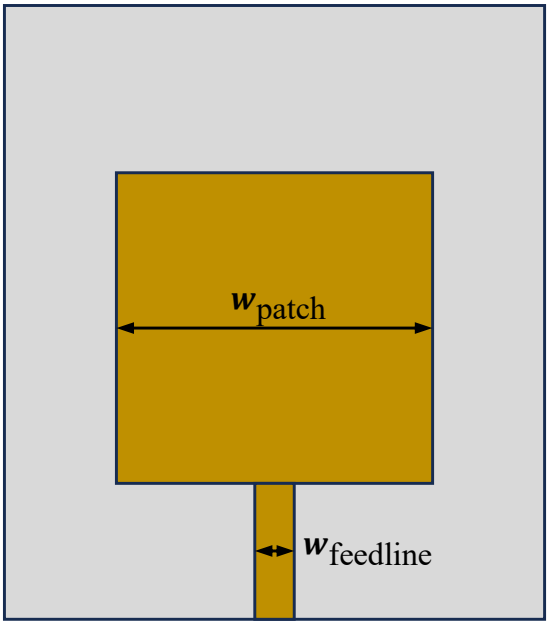


Figure 1.1: An illustration of a simple patch antenna design.

Moreover, simple methods like parametric sweep can be used to search for an initial design for antenna design cases with few design parameters. Nevertheless, this is not usually the case for most modern antenna designs, which usually have many design parameters and complex interdependencies. Traditional approaches are mostly considered trial-and-error practices and have limited applicability on complex or unconventional modern antenna design structures, thus not guaranteeing the success of the antenna design in these cases.

Besides, optimisation techniques are proven efficient or available in other optimisation problems. These off-the-shelf methods, however, are not often suitable for antenna optimisations. Most antennas involve complex EM interactions, and non-linear, multimodal and discontinuous behaviours usually characterise the design spaces. Furthermore, unlike most function evaluations with negligible computational cost, the evaluation of antennas requires EM simulations, which involve partial differential equation (PDE) solvers and are computationally expensive. Therefore, off-the-shelf optimisation methods are not often capable of efficiently handling antenna optimisation problems (Werner et al. 2014).

1.3 Antenna design exploration using optimisation techniques

Today, there are many CAD/CEM platforms available, such as ANSYS EDT/HFSS (ANSYS Accessed 2024), CST Microwave Studio (CST-MWS) (CST Accessed 2024) and MATLAB Antenna Toolbox (MathWorks Accessed 2024), etc. The optimisation methods can be categorised as follows,

- **Standard evolutionary algorithms (EAs)**, such as particle swarm optimisation (PSO), differential evolution (DE), genetic algorithm (GA) and simulated annealing (SA). These methods are classical, statistically grounded and usually decades old (Storn and Price 1997; Kennedy and Eberhart 1995; Kirkpatrick et al. 1983; Holland 1984; Lazaridis et al. 2016; Zaharis et al. 2017). Covariance matrix adaptation evolution strategy (CMA-ES) and SHERPA are two revised EAs and are relatively recent (Gregory et al. 2011; Tech 2008).
- **Standard local optimisation**, such as classical Powell and grid search. These methods usually have limited exploration capability (Powell 1964; Lewis et al. 2000; Powell 2007; Bergstra and Bengio 2012; Huang et al. 2012).
- **Derivative-based local search**, such as quasi-Newton method and sequential quadratic programming (SQP) (Dennis and Moré 1977; Boggs and Tolle 1995; Gill and Wong 2011; Koziel and Pietrenko-Dabrowska 2019b; Pietrenko-Dabrowska and Koziel 2020b).
- **Derivative-free local search**, such as Nelder-Mead Simplex Algorithm and Pattern Search (Lewis et al. 2000; Powell 2007; Singer and Nelder 2009; Audet and Dennis Jr 2002).

- **Surrogate model-based local optimisation**, such as trust region method and interpolated quasi-Newton method (Dennis and Moré 1977; Yuan 2000; Koziel and Unnsteinsson 2018).
- **Surrogate model-based global optimisation**, the SADEA series (Liu et al. 2013, 2017b; Akinsolu et al. 2019; Liu et al. 2021; Liu et al. 2022a) discussed in Section 1.5, and other surrogate model-based global optimisation (Wu et al. 2020; Koziel et al. 2021, 2014; Zhou et al. 2020).

1.3.1 Antenna design optimisation using EAs

The history of antenna design automation can be traced back to mid-1990s when most of practitioners utilises GAs to optimise their EM devices and the optimised antennas are then called evolved antenna (Michielssen et al. 1993; Altshuler and Linden 1997; Linden and Altshuler 1996; Haupt 1995). From then on, more and more antenna designs are driven by GAs (Hornby et al. 2006; Ares-Pena et al. 1999; Marcano and Durán 2000), and by other similar automatic optimisation approaches including PSO (Khodier and Christodoulou 2005; Liu 2005) and DE (Kurup et al. 2003; Goudos et al. 2011). Figure 1.2 shows a general workflow of antenna design optimisation by evolutionary algorithms. The optimisation starts with initial set-ups, usually samplings and pre-processing, and loops over specific search mechanisms until a satisfied design is found and the design specifications are met (Simon 2013).

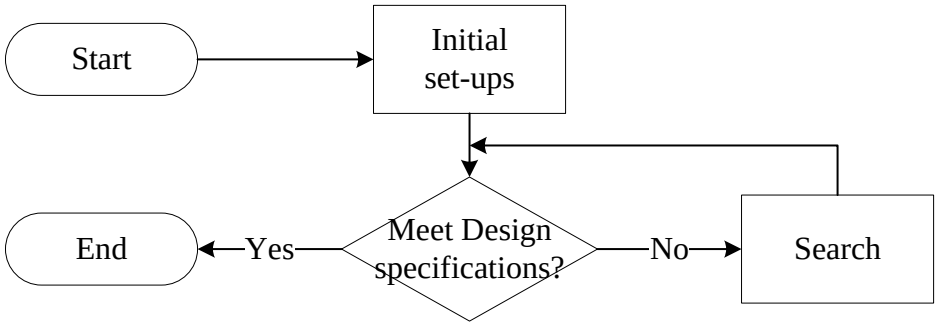


Figure 1.2: A flowchart of antenna design by evolutionary algorithms.

Antenna design exploration usually involves from a few (e.g. conventional microwave antennas) to multiple dozens (e.g. 5G base station antennas) of parameters (Liu et al. 2014e, 2017d,c; Wen et al. 2017). According to the nature of the antenna structure, the parameters can be discrete and continuous. Sometimes, the design parameters have geometric constraints or interdependencies in between them. Typically, antenna

design optimisation problems search for design values that reach the global minimum or maximum for one or more design specifications, as demonstrated in Figure 1.2. This can be done by using local or global optimisation algorithms. Global optimisation algorithms, particularly EAs, are frequently used in antenna optimisation problems (Hoorfar 2007; Lake et al. 2013; Deb et al. 2017).

1.3.2 Antenna design local optimisation

Local optimisation of antenna design is essential for refining antenna designs and improving performance characteristics. Given an initial antenna design obtained by theoretical calculations, previous designs or global optimisation, the local design algorithm tunes the antenna structure to achieve better performance. Antenna design local optimisation algorithms are usually built based on numerical optimisation methods such as trust region-based optimisation algorithms (Koziel et al. 2018; Koziel and Pietrenko-Dabrowska 2019b; Pietrenko-Dabrowska and Koziel 2020b), SQP algorithm (Li et al. 1997), and Quasi-Newton method (Chakraborty et al. 2023).

Most antenna design optimisation algorithms do not take process variations or fabrication errors into account. However, discrepancies are likely between the nominal optimised antenna structure and the actually fabricated antenna structure. Process variation-aware or yield-driven optimisation is usually used as a final step for a robust antenna design and to ensure a complete design closure. The variation-aware optimisation aims to maximise the yield or the probability that the fabricated antennas meet the pre-set antenna design specifications, given an assumed probability distribution of fabrication error around its nominal optimal design. For such purpose, antenna design local optimisation algorithms are usually applied (Swidzinski and Chang 2000; Koziel and Bekasiewicz 2018; Koziel and Pietrenko-Dabrowska 2019a; Pietrenko-Dabrowska et al. 2020; Zhang et al. 2022).

1.3.3 Multi-fidelity antenna optimisation

The key idea behind multi-fidelity antenna optimisation is to evaluate "non-promising" antenna designs using rough but computationally cheap low-fidelity models while performing accurate and high-fidelity searches around "promising" antenna designs suggested by low-fidelity models. The computationally cheap low-fidelity models include vari-

ous surrogate models and EM model (Koziel and Bekasiewicz 2016; Song et al. 2019). Moreover, the multi-fidelity optimisation is applied to antenna design optimisation problem (Chen et al. 2022b). The low-fidelity antenna evaluation is validated with high-fidelity EM simulation during the antenna optimisation. Furthermore, multi-fidelity can be employed dynamically as variable-fidelity models of a given antenna structure during optimisation, which further improves the surrogate modelling and overall optimisation efficiency (Pietrenko-Dabrowska and Koziel 2020a; Pietrenko-Dabrowska et al. 2022). At the end of the optimisation, co-Kriging is applied to combine low- and high-fidelity simulation results to manage the model discrepancies, which helps to eliminate the low-fidelity model correction process (Pietrenko-Dabrowska et al. 2022). The second generation of the SADEA series combines surrogate model-assisted low-fidelity global search, and local search is another multi-fidelity optimisation method that can efficiently handle fidelity discrepancies (Liu et al. 2017b), and this is discussed later in Section 1.5.

1.3.4 Domain knowledge-assisted antenna optimisation

Case-by-case antenna design knowledge can be applied to antenna design optimisation to reduce the computational cost during the optimisation. For instance, the knowledge of active base elements (ABEs) of antenna arrays and their patterns can be applied with the Gaussian process (GP) predictions of the ABE geometries and their corresponding excitation of the sub-arrays and helps the antenna array design exploration (Wu et al. 2022). For a low-cost antenna and array design with robustness, analyses including worst-case analysis, maximum input tolerance hypervolume search mechanism and robust optimisation are applied to guide the optimisation process, and the proposed method is applied in a multi-objective microstrip patch antenna optimisation study (Wu et al. 2021). These analyses rely on the specific domain knowledge about the design sample in the design space and the output tolerance region. Such domain knowledge enables the identification and application of the appropriate optimisation techniques for given antenna design challenges, thereby ensuring the development of robust, efficient, and cost-effective antenna systems.

1.4 Surrogate model assisted antenna design optimisation

1.4.1 Challenges and difficulties of the traditional antenna design optimisation methods

Recently, antennas' physical designs have become increasingly complex, the design specifications are becoming increasingly stringent and challenging, and relations between the physical designs and their radiation patterns are more and more unpredictable. On the other hand, the engineering design flow of such developing antenna structures are merely changed or upgraded accordingly. Typically, a top-down flow, which divides the design task into several building blocks and allocates design specifications to them, is still widely and frequently used (Crepaldi et al. 2014). Such traditional design approaches require much effort from the antenna designers and can cause significant delays from product conceptualisation to antenna fabrication. With the increase in complexity of antenna design structures and design specifications and stringent time-to-market requirement, such traditional trial-and-error design methods are gradually replaced by advanced intelligent methods (Arora et al. 1995; Floudas and Gounaris 2009; Rios and Sahinidis 2013), like introduced in Subsection 1.3. Amongst these methods, PSO and DE are the most suitable approaches for their outstanding searching capability in global optimisation and not requiring initial solutions (Lake et al. 2013; Kennedy 2011; Storn and Price 1997; Zaharis et al. 2017; Arora et al. 1995; Fan et al. 2008; Rocca et al. 2011). However, even though taking the advantages of the capability of the search engine (DE and PSO), to obtain optimal antenna design with reasonable time consumption remains challenging. EM simulations are required to evaluate the performance of antennas accurately. The EM simulations involve PDE solvers, so they are characteristically computationally expensive (Keyes et al. 2013). Therefore, canonical DE and PSO, which often require many function evaluations to search for the optimal or near-optimal (John and Ammann 2009), are not ideal methods for antenna design optimisation.

1.4.2 Opportunities of surrogate model-assisted design optimisation framework

To solve the problems stated in Subsection 1.4.1, based on a surrogate model-assisted evolutionary algorithm (SAEA), surrogate model-based optimisation (SBO) (Liu et al. 2017c; Hawe and Sykulski 2008; Couckuyt et al. 2010) was adopted. Surrogate models are often statistical models or simple machine learning models that are relatively computationally cheap to build and use, simultaneously offering satisfying results. SBO methods build surrogate models to replace computationally expensive exact function evaluations, EM simulation in this case. Various techniques, such as neural networks, support vector machine, Gaussian process, radial basis function and polynomial regression, are proposed to build the surrogate models (Myers and Myers 1990; Jin 2005; Ong et al. 2008; Giunta and Watson 1998; Giannakoglou 2002; Emmerich et al. 2006). Amongst these methods, Gaussian process (or Kriging) and RBF are outperforming others as surrogate models (Jin et al. 2001; Giunta and Watson 1998).

With the help of the computationally cheap surrogate models, most computationally expensive EM simulations are avoided. Therefore, the quality of the prediction results is essential in this case. However, imaginably, the prediction results from the surrogate models are not usually accurate as the surrogate models are approximated models and, therefore, uncertain. Furthermore, the search should be guided to highlight the potential for improvement. Thus, the capability to handle uncertainty is needed in SBO. Several techniques are available to handle uncertainty in SBO, such as probability of improvement, expected improvement, infill sampling and lower confidence bound (Parr et al. 2012; Zhan and Xing 2020; Budak et al. 2021).

For most surrogate models, the predicted value of a given point is primarily determined by the true values of its surrounding points. Moreover, if the given point has dense surround points and their true values are close together, the model uncertainty, or the potential in this context, of the prediction of the given point should be low. On the other hand, the fact that a given point has a few points next to it logically implies a significant potential. This is discussed in detail and analysed numerically in Section 3.4.

Recently, some novel optimisation algorithms have been proposed. For instance, inspired by generative adversarial networks (GANs), a novel generative algorithm is proposed with the use of support vector classifier (SVC) in a unified evolutionary optimisation framework for antenna design automation using broadband antenna and dual resonance as examples (Zhong et al. 2022). The trained generator and discriminator from the GAN, along with the SVC, work together to predict the performance of antennas. Another study validates accelerated gradient-based optimisation with response feature methodology and numerical derivatives, with dual-band and tri-band microstrip patch antenna optimisations (Pietrenko-Dabrowska and Koziel 2020b, 2023).

1.5 The SADEA series

The initial surrogate model-assisted differential evolution for antenna design exploration (SADEA) was proposed in 2013 and was then one of the state-of-the-art antenna optimisation algorithms (Liu et al. 2014c). The algorithm utilises GP regression as the surrogate model to estimate the antenna performances of the candidate designs, saving many computationally expensive EM simulations. Moreover, the surrogate model is constructed considering that it should be trained so it can predict values for candidate designs in a global landscape, that it should be accurate enough to support the appropriate selection of candidate designs during the search, and that it should require as few EM simulations as possible. The initial SADEA sets the basis for the later upgrades.

Following the initial SADEA, its second generation, SADEA-II, was proposed in 2017 to amend the initial SADEA with a data mining phase handling multi-fidelity discrepancy between simulation models of different fidelities and with a surrogate model-assisted combined global and local search mechanism during the exploration (Liu et al. 2017b).

The third generation of SADEA, parallel SAEAs for electromagnetic design (PSAED, sometimes called parallel-SADEA, or P-SADEA), was proposed in 2019 to allow the GP model to be built for each candidate designs with its closest samples as training data for the more accurate model predictions (Akinsolu et al. 2019; Akinsolu 2019). PSAED also allows multiple DE mutation operators with different characteristics to be used adaptively to use diversity produced from multiple mutation operations.

And then proposed in 2020, the fourth generation of SADEA, training cost reduced SADEA (TR-SADEA), which trains GP models for clustered candidate designs instead of for every candidate design, saving much training time, particularly for optimisation of antennas with many design parameters (Liu et al. 2021). TR-SADEA also utilises the radial basis function (RBF)-based surrogate model to assist the search adaptively during the early exploration stage, which speeds up the optimisation.

1.6 Challenges and opportunities in surrogate model assisted antenna design optimisation

1.6.1 Challenges and difficulties of the existing methods

Like mentioned, the design specifications of modern antennas are becoming increasingly stringent and compact antennas are always preferable in many context. The recent development of 5G require many antenna systems to work with (Khan et al. 2018). There are also wearable devices and body-centric wireless communication requires bespoke antenna designs (Paracha et al. 2019; Danjuma et al. 2020). What is more, the topologically irregular structures are getting more attention as driven and parasitic structural elements for certain antennas such as the crossed-dipole antenna with spline-shaped patches for navigation satellite system (GNSS), triple band and quadruple band applications (Hussine et al. 2017; Alieldin et al. 2018).

Optimisation of the antenna structure with increased complexness requires advanced and intelligent optimisation algorithms accordingly. The latest SADEA, TR-SADEA, firstly addresses the challenge of increased model training cost by adopting a GP model sharing mechanism to reduce the necessary number of GP model training and a self-adaptive surrogate model-assisted local optimisation technique to improve the convergence speed, particularly during the early stage of the optimisation. Moreover, it successfully optimises complex base station antennas (Liu et al. 2021). However, imaginably, training a model to predict a cluster of points decreases the accuracy of the prediction. Furthermore, for antennas that have multiple tens of design parameters, to train such very high dimensional GP models is still computationally expensive due to the “curse of dimensionality” (Rasmussen and Williams 2006), even with the model sharing mechanism that decreases the number of GP model training.

In the evolution of the SADEA series, as discussed in Section 1.5, the main innovations lie in the model management methods and search operators. Regarding the surrogate model, not only the SADEA series but also other surrogate model-based optimisation methods of a similar kind utilise GP as their surrogate model (Wu et al. 2020; Koziel et al. 2021, 2014; Zhou et al. 2020). An accurate machine learning model is essential to be the surrogate for the EM optimisation landscape, which is often multi-modal. GP stands out from the regular machine learning models for its learning ability, requiring few empirical parameters and providing statistically ground uncertainty quantification.

However, GP has drawbacks. The training cost of GP grows cubically concerning the number of training samples, which is highly dependent on the number of design variables (Rasmussen and Williams 2006). Moreover, because GP predicts individual design specification values separated, the number of GP models required, and hence the training time, grows linearly concerning the number of design specifications. Considering antenna optimisation problems with a few design variables and design specifications, GP training time is short. However, for antenna optimisation problems with more than 20 design variables and multiple design specifications, TR-SADEA uses one to two days for GP model training (approximately 90% of the training time reduction compared with standard SADEA). Such training cost is practical but not desirable (Liu et al. 2021).

If GP is not always ideal as the surrogate model under the background of SAEAs, quantitative assessments for its alternatives are needed to find the better ones. The following three key factors should be considered when evaluating surrogate model alternatives: (1) surrogate model prediction accuracy, (2) the prediction uncertainty quantification, and (3) the training cost, particularly when the number of decision variables becomes large. The surrogate model prediction accuracy is directly affecting convergence speed. A highly accurate surrogate model assists the EA search engine in the efficient evolutionary search. Utilisation of the prediction uncertainty quantification of individual solutions (instead of the overall approximation error of the surrogate model) is vital for SAEAs not to converge into local optima (Cai et al. 2019). Also, prescreening methods (infill criteria or acquisition functions) are proposed (Jones et al. 1998; Emmerich et al. 2006), using the prediction uncertainty quantification to promote global exploration. Regarding the training cost, for small-scale optimisation problems, the machine learning cost is often low and negligible for most machine learning models. Nevertheless, the training costs become challenging for some machine learning models when the number of decision variables grows to several tens or even more. Sometimes, it may be even longer than the real function evaluation itself (Liu et al. 2021) and is critical for SAEAs.

Popular and widely used machine learning that can be considered as surrogate model alternatives under SAEAs background and Bayesian optimisation include the Gaussian process (GP), radial basis function (RBF), artificial neural networks (ANNs), and ensemble methods. They are reviewed and discussed below, considering the above three factors.

- **GP** (Rasmussen and Williams 2006): GP, sometimes called Kriging, is a statistically sound machine learning method. It has very few empirical parameters and is usually an accurate model. Moreover, a statistical prediction uncertainty quantification for each candidate solution (i.e., variances) is provided. Recently, student t-process (Shah et al. 2013) is proposed, improving the covariance and performing better than Kriging in small-scale Bayesian optimisation. Non-stationary GP is reviewed and investigated under the background of SAEAs (Hu et al. 2021), showing competitive performance. The major drawback of GP is the training time complexity under SAEAs (Guo et al. 2018). The computational time complexity of GP modelling is $O(N_G n^3 d)$ (Emmerich et al. 2006), where N_G is the number of iterations spent in model parameter optimisation, n is the number of the training sample, and d is the number of decision variables. n is dependent on d to construct reliable GP models. The computational cost of GP modelling can be high, particularly when the number of decision variables grows to several tens or more (Liu et al. 2021). Alternative methods, including sparse GP (Bauer et al. 2016), simplified GP (Fu et al. 2022), and training samples sharing method in model management (Liu et al. 2021) are proposed to reduce the cost of surrogate modelling. The drawback of these GP variants is that the prediction accuracy is generally compromised.
- **RBF**: Compared to GP, the training cost of RBF is much lower even for high-dimensional problems (Powell 1992). However, the major drawback of RBF is that it does not provide statistical prediction uncertainty quantification for each candidate solution. Prescreening methods, assisting SAEAs in converging to global optima, are therefore difficult to use. Although there are successful RBF-based SAEAs (Chen et al. 2022a), many SAEA research works from both the EA domain (Kudela and Matousek 2023; Cai et al. 2019; Liu et al. 2022b; Wang et al. 2022) and engineering optimisation domain (Liu et al. 2021) show that combining RBF-assisted search with GP-assisted search or other machine learning alternatives-assisted ones is needed for global optimisation, particularly for complex and real-world engineering design problems. New RBF-related methods that provide prediction uncertainty quantification are investigated (Qin et al. 2021).

- **ANNs**: Standard multi-layer perceptron (MLP) ANN is not widely used in SAEAs as many empirical parameters need to be tuned. RBF neural networks (RBFNN), on the other hand, are used in SAEAs (Guo et al. 2018) as it has much fewer empirical parameters to be tuned and more stable predicted values. However, the RBFNN can also not provide the prediction uncertainty quantification for each candidate solution. One way to generate prediction uncertainty for each candidate solution when ANN is the surrogate model is the dropout method (Srivastava et al. 2014). The original purpose of the dropout method is to prevent overfitting and to improve generalisation error for deep networks by randomly dropping out nodes from the neural network during training. Dropout can be treated as an ensemble of multiple ANNs, and the variance among multiple output prediction results can naturally be obtained. (Gal and Ghahramani 2016) first introduces the method to Bayesian optimisation to allow these ANN-like models to provide uncertainty quantification and attract much attention. (Guo et al. 2021) applies the dropout method to SAEAs. On the other hand, criticism also appears for the rigour of the dropout method as the statistical model uncertainty is not necessarily the variance obtained by dropping out nodes from the neural network, even though it does generate uncertainty estimation (Folgoc et al. 2021).
- Ensemble methods: Two kinds of ensemble methods are usually used in SAEAs as the surrogate models. The first kind uses multiple machine learning alternatives for different purposes (Cai et al. 2019). For example, (Wang et al. 2022) uses the global RBF model to estimate the search trend for global optimisation, while local GP models provide the exploration ability. The other kind uses ensemble learning techniques, e.g., bootstrap, to improve the prediction accuracy and to generate "prediction uncertainty" (the variance between predicted values of multiple learners in this case) (Guo et al. 2018; Wang et al. 2018). Successful ensemble methods show comparable performance to the GP model regarding prediction accuracy but with much less training cost (Guo et al. 2018). In a noteworthy manner, The design of an ensemble method is often ad hoc.

1.6.2 Novelties and contributions

Generally speaking, the long GP training time affects the overall convergence speed. This research aims to seek a new machine learning core that can provide what GP provides but with less computational expenses to replace GP in SADEA and decrease the overall time. Due to this, this research is focused on seeking a different machine learning surrogate core to replace the GP-based surrogate modelling and introduce

a method using the new surrogate model. The goal is to largely improve both the convergence speed (i.e., the number of EM simulations needed to obtain the optimal design) and the training cost of surrogate modelling and providing a unified method for antennas with various design variables and specifications.

A natural question is whether there exists a machine learning alternative that has high prediction accuracy, statistically grounded prediction uncertainty estimation, and reasonable training costs for SAEAs. To answer the question, this thesis investigates Bayesian neural networks (BNNs) (Goan and Fookes 2020) under the background of SAEAs. Till now, using BNN in SAEAs has yet to be paid much attention to. To my knowledge, the first SAEAs built upon BNN are in (Briffoteaux et al. 2020), the pioneering work of this area. The dropout method is used in BNN, and the expected improvement (EI) prescreening method is used. (Liu et al. 2022a) employs BNN without dropout. A new self-adaptive lower confidence bound (LCB) prescreening method shows convergence in only half the number of iterations needed than that of GP under the same SAEAs framework for complex real-world antenna optimisation problems with up to 20 design variables. (Gao et al. 2019) employs BNN in multi-objective SAEAs and shows an effective result for a real-world problem. Hence, a behavioural study is presented for the BNN as a potential machine learning core, and a new method, called the self-adaptive Bayesian neural network surrogate model-assisted DE for antenna design exploration (SB-SADEA), is proposed.

The critical contribution of the research includes (1) a behavioural study of BNN as a surrogate model under SAEAs, (2) the introduction to Bayesian neural networks (BNNs)-based antenna surrogate modelling method into the area of surrogate model-based antenna evolutionary optimisation as a substitution of GP and (3) the introduction to a bespoke self-adaptive lower confidence bound (LCB) prescreening method for antenna design landscape making use of the BNNs prediction results.

Through the experiments in Chapter 5, it can be seen that using BNNs as the machine learning core in SADEA and the self-adaptive LCB prescreening method as part of the model management requires less computation cost and, at the same time decrease the number of EM simulation required during the optimisation iterations. Through the experiments in wider fields in Chapter 6, it can be seen that the proposed BNNs-based SADEA has potential availability not only in antenna optimisation but also in wider fields.

1.7 List of publications

- **Yushi Liu**, Bo Liu, Masood Ur-Rehman, Muhammad Ali Imran, Mobayode O Akinsolu, Peter Excell and Qiang Hua, 'An efficient method for antenna design based on a self-adaptive Bayesian neural network-assisted global optimization technique', *IEEE Transactions on Antennas and Propagation* 70.12, pp. 11375-11388. December 2022.
- **Yushi Liu**, Muhammad Ali Imran, Masood Ur-Rehman and Bo Liu, 'Behavioral Study of Bayesian Neural Networks in Surrogate Model Assisted Evolutionary Algorithms', submitted to *IEEE Access*
- Hsi-Tseng Chou, Rui-Zhe Wu, Mobayode O Akinsolu, **Yushi Liu** and Bo Liu, 'Radiation optimization for phased arrays of antennas incorporating the constraints of active reflection coefficients', *IEEE Transactions on Antennas and Propagation* 70.12, pp. 11375-11388. December 2022.
- Zhenghui Li, **Yushi Liu**, Bo Liu, Julien Le Kernec and Shufan Yang, 'A holistic human activity recognition optimisation using AI techniques', *IET Radar, Sonar & Navigation* 18.2, pp. 256–265. September 2023.
- Yijia Hao, **Yushi Liu** and Bo Liu, George Amarantidis and Rami Ghannam, 'Integrating AI in Engineering Education: A Systematic Review and Student-Informed Module Design for UK Students', submitted to *IEEE Transactions on Education*

Chapter 2

Background knowledge

In this chapter, the building blocks of this research are introduced. Section 2.1 technically introduces the concept of global and local optimisation in detail and its relation with antenna design optimisation. Section 2.2 introduces evolutionary algorithms heuristically and mathematically. Following the reviews in Section 1.4, Section 2.3 introduces several potential surrogate modelling methods in more technical detail. Section 2.5 introduces Bayesian optimisation in technical details. Section 2.6 introduces surrogate model-assisted evolutionary algorithms based on the previous sections in the chapter. Section 2.7 introduces the surrogate model management of the evolutionary algorithms.

2.1 Global optimisation, local optimisation and antenna design optimisation

Based on the different optimisation goals, optimisation can be classified into two categories: single-objective optimisation and multi-objective optimisation. Single-objective optimisation runs towards a single objective; thus, the optimisation result is usually a single optimal solution. The optimisation goal of multi-objective optimisation is a Pareto front, which contains a set of optimal trade-offs between the objectives. Even though antenna optimisation can have multiple design specifications, multi-objective optimisation is not usually the case of antenna optimisation as the preference of design specifications is known beforehand, and an optimal antenna design – rather than a set of them having different trade-offs – is desired based on the design specifications (Liu et al. 2014a; Villiers and Koziel 2018). The multiple design specifications are usually

transformed into a single objective by weighted summation, also called aggregation method (Chiandussi et al. 2012). Therefore, in this work of antenna design automation, the optimisation refers to single-objective optimisation. An optimisation problem can be either a minimisation problem or a maximisation problem. As minimisation and maximisation are mutually interchangeable, only minimisation setups are used in the introduction to optimisation in this chapter.

2.1.1 Optimisation fundamentals

An unconstrained single-objective optimisation problem is described in its mathematical minimisation form as

$$\begin{aligned} &\text{minimise } f(x) \\ &\text{s.t. } x \in [lb, ub]^d, \end{aligned} \tag{2.1}$$

where x is the decision variable, lb and ub are the lower and upper bounds of the decision variable, respectively, and s is the dimension of x. $f(x)$ is the objective function and can also be the aggregated form (or weighted summation) of n objective functions as

$$f(x) = w_1 f_1(x) + w_2 f_2(x) + \cdots + w_n f_n(x) \tag{2.2}$$

where from $f_1(x)$ to $f_n(x)$ are the n objective functions and from w_1 to w_n are their corresponding weights. Once the minimisation completes, the optimal $x = x^*$ is found and considered the optimal decision variable that minimises the objective function(s), i.e., $f(x^*)$ reaches the minimum possible value.

2.1.2 Constrained optimisation

A constrained optimisation problems is described as

$$\begin{aligned} &\text{minimise } f(x) \\ &\text{s.t. } g_i(x) \leq 0, \\ &\quad x \in [lb, ub]^d, \end{aligned} \tag{2.3}$$

where $f(x)$ is, similarly in (2.1), the objective function, and $g_i(x)$ are the constraints. Objectives and constraints are two main components of constrained optimisation problems. For the minimisation objective function $f(x)$, the less it is, the better the decision variable x is. The constraints $g(x)$ can be satisfying or not satisfying. This is to say,

for two different given decision variables x_1 and x_2, $g(x_1) = -100$ is not better than $g(x_2) = -1$, as they both satisfy $g(x) \leq 0$. On the other hand, for a given x_3, $g(x_3) > 0$ violates the constraint, making x_3 an undesired solution, even when $f(x_3)$ has a minimal value. In a word, a constrained minimisation problem is to find a x^* so that the value of $f(x^*)$ is minimal while having all constraints $g_i(x^*)$ satisfied.

2.1.3 Antenna design optimisation

Modern antenna design often relies on optimisation algorithms, as an antenna design structure can be represented by a parameterised model, and its design specifications can be fitted in the optimisation problem formation. Popular design specifications include the reflection coefficient and realised gain over a certain bandwidth of a microwave antenna, axial ratio, total efficiency, and other factors. Depending on specific needs, some are suitable for objectives while others are suitable for constraints (Tentzeris et al. 1998; Liu et al. 2017d). Antenna design optimisations have multiple design specifications; considering the preference or priority of the specifications and the magnitudes of the objective and constraints, different weights can be assigned to the performances to form a single objective function to fit in the single objective optimisation. As such, the optimisation methodology can be applied to antenna design structure optimisation.

2.2 Evolutionary algorithms

Like mentioned in Section 1.3, evolutionary algorithms (EAs) are frequently used in antenna design optimisation. For the general usage of an EA, an antenna designer practically expect the algorithm to fulfil the following three requirements (Storn and Price 1997) due to the nature of optimising antennas or EM devices.

- The EA has the capability of handling non-differentiable, non-linear and multimodal landscapes of antenna optimisation.
- The EA allows parallel fine evaluations of antenna designs.
- The EA has good convergence properties, i.e., the EA ends up converging to global minimum in most of the independent runs.

2.2.1 The heuristics of evolutionary algorithms

EAs are inspired by the principles of biological evolution and natural selection and is commonly used in optimisation. These algorithms mimic the process of natural selection to search for optimal solutions to complex problems through an iterative process. These algorithms undergo a continuous refinement process inspired by biological evolution, aiming to adapt and improve solutions over successive generations. In other words, they navigate the vast and complex optimisation landscape of potential solutions, strategically probing and tuning candidate solutions to converge towards global optimum. Amongst the different types of EAs, the heuristics play a crucial role in guiding the search process by efficiently exploring the solution space with precision and efficacy (Bäck and Schwefel 1993; Eiben et al. 2015).

At the heart of evolutionary algorithms rely on some key concepts, including populations, individuals, fitness evaluation, selection, crossover, and mutation. First, a population of potential solutions, or individuals, is randomly generated within the solution space. In the population, each individual represents a potential solution to the optimisation problem, encoded in a format suitable for computational processing. The fitness of each individual is then evaluated based on a predefined objective function that quantifies the quality of the solution (Bäck and Schwefel 1993; Eiben et al. 2015).

Heuristics in evolutionary algorithms guide the iterative improvement process by determining which individuals are selected for reproduction and how their genetic information is combined to produce new offspring. Selection mechanisms, such as the tournament selection method or roulette wheel selection, mimic the survival of the fittest principle by favouring individuals with higher fitness values for reproduction (Michalewicz 1996; Vikhar 2016). The effectiveness of heuristics in evolutionary algorithms relies on their ability to balance exploration and exploitation of the solution space. Exploration involves searching for new regions of the solution space to discover potentially better solutions, while exploitation focuses on refining promising solutions to improve their quality (Michalewicz 1996; Raidl and Gottlieb 2005; Bozorg-Haddad et al. 2017).

The crossover and mutation operators are essential for introducing diversity into the population by recombining genetic information from selected individuals and introducing random changes, respectively (Spears 1993). Crossover involves the exchange of genes between parent individuals to produce offspring with characteristics inherited

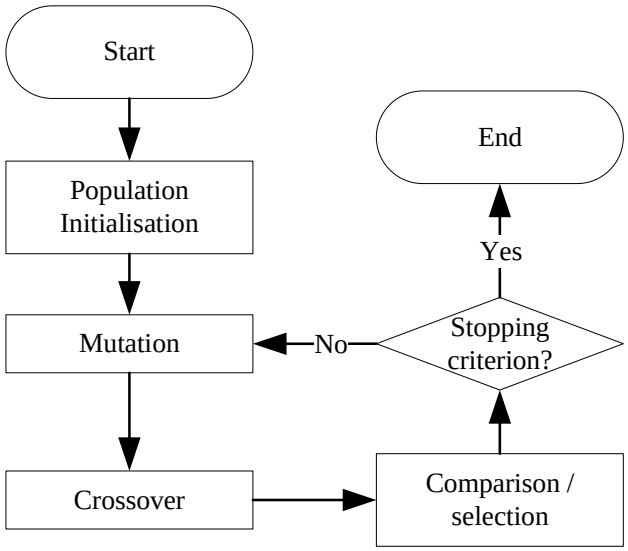


Figure 2.1: Flow diagram of differential evolution.

from both parents (Wu et al. 1997). The process helps to increase the diversity of the population and generate new solutions. On the other hand, mutation introduces random changes in the genetic information of individuals, preventing premature convergence and encouraging exploration in the search. Both processes are critical for the success of EAs, as they allow the algorithm to avoid premature convergence and to search for better solutions (Spears 1993; Wu et al. 1997).

Overall, heuristics in evolutionary algorithms play a vital role in efficiently navigating the solution space, facilitating the discovery of high-quality solutions to complex optimisation problems. Evolutionary algorithms equipped with effective heuristics offer powerful tools for solving a wide range of real-world optimisation problems in various domains by emulating the principles of natural selection and genetic variation.

2.2.2 Differential evolution

Differential evolution (DE) is one kind of EAs, and it outperforms many other EAs for optimisation in continuous space. The DE utilises differential operators to generate new candidate solutions and a one-on-one comparison scheme to greedily select better candidate solutions during the operations (Storn and Price 1997; Storn et al. 2005).Figure 2.1 illustrates the flow diagram of the DE, and a brief technical introduction to DE is as follows.

The population composed of N individual solutions and each solution, or target vector is denoted by $x = (x_1, \ldots, x_d) \in R^d$. To create a child solution $u = (u_1, \ldots, u_d)$, firstly, mutation happens to generate a mutation vector $v = (v_1, \ldots, v_d)$. There are several mutation strategies under DE, such as the standard *DE/rand/1* strategy, expressed mathematically as

$$v^i = x^{r_1} + F \cdot (x^{r_2} - x^{r_3}) \tag{2.4}$$

where indexes $r_1, r_2, r_3 \in \{1, 2, 3, \cdots, \mathrm{N}\}$ are sampled from a random uniform distribution without replacement, x_r is the r^{th} sample in the population, and mutation rate $F \in (0, 2]$ controls the amplification of the differential variation term $(x^{r_2} - x^{r_3})$, the *DE/best/1* strategy, expressed as

$$v^i = x^{\text{best}} + F \cdot (x^{r_1} - x^{r_2}) \tag{2.5}$$

where x^{best} is the current best candidate solution in the population, the *DE/current-to-best/1* strategy, expressed as

$$v^i = x^i + F \cdot (x^{\text{best}} - x^i) + F \cdot (x^{r_1} - x^{r_2}) \tag{2.6}$$

and the *DE/best/2* strategy, expressed as

$$v^i = x^{\text{best}} + F \cdot (x^{r_1} + x^{r_2} - x^{r_3} - x^{r_4}) \tag{2.7}$$

etc. To have more samples involved in the mutation vector generation, like in (2.7), may improve the population diversity, particularly for large N. The *DE/current-to-best/1* strategy illustrated in (2.6) is applied in this work.

In order to maintain the diversity in the population, a child solution vector u is obtained mathematically as

$$u_j = \begin{cases} v_j, & \text{if } r \leq CR \text{ or } j = j_{\text{rand}} \\ x_j, & \text{otherwise} \end{cases} \tag{2.8}$$
$$j = 1, 2, \cdots, d$$

where u_j, v_j and x_j are the j^{th} element of the child solution vector, mutation vector and target vector in the population, respective, the r is a random number sampled from a standard uniform distribution, $CR \in [0, 1]$ is the crossover rate, j_{rand} is the random integer sampled uniformly from $\{1, \ldots, d\}$.

To decide whether or not a child solution vector should be included as a member of the population in the next generation, such a child solution vector is compared to its corresponding target vector using the greedy criterion. If for a child solution vector u_k and a target vector x_k, regarding the cost function value, it has $f(u_k) < f(x_k)$ for a minimisation problem, like in (2.1) and (2.3), the child solution vector u_k should replace the target vector x_k in the next generation population. Otherwise, the old target vector x_k is retained in the population.

Theoretically, by looping over the above operations, the optimisation converges to the global minimum. The iterative optimisation can also be terminated once a minimised value is reached, as shown in Figure 2.1.

2.2.3 Genetic algorithm

GAs usually handle optimisation problems whose solutions are represented as fixed-length strings of symbols for scheduling and routing optimisation problems or as real-valued vectors for parameter optimisation. Regarding the selection mechanism, GAs often utilise roulette wheel selection or tournament selection.

Similar to DE introduced in Subsection 2.2.2, random samples are drawn and evaluated to form initialisation. Following the initialisation, roulette wheel selection is typically used to select parent generation from the samples in the initialisation. The roulette wheel selection mechanism can be expressed as

$$P_{x_i} = \frac{f_{\text{fitness}}(x_i)}{\sum_{j=1}^{N} f_{\text{fitness}}(x_j)}, \tag{2.9}$$

where N is the population size, $f_{\text{fitness}}(x)$ is a function fitness value of parent individual x, and P_{x_i} is the probability for the parent individual x_i being selected from the pool. This implies a higher probability of being selected for the better parent individuals as they have higher fitness values in the maximisation setting.

Regarding crossover operation, unlike DE, GAs randomly select a crossover point, divide the genetic sequence into two sections and allow exchanges of genetic sections for every two parent individuals. This operation can be expressed in pseudo-code as

$$\text{Child}_k \leftarrow \text{Parent}_{p,1\sim s}, \text{Parent}_{q,s+1\sim D}, \tag{2.10}$$

where s is the crossover point, D is the length of the genetic sequence, $\text{Parent}_{p,1\sim s}$ represent the first s genetic information of the parent individual p, and $\text{Parent}_{q,s+1\sim D}$ represent the last $D-(s+1)$ genetic information of the parent individual q. The concatenation of the two sectioned sequences forms Child_k.

Mutation is then applied to the generated children to maintain diversity for the optimisation. This operation here is similar to that of DE if the genetic information is real number values. For information coded in binary digits, mutation means to reverse the digits in practice.

The replacement and the termination criteria are similar to those of DE.

2.2.4 Particle swarm optimisation

PSO is another population-based stochastic evolutionary algorithm for optimisation. PSO is inspired by animals' social behaviours such as fish schooling and bird flocking (Kennedy and Eberhart 1995). Unlike DE and GA introduced in Subsections 2.2.2 and 2.2.3, respectively, which search for optimum solutions based on the decision variables and their fitness values, PSO introduces an additional concept, velocity, to assist the search along with the decision variable, or position. Furthermore, PSO maintains a fixed number of individuals or particles throughout the optimisation process and no individuals are eliminated or removed from the population. As no individuals are removed from the population and the traces of the individuals are known, individuals' best decision variable, which refers to the decision variable with the best fitness value that an individual achieves up the current iteration, can be obtained anytime. The global best solution, similarly to that of DE, refers to the best fitness value among all individuals up to the current iteration.

At the beginning, decision variables and velocity vectors are randomly initialised, and the fitness values are obtained. In the optimisation loop, for each individual, update the velocity vector and position vector by

$$\begin{cases} v_i(t+1) = wv_i(t) + c_1 r_(p_i x_i) + c_2 r_2 (g - x_i) \\ x_i(t+1) = x_i(t) + v_i(t+1) \end{cases} \tag{2.11}$$

where x_i is the position or solution variable of the i-th individual, t is the number of current iteration, w is the inertia weight empirically to be set to 1 initially (Shi and Eberhart 1998) and decreased or adaptively varied with the optimisation, c_1 and $c2$ are acceleration coefficients representing the relative influence from the individual best position and global best position, r_1 and r_2 are two random values sampled from the standard uniform distribution, p_i is the individual best position of the i-th individual and g is the global current best position. After the position vector v_i and velocity vector x_i update, the individual best position p_i and the global current best position g are updated before the next iteration. Similarly, the optimisation will theoretically converge to the global optimum when the population has little diversity.

2.3 Surrogate modelling

Surrogate models, sometimes called surrogates, metamodels or response surface models, often refer to simplified mathematical approximations of computationally expensive and particularly simulation-based processes. They often mimic the behaviours or mapping of the original models with significantly reduced computational costs. Mathematically, given a computationally expensive function

$$f(x),$$

a computationally cheap surrogate model

$$\hat{f}(x) \tag{2.12}$$

that has approximate mapping can be built to replace the former one wherever necessary or appropriate. Therefore, surrogate models are instrumental in expensive optimisation and many other analyses with time-consuming processes involved. Some common surrogate models include

- Kriging (Gaussian process regression).
- Radial basis function interpolation.
- Artificial neural networks.
- Polynomial regression.
- Support vector regression.
- Response surface method.

Kriging, radial basis function interpolation, artificial neural networks, and polynomial regression are frequently used in antenna optimisation to mimic the characterisation of antennas and are introduced in detail in this section.

2.3.1 Gaussian process regression

In Bayesian optimisation, Gaussian process regression is sometimes called Gaussian process (GP) or Kriging. The basic principle of GP is as follows (Rasmussen and Williams 2006). In the following, superscript refers to the index of a sample in the dataset, and subscript refers to the index of a variable in a sample. Giving n observations ($x = (x^1, \ldots, x^n)$ and $y = (y^1, \ldots, y^n)$), GP assumes that $y(x)$ is a sample of a Gaussian distributed stochastic process with mean μ and variance σ. GP then predicts the value of $y(x)$ for a new x using its relation with the n observations. For example, a correlation function can be described as

$$\begin{aligned} &\mathrm{Corr}(x^i, x^j) = \exp(-\textstyle\sum_{l=1}^{d} \theta_l |x_l^i - x_l^j|^{p_l}) \\ &\theta_l > 0, 1 \leq p_l \leq 2 \end{aligned} \tag{2.13}$$

where d is the dimension of x. θ_l and p_l are hyper-parameters, which are determined by maximising the likelihood function in (2.14).

$$\frac{1}{(2\pi\sigma^2)^{n/2}\sqrt{\det(R)}} \exp\left[-\frac{(y-\mu I)^T R^{-1}(y-\mu I)}{2\sigma^2}\right] \tag{2.14}$$

where R is a $n \times n$ covariance matrix and I is a $n \times 1$ vector having all its elements as unity. By maximising the likelihood function that $y = y^i$ at $x = x^i (i = 1, \ldots, n)$ and handling the prediction uncertainty based on the best linear unbiased prediction, for a new point x^*, the predicted value and prediction uncertainty are $\hat{y}(x^*)$ and $\hat{s}^2(x^*)$, which are expressed mathematically as

$$\hat{y}(x^*) = \mu + r^T R^{-1}(y - I\mu) \tag{2.15}$$

where

$$\begin{cases} R_{i,j} = \mathrm{Corr}(x^i, x^j),\ i, j = 1, 2, \ldots, n \\ r = [\mathrm{Corr}(x^*, x^1), \mathrm{Corr}(x^*, x^2), \ldots, \mathrm{Corr}(x^*, x^n)] \\ \hat{\mu} = (I^T R^{-1} y)(I^T R^{-1} I)^{-1} \end{cases} \tag{2.16}$$

and

$$\hat{s}^2(x^*) = \hat{\sigma}^2[1 - r^T R^{-1} r + (I - r^T R^{-1} r)^2 (I^T R^{-1} I)^{-1}] \tag{2.17}$$

where

$$\hat{\sigma}^2 = (y - I\widehat{\mu})^T R^{-1}(y - I\widehat{\mu})n^{-1}. \tag{2.18}$$

Based on the above, two advantages of GP include:

1. There are almost no empirical parameters in GP modelling except deciding the type of correlation function; A few appropriate correlation functions are already found by antenna surrogate modelling researchers (Wu et al. 2020; Koziel et al. 2021, 2014; Liu et al. 2014c). Hence, overfitting or under-fitting like artificial neural networks (ANNs) is less likely to happen, which improves the prediction quality.
2. The prediction uncertainty (2.17) is statistically grounded, which can play an important role when judging the full potential of a candidate antenna design in prescreening or acquisition function.

With the prediction uncertainty, widely used prescreening methods include expected improvement (Jones et al. 1998), probability of improvement (Ulmer et al. 2003), and lower confidence bound (LCB) (Dennis and Torczon 1997). LCB is the fundamental of the new prescreening method in this paper and is introduced as follows. Given the objective function $y(x)$ has a predictive distribution of $N(\hat{y}(x), \hat{s}^2(x))$, an LCB prescreening of $y(x)$ is:

$$\begin{aligned} &y_{\text{lcb}}(x) = \hat{y}(x) - \omega\hat{s}(x) \\ &\omega \in [0, 3] \end{aligned} \tag{2.19}$$

where ω is a constant, and is often set to 2 in many algorithms in the AI domain (Emmerich et al. 2006), and is applicable to antenna problems (Liu et al. 2014c).

However, the main drawback of GP is its training cost. In online surrogate model-assisted antenna global optimisation, the total training time of GP models in the optimisation process can be estimated as $T_{GP} \times N_{specs} \times N_{pop} \times N_{it}$ (Liu et al. 2021), where T_{GP} is the training time of each GP model, N_{specs} is the number of specifications, N_{pop} is the number of candidate designs in a population, and N_{it} is the number of iterations in antenna design optimisation.

For a GP model, the computational complexity is $O(N_o n^3 d)$ (Emmerich et al. 2006), where N_o is the number of iterations spent in hyper-parameter optimisation (i.e., (2.14)) and n is the number of training data points. n is highly affected by d to construct a reliable surrogate model. (Liu et al. 2014c, 2021) shows that at least $4 \times d$ training data points are needed for antenna problems. Often, when d reaches 20, T_{GP} could be

in minutes for a normal computer and then grows cubically (Liu et al. 2021). Also, to maintain the exploration ability, N_{pop} is also highly affected by d (e.g., often at least $4 \times d$ when using DE operators). This makes the GP modelling time in antenna optimisation longer when d is large and could be even longer than the EM simulation time.

2.3.2 Radial basis function

Radial basis function (RBF) (Powell 1992) is another popular machine learning method for surrogate modelling. Its learning ability is not as high as GP, but its training is computationally much cheaper (Regis 2013). For several complex antenna test cases, the RBF modelling time for a candidate design is often less than 30 seconds using a normal desktop workstation, even if 1000 training data points are used. Given a set of observations $x = (x^1, \ldots, x^n)$ and $y = (y^1, \ldots, y^n)$, RBF predicts the function value $y(x^*)$ at a new point x^* as:

$$\hat{y}(x^*) = \sum_{i=1}^{n} \lambda_i \phi(||x^* - x^i||) + p(x) \tag{2.20}$$

where λ are the coefficients, $p(x)$ is a linear polynomial with d variables and $p(x) = \sum_{j=1}^{d+1} b_j x_j$. ϕ is a basis function, which is monotonic. In this implementation, the cubic form, $\phi(x) = x^3$ is used. To fit this model, the hyperparameters $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_n]^T$ and $B = [b_1, b_2, \ldots, b_{d+1}]^T$ can be calculated by solving:

$$\begin{bmatrix} \Phi & P_r \\ P_r^T & 0_{(d+1)\times(d+1)} \end{bmatrix} \begin{bmatrix} \lambda \\ B \end{bmatrix} = \begin{bmatrix} y \\ 0_{d+1} \end{bmatrix} \tag{2.21}$$

where $\Phi \in R^{n \times n}$ and $\Phi_{ij} = \phi(||x^i - x^j||)$, $i, j = 1, 2, \ldots, n$. $P_r \in R^{n \times (d+1)}$ and the i^{th} row of P_r is $[1, x^i]$.

2.3.3 Artificial neural networks

Artificial neural network (ANN), sometimes called neural network or neural net, is a popular machine learning method. ANN can be used as a surrogate model in regression, as its forward propagation is computationally cheap.

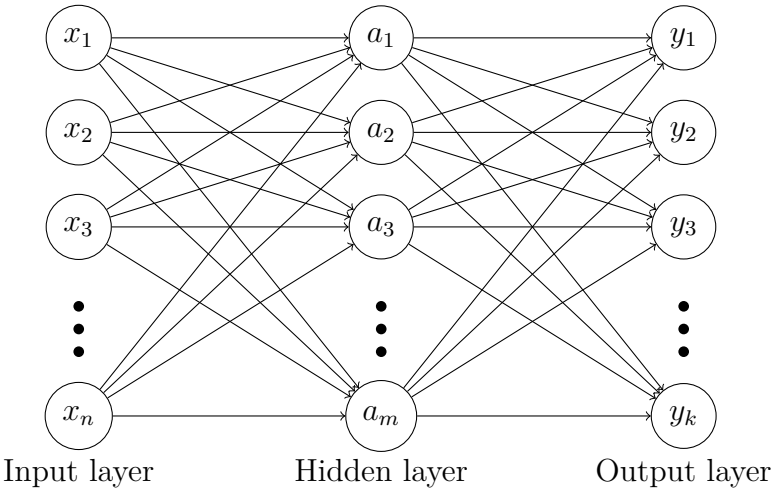


Figure 2.2: Illustration of an artificial neural network.

Figure 2.2 shows an example of an ANN with three layers (one input layer, one hidden layer and one output layer). The output of the i-th node in the hidden layer is

$$a_i = \sum_{j=1}^{n} w_{ij} x_j \tag{2.22}$$

where x_j denotes the j-th node in the input layer, and w_{ij} denotes the weight on the directed connection from x_j in the input layer to a_i in the next layer. x and a are both vectors. And for the layer as a function, it can be represented as

$$a = f^{(x \to a)}(x). \tag{2.23}$$

Overall, the entire ANN model in Figure 2.2 is then represented mathematically as a chain of functions as

$$y = f^{(x \to a)}(f^{(a \to y)}(x)) = f^{(x \to y)}(x) = f^{(\mathrm{ANN})}(x) \tag{2.24}$$

where $f^{(\mathrm{ANN})}$ is a computationally cheap candidate approximation of the real mapping, or f^*. In order to let $f^{(\mathrm{ANN})}$ to approximate f^*, the training of the ANN basically update w_{ij} iteratively with

$$\begin{aligned} \Delta w_{k,G} &= \eta \frac{\partial E}{\partial w_{k,G}} \\ w_{k,G+1} &= w_{k,G} - \Delta w_{k,G} \end{aligned} \tag{2.25}$$

where $w_{k,G}$ is an arbitrary weight in the ANN after G generations of training, E is the differentiable loss function and η is the learning rate.

Advanced ANN often includes activation functions to introduce non-liberality. Various optimisers are proposed to accelerate the ANN training.

2.3.4 Multivariate polynomial regression

Polynomial regression is a type of regression analysis technique that models the relationship between a single input variable and an output variable in complex systems or processes. The relationship is modelled as an n-degree polynomial function.

The general form of a polynomial regression with degree n is mathematically expressed as

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_n x^n + \epsilon \tag{2.26}$$

where x is the input variable, y is the output variable, β_0, β_1, $\cdots$, β_n are the coefficients of the polynomial terms, and ϵ is the error term representing the variability or noise of the data. Typically, the lease square method is used to estimate the coefficients β_0, β_1, $\cdots$, β_n. Once the coefficients are estimated, the polynomial regression function (2.26) can be used to approximate the relationship between the single input variable and the output variable from complex mappings.

Furthermore, the standard polynomial regression expressed in (2.26) can be extended in its multivariate form, which is expressed as

$$\begin{aligned} y = {} & \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \\ & + \sum_{i_{2,1}=1}^{k} \sum_{i_{2,2}=1}^{k} \beta_{i_{2,1} i_{2,2}} x_{i_{2,1}} x_{i_{2,2}} \\ & + \cdots \\ & + \sum_{i_{n,1}=1}^{k} \sum_{i_{n,2}=1}^{k} \cdots \sum_{i_{n,n}=1}^{k} \beta_{i_{n,1}, i_{n,2}, \cdots, i_{n,n}} x_{i_{n,1}} x_{i_{n,2}} \cdots x_{i_{n,n}} + \epsilon \end{aligned} \tag{2.27}$$

where x_1, x_2, $\cdots$, x_k are the inputs variables, β_2 is the interaction coefficient associated with the 2^{nd} input variable x_2, $\beta_{i_{2,1} i_{2,2}}$ is the 2-way interaction coefficient associated with the $i_{2,1}$-th variable $x_{i_{2,1}}$ and $i_{2,2}$-th input variable $x_{i_{2,2}}$, and $\beta_{i_{n,1}, i_{n,2}, \cdots, i_{n,n}}$ is the n-way interaction coefficient associated with the $i_{n,1}$-th input variable $x_{i_{n,1}}$, $i_{n,2}$-th input variable $x_{i_{n,2}}$, $\cdots$, $i_{n,n}$-th input variable input variable $x_{i_{n,n}}$.

Similarly to the univariate polynomial terms, least square methods are used to estimate the polynomial coefficients so that the loss between calculated $\hat{y}$ and the real y is minimised. Multivariate polynomial regression allows for modelling complex relationships between multiple input variables and a single output variable. It captures the interaction effects among the input variables and non-linearity in the complex relationship.

However, the computation complexity increases with the increased dimensionality of input variables. This increased dimensionality of the multivariate polynomial regression requires a large amount of data samples and time to estimate the coefficients accurately, which is a particular drawback of a surrogate model.

2.4 Prescreening methods

The prescreening method, sometimes called acquisition function or infill criterion, is applied to filter out unnecessary candidate solutions and select promising candidate solutions P_{selected} from the population of candidate solutions P for the later function evaluation

$$\begin{aligned} &f(x_i) \\ &x_i \in P_{\text{selected}} \\ &P_{\text{selected}} \subset P. \end{aligned} \tag{2.28}$$

The prescreening methods utilise information the surrogate models provide to identify promising candidate solutions out of a set of solutions, thereby reducing the number of function evaluations needed overall. Prescreening can be realised differently for different usages of the information. Some widely used prescreening methods include

- Expected improvement.
- Probability of improvement.
- Lower confidence bound.

They are introduced below in this section.

2.4.1 Expected improvement

The expected improvement is a prescreening method that selects candidate solutions likely to improve over the current best solution. It selects candidate solutions by quantifying their potential improvements over the current best solution, considering both the predicted mean and prediction uncertainty.

The expected improvement for a candidate solution x is defined as the expected value of the improvement over the current best solution. This can be expressed mathematically as

$$EI(x) = \mathbb{E}[\max(y_{\text{best}} - \hat{y}(x), 0)] \tag{2.29}$$

where

$$y_{\text{best}} = f(x_{\text{best}})$$

and x_{best} is the current best solution. This can be calculated as the integral of the improvement function over the region where the surrogate model predicts a potential improvement over the current best solution, which is expressed as

$$EI(x) = \int_{-\infty}^{\infty} \max(y_{\text{best}} - \hat{y}(x), 0) \cdot p(\hat{f}(x)) d\hat{f}(x). \tag{2.30}$$

Alternatively, for the Gaussian process regression surrogate model, the expected improvement can be calculated analytically as

$$EI(x) = (\mu(x) - y_{\text{best}}) \cdot \Phi(\frac{\mu - y_{\text{best}}}{\sigma(x)}) + \sigma(x) \cdot \phi(\frac{\mu - y_{\text{best}}}{\sigma(x)}). \tag{2.31}$$

where $\mu(x)$ and $\sigma(x)$ are the predicted mean and the prediction accuracy of the candidate solution x, respectively and Φ and ϕ are the cumulative distribution function and probability density function of the standard normal distribution, respectively. In the end, one or more candidate solutions are selected based on the expected improvement value $EI(x)$. Candidates with higher expected improvement values are more likely to improve over the current best solution.

2.4.2 Probability of improvement

Similarly, the probability of improvement quantifies the probability that a candidate solution yields a better objective function value than the current best candidate solution based on both predicted mean and prediction uncertainty.

The probability of improvement of a candidate solution x is defined as the probability that the surrogate model predicts an improvement over the current best solution, which is mathematically expressed as

$$PI(x) = P(\hat{f}(x) \geq y_{\text{best}}). \tag{2.32}$$

This can be calculated as the integral of the probability density function of the surrogate model predictions over the region where the predictions are greater than or equal to the current best solution, which is expressed as

$$PI(x) = \int_{y_{\text{best}}}^{\infty} p(\hat{f}(x))d\hat{f}(x). \tag{2.33}$$

Alternatively, for the Gaussian process regression surrogate model, the probability of improvement can be analytically calculated as

$$PI(x) = \Phi(\frac{\mu - y_{\text{best}}}{\sigma(x)}). \tag{2.34}$$

In the end, one or more candidate solutions are selected based on the expected improvement value $PI(x)$. Candidates with a higher probability of improvement are more likely to improve over the current best solution.

2.4.3 Lower confidence bound

The lower confidence bound (LCB) method combines the quantified information, predicted mean and prediction uncertainty provided by the surrogate models to assess the potential of candidate solutions. LCB selects candidate solutions that are likely to have less fitness values. It is also based on the uncertainty information provided to the surrogate model. It helps to balance exploration and exploitation by favouring solutions with low predicted values and high prediction uncertainties. Mathematically, the LCB of a candidate solution x is expressed as

$$LCB(x) = \mu(x) - \beta\sigma(x) \tag{2.35}$$

where β is a user-defined parameter controlling the exploration versus exploitation trade-off. A higher β leads to more exploration, while a lower β leads to more exploitation.

Similarly, in the end, one or more candidate solutions are selected based on the LCB $LCB(x)$. Candidates with less LCB values are more promising than those with greater values and thus are prioritised for expensive function evaluation.

2.5 Bayesian Optimisation

Bayesian optimisation is a sequential model-based optimisation technique used to work on global optimisation of black-box functions that does not assume forms and parameterisation (Mockus 2005), like described in (2.1) as a general form. It combines probabilistic models like GP with prescreening methods to efficiently explore and exploit the search space for the global optima.

Mathematically, like described in (2.1), given an objective function $f(x)$, Bayesian optimisation maintains a probabilistic model

$$p(f|\mathcal{D})$$

of the objective function given observation $\mathcal{D}$. Typically, GP is used as the parametric probabilistic model in Bayesian optimisation for its flexibility and ability to model uncertainty. The prior distribution of GO is specified by the mean μ and a covariance matrix R, mathematically expressed in (2.13). These are discussed in detail in Subsection 2.3.1.

The prescreening method is applied in Bayesian optimisation to balance the exploration and exploitation in the search space. Bayesian optimisation proceeds iteratively by obtaining new samples in the search space with the help of the prescreening method. By prescreening, the new sample observation

$$(x_{\text{new}}, y_{\text{new}})$$

is obtained and to be used to update the probabilistic model, incorporating the new information into the posterior distribution. This is discussed technically in section 3.4.

Bayesian optimisation terminates when satisfying predefined stopping criteria, which typically include limited iterations (or computation budgets) and satisfied optimisation requirements. This is reflected in Figure 4.1 in Section 4.1.

Bayesian optimisation is particularly advantageous for expensive optimisation, whose computational cost of function evaluation is not negligible. By leveraging probabilistic models like GP, Bayesian optimisation can efficiently explore the search space, focusing on the region where the objective function is likely to improve or is uncertain and thus

has the potential to improve, which allows for a guided and effective exploration of the search space comparing to the traditional random search. Bayesian optimisation is a sequential optimisation technique, where new sample points are obtained iteratively online, which allows the optimiser to adaptively search for optimum solutions and avoid regions that are unlikely to contain better solutions. As a result, Bayesian optimisation requires fewer function evaluations than other optimisation methods. Owing to these, Bayesian optimisation is well suited for expensive optimisation tasks where function evaluations are costly or time-consuming.

2.6 Surrogate model-assisted evolutionary algorithms

In SAEAs, EAs are considered search engines. Despite being efficient, as introduced in 2.6, these search engines usually require a certain amount of function evaluations in each iteration to evaluate the candidate solutions of the next generation. Therefore, to improve the efficiency of the search engines, surrogate models are introduced to assist them in forming surrogate model-assisted EAs. Instead of directly evaluating the objective function $f(x)$ for every candidate solution, SAEA utilises the surrogate model

$$\hat{f}(x)$$

to estimate objective function values like discussed in (2.12), and candidate solutions are prescreened with the estimated values for the function evaluations.

The surrogate model $\hat{f}(x)$ is usually built based on data samples yielded from computationally expensive processes. Ideally, the surrogate model $\hat{f}(x)$ is sufficiently approximate to the computationally expensive processes $f(x)$. If so, the efficiency or the number of function evaluations needed for SAEA in optimisation tasks is close or comparable to that of only using EA search engines. In reality, the surrogate model $\hat{f}(x)$ cannot perfectly fit the real computationally expensive process $f(x)$. It is common to observe from little to medium prediction error for $\hat{f}(x)$, which naturally slows down the convergence by some generations, compared to using EAs directly. In this case, the trade-off is between the number of generations before converge and the number of function evaluations in each generation. Moreover, it has been validated by many researches that, from EAs to SAEAs, the number of function evaluations saved in each generation and the number of overused generations are disproportional. SAEAs require

less overall number of function evaluations than the EAs (Jin 2011), with little additional computation for surrogate model training and negligible additional computation for running the search engines. Thus, also taking advantage of Bayesian optimisation, SAEAs are efficient, particularly for expensive optimisation tasks.

Practically, the trained surrogate model fidelity can be poor at the beginning due to the lack of training samples available. As the EAs iteratively progress, more and more actual function evaluations of solutions are performed, and the input-output pairs are appended to the training dataset accordingly. Therefore, the fidelity of the surrogate model is gradually improved with more training samples involved, with the iterative optimisation proceeding.

Besides, some surrogate models assist in optimisation by balancing exploration and exploitation by guiding the selection of candidate solutions in prescreening. Adaptive strategies can also be employed to adjust the exploration versus exploitation trade-off based on the surrogate model predictions, the expected improvements in solutions and so on (Liu et al. 2012; Liu et al. 2022a).

2.7 Surrogate model management

A trade-off between the quality of the surrogate model and the number of expensive function evaluations for training sample acquisition is presented in most SAEAs. In SAEAs, as discussed in sections 2.3 and 2.6, ideally, if the surrogate model $\hat{f}(x)$ is sufficiently approximate to the original expensive function evaluation $f(x)$, i.e., the quality of the surrogate model is high, the predictions of the surrogate model are accurate, and hence the optimisation efficiency is high. On the other hand, the need for sufficient samples for training the surrogates is a burden for optimisation efficiency, as the training sample acquisition is computationally expensive due to the nature of the complex processes. Therefore, the surrogate model management is considered to find an appropriate trade-off for satisfying surrogate model quality requiring as few training samples as possible (Jin 2011). There are several surrogate model management methods available for SAEAs. They include

- Trust-region search (TRS) framework (Zhou et al. 2006; Lim et al. 2009).
- Generation control (GC) framework (Jin 2005).
- Prescreening-based framework (Emmerich et al. 2006).

- Surrogate model-aware evolutionary search (SMAS) framework (Liu et al. 2014b).

TLS and GC are generally compatible with standard EA operators yet require more computationally expensive function evaluation (Zhou et al. 2006; Lim et al. 2009). On the other hand, the SMAS framework, one of the state-of-the-art SAEA frameworks, borrows ideas from prescreening and manages the evolutionary search appropriately with the high quality of the surrogate model and high efficiency of the optimisation at the same time. In particular, it has been widely applied to EM design optimisations. Several state-of-the-art EM designs optimisation, including mm-wave integrated circuits, antennas, filters, and MEMS designs optimisation, borrow ideas from SMAS, and their optimisations outperform other SAEAs in regard to optimisation efficiency (Liu et al. 2014c,e, 2017c). This is discussed in detail in Chapter 4.

Chapter 3

Bayesian neural networks' behaviour in SAEAs

From a problems solving and pragmatic perspective, Section 3.1 recalls the tricky points in modern antenna design optimisation from Chapter 1, Section 3.2 introduces the BNNs and demonstrates that adapting BNNs is an appropriate solution to the issues raised in Section 3.1 through a set of behavioural studies in the following sections. Section 3.3 discussed the test problems selection for experiments in this Chapter. Section 3.4 demonstrates the prediction performance of BNNs and the co-work with multiple prescreening methods. Section 3.5 demonstrates the performance comparison of BNN versus GP, and BNN versus NN with dropout (NNDO) separately.

The experiments are conducted on a workstation with an AMD Ryzen™ Threadripper™ PRO (2.7GHz) and an NVIDIA® RTX™ A4000 GPU. 80 MATLAB parallel workers are activated for GP/BNN-based surrogate model training.

3.1 Motivation

As introduced in Subsection 1.4.1 and in Subsection 1.6.1, there are multiple issues that hurdle the design optimisation of modern antennas. Like mentioned, modern antenna designs can have topologically irregular geometries and more complex structures and components, and they often require more design parameters to define the parametric antenna model. Antenna models with higher number of deign parameters means higher dimensionality under optimisation framework. Due to the "curse of dimensionality",

GPs are not always efficient as surrogate models under SAEAs. Beside the issue with the increasing number of design parameters, modern antenna design specifications are also increasingly stringent. Multiple design specifications can be required simultaneously in the antenna design optimisation. But for surrogate models like GPs which predict a single output at a time, the computational cost for training increases linearly with respect to the the number of design specifications as the number of surrogate models is increased correspondingly.

An alternative surrogate model that can predict multiple outputs at once and that is not sensitive to the dimensionality of the antenna deign optimisation is desired. By also investigating the previous working surrogate models, the essential characteristics of a potential surrogate model alternative should include

- High prediction accuracy.
- Statistically uncertainty quantification.
- Low training cost, particularly for high dimensional modelling.

These are discussed in details in Section 3.2. Since mathematical proof is complex, an empirical study of employing BNN in an SAEA framework and analysing the performance is conducted and presented in the following sections in this chapter.

Considering that many SAEA frameworks have been available in recent years, with various EA operators and model management methods, it is impossible to use all of them. This thesis does not aim to prove the superiority of BNN as the surrogate model over other counterparts. Instead, as said in Chapter 1, the paper aims to show the potential of BNN as the surrogate model by some observations, inspiring attention to carry out consecutive research in BNN-based SAEAs. Hence, results from a typical SAEA framework can achieve our goal. The SMAS framework (Liu et al. 2013, 2014d) is selected as a typical one and employed in this thesis and the reason for choosing the SMAS framework is discussed in details in Section 4.1. And thus, different prescreening methods and machine learning alternatives will be used under the SMAS framework in the following, which are presented in Sections 3.4 and 3.5.

3.2 Bayesian neural networks

Consider input variables x and output variables y in a regression problem setting. If an ANN model is built and the model parameters are $\theta = [w_1, \ldots, w_j, b_1, \ldots, b_k]$, where w are the weights and b are the biases. In an MLP ANN, each layer carries a linear transformation, followed by a nonlinear activation function that introduces nonlinearity into the layer. The MLP ANN training optimises the loss function, which is often the log-likelihood of the training samples, i.e., maximise $\sum_{i=1}^{n} \log(p(x^i; \theta))$ with a regularisation term. The θ converges to fixed values during the MLP ANN training, and the converged θ is then used for prediction given inputs.

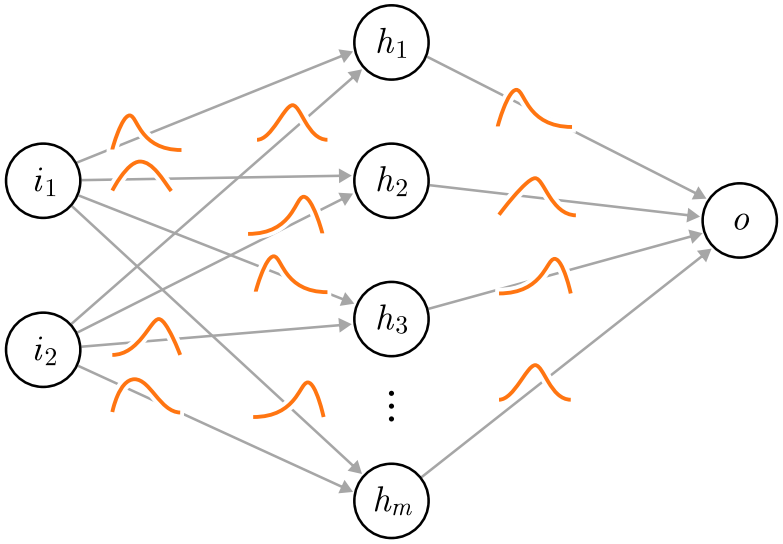


Figure 3.1: An illustrative figure of a basic BNN.

The basic network structure of BNN is similar to the standard ANN, where θ for BNN are random variables with the probability distribution $p(\theta)$. Figure 3.1 illustrates the structure of a BNN. i, h and o represent the neurons on input, hidden and output layers, respectively. In the following, D denotes the training set, and D_x and D_y denote the training inputs and training outputs, respectively. By applying Bayes' theorem, the Bayesian posterior can be expressed mathematically as:

$$p(\theta|D) = \frac{p(D_y|D_x, \theta)p(\theta)}{\int_\theta p(D_y|D_x, \theta')p(\theta')\mathrm{d}\theta'} \propto p(D_y|D_x, \theta)p(\theta), \tag{3.1}$$

where $p(D_y|D_x, \theta)$ denotes the likelihood, $p(\theta)$ denotes the prior, the denominator represents the evidence, and $p(\theta|D)$ denotes the posterior. The posterior is what we acquire in order to obtain the predicted value and prediction uncertainty quantification. Obtaining $p(\theta|D)$ by standard sampling is intractable. Hence, the variational inference method by Markov chain Monte Carlo (MCMC) is employed to make the training time practical. The variational inference method (Blei et al. 2017) is introduced in the following.

Regarding variational inference, a new distribution of the model parameters $q(\phi)$, called a variational distribution, is introduced to approximate $p(\theta|D)$. By minimising the Kullback-Leibler (KL) divergence (Joyce 2011), D_{KL}, between $q(\phi)$ and $p(\theta|D)$, the closest distribution can be obtained to substitute the posterior. Compared to the posterior, $q(\phi)$ has fewer parameters, often including means and variances of a multivariate Gaussian distribution. Moreover, $q(\phi)$ is optimised in the BNN training.

Now, the evidence lower bound (ELBO) (Blei et al. 2017) is introduced as an alternative and easily derived formula as

$$ELBO = E_{\phi \sim q}(\log p(D_y|D_x; \phi)) - D_{\mathrm{KL}}(q(\phi)||p(\theta)). \tag{3.2}$$

Maximising the ELBO is equivalent to maximising a lower bound on the log-likelihood of the training samples. In (3.2), the first term E represents Shannon entropy, which means the sum of the expected log-likelihood of the training samples, and it can be obtained by sampling, and the second term is the regularisation loss term provided by the KL divergence, which is a closed form of Gaussian distribution. After the optimisation, the posterior, $p(\theta|D)$, is approximated, and the BNN is trained and ready to be used.

When performing BNN prediction, given the posterior, $p(\theta|D)$, the model's prediction uncertainty quantification can be obtained via $p(y|x, D)$. Practically, this is done by sampling (Gal and Ghahramani 2015), which is expressed mathematically as

$$\theta \sim p(\theta|D). \tag{3.3}$$

The predicted values are averaged BNN model output samples, which is expressed mathematically as

$$\hat{y} = \frac{1}{|\Theta|} \sum_{\theta_i \in \Theta} \Phi_{\theta_i}(x), \tag{3.4}$$

where Θ denotes a set containing all sampled θ $|\Theta|$ denotes the size of the set, $\Phi_\theta(x)$ denotes the BNN model and $\hat{y}$ denotes the estimated output. Moreover, the uncertainty quantification is calculated by using the covariance matrix $\Sigma_{y|x,D}$, which is expressed mathematically as

$$\Sigma_{y|x,D} = \frac{1}{|\Theta| - 1} \sum_{\theta_i \in \Theta} (\Phi_{\theta_i}(x) - \hat{y})(\Phi_{\theta_i}(x) - \hat{y})^T. \tag{3.5}$$

The properties of the BNN model in terms of the three key factors in Chapter 1 for a machine learning alternative used in SAEAs are analysed as follows,

- Prediction accuracy. BNN can be interpreted as a special case of ensemble methods with multiple ANN ensembles (Zhou 2012). Ensemble methods are well known for taking advantage of aggregating multiple averaged and independent predictors that may outperform a single expert predictor, given the same training information (Breiman 1996). A few ensemble learning methods using ad-hoc learners offer high performance under SAEAs settings (Guo et al. 2018; Wang et al. 2018). Hence, it is natural to infer that BNNs' stochastic components or ensembles improve a standard ANN in a similar way. Moreover, it is observed that BNN, to a large extent, avoids overfitting when learning from a small dataset. A small dataset aligns with the SAEA nature, where only a limited number of real and expensive processes or function evaluations are available for surrogate modelling.
- Prediction uncertainty quantification is available for individual candidate solutions. Compared to the ad-hoc ensemble models and dropout-based methods, the prediction uncertainty quantification of BNN is statistically validated (Westermann and Evins 2021; Jospin et al. 2022). Although estimated by sampling, the BNN model's prediction uncertainty quantification can be derived by $p(y|x, D) = \sum_\theta p(y|x, \theta')p(\theta'|D)\mathrm{d}\theta'$.
- Training cost. Even when the number of decision variables is large (e.g. over 50), BNN training costs are still affordable. For a BNN with two hidden layers in between the input layer and output layer, the computational complexity is $O(N_b d(d + m)^2 s)$, where d denotes the number of input variables, m denotes the number of output variables, s denotes the sampling cost and N_b denotes the number of iterations in training. This Asymptotic computation complexity is much lower than that of ordinary GP. Moreover, for multiple outputs, GP needs to build separate models for each output, whereas BNN can provide multiple outputs on a single BNN prediction, further improving the training efficiency.

The above analysis is empirically verified in Sections 3.4 and 3.5. Furthermore, BNN is not sensitive to hyperparameters thanks to the characteristics of ensemble learning, which is an advantage compared to many other standard ANNs. In this thesis, the BNN is defined with the hyperparameters below and is used for all test problems. The BNN structure has two hidden layers between the input and output layers. There are d and m neurons in the input layer and in the output layers, respectively, to match the numbers of inputs and outputs. The numbers of neurons in the hidden layers are $2d$ (the first hidden layer) and $\max([d, 2m])$ (the second hidden layer), respectively. Moreover, the prior standard deviation defaults to 0.1. The Adam optimiser is used for training with an initial learning rate of 0.05 and a decay rate of 0.999 in every step of the model parameter updates.

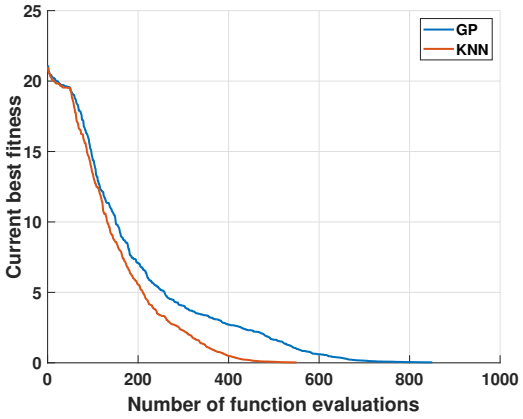
3.3 Selection of the test problem set

To evaluate the performance of an SAEA, both the exploration capability and convergence speed need to be considered. As mentioned in Subsection 2.2.1, there is a trade-off between the two – the greater exploration capability often comes with reduced convergence speed. Also, to approximate a landscape, the more multi-modal, nonlinear, and complex it is, the more training sample points are required to obtain good prediction accuracy and the lower its efficiency. Although the above limitation is somewhat released in SAEAs for the smoothing effect of surrogate models in them (Sun et al. 2015) as only a reasonable ranking of the candidate designs is required sometimes, the limitation generally holds true. In other words, every well-designed SAEA shows an inevitable trade-off between exploration capability and convergence efficiency. For example, optimisation problems with more complex landscapes prefer SAEAs with higher exploration capability but lower efficiency and vice versa.

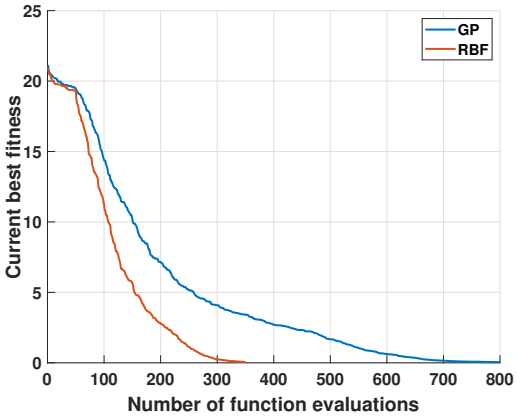
For algorithm study and comparison, mathematical benchmark problems are widely used. They include the Sphere function and Ellipsoid functions as uni-modal functions, Ackley function, Griewank function, Rastrigin function and Rosenbrock function, and some very complex hybrid functions as multi-modal functions (Jamil et al. 2013; Hussain et al. 2017). However, empirically, as said in Chapter 4, real-world engineering problems needing simulations and complex calculations are the primary resources of computationally expensive optimisation problems, which is the benchmark that should be used to test SAEAs. Unfortunately, the characteristics of the above mathemat-

ical benchmark functions and many real-world problems are sometimes mismatched. Considering the exploration capability and convergence efficiency mentioned above, whether the trade-offs reflected on the landscape characteristics are appropriate remains an issue.

Although many real-world simulation-based engineering design problems are multi-modal (Zaharis and Yioultsis 2011; Liu et al. 2017a), very rugged optimisation landscapes, for example, the Rastrigin function (Dieterich and Hartke 2012) and complex hybrid functions, which tend to be relatively discontinuous, seldom appear. The reason is that numerical techniques in the simulation usually solve PDEs, such as Maxwell's equations and Navier-Stokes equations, and such very rugged landscapes can hardly be the result of these processes. Even testing SAEAs against the Ackley function, whose complexity is close to real-world multi-modal problems, is one-sided.



(a) Comparison between GP (95% success rate) and KNN model (95% success rate) under the SMAS framework



(b) Comparison between GP (95% success rate) and RBF model (65% success rate) under the SMAS framework

Figure 3.2: Limitations and usefulness of the Ackley function (20 runs, only successful runs are drawn).

Table 3.1: The selected test problems

Category	Test problems
A	F1: 10-D Ackley optimisation F2: 10-D Griewank optimisation
B	F3: Circular antenna array (Das and Suganthan 2010) optimisation (12-D) F4: 12-resonator diplexer coupling matrix (Wu et al. 2018; Yu et al. 2020) optimisation (27-D)
C	F5: quadruple band mm-wave antenna (Ur-Rehman et al. 2018) optimisation (20-D) F6: Microstrip patch antenna optimisation (62-D) (Jayasinghe et al. 2015)

Figure 3.2a shows optimisation convergence trends of using the SMAS framework with two surrogate modelling alternatives against the 10-dimensional Ackley benchmark function. The two surrogate models are GP and K-nearest neighbour (KNN). When KNN is used as the surrogate model, the average of the Ackley function values of the 20 nearest candidates for a candidate solution is assigned as the predicted value for that candidate solution. We consider reaching a threshold objective function value of 0.5 in 1000 function evaluations as a successful optimisation. The reason and necessity to define a successful run are explained in the next section.

It can be seen that using the KNN as the surrogate model, the SMAS framework-based optimisation converges even faster than that of using the GP as the surrogate model, and both have a 95% success rate. However, the KNN model hardly succeeds in any real-world problems in (Akinsolu et al. 2019; Liu et al. 2021; Liu et al. 2022a). Note that this does not mean that the Ackley function is not useful as a benchmark function. As can be seen in Figure 3.2b, under the SMAS framework, when using the RBF as the surrogate model, the success rate is 65% compared to 95% when using the GP as the surrogate model. This observation broadly aligns with the fact that prescreening introduces additional exploration capability into the search that is discussed in Section 2.5, so it is natural to believe that GP with prescreening methods has a higher exploration capability than using RBF directly alone, whose exploration capability usually needs to be explicitly improved for multi-modal problems.

Note that this section aims to discuss and select appropriate test problems that can better reflect the background problems of expensive engineering optimisation for SAEAs for studying the behaviour of the BNN model instead of specifically focusing on an in-depth investigation of the limitations of existing popular mathematical benchmark test problems. The nearest test problems to engineering design problems are themselves. Hence, the mathematical benchmark problems are only used when considering the computing overhead for statistical analysis and are supplemented by real-world problems. Moreover, when the two results contradict each other, results from testing against real-world problems should be prioritised for the discussion. Four EM design problems are used in this paper as EM simulation is a typical computationally expensive evaluation.

Table 3.1 lists the six test problems used in the following sections, and the six test problems can be classified into three categories, namely, A, B and C. Category A has two typical mathematical benchmark functions, the Ackley function and the Griewank function, and their dimensionalities are set to 10. The functionality of category A is to test low-dimensional problems using mathematical benchmark functions that could

be near to real-world problems. Most real-world engineering applications have design variables between 5 and 20 regarding dimensionality. Therefore, higher-dimensional mathematical benchmark problems are not considered in the test. Their details are shown in the Appendix.

Category B has two simplified real-world problems, namely, a 12-dimensional circular antenna array optimisation problem and a 27-dimensional 12-resonator diplexer coupling matrix optimisation problem. For the first optimisation problem, computationally cheap and simple mathematical calculations for array design are applied to the simulation to simplify the computationally expensive EM simulation, and it can be used as the coarse model in practice. In the simplified problem, although the optimisation landscape is made smoother and less complex, it still reflects most characteristics of the optimisation landscape of EM simulation-based engineering design problems. The first problem has one optimisation objective. The second problem, the 27-dimensional diplexer coupling matrix optimisation problem, is a uni-modal problem included in the test problems to improve diversity. Theoretically, diplexer coupling matrix global optimisation problem is multi-modal, but practically, there are engineering methods, such as the filter design knowledge value used in this test problem (Hong and Lancaster 2004; Skaik et al. 2011; Cameron 2003), to find a good starting point, which makes the search starts around a convex region near the global optimum in the optimisation landscape and thus making it a uni-modal optimisation problem. Technically, based on the starting point, relatively narrow intervals are considered for the search ranges, and the problem is then simplified to a uni-modal problem. The second problem has one optimisation objective. Their details are shown in the Appendix.

Category C has two real-world EM device design problems, namely, a 20-dimensional quadruple band 5G mm-wave antenna and a 62-dimensional microstrip patch antenna, and they are both simulated by CST-MWS. These problems are computationally expensive, but it is essential to include them as they directly reflect the characteristics of such problems. Both of the problems are multi-modal. For the first problem, there are 12 design requirements, and once these design specifications are reached, the optimisation terminates. Considering the dimensionality, engineering problems with larger than 100 design variables in real-world practice are rare as they are routinely decomposed into small sub-problems. The 62 design variable case study also investigates a new EM design methodology, a variant of digitally coded antennas. By randomly changing the vertical positions of microstrip patch units, novel shapes can be formed with a significant level of freedom to satisfy specific design specifications. Therefore, the number of design variables is high in this specific case study. The second test problem has one optimisation objective. Their details are shown in Subsection 5.2 and 5.4.

In the following sections, test problems F1-F6 are used to carry out experiments. All experiments are conducted on a workstation with an AMD Ryzen™ Threadripper™ PRO (2.7GHz) CPU and an NVIDIA® RTX™ A4000 GPU. 80 MATLAB parallel workers are available for GP/BNN-based surrogate model training.

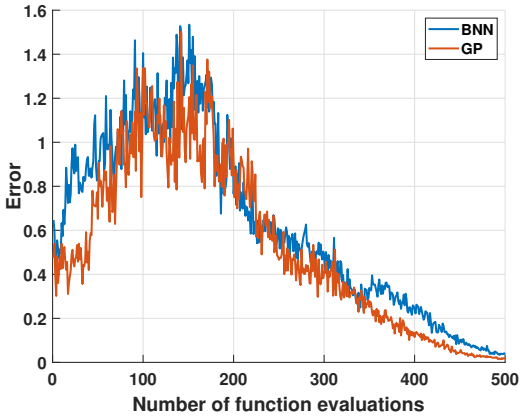
3.4 Characteristics of BNN prediction and its co-work with prescreening methods

3.4.1 Characteristics of BNN prediction

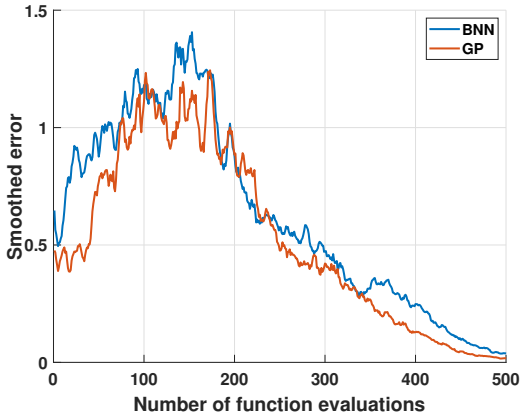
In this subsection, the prediction accuracy and uncertainty quantification of the BNN model are investigated. GP is considered as the reference surrogate. First, under the SMAS framework, the LCB prescreening method is used with $\omega = 2$ after the GP surrogate model prediction. Twenty optimisation experiments are carried out for F1-F4. As F1-F4 shows the same conclusion, a typical optimisation convergence trend from F1 is selected, and the population in each iteration throughout the convergence is obtained as the data to be analysed. Due to the unaffordable computation cost for F5 and F6 here, where more than 1000 EM simulations for each run are needed, these two test cases are not included in this experiment.

Figure 3.3a shows the trends of averaged Euclidean distance between the predicted value of each candidate solution and its true function value in the parent population of each iteration throughout the optimisation convergence. The plotting rule applies to both GP prediction and BNN prediction using the same training data set. Figure 3.3b shows the smoothed curve of Figure 3.3a using a moving average filter with a span 20. Figure 3.3c shows the prediction uncertainty quantification, as described by standard deviation.

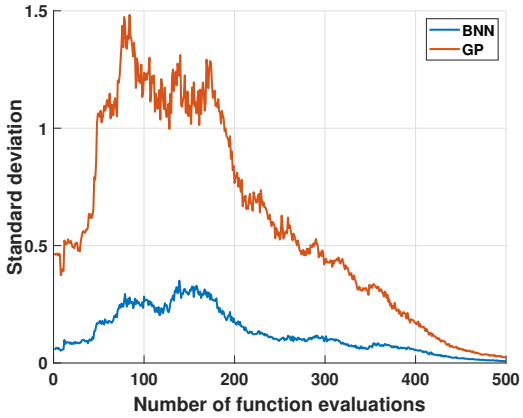
From the plots, the following can be observed. (1) As for the predicted values, the prediction accuracy of the GP model and BNN model are comparable throughout the optimisation. However, GP shows a slight advantage. This can be explained as when the number of training samples is less than sufficient for BNN training, particularly at the beginning phase, the GP model slightly outperforms the BNN model in prediction for having a more solid mathematical background. When the number of training samples



(a) Average error to ground truth for the population in each iteration (GP vs. BNN)



(b) Smoothed curve of (a)



(c) Average prediction uncertainty (standard deviation) for the population in each iteration (GP vs. BNN)

Figure 3.3: GP and BNN predicted values and prediction uncertainty comparison using the F1 test problem.

increases in the later iterations, the prediction accuracy of BNN predictions improves relatively compared to that of GP; (2) In regards to the prediction uncertainty quantification, the GP and BNN models show a similar trend throughout the optimisation process. This tendency can be explained as the exploration capability dominating at the early stage of the optimisation, so the prediction uncertainty is high, while at the late stage of the optimisation, it is converging, and the samples are close to each other. Note that the standard deviations of BNN predictions are much less than that of GP predictions. Given that the capability of GP prediction uncertainty is widely recognised and is proved by many successful SAEA applications and given that the similar tendency in Figure 3.3c being observed, the BNN prediction uncertainty is likely to be helpful in prescreening under SAEAs. The standard deviations with a similar trend but different magnitude open the investigation of the co-work of BNN and prescreening methods.

3.4.2 The characteristics of the BNN model working with prescreening methods

Prescreening methods make use of prediction uncertainty quantification to promote exploration. The three most widely used prescreening methods are EI, PI, and LCB, which are investigated in this subsection. EI and PI do not have hyperparameters involved, and their extents of exploration are fixed. LCB, however, has a variable coefficient, ω, which determines the extent of the exploration. In this subsection, (1) for the global optimisation, the characteristics of avoiding premature convergence by the BNN as the surrogate model under SAEAs when it comes to working with the prescreening methods, and (2) the exploration performance of the BNN surrogate model under SAEAs when it comes to working with EI, PI and LCB, are investigated.

First, the metric for success rate is defined before the empirical studies. Often, the averaged convergence curves over several runs are used to compare different algorithms in the SAEA domain. It is assumed that premature convergence in some optimisation can be reflected by the averaged convergence curve to have a worse averaged fitness value and slow convergence. Despite being correct and used in some cases (e.g., F4 and F6, in this chapter), the background of simulation-based engineering design optimisation, which is a primary resource for computationally expensive optimisation, is supplemented for discussion below.

In engineering design, the primary goal is to meet the preset design specifications. Otherwise, the convergence trend is not valuable for the quantitative convergence investigation if the design requirements are not satisfied for a long time. Here, we define success rate as the percentage of repeated runs that end up satisfying the preset design requirements. Therefore, for example, in practical engineering optimisation, an algorithm that converges slowly but has a high success rate (e.g., 90%) is superior to others with much faster convergence but a low success rate (e.g., 60%). Considering the design requirements and success rate, we set threshold values for F1, F2, F3, and F5. When a run reaches the threshold value, it is considered a successful optimisation. The success rates are considered in comparisons.

For F1 and F2, the threshold value is set to be 0.5, assuming that reaching a value below 0.5 satisfies the Ackley and Griewank functions. F3, which is originally from the CEC 2011 competition (Das and Suganthan 2010), is the aggregation of several array performances. Although each performance has a separate specification in real practice,

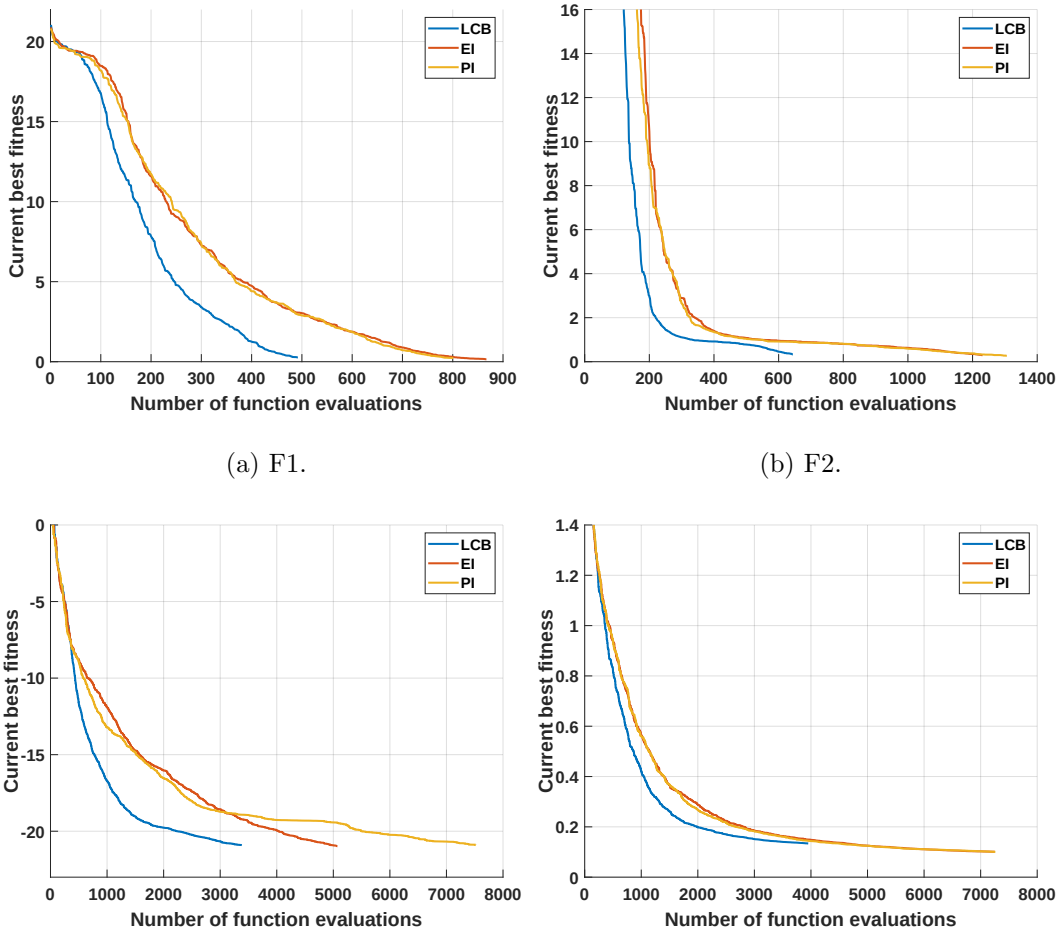
the overall fitness value -20 is used as the threshold as designs with function values less than -20 usually have acceptable values for all specifications. F4 is a uni-modal problem, and a sufficient number of iterations is allowed, so no threshold is necessary for this case. F5 is a practical antenna design problem, and it has multiple design requirements simultaneously. F6 is an open research question that explores novel designs that can possibly lead to maximum bandwidth, so it has a single design objective function, and no threshold is used in this case either.

For LCB, the value of ω trades off the exploration capability and convergence efficiency. Regarding the value of ω to represent the performance of LCB with the BNN model, a parametric sweeping for the value of ω is carried out from 0, using 0.5 as the step size, and finding the smallest possible value of ω, which theoretically has the best convergence efficiency, that has over 90% success rate for F1, F2, F3 and F5. As such, the obtained ω leads to sufficient but not excessive exploration capability. ω is set to 0 for F4 as it is uni-modal, and for F6 as it is an open research problem, and the antenna design problem formation already allows certain levels of exploration, and practically, using $\omega = 0$ already obtains promising findings from the microwave engineering point of view.

Table 3.2: Performance of EI, PI and LCB when co-working with BNN

Problem	Method	Best	Worst	Mean	Std
F1	EI	2.2624	4.0267	3.0715	0.4405
	PI	1.6517	3.7445	2.9653	0.4988
	LCB	**0.1113**	**0.5953**	**0.2668**	0.1267
F2	EI	0.7544	1.0109	0.9318	0.0695
	PI	0.5110	0.9954	0.9036	0.1170
	LCB	**0.0311**	**0.7661**	**0.3321**	0.2812
F3	EI	-21.5800	**-13.4888**	-19.2204	3.1050
	PI	-21.5099	-10.5363	-18.9222	4.1176
	LCB	**-21.6270**	-12.9220	**-20.9069**	1.9572
F4	EI	0.0814	0.2390	0.1495	0.0409
	PI	0.0893	**0.2045**	0.1446	0.0322
	LCB	**0.0557**	0.2130	**0.1339**	0.0462

F1-F4 are used to compare EI, PI, and LCB with the ω obtained by the above criteria, and 20 optimisation runs are carried out. Figure 3.4 shows the averaged convergence curves. Because the properties of various prescreening methods are studied when co-working with the BNN model, instead of observing the same number of function evaluations, the results after convergence are needed. Therefore, Hopkins test (Hopkins and Skellam 1954) is introduced as the stopping criterion for F1-F4, and the optimisation



(a) F1. (b) F2.

(c) F3. (d) F4.

Figure 3.4: Performance comparisons among LCB, EI and PI working with BNN (average of 20 runs, and only successful runs are drawn. All success rates are higher than 90%.)

terminates when the H-measure values reach 0.995. This leads to different iterations and function evaluations for each alternative. On the other hand, for comparison purposes, the statistical results shown in Table 3.2 utilise the standard stopping criterion determined by the number of iterations of the fastest converging method.

The following can be observed from the experiment results. (1) The BNN-assisted SAEA obtains high-performance results when co-working with all three prescreening methods. According to Figure 3.4 and Table II, the obtained solutions are close to the global optimum or the desired fitness value when H-measure values reach 0.995. (2) Co-working with LCB performs better than EI and PI when using an appropriate ω. (3) Using the BNN as the surrogate model, when ω is set as 0.5, success rates for F1, F2 and F3 are 100%, 100% and 95%, respectively. F4 is a uni-modal problem, so optimisation should always be successful. This indicates that the magnitudes of prediction uncertainty of BNN are not too little for SAEAs to carry out global optimisation.

3.5 Comparison of BNN, GP and the NNDO method in SAEA

This subsection empirically compares the BNN model with other popular machine learning alternatives under SAEAs. In Chapter 1, GP models, RBF models, ANN models with the NNDO method and ensemble methods are reviewed as frequently used machine learning alternatives. Among them, the RBF model is not included in the comparison as the RBF model, as discussed in Section 3.3, cannot provide prediction uncertainty quantification for each candidate solution, which highly influences the exploration ability of an SAEA. Hence, many successful RBF-based SAEAs need ad-hoc supplements to "generate" prediction uncertainty (Qin et al. 2021). There are various ensemble methods, and the design is ad-hoc. Although ensemble methods with various algorithms are possible for SAEAs, this thesis discusses and compares the characteristics of a single machine learning alternative. The NNDO method is popular in the machine learning domain (Hara et al. 2016). Therefore, GP models and the NNDO methods are selected as the machine learning alternative for the comparison.

With the test problems selected in Section 3.3 and the SMAS framework introduced in Chapter 4, LCB is used for BNN, GP and the NNDO method, and the idea mentioned in Section 3.4 is borrowed. For each problem, carry out a parametric sweep for the value of ω, starting from 0, with increments of 0.5 until the success rate reaches 90%. In this case, the obtained value of ω provides sufficient but not excessive exploration capability. In other words, the obtained ω does not significantly slow the convergence for these particular problems. The values of ω for F1-F6 and the machine learning alternatives are obtained through case-by-case experiments and thus are case-dependent. The ω values are fixed for the test problems and for various machine learning alternatives, and the convergence efficiency and the quality of the optimal solutions are obtained for statistical comparison.

With respect to settings and hyperparameters, for GP models, standard GP (Rasmussen and Williams 2006) is used in the section. Sparse GP (Bauer et al. 2016) or similar methods are not used as they sacrifice the prediction accuracy, while TP (Shah et al. 2013) is not used as it does not address the issue of the high training cost of GP modelling. For the NNDO method, the ANN structure is similar to that of BNN, and

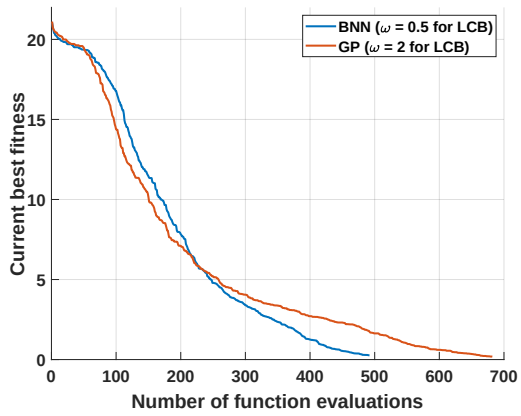
the dropout rate is 0.2, as suggested by (Gal and Ghahramani 2016). Considering the computing cost, 20 optimisation runs are carried out for F1-F4, and 10 optimisation runs for F4 and F5. More optimisation runs are unaffordable as real EM simulations are used for F5 and F6, and they cost more than two months on multiple workstations.

Hopkins test (Hopkins and Skellam 1954) is applied as the stopping criterion for F1, F2, F3 and F4, and the optimisation terminates when the H-measure values reach 0.995. For the real-world problem F5, the satisfaction of the design requirements is considered the stopping criterion. For the real-world problem F6, a computation budget of 1250 EM simulations is given to the optimisation as the stopping criterion. Considering that there are 62 design variables, it is unaffordable to use a high H-measure value. Given that the problem does not have design specifications, satisfactory results can be obtained using 1250 EM simulations from a microwave engineering point of view.

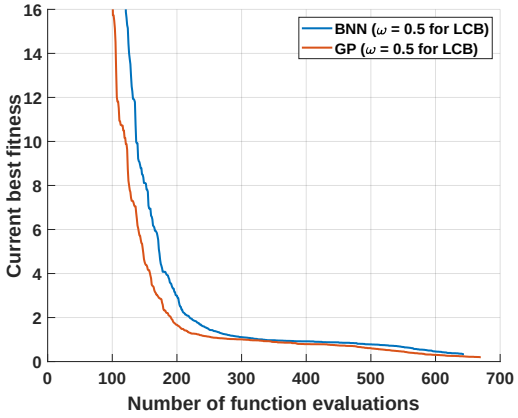
Table 3.3: Comparison between GP and BNN as surrogate models

Problem	Method	Best	Worst	Mean	Std
F1	BNN	**0.1113**	**0.5953**	**0.2668**	0.1267
	GP	0.3237	3.3887	1.7409	0.9379
F2	BNN	0.0311	**0.7661**	0.3321	0.2812
	GP	**0.0240**	0.9311	**0.2341**	0.2843
F3	BNN	**-21.6270**	**-12.9220**	**-20.9069**	1.9572
	GP	-21.3830	-8.6491	-16.6069	5.0281
F4	BNN	**0.0557**	0.2130	0.1339	0.0462
	GP	0.0567	**0.1698**	**0.1124**	0.0345
F5	BNN	**641**	**1567**	**1126.1**	274.1
	GP	1196	3016	2230.9	610.9
F6	BNN	**-60.7676**	**-55.0107**	**-58.4222**	1.6082
	GP	N.A.	N.A.	N.A.	N.A.

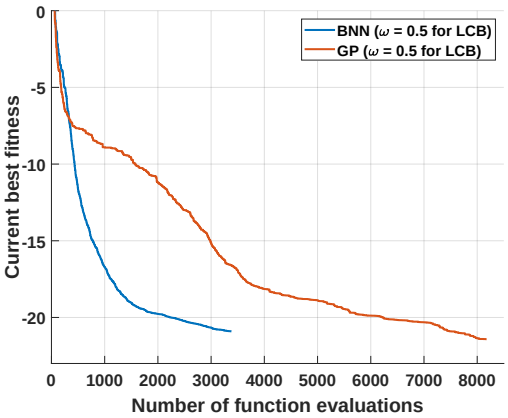
[1] For F5, the shown values are the number of EM simulations needed to meet the design specifications. For other test problems, the shown values are the fitness values when the computation budget is exhausted. The computation budget is decided by the faster-converging alternative.



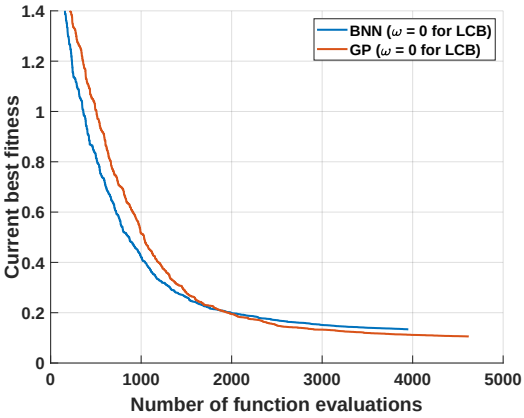
(a) BNN versus GP (F1).



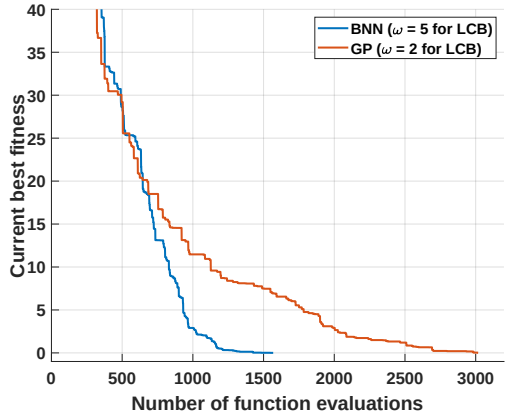
(b) BNN versus GP (F2).



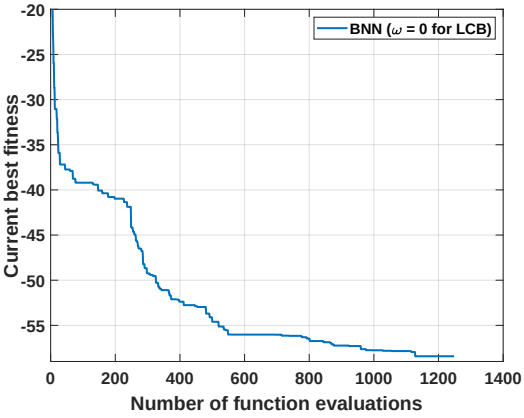
(c) BNN versus GP (F3).



(d) BNN versus GP (F4).



(e) BNN versus GP (F5).



(f) BNN (F6).

Figure 3.5: Performance comparison between BNN and GP (Only the average of successful runs are drawn. All success rates are higher than 90%.)

Table 3.4: Total training and simulation time of GP and BNN for the test problems

	BNN-assisted optimisation			GP-assisted optimisation		
Problems	Sim. time (h)	Training time (h)	Total optim. time (h)	Sim. time	Training time (h)	Total optim. time (h)
F1	N.A.	0.62	0.62	N.A.	0.12	**0.12**
F2	N.A.	0.90	0.90	N.A.	0.14	**0.14**
F3	N.A.	3.52	3.52	N.A.	1.55	**1.55**
F4	N.A.	6.89	**6.89**	N.A.	146.4	146.4
F5	44.10	2.44	**46.54**	87.38	111.14	198.52
F6	50.96	13.66	**64.62**		unaffordable	

a. The simulation time for F5 is different from that of GP and BNN because the necessary number of EM simulations to meet the design specifications are different.

3.5.1 Comparison between BNN and GP models

For the comparison between BNN model and the GP model, Figure 3.5 shows the convergence trends with the stopping criterion mentioned above. Table 3.3 shows the statistics, in which, except for F5, which compares the number of function evaluations to satisfy the constraints, for all other problems, the number of function evaluations of the faster-converging alternative is specified for both alternatives to compare the fitness values. The simulation and training time are shown in Table 3.4, considering the convergence of both alternatives.

The following can be observed from the test results. (1) Under the SMAS framework, SAEAs with the BNN model and GP model as the surrogate can obtain promising results for all the test problems but F6, for which GP training is unaffordable. (2) Among the multi-modal mathematical benchmark problems, BNN shows a clear advantage for F1 – approximately 20% less function evaluations needed for convergence – over GP and comparable performance for F2 to GP. (3) Among multi-modal real-world problems, BNN shows a clear advantage over GP – more than 50% less function evaluations for F3 and F5. (4) For the uni-modal problem F4, BNN shows a slightly slower convergence speed than that of GP. (5) Regarding the cost of training, GP has advantages for low-dimensional problems (i.e., F1, F2, F3). On the other hand, for high-dimensional problems like F4, which has 30 decision variables, the GP training cost is much worse than that of BNN due to the "curse of dimensionality". Furthermore, when there are multiple sub-functions to learn in the problem (i.e., F5), separate models need to be built if GP is the surrogate. For BNN, however, only a single network is needed, which is much computationally cheaper (see Section 3.2). For F6, as it has 62 decision variables, which is many, modelling with GP is unaffordable, while modelling with BNN still works in a practical time frame.

BNN's potential is identified from the above observations. In regards to solution quality, BNN is comparable to GP. Furthermore, it has the advantage of better convergence efficiency. This may come from the comparable accuracy and smooth prediction uncertainty of the BNN model compared to the GP model. By primary observation, at the early stage of the convergence, BNN and GP often show close performance, while BNN sometimes outperforms GP in the later stage. This can be explained as the comparably smaller prediction uncertainty decreasing the exploration and increasing the convergence efficiency. Fortunately, it is still sufficient in the later stage of the optimisation to avoid premature convergence.

On the other hand, when decreasing the values of ω for GP and BNN leads to a decreased success rate. This is evidenced by F1 and F5, where the success rate can drop to 70% with a smaller ω value for GP. This is also demonstrated in Figure 3.3c in Subsection 3.4. When dividing the prediction uncertainty values of GP to match the late-stage convergence characteristics of the BNN model, the uncertainty at the middle stage is still unnecessarily large if assuming that of BNN is sufficient. Therefore, an assumption can be made as follows: the prediction uncertainty quantification of BNN is advantageous compared to that of GP when being used as the surrogate in some SAEAs.

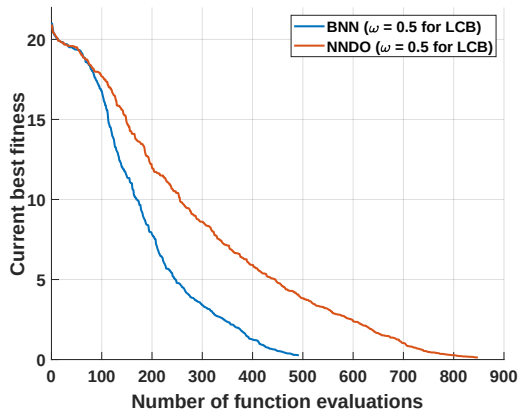
Table 3.5: Comparison between ANN with drop-out and BNN as surrogate models

Problem	Method	Best	Worst	Mean	Std
F1	BNN	**0.1113**	**0.5953**	**0.2668**	0.1267
	NNDO	2.6145	5.1328	3.9813	0.7870
F2	BNN	**0.0311**	**0.7661**	**0.3321**	0.2812
	NNDO	0.7710	1.0850	0.9867	0.0884
F3	BNN	**-21.6270**	**-12.9220**	**-20.9069**	1.9572
	NNDO	-21.1786	-12.7178	-14.8973	2.6706
F4	BNN	**0.0557**	**0.2130**	**0.1339**	0.0462
	NNDO	0.1839	0.3572	0.2534	0.0433
F5	BNN	**641**	**1567**	**1126.1**	274.1
	NNDO	1057	4600	2238.3	1272.3
F6	BNN	**-60.7676**	**-55.0107**	**-58.4222**	1.6082
	NNDO	-60.5544	-52.3454	-57.3454	2.9107

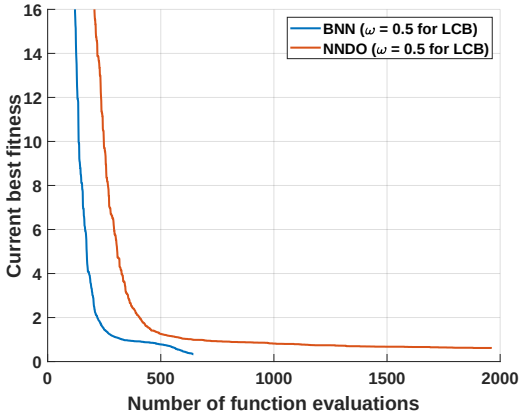
[1] For F5, the shown values are the number of EM simulations needed to meet the design specifications. For other test problems, the shown values are the fitness values when the computation budget is exhausted. The computation budget is decided by the faster-converging alternative.

3.5.2 Comparison between BNN and the NNDO method

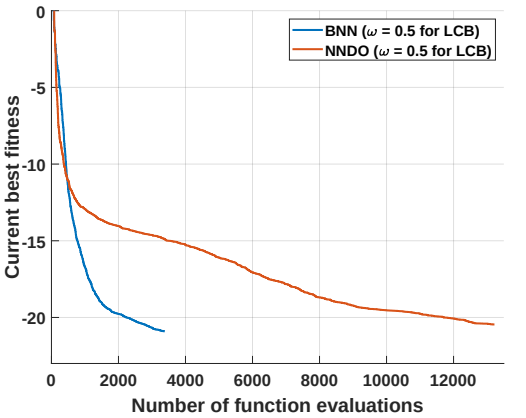
The NNDO method is popular in the deep learning domain, and it is widely used for Bayesian optimisation and discussed by the machine learning community (Gal and Ghahramani 2016). However, BNN is considered computationally expensive regarding the cost of training (Jia et al. 2020). Fortunately, BNN is computationally expensive, mainly under the background of deep networks involving large training datasets. In



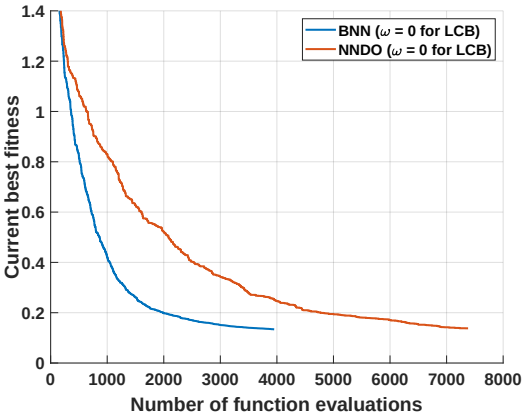
(a) BNN versus NNDO (F1).



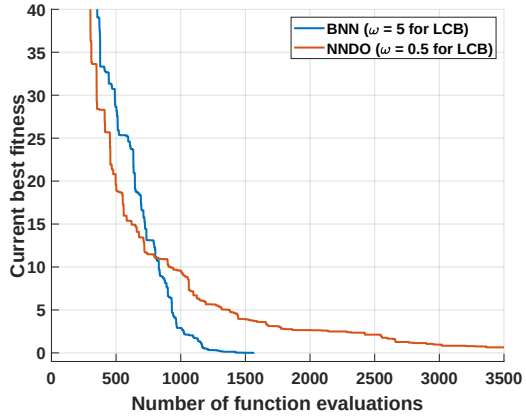
(b) BNN versus NNDO (F2).



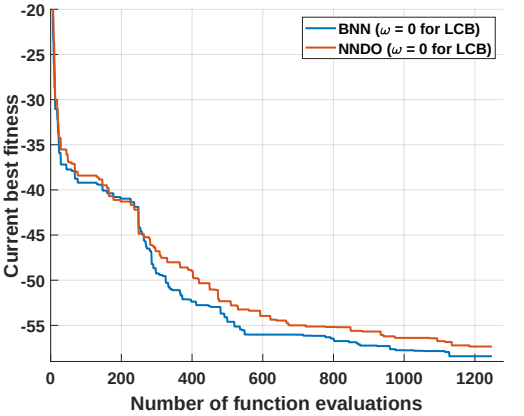
(c) BNN versus NNDO (F3).



(d) BNN versus NNDO (F4).



(e) BNN versus NNDO (F5).



(f) BNN versus NNDO (F6).

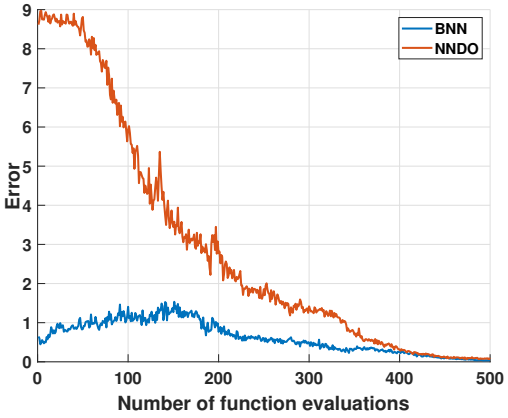
Figure 3.6: Performance comparison between BNN and NNDO (Only the average of successful runs are drawn. All success rates are higher than 90%.)

Table 3.6: Total training and simulation time of NNDO and BNN for the test problems

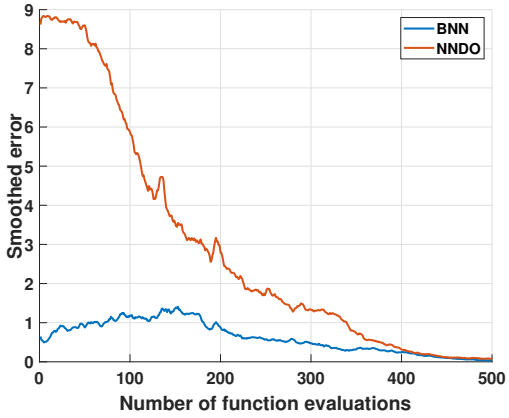
	BNN-assisted optimisation			Drop-out-assisted optimisation		
Problems	Sim. time (h)	Training optim. time (h)	Total time (h)	Sim. time	Training time (h)	Total optim. time (h)
F1	N.A.	0.69	**0.69**	N.A.	0.94	0.94
F2	N.A.	0.90	**0.90**	N.A	3.29	3.29
F3	N.A.	3.52	**3.52**	N.A.	29.56	29.56
F4	N.A.	6.89	**6.89**	N.A.	19.52	19.52
F5	44.10	2.44	**46.54**	91.57	7.90	99.47
F6	50.96	13.66	**64.62**	50.96	19.50	70.46

a. The simulation time for F5 is different because the numbers of EM simulations needed using the two surrogate models are different.
b. The simulation time for F6 is identical because the computational budgets are the same for F6 using the two surrogate models, and the averaged simulation time is used.

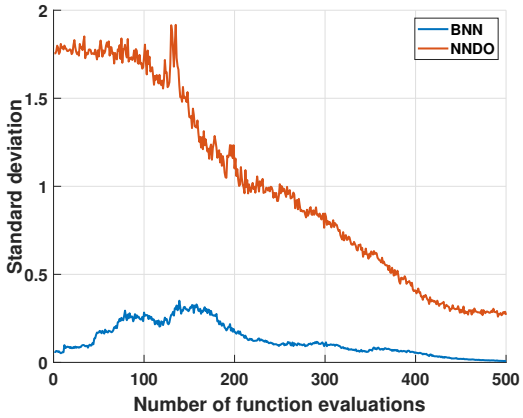
contrast, the surrogate model training under SAEAs ought to be done with "small" data rather than big data. In fact, the main reason such optimisation problems need SAEAs is that their function evaluations are computationally expensive; thus, having large datasets is impossible in the first place. Hence, the comparison between the performance of BNN and NNDO under SAEAs is presented. The comparison and the computation budget are the same as in Subsection 3.5.1. Figure 3.6 shows the convergence trends, and Table 3.5 shows the corresponding statistics. Table 3.6 shows the simulation and training time.



(a) Average error to ground truth for the population in each iteration (NNDO vs. BNN)



(b) Smoothed curve of (a)



(c) Average prediction uncertainty (standard deviation) for the population in each iteration (NNDO vs. BNN)

Figure 3.7: NNDO and BNN predicted values and prediction uncertainty comparison using the F1 test problem.

The test results show that the convergence efficiency of the NNDO method is worse than that of BNN and GP for all test problems. This can be explained with the prediction accuracy and prediction uncertainty quantification experiments. Similar to Section 3.4, for the NNDO method, similar to the experiment in Subsection 3.4.1, Figure 3.7a shows the averaged Euclidean distance between the predicted value of each candidate

solution and its true function value in the parent population of each iteration throughout the optimisation convergence and Figure 3.7b shows the prediction uncertainty quantification as described by the standard deviation. A typical run of F1 is used similar to Subsection 3.4.1.

From the experiment, it can be observed that the predicted values and prediction uncertainty quantification of the NNDO method are much worse than those of the BNN model. This might be because it is not fully validated mathematically and of less rigour for the NNDO method (Folgoc et al. 2021) compared to BNN, although the NNDO method can also be considered as an ensemble. Regarding the cost of training, both methods show similar performance. The total training time of NNDO in Table 3.6 is long because more iterations are needed. Therefore, under the SAEA background, the BNN model works better than the NNDO method as a surrogate.

3.5.3 Comparison discussion

In conclusion, the BNN model has good performance for all three critical factors, (1) surrogate model prediction accuracy, (2) the availability of prediction uncertainty quantification, and (3) the cost of training, while one of them is often a challenge for existing popular machine learning alternatives as surrogate under SAEAs. Regarding prediction accuracy, BNN is comparable to GP based on the experiments. However, empirically, its prediction uncertainty yields better convergence performance than the GP model under the SMAS framework. Regarding the cost of training, using BNN handles the training time issue of GP when the dimensionality is large. Compared to RBF, the primary advantage of BNN is the availability of prediction uncertainty estimation. Although the cost of training for BNN is larger than that of RBF, it is acceptable in practice even when the dimensionality is large. The major challenge of the NNDO method is the quality of prediction accuracy and prediction uncertainty estimation, and BNN outperforms the NNDO in this aspect.

Therefore, although BNN is not the best in prediction accuracy and training cost, BNN is competitive when used in SAEAs thanks to its good performance in all three critical factors and its high quality in prediction uncertainty quantification.

The above observations are based on a typical SAEA framework and six test problems, and they are carefully selected. Hence, it is too early to provide a complete conclusion about the superiority of the BNN model as a surrogate model for all, which is not the purpose of this research. Instead, the aim is to demonstrate the potential of BNN, which is largely overlooked in the SAEA community, and to inspire investigations considering various SAEA frameworks and test problems.

Chapter 4

Bayesian neural network based self-adaptive surrogate model-assisted evolutionary algorithm for antenna design exploration

Based on the behavioural study of Bayesian neural networks discussed in Chapter 3, this chapter introduces a surrogate model-assisted evolutionary algorithm for antenna design exploration based on Bayesian neural networks. Section 4.1 introduces the overall flow of the algorithm framework. The details of its two major innovations are introduced in the following sections. Section 4.2 introduces the way Bayesian neural networks are applied to the proposed algorithm and illustrates the performance of the Bayesian neural networks with a pilot experiment. The proposed algorithm also utilises a self-adaptive lower confidence bound method for the prescreening, which is discussed in details in Section 4.3. By the end of the Chapter, Section 4.4 lists all the parameters involved in the algorithm.

4.1 The algorithm framework

In Chapter 3, the properties of BNN are illustrated through comparisons with other popular machine learning surrogate alternatives for SAEAs. As in Chapter 3, the SMAS framework (Liu et al. 2013, 2014d) is selected as a typical one and employed in this thesis as one of the most important building block of the proposed framework – SB-SADEA. The proposed framework should work universally for different antenna cases with various number of design variables and design specifications. Figure 4.1 shows the SB-SADEA framework, and a brief description is as follows.

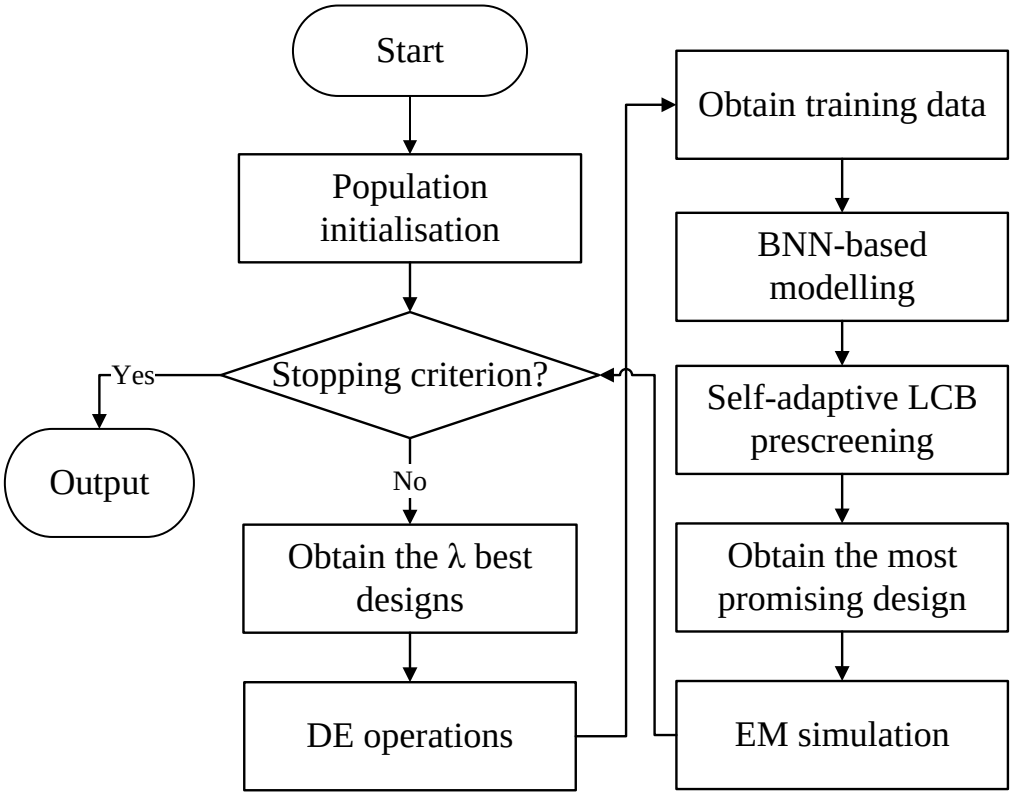


Figure 4.1: Flow diagram of the SMAS framework

Step 1: Obtain α candidate designs from the design space $[LB, UB]^d$ (LB and UB are the lower and upper bounds of the design variables, respectively) using Latin Hypercube sampling (Stein 1987; Queipo et al. 2005) and carry out EM simulation to obtain their performance values. Eventually, these samples form the initial population database.

Step 2: If a preset stopping criterion is met (e.g., satisfied antenna design specifications, exhausted computation budgets), output the current best candidate design from the population database; otherwise, go to Step 3.

Step 3: Obtain the λ best candidate designs from the population database to form a parent population P.

Step 4: Apply the DE/current-to-best/1 operator (Storn and Price 1997) on P to generate λ child solutions.

Step 5: For each child solution, obtain τ nearest samples (based on Euclidean distance) from the database as the training samples and train a BNN-based surrogate model with them (see Section 4.2).

Step 6: Prescreen the child solutions generated in Step 4 using the BNN-based surrogate model and the self-adaptive LCB method (see Section 4.3).

Step 7: Carry out an EM simulation to the estimated best child solution from Step 6. Append this evaluated candidate design and performance values into the database as a new sample. Go back to Step 2.

The SMAS framework was introduced to the SAEA domain in 2013 (Liu et al. 2013). SMAS aims to improve the appropriateness of the locations of training samples and to utilise the population database better to improve prediction accuracy. Because the λ current best candidate solutions determine the current population in each iteration, the search is guided to the promising region of the optimisation space. As only one candidate solution is modified in the parent population in each iteration, the suggested best candidate in the child solutions in several consecutive iterations is assumed to be closer to each other than using a standard EA parent population updating scheme. The suggested best candidate solution by prediction is input to the real function evaluation and thus serves as a training sample in the later iterations. Therefore, the training samples around the current promising area are likely denser than those in SAEAs using the standard EA parent population updating scheme, which may spread in wider regions of the optimisation landscape. Being trained with the same amount of training samples, the surrogate models are expected to have higher accuracy considering the samples of these training data points (Liu et al. 2013, 2014d; Akinsolu et al. 2019). On the other hand, to compensate for the exploration capability, a larger DE scaling factor is used. The idea of SMAS has been adopted by several recent successful SAEAs, e.g., (Yang et al. 2019; Chen et al. 2021; Luo et al. 2018; Zhan and Xing 2021).

The SMAS framework is also known for its engineering applications. Due to "no free lunch", all SAEA frameworks have trade-offs between the exploration ability and the convergence efficiency, which is usually reflected by the number of real function evaluations used. Therefore, their performance is different for different characteristics of the optimisation landscape. SAEAs are mainly used for computationally expensive optimisation problems, and a main resource for computationally expensive evaluations is complex numerical techniques and processes such as finite element analysis (FEA) to solve PDEs, frequently appearing in engineering design and analysis. Therefore, the engineering background provides the standards to trade off the exploration ability and

the convergence efficiency. Although engineering domain knowledge plays an essential role to support SAEAs, SAEAs borrowing ideas from the SMAS framework show effectiveness for real-world engineering design, e.g., (Liu et al. 2021; Xue et al. 2022; Liu et al. 2017a).

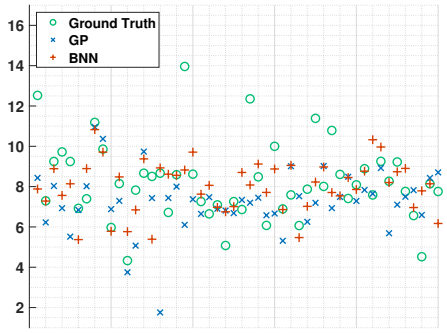
The following parameters are used for the SMAS framework according to (Liu et al. 2013, 2014d) is also applied to AB-SADEA. They are: $\alpha = 5 \times d$, $\lambda = 5 \times d$, $\tau = 5 \times d$, $F = 0.8$, $CR = 0.8$, where d is the number of decision variables, F and CR are DE crossover and mutation rate.

4.2 Bayesian neural network as the surrogate model

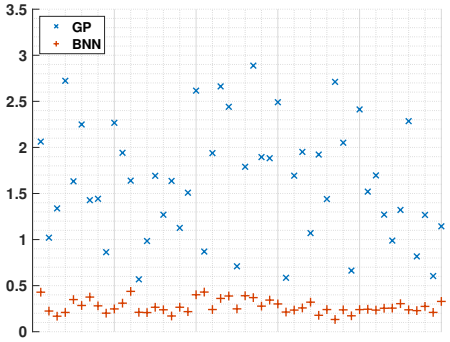
BNN is a novel surrogate model under SAEAs, and its data characteristics could be different from those of GP concerning the predicted values and prediction uncertainty quantification. A characteristics study of BNN is needed to propose a prescreening method that can avoid premature convergence and improve the BNN-based model's convergence efficiency. Using the compact UWB slotted monopole antenna optimisation case study in Section 5.1 as an example, to demonstrate the performance of GP and BNN, Figure 4.2 shows the predicted values and prediction uncertainty for three sample populations of candidate designs during the early, middle, and late phases of the optimisation process. The predictions are made on $\max(|S_{11}|)$. It can be observed that:

- Regarding the predicted values, the BNN-based and GP-based models are comparable, and both show reasonably high prediction accuracy compared to the ground truth simulated values considering all three sample populations at different phases.
- Regarding the prediction uncertainty quantification, the BNN-based model shows much lower values than that of GP, and the difference between them is much more significant during the later phase. For example, when the optimisation is at its early phase, the BNN prediction uncertainty is roughly at the magnitude of 0.2-0.3, while the GP prediction uncertainty is roughly at the magnitude of 1.5-2. During late phase populations, 0.05 for BNN and 0.05 for GP.

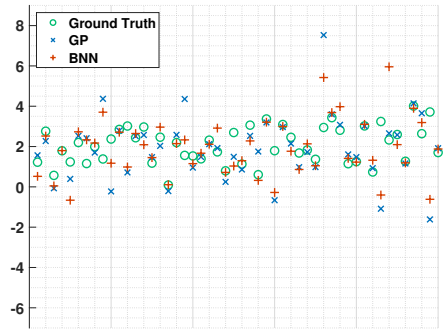
Therefore, a different LCB ratio should be used instead of directly borrowing from GP when BNNs are the surrogate model under SAEAs.



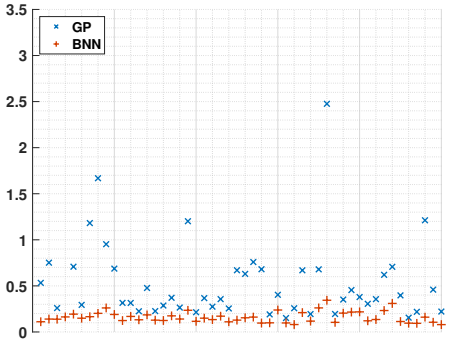
(a) Early stage S_{11} predicted values.



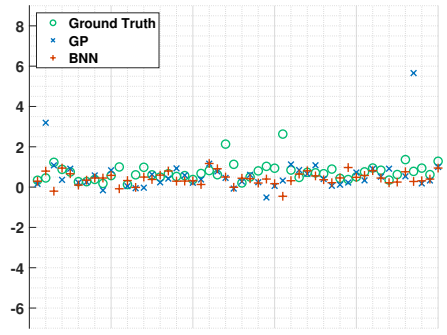
(b) Early stage S_{11} prediction uncertainties (standard deviation).



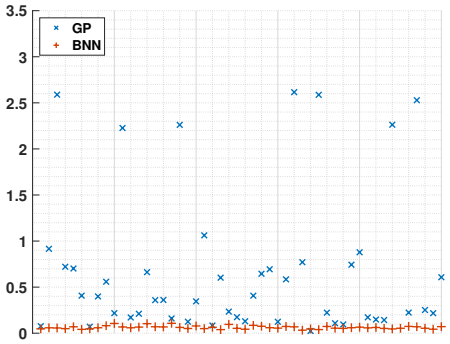
(c) Middle stage S_{11} predicted values.



(d) Middle stage S_{11} prediction uncertainties (standard deviation).



(e) Late stage S_{11} predicted values.



(f) Late stage S_{11} prediction uncertainties (standard deviation).

Figure 4.2: GP and BNN predicted values and prediction uncertainty during early, middle and late stage of the optimisation (ground truth is from EM simulations).

Furthermore, following Section 3.2, due to the significantly reduced cost of training for the BNN, inspired by deeply supervised nets (Lee et al. 2015; Zhou et al. 2018) for image recognition, an idea we call "fine supervision"is proposed. Often, the antenna response over the operating band is considered as a whole, and only a maximum or a minimum is obtained as the antenna performance (e.g., $\max(|S_{11}|)$). By extracting only a point to represent an array of values on the operating band, much information is omitted. In the proposed fine supervision, the frequency response curve is separated into several sections (e.g., by the number of resonances). For each section of the frequency response curve, its maximum or minimum value is obtained for later use. This is helpful for cases like UWB antennas whose frequency responses are broad in frequency and are dividable. In this way, much more information is extracted and included in future model training, with the cost of the increased number of design specifications. This is a significant burden for GP modelling as the number of surrogate models linearly increases with the number of design specifications because GPs output multiple performance predictions separately. However, for BNN, this is affordable as only the number of neurons in the output layer is increased. This is discussed in Section 3.2 with asymptotic analysis and is verified by the quadruple-band 5G mm-wave antenna optimisation case study in Section 5.2.

4.3 The self-adaptive lower confidence bound mechanism

An ad hoc prescreening method (Step 6 in SB-SADEA in Section 4.1) is often needed for a machine learning core, considering the data characteristics. In this thesis, the prescreening for the BNN-based surrogate model is analysed. Thanks to the investigation in the above Section 4.2, it is primarily concluded that a different LCB ratio should be used when the BNN-based surrogate model is used under SAEAs. Usually, the reason for a surrogate model-assisted antenna global optimisation method to converge into local optima is the lack of exploration ability. In the optimisation theory, the exploration capability refers to exploring the search region currently unknown to the search engine, while the exploitation capability refers to searching for the optimum in the search region with sufficient information. Antenna design optimisation landscapes are usually multi-modal and highly complex, and thus, strong exploration capability is required (Akinsolu et al. 2019). Utilising the prediction uncertainty quantification is essential for exploration, which is why prescreening methods exist. With the prediction uncertainty obtained, widely used prescreening methods including EI (Jones et al.

1998), PI (Ulmer et al. 2003), and LCB (Dennis and Torczon 1997). However, there are no hyperparameters controlling the extent of exploration for the popular EI (Jones et al. 1998) and PI (Ulmer et al. 2003) prescreening methods, but the prediction uncertainty obtained by the BNN-based models is often small. Therefore, it is unsurprising that using a BNN-based model often leads to premature convergence for antenna cases compared to the GP model. On the other hand, LCB is the fundamental of the new prescreening method in this thesis and is introduced as follows. Given the objective function $y(x)$ has a predictive distribution of $N(\hat{y}(x), \hat{s}^2(x))$, an LCB prescreening of $y(x)$ is:

$$y_{\text{lcb}}(x) = \hat{y}(x) - \omega \hat{s}(x) \tag{4.1}$$

where ω is a constant and is often set to 2 in many algorithms in the AI domain (Emmerich et al. 2006), and applies to antenna problems (Liu et al. 2014c). A solution to the above-mentioned premature convergence issue is to use the LCB prescreening method (4.1) (Dennis and Torczon 1997), which has a hyperparameter ω involved to control the extent of exploration. The value of ω can be set to empirical values resulting from experiments with various antenna design cases, and the recommended value is 14. Assigning a large value to ω can promote the exploration ability while focusing on exploration inevitably slows down the convergence, which means more EM simulations are needed due to no free launch. Therefore, a novel prescreening mechanism to obtain the appropriate trade-off, called self-adaptive LCB, is proposed. Given the λ current best candidate designs P_b, and an array S, where $S_i (i = 1, 2, \ldots, k)$ is assigned with the smallest distance between the current best candidate design by prediction and its closed candidate designs in P_b for the i-th iteration, self-adaptive LCB (Step 6 in SB-SADEA in Section 4.1) works as follows.

Step 1: Obtain the best candidate design x_b from the child population generated in Step 4 of SB-SADEA using the BNN-based model predicted values in Step 5 of SB-SADEA.

Step 2: Calculate the distance between x_b and each individual in P_b and find the smallest distance, S_{k+1}.

Step 3: Take the last 10 elements in S to form S_z, $z = k - 9, \ldots, k$, check if S_{k+1} from Step 2 is less than $\bar{S}_z - 0.5 \times \sigma_{S_z}$, where σ_{S_z} is the standard deviation of $\boldsymbol{S_z}$. If yes, go to Step 4; Otherwise, output x_b.

Step 4: Prescreen the child population using the LCB method (4.1) with the recommended ω value.

Step 5: Output the best candidate design according to LCB values.

Some explanations are as follows.

- In order to obtain the most promising candidate design from the child population, the self-adaptive LCB method adaptively uses the predicted value by the BNN models and the LCB prescreened BNN models output values. The former enables exploitation to improve the convergence speed, while the latter enables exploration to avoid premature convergence.
- Whether the algorithm has sufficient exploration ability or not highly depends on the diversity in P_b (Step 3 of SB-SADEA in Section 4.1). Hence, the predicted values are used when the diversity is reasonably high, while LCB prescreening is imposed when the diversity is low.
- The metric of determining the extent of introduced diversity is to compare the smallest distance to any individual in P_b with those in the ten recent iterations. Assuming $S_i(i = 1, 2, \ldots, k)$ is Gaussian distributed, the $0.5 \times \sigma$ value is used as the threshold to find those introducing low diversity to P_b when using them. The diversity is considered low when the closest distance is lower than most of those in the recent iterations and vice versa.

4.4 Parameter Settings

Compared to standard SADEA (Liu et al. 2014c), SB-SADEA only introduces one new parameter, ω, in the self-adaptive LCB method. In order to determine a good empirical value for ω, various antennas (from having fewer than ten design variables to having 45 design variables, from having a few specifications to having more than 20 specifications) are used for the experiments. Resulting from the experiments, ω is suggested to be 14 for convergence efficiency and avoiding premature convergence. Although when using a smaller value of ω, the two case studies shown in Figure 3.2 can obtain the optimal design much faster with a 100% success rate, optimisation for some other antennas can converge to local minima. Our experimental results show that $\omega = 14$ is safe to use, although, for antennas that are easy to optimise, a less ω can help to improve the speed. In BNN modelling, the network hyperparameters are pre-decided by the rules of thumb and do not need the users to alter. For all other parameters, the setting rule in other SADEA versions is applicable to SB-SADEA as well, which are: $\alpha = 4 \times d$, $\lambda = 4 \times d$, $\tau = 4$, $F = 0.8$, $CR = 0.8$. They are used in all the test cases in Chapters 5 and 6.

Chapter 5

Experimental results and verification

SB-SADEA is tested against more than ten complex real world antenna optimisation cases and the conclusion is consistent for different antennas with various characteristics. In this chapter, four of them are investigated and demonstrated in details. Section 5.1 demonstrates the optimisation of an ultra-wide band (UWB) slotted monopole antenna that can be used in microwave imaging (Danjuma et al. 2020). The antenna has ten design variables and three design specifications. The design optimisation of this antenna is challenging as it requires proper physical placement and integration of its antenna structure on a compact dimension, as well as compact components on the printed circuit board. The computational cost of machine learning using most methods is often little for antennas with such numbers of design variables and design specifications. Hence, this case study only aims to test SB-SADEA's convergence speed or the number of EM simulations needed to obtain the optimal design when the antenna design specifications are stringent. Section 5.2 demonstrates a quadruple-band mm-wave antenna for wearable 5G and beyond applications (Ur-Rehman et al. 2018). It has 20 design variables and 12 design specifications. The design optimisation of a high-performance 5G mm-wave antenna is often difficult (Khan et al. 2020), and this case study has particular difficulty due to its low profile, low maintenance, lightweight and compact size, with a simple off-centred microstrip feeding structure. Furthermore, maintaining multi-band and high gain operations for body-centric wireless communications at mm-wave frequencies in wearable conditions increases the design sensitivity and difficulty. Considering the number of design variables and design specifications, the computational cost of machine learning can be expensive for the GP-based methods. This antenna is selected as a typical example of antennas with similar numbers of design variables and design specifications for SB-SADEA testing. This case study aims to test SB-SADEA's performance for its convergence speed and computational

cost of machine learning. Section 5.3 demonstrates the optimisation of a 5G outdoor base station antenna (5G-OBSA). The antenna optimisation has 23 design variables and 18 design specifications. The 5G-OBSA operates in 3.3-5 GHz for 5G communications and is also applied for dual-linear polarisation base station antenna arrays. The design optimisation of the 5G-OBSA is challenging as the antenna needs to maintain a stable efficiency, realised gain, broadside radiation pattern and beamwidth over the frequency band of interest. Considering the two-port isolation for the dual linear polarisation design makes the design even harder. A compact design of such antennas is often preferable in real-world applications, which often require multiple inputs, multiple outputs (MIMO) and smart-beam performance (Hong et al. 2017). Similarly, this case study also aims to test SB-SADEA's performance in terms of convergence speed and computational cost of machine learning. Section 5.4 demonstrates the optimisation of a quasi-digitally coded microstrip patch antenna (Jayasinghe et al. 2015). This is a 62-dimensional antenna optimisation problem. The training cost of using the previous methods is unaffordable and thus the major challenge of the optimisation. Engineering optimisation problems with more than 100 design variables are rare, as such designs are usually decomposed into sub-problems before solving them. Therefore, the optimisation of a 62-D antenna design practically shows the performance of the proposed method when optimising a very high-dimensional but possible antenna design. This case study mainly aims to demonstrate the cost of machine learning when the optimisation dimensionality is very high.

As no initial designs are provided for all four antenna cases, the search ranges for the design exploration are relatively wide and restricted by compact size specifications in certain case studies. The SADEA series is a series of stochastic algorithms, and more than five independent runs are carried out for all four case studies. P-SADEA (Akinsolu et al. 2019) is selected as the reference algorithm for case study 1, and TR-SADEA (Liu et al. 2021) is selected as the reference algorithm for case studies 2 and 3. PSO, one of the most popular EAs for antenna design exploration, is also used as a reference algorithm for case studies 2 and 3.

The experiments are conducted on a workstation with an AMD Ryzen™ Threadripper™ PRO (2.7GHz) and an NVIDIA® RTX™ A4000 GPU. 80 MATLAB parallel workers are activated for GP/BNN-based surrogate model training.

5.1 Case study 1: Compact UWB slotted monopole antenna optimisation

5.1.1 Engineering background

Figure 5.1 shows an illustrative layout of the slotted monopole antenna. The antenna is implemented on an FR-4 substrate with a relative permittivity of 4.4, a loss tangent of 0.02, and a thickness of 0.8 mm. It comprises two uniform rectangular metal planes separated by the microstrip line and a driven circular patch radiator. Two slots are fused at the centre of the driven circular patch radiator to form a quasi-cross slot, and the patch slot helps control the current flow on the surface. At the same time, the rectangular planes play the role of co-planar partial ground.

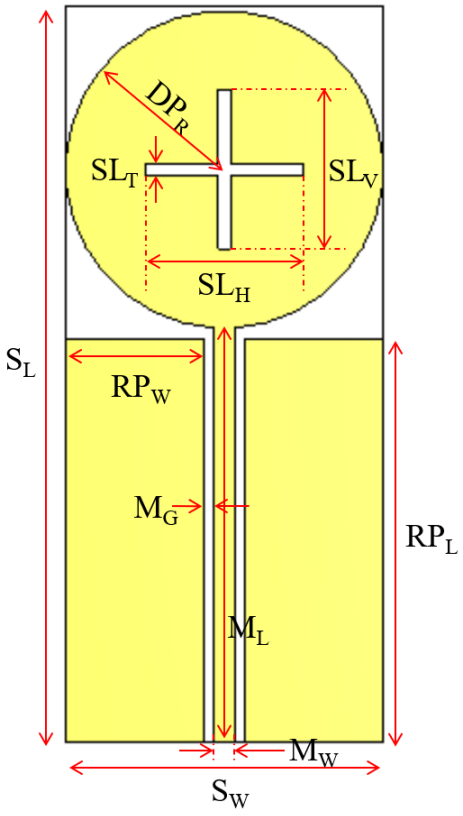


Figure 5.1: Layout of the compact UWB slotted monopole antenna.

The slotted monopole antenna is modelled and discretised in CST-MWS with over 162,000 mesh cells. Each EM simulation of the antenna takes around 45 seconds on average. The design variables shown in Figure 5.1 and their search ranges in Table 5.1 are considered for optimising the slotted monopole antenna. The optimisation goal is to minimise the fitness function, F_{monopole}, to satisfy the design specifications shown in

Table 5.1: Search ranges of the design variables and the optimal design by SB-SADEA (All sizes in mm)

Parameters	Lower bound	Upper bound	SB-SADEA Optimum
Circular patch radius (DP_R)	2	25	7.14
Substrate width (S_W)	$2\times DP_R$	$3\times DP_R$	14.40
Width of slot throat (SL_T)	0	$2\times DP_R$	5.77
Vertical slots' depth (SL_V)	0	$2\times DP_R$	5.36
Horizontal slots' depth (SL_H)	0	$2\times DP_R$	0.79
Microstrip length (M_L)	RP_L	50	26.21
Partial ground plane length (RP_L)	DP_R	M_L	8.21
Microstrip width (M_W)	0.50	7.50	1.20
Microstrip gap (M_G)	0	DP_R-0.5$\times$ M_W	0.34
Feed guide width (P_W)	$6\times M_W$	$10\times M_W$	10.68
Substrate width (S_L) = $M_L + 2\times DP_R + 0.2$ (mm)			
Partial ground plane width (RP_W) = $(S_W - 2\times M_G - M_W) \div 2$ (mm)			

Table 5.2, mathematically,

$$\begin{aligned} F_{\text{monopole}} =& w_1 \times \max(|S_{11}| + 10, 0) \\ &+ w_2 \times \max(G_{\max} - 3, 0) \\ &+ w_3 \times \max(1 - G_{\min}, 0) \end{aligned} \tag{5.1}$$

where w_1, w_2 and w_3 are the penalty coefficients set to 1, 50 and 50, respectively. When the EM simulation performances of the optimal design meet all the design specifications listed in Table 5.2, F_{monopole} is equal to 0. For SB-SADEA and all other reference methods except PSO, ten independent runs are carried out. Three optimisation runs are carried out for PSO to verify the efficiency comparison sufficiently.

Table 5.2: Design specifications and the performance of a typical optimal design obtained by SB-SADEA

Item	Specification	SB-SADEA Optimum
Maximum Reflection Coefficient ($\lvert S_{11}\rvert$) (3.1 to 10.6 GHz)	$\leq$ -10 dB	-10.13 dB
Maximum Realized Gain ($G_{\max}$)	$\leq$ 3 dB	2.90 dB
Minimum Realized Gain ($G_{\min}$)	$\geq$ 1 dB	1.19 dB

Table 5.3: Number of EM simulations (average number) used to satisfy the specifications for different methods

	SB-SADEA	GP-ALCB	FBN-LCB	BN-ALCB	GP-LCB
ML models	BNN	GP	BNN	BNN	GP
Fine-supervision	Yes	No	Yes	No	No
Prescreening	AdapLCB	AdapLCB	LCB	AdapLCB	LCB
number of EM simulations	**924**	1262	1329	1104	1991

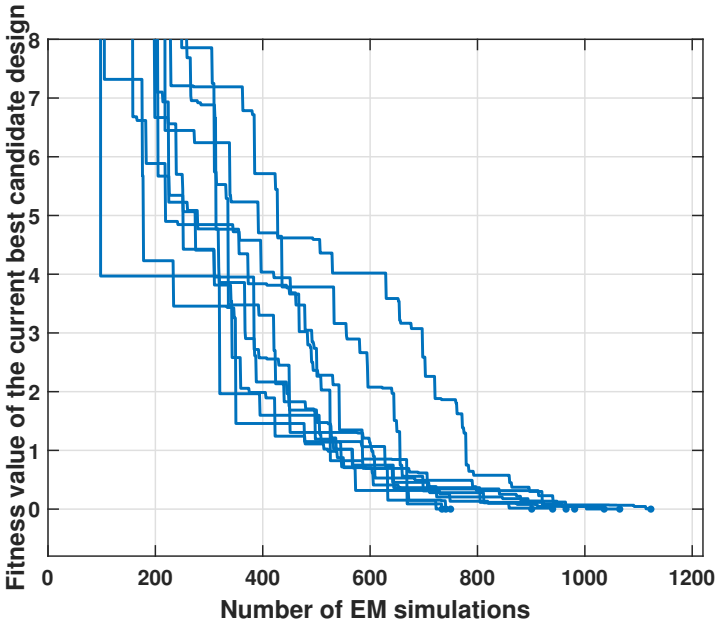
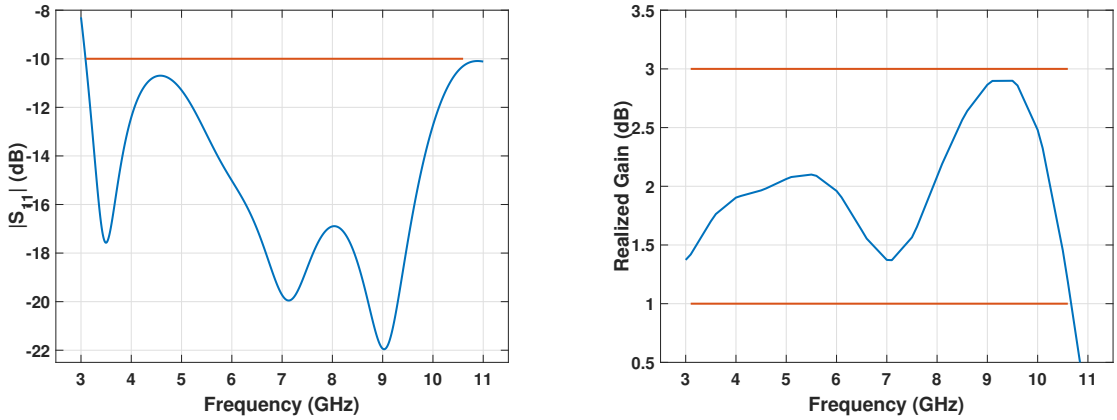


Figure 5.2: Convergence trends of slotted monopole antenna optimisation using SB-SADEA.

5.1.2 Optimisation results and discussion

In all ten optimisation runs, SB-SADEA can satisfy the design specifications shown in Table 5.2 using an average of 925 EM simulations (11.5 hours). Figure 5.2 shows the convergence trends. Figure 5.3 shows the simulation performance, including the reflection coefficient and the realised gain, of a typical optimal design shown in Table 5.2. The antenna size shrinks to approximately 60% compared with a state-of-the-art reference design (Yeboah-Akowuah et al. 2017).

As discussed earlier, as one of the state-of-the-art algorithms for medium-scale antennas with stringent specifications, P-SADEA is considered the reference method. P-SADEA also shows a 100% success rate, but it uses an average of 1575 EM simulations to satisfy all the specifications. Therefore, SB-SADEA saves approximately 40% of the EM simulations compared to P-SADEA. Note that compared to standard SADEA (Liu et al. 2014c), P-SADEA improves the convergence speed at the cost of time consumption at surrogate modelling phase (Akinsolu et al. 2019; Liu et al. 2018) by its new model management framework. SB-SADEA, however, only utilises the standard model management framework (Liu et al. 2014c), and the comparison result shows the effectiveness of BNN-based modelling and self-adaptive LCB techniques. Furthermore, the surrogate modelling is fully compatible with the model management framework proposed with P-SADEA, which can form an even faster algorithm.



(a) Reflection coefficient of the optimal design (b) Realized gain of the optimal design

Figure 5.3: Response of the optimal design obtained by SB-SADEA

To verify the effectiveness of the BNN-based antenna modelling, including the fine supervision and the self-adaptive LCB-based prescreening, more comparisons are shown in Table 5.3. When the GP model is applied, the ω value for LCB is set to 2, as in other SADEA versions, instead of 14 for the BNN-based model.

The following conclusions can be drawn from Table 5.3: (1) By comparing SB-SADEA with GP-ALCB when both utilise the self-adaptive LCB-based prescreening, about 25% EM simulations are saved by the BNN-based surrogate modelling compared to its GP-based counterpart. (2) By comparing SB-SADEA with FBN-LCB when both utilise the BNN-based surrogate modelling, about 30% EM simulations are saved for the effectiveness of the self-adaptive LCB prescreening and its co-working with the BNN-based model. As discussed in Section 4.2, for the BNN-based model, the prediction uncertainty is less than that of GP, and a larger ω is needed in LCB prescreening to guarantee a consistent exploration ability, which inevitably slows down the convergence speed. Hence, the self-adaptive LCB technique is essential for the BNN-based model. (3) By comparing SB-SADEA with BN-ALCB, when the only difference is the use of fine supervision, roughly 15% EM simulations are saved, showing the effectiveness of fine supervision. (4) GP-LCB (i.e., standard SADEA) is the slowest reference algorithm among all, and SB-SADEA decreases 53% of the necessary EM simulations to obtain the optimal design, showing the combined effectiveness of the BNN-based antenna surrogate model and the self-adaptive LCB method.

In the three PSO runs, the specifications on realised gain are satisfied, but the specification on $\max(|S_{11}|)$ is not, and the average value is -5.2 dB. The failure of the design optimisation can be attributed to the structure's compactness and the design specifications' stringency. Considering all these comparisons, this case study verifies the advantages of SB-SADEA in terms of convergence speed.

5.2 Case study 2: Quadruple-band 5G mm-wave antenna optimisation

5.2.1 Engineering background

This antenna is designed to present a quadruple-band operation with significant band discrimination and high gain at mm-wave frequencies of 28 GHz, 38 GHz, 50 GHz, and 60 GHz. The design goals for all four operating bands mentioned include achieving a minimum realised gain of 4.5 dB and a minimum total efficiency of 80%. This low-profile antenna uses a patch geometry combining a squared patch with an L- and an F-shaped slot on a Rogers RT/Duroid 5880 substrate of 0.254 mm thickness, a loss tangent of 0.0009, and a relative permittivity of 2.2. Moreover, the single layer 5.1 mm $\times$ 5 mm $\times$ 0.254 mm antenna is excited by a 50Ω off-centred single-feed microstrip line. As shown in Figure 5.4, the slots positioned close to the patch edges make the current mostly concentric there and yield inductive and capacitive sensing effects, resulting in the multi-frequency operation (Ur-Rehman et al. 2018).

The quadruple-band mm-wave antenna is modelled and discretised in CST-MWS with nearly 300,000 mesh cells in total. Each EM simulation costs about 2.2 minutes on average. For the optimisation of the targeted antenna, there are 20 design variables and the they are shown in Figure 5.4 and listed in Table 5.4 along with their search ranges. There are 12 design specifications and the optimisation goal is to minimise the fitness function, $F_{\text{mm-wave}}$, to satisfy the design specifications shown in Table 5.5,

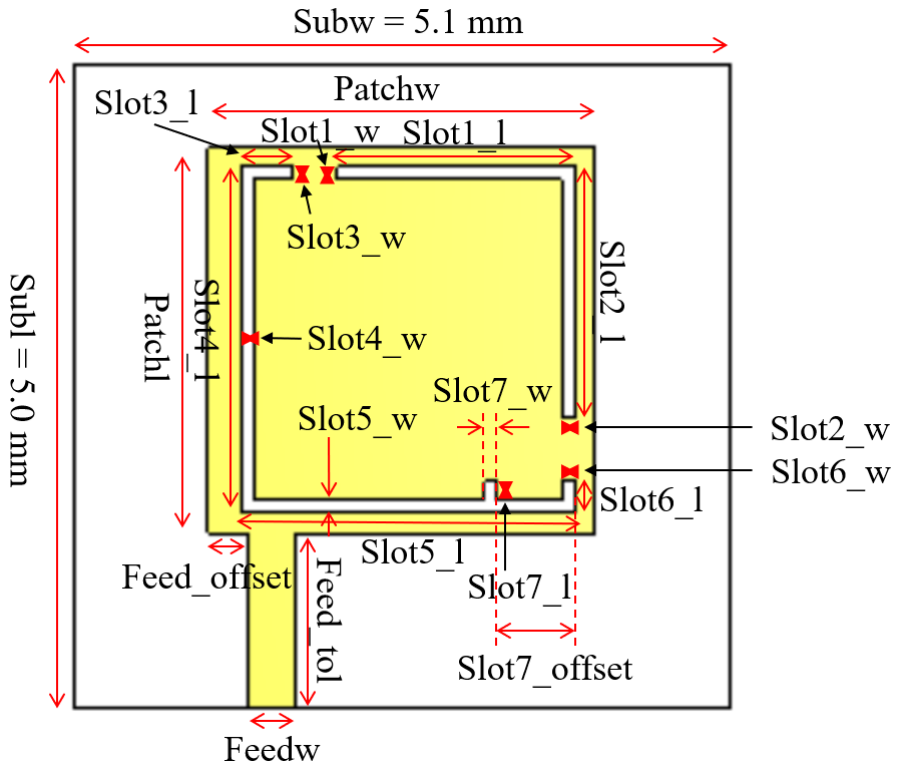


Figure 5.4: The layout of the quadruple-band 5G mm-wave antenna.

mathematically,

$$
\begin{aligned}
F_{\text{mm-wave}} = & \sum_{i=1}^{4} w_1 \times \max(|S_{11}^i| + 10, 0) \\
& + \sum_{i=1}^{4} w_2 \times \max(4.5 - G_{\text{min}}^i, 0) \\
& + \sum_{i=1}^{4} w_3 \times \max(0.8 - E_{\text{total}}^i, 0)
\end{aligned}
\tag{5.2}
$$

where i is the index for the current frequency band out of the 4 frequency bands. and $S_{11}^i(x)$, $G_{\text{min}}^i(x)$ and $E_{\text{total}}^i(x)$ are the S-parameters, minimum gain and the total efficiency given an antenna structure x, within the i-th operating band, respectively. w_1, w_2 and w_3 are the penalty coefficients set to 1, 50 and 50, respectively. When all the design specifications in Table 5.5 are satisfied, $F_{\text{mm-wave}}$ is equal to 0.

Table 5.4: Search ranges of the design variables and a typical optimal design obtained by SB-SADEA (All sizes in mm)

Variable	Lower bound	Upper bound	SB-SADEA Optimum
slot1_w	0	3	0.059
slot2_w	0	3	0.72
slot3_w	0	3	0.033
slot4_w	0	3	2.23
slot5_w	-3	0.2	-0.71
slot6_w	-3	0.2	-0.081
slot7_w	-2.2	0.9	-1.34
slot7_offset	0	2.5	2.24
feedw	0.1	0.45	0.18
feed_offset	0	3-feedw	0.017
slot1_l	0	3	0.18
slot2_l	0	3	1.98
slot3_l	0	3	1.95
slot4_l	0	3	0.60
slot5_l	0	3	1.40
slot6_l	0	3	2.15
slot7_l	0	3	0.87
feed_tol	0	3	2.02
patchl	0.5	4.3	4.26
patchw	0.5	5	4.55

5.2.2 Optimisation results and discussion

Five independent runs of optimisation experiments are carried out to test the proposed SB-SADEA. They all satisfy the design specifications listed in Table 5.5, with an average of approximately 1,200 EM simulations. Figure 5.5 shows the convergence trends of the five optimisation runs. Figure 5.6 shows the reflection coefficient, the realised gain, and the total efficiency of a typical optimal design in Table 5.4, obtained using SB-SADEA.

As discussed earlier, TR-SADEA (Liu et al. 2021) is selected as the reference optimisation algorithm. In all five runs, although TR-SADEA achieves a 100% success rate as well, it utilises an average of 2426 EM simulations, which doubles that of using SB-SADEA. Hence, SB-SADEA decreases the number of EM simulations of the optimisation by more than 50% compared to TR-SADEA in this case study, verifying the advantages in convergence speed again.

Table 5.5: Design specifications and the performance of a typical optimal design obtained by SB-SADEA

Items	Specification	SB-SADEA Optimum
Maximum in-band reflection coefficients ($\|S11\|$) (27.75 to 28.25 GHz)	$\leq$ -10 dB	-12.28 dB
Maximum in-band reflection coefficients ($\|S11\|$) (37.75 to 38.25 GHz)	$\leq$ -10 dB	-13.04 dB
Maximum in-band reflection coefficients ($\|S11\|$) (49.75 to 50.25 GHz)	$\leq$ -10 dB	-10.54 dB
Maximum in-band reflection coefficients ($\|S11\|$) (59.75 to 60.25 GHz)	$\leq$ -10 dB	-16.18 dB
Minimum in-band realised gain (G_{min}) (27.75 to 28.25 GHz)	$\geq$ 4.5 dB	5.67 dB
Minimum in-band realised gain (G_{min}) (37.75 to 38.25 GHz)	$\geq$ 4.5 dB	4.88 dB
Minimum in-band realised gain (G_{min}) (49.75 to 50.25 GHz)	$\geq$ 4.5 dB	6.75 dB
Minimum in-band realised gain (G_{min}) (59.75 to 60.25 GHz)	$\geq$ 4.5 dB	7.01 dB
Minimum in-band total efficiency (E_{tot}) (27.75 to 28.25 GHz)	$\geq$ 80%	82.4%
Minimum in-band total efficiency (E_{tot}) (37.75 to 38.25 GHz)	$\geq$ 80%	86.2%
Minimum in-band total efficiency (E_{tot}) (49.75 to 50.25 GHz)	$\geq$ 80%	84.4%
Minimum in-band total efficiency (E_{tot}) (59.75 to 60.25 GHz)	$\geq$ 80%	89.3%

Also, this case study compares the training cost of machine learning of SB-SADEA with TR-SADEA. As introduced in Section 1.5, TR-SADEA is proposed specifically for antennas with many design variables, at which GP modelling time becomes a challenge due to "curse of dimensionality". With its GP model sharing mechanism applied, TR-SADEA often reduces the time consumption in the surrogate modelling phase by 90% (Liu et al. 2021), particularly when the number of design variables is very high. Still, for the targeted antenna in this case study, an average of over 426,000 GP surrogate models are built and trained during the optimisation using TR-SADEA, taking 12.0 hours on average. By using TR-SADEA, this time consumption is practical but not desirable. With BNN-based surrogate modelling applied in SB-SADEA, the surrogate modelling training phase in the optimisation only takes an average of 1.6 hours. Table 5.6 shows the number of EM simulations used, the number of surrogate models trained, and the total time consumption for both methods. The data in the table are all averaged values of the five independent runs. The significant reduction in the training cost of machine learning of SB-SADEA is also highlighted in the table. The total optimisation time decreased by more than half compared to the referenced method, TR-SADEA.

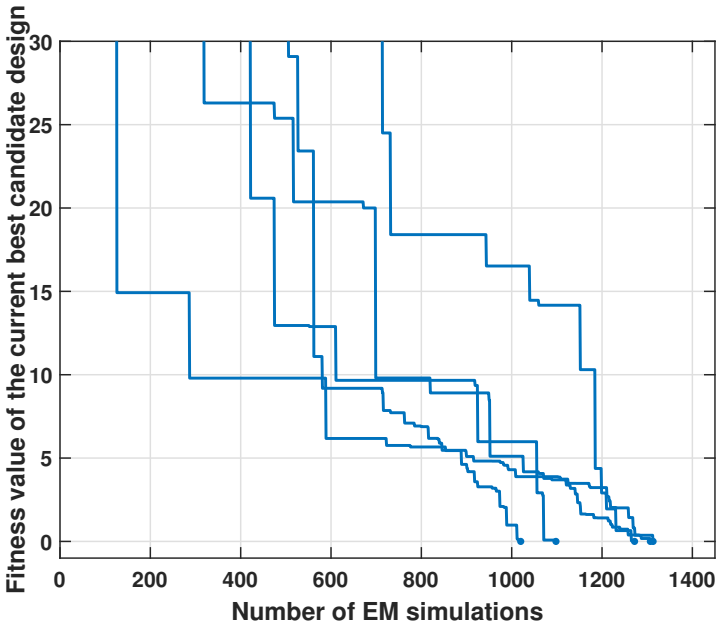


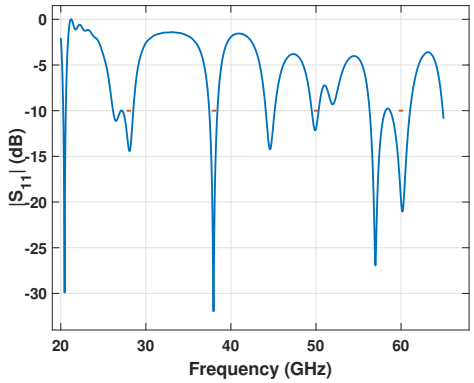
Figure 5.5: Convergence trends of SB-SADEA (5 runs).

Table 5.6: Comparison between SB-SADEA and TR-SADEA (average values)

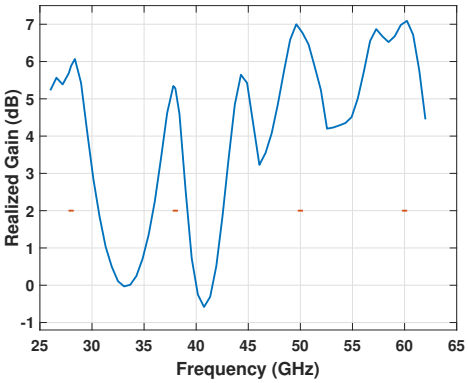
	SB-SADEA	TR-SADEA
ML models	BNN	GP (model sharing)
Fine-supervision	Yes	Yes
Prescreening	AdapLCB	LCB
Number of surrogate models	**102,000**	426,000
Modelling time (hours)	**1.6**	12.0
Number of EM simulations	**1202**	2426
Total optimisation time (hours)	**48.5**	104.3

Using DE, the optimisation experiments are carried out for the quadruple-band 5G mm-wave antenna. After two weeks of optimisation, all the reflection coefficient specifications are still being optimised, and only half of the gain and total efficiency specifications are satisfied. A longer run may improve the performance, but the optimisation time should be shorter for practical use. Considering all the evidence from the comparisons, this case study verifies the advantages of SB-SADEA regarding the number of EM simulations needed and the training cost of machine learning.

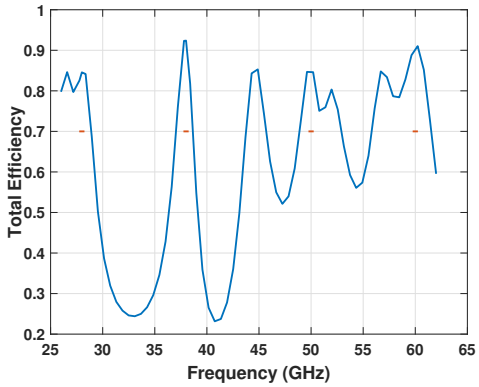
As the quadruple-band 5G mm-wave antenna optimisation also serves as the F5 of the test problems, more experiments and discussion are also in Chapter 3.



(a) Reflection coefficient of the optimal design



(b) Realised gain of the optimal design



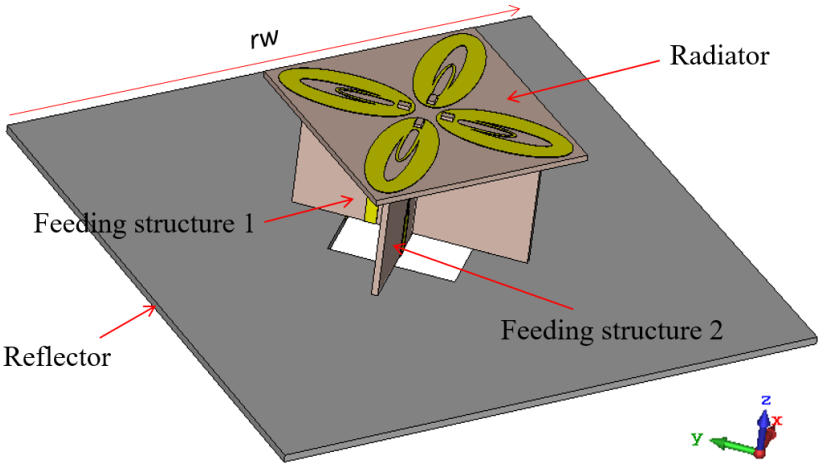
(c) Total efficiency of the optimal design

Figure 5.6: Responses of the optimal design obtained by SB-SADEA

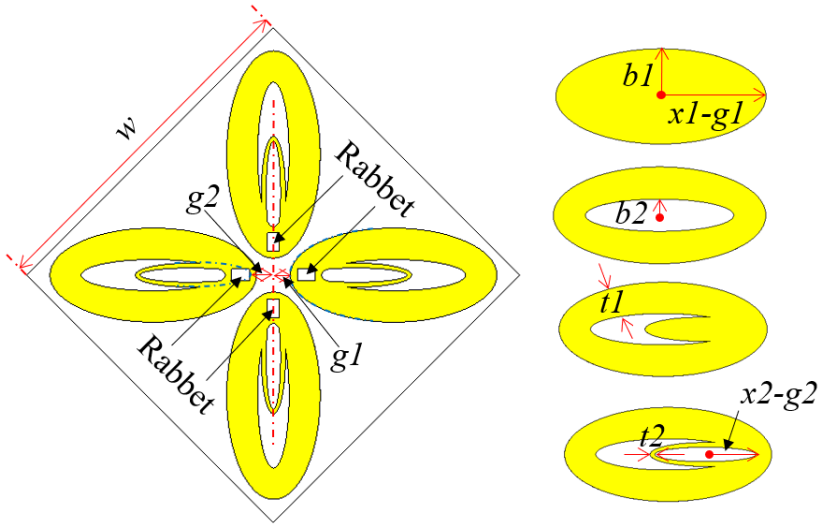
5.3 Case study 3: 5-G outdoor base station antenna optimisation

5.3.1 Engineering background

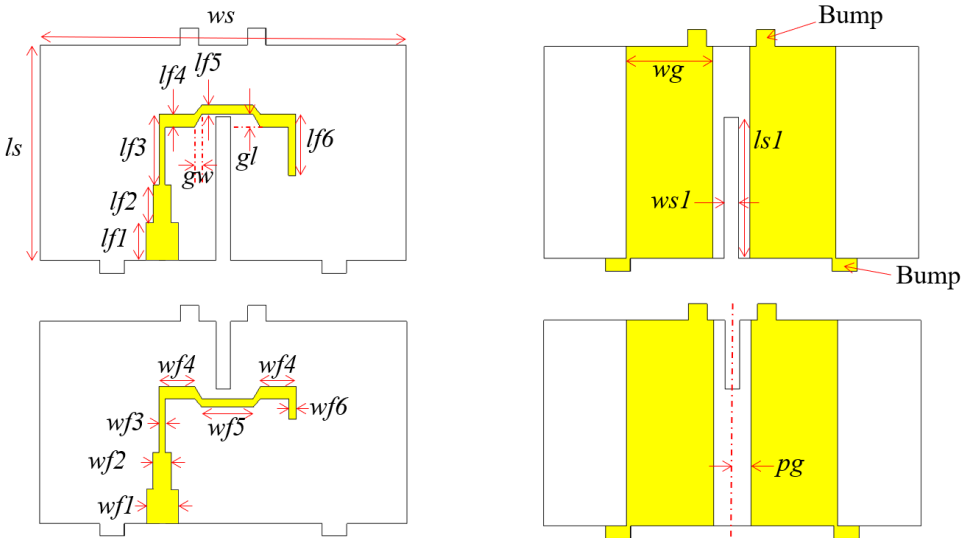
Figure 5.7 shows the layout of the 5-G outdoor base station antenna (5G-OBSA). The antenna consists of one double oval-shaped radiator, two Γ-shaped feeding structures and one reflector. The radiator and the feeding structures are printed on a 0.8-mm-thick FR-4 substrate with relative permittivity of 4.4 and a loss tangent of 0.02. The crossed-dipole arms are arranged diagonally on the substrate to ensure a ±45 polarisation, with each arm having inner and outer oval geometries. The Γ-shaped feeding structures are printed on the substrate, and the two patches serve as the ground plane at the back of the substrate. The inner coaxial cable is connected to the Γ-shaped feeding structure,



(a) 3D view of the overall 5-G outdoor base station antenna structure. This figure is a screenshot from CST-MWS.



(b) Crossed-dipole radiator.



(c) Front view of the feeding structures.

(d) Back view of the feeding structures.

Figure 5.7: 5-G outdoor base station antenna illustration. This figure is modified from (Liu et al. 2021).

while the outer coaxial cable is connected to the patches. The two Γ-shaped feeding structure middle parts are designed differently to prevent intersection. By soldering the two patches to the central radiator and reflector using bumps and rabbets, the reflector base achieves a more effective unidirectional radiation pattern.

The 5G-OBSA is modelled and discretised in CST-MWS with a mesh density of 15 cells per wavelength so it has approximately 2,630,000 mesh cells in total. An EM simulation of the 5G-OBSA using the CST-MWS take around 10 minutes on average.

Table 5.7: Search ranges of the design variables and the optimal design by SB-SADEA (all sizes in mm)

Variables	Lower bound	Upper bound	SB-SADEA optimum
$lf1$	1.00	10.00	1.95
$lf2$	1.00	10.00	1.71
$lf3$	1.00	10.00	3.00
$lf4$	0.10	1.50	0.14
$lf5$	0.10	1.50	0.22
$lf6$	0.10	15.00	4.61
$wf2$	0.10	1.50	0.97
$wf3$	0.10	1.50	0.35
$wf4$	0.10	5.00	2.27
$wf5$	0.10	5.00	2.53
$wf6$	0.10	1.50	0.52
gw	0.10	1.50	0.63
ls	16.00	30.00	16.32
$x1$	7.50	15.00	12.04
$b1$	3.00	5.00	4.69
$t1$	0.50	3.00	2.58
$x2$	3.00	8.00	5.54
$b2$	0.50	3.00	2.02
$t2$	0.20	0.80	0.61
$g1$	1.00	3.00	1.94
$g2$	1.00	3.00	2.58
wg	5.00	15.00	6.74
rw	60.00	85.00	84.45

Table 5.7 lists the 23 design variables and the search ranges for the 5G-OBSA in this case study. For the optimisation of the targeted antenna, the design variables shown in Figure 5.7 are tuned within their search ranges listed in Table 5.7. Eventually, an optimal antenna design is expected as the output of the case study.

Table 5.8: Design specifications and the performance of an optimal design

Item	Specification	SB-SADEA Optimum
Maximum reflection coefficient (S_{11}) (3.3 to 3.8 GHz)	$\leq$ -10 dB	-12.04 dB
Maximum reflection coefficient (S_{11}) (4.8 to 5.0 GHz)	$\leq$ -10 dB	-12.41 dB
Maximum reflection coefficient (S_{22}) (3.3 to 3.8 GHz)	$\leq$ -10 dB	-11.68 dB
Maximum reflection coefficient (S_{22}) (4.8 to 5.0 GHz)	$\leq$ -10 dB	-12.19 dB
Maximum transmission coefficient (S_{12}) (3.3 to 3.8 GHz)	$\leq$ -20 dB	-24.55 dB
Maximum transmission coefficient (S_{12}) (4.8 to 5.0 GHz)	$\leq$ -20 dB	-36.35 dB
Minimum realised gain (G) (3.3 to 3.8 GHz)	$\geq$ 5 dBi	8.00 dBi
Minimum realised gain (G) (4.8 to 5.0 GHz)	$\geq$ 5 dBi	7.87 dBi
Minimum front-to-back ratio (FBR) (3.3 to 3.8 GHz)	$\geq$ 15 dB	15.89 dB
Minimum front-to-back ratio (FBR) (4.8 to 5.0 GHz)	$\geq$ 15 dB	15.51 dB
Minimum half-power beam-width ($HPBW_l$) (3.3 to 3.8 GHz)	$\geq 60°$	68.68°
Maximum half-power beam-width ($HPBW_u$) (3.3 to 3.8 GHz)	$\leq 70°$	69.81°
Minimum half-power beam-width ($HPBW_l$) (4.8 to 5.0 GHz)	$\geq 60°$	66.13°
Maximum half-power beam-width ($HPBW_u$) (4.8 to 5.0 GHz)	$\leq 70°$	69.27°
Number of resonance (NR_1) (3.3 to 3.8 GHz) (if S_{11} > -10 dB)	$\geq$ 1	Defaulted to 1
Number of resonance (NR_1) (4.8 to 5.0 GHz) (if S_{11} > -10 dB)	$\geq$ 1	Defaulted to 1
Number of resonance (NR_2) (3.3 to 3.8 GHz) (if S_{22} > -10 dB)	$\geq$ 1	Defaulted to 1
Number of resonance (NR_2) (4.8 to 5.0 GHz) (if S_{22} > -10 dB)	$\geq$ 1	Defaulted to 1

There are 18 design specifications for this antenna and they are listed in Table 5.8. They include the maximum reflection coefficients (S_{11}, S_{22} and S_{12}), minimum realised gains, minimum front-to-back ratio, minimum and maximum half-power beam-width ($HPBW$) and number of resonance (NR_1 for S_{11} and NR_2 for S_{22}), along the two operational bands. In this case study, the optimisation goal is to find an antenna that satisfies all the design specifications. To form a single-objective optimisation problem, the 18 design specifications are merged into a single fitness function by obtaining a weighted summation like mentioned in 2.1. When all design requirements are met, the fitness function is equal to zero. The fitness function can be expressed mathematically as

$$\begin{aligned} F_{\text{OBSA}} =& w_1 \sum_{i=1}^{2} [\max(|S_{11}^i| + 10, 0) + \max(|S_{22}^i| + 10, 0) + \max(|S_{12}^i| + 20, 0)] \\ &+ w_2 \sum_{i=1}^{2} [\max(5 - G^i, 0) + \max(15 - FBR^i, 0) + \max(60 - HPBW_l^i, 0) \\ &+ \max(HPBW_u^i - 70, 0) + \max(1 - NR_1^i, 0) + \max(1 - NR_2^i, 0)] \end{aligned} \tag{5.3}$$

where i indicates the i-th band for the dual-band operations of the 5G-OBSA, and w_1 and w_2 are the penalty coefficients. NR_1^i and NR_2^i are defaulted to be 1 subject to the conditions that $S_{11}^i \leq -10dB$ and $S_{22}^i \leq -10dB$ in (5.3) for NR_1^i and NR_2^i, respectively. w_1 and w_2 are set to be 1 and 50, respectively, so that the scale of the weighted antenna performances are in the same magnitude, and that all design specifications are considered at the same time with the same priority.

5.3.2 Optimisation results and discussion

Figure 5.8 shows the convergence trends of six runs of independent optimisation the 5G-OBSA using SB-SADEA. All optimisations complete with the specifications listed in Table 5.8 satisfied. The optimisations complete in 3.5 days on average.

Figure 5.9 illustrates the reflection coefficients, transmission coefficients, realised gains, front-to-back ratio and half-power beamwidth of the typical optimal design mentioned in Table 5.8, as well as the design specifications in orange lines. It can be seen that the specific typical optimal design meets all the design specifications by referencing the response curves and the design specification lines.

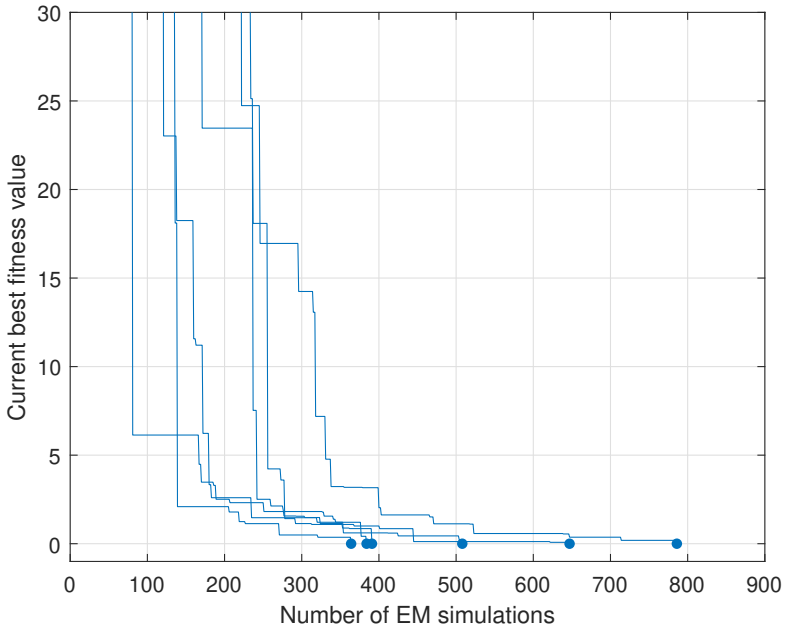
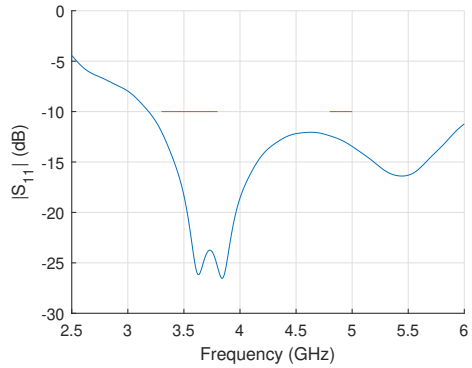


Figure 5.8: Convergence trends of SB-SADEA (six runs).

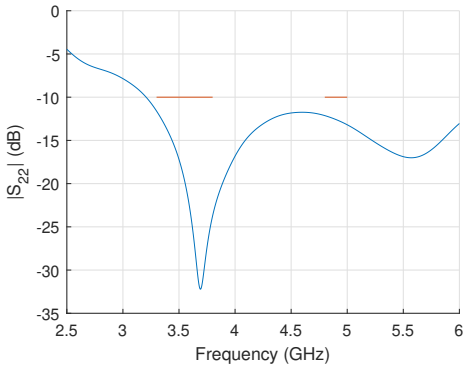
5.4 Case study 4: Quasi-digitally coded microstrip patch antenna optimisation

5.4.1 Engineering background

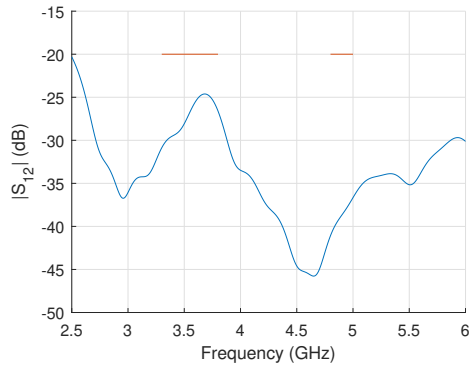
Figure 5.10 shows an illustrative layout of the quasi-digitally coded microstrip patch antenna (qDMPA). The qDMPA is constructed on a 46 mm × 57 mm rectangular substrate with patch shape similar to a rectangular patch resonating in around 2.2 GHz. A 50 Ω coaxial cable is used in the design as the feed. The qDMPA has 62 design variables, including the vertical shifting values of the microstrip patches and the co-ordinate position of the feed. As shown in the figure, the small patches can be shifted on the substrate with a high level of freedom and flexibility, thus the antenna can resonates at multiple frequencies (Jayasinghe et al. 2015). Therefore, the microstrip patch structure are frequently used by researchers to develop multi-band antennas. Also, like mentioned in Subsection 3.3, such high dimensional design optimisation problem investigates a new design methodology, which is a variant of digitally coded antennas. By randomly shifting the coordinate positions of the microstrip patches, novel and unexpected geometries can be explored in order to satisfy given design specifications.



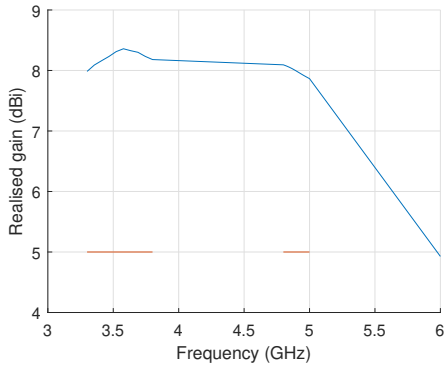
(a) Reflection coefficient S_{11} of the optimal design



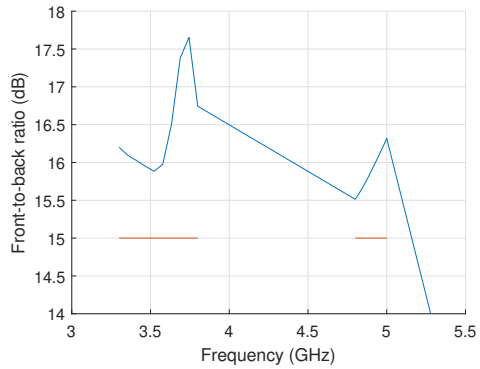
(b) Reflection coefficient S_{22} of the optimal design



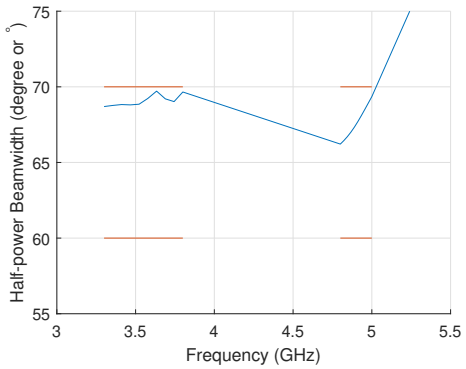
(c) Transmission coefficient S_{12} of the optimal design



(d) Realised gain of the optimal design



(e) Front-to-back ratio of the optimal design



(f) Half-power beamwidth of the optimal design

Figure 5.9: Responses of the optimal design obtained by SB-SADEA.

Table 5.9: Search ranges of the design variables and a known optimal design obtained by SB-SADEA (All sizes in mm)

No.	Lower bound	Upper bound	Optimum	No.	Lower bound	Upper bound	Optimum
1	4	10	7.9894	32	0	50	11.6075
2	-16	-10	-14.8630	33	0	50	29.5680
3	6	10	7.5761	34	0	50	3.6776
4	6	10	9.2878	35	0	50	13.5471
5	0	50	2.4054	36	0	50	14.6753
6	0	50	44.9459	37	0	50	28.1567
7	0	50	0.7049	38	0	50	43.0647
8	0	50	3.1079	39	-0.1	50	35.1182
9	0	50	30.6992	40	-0.1	50	41.9617
10	0	50	27.1247	41	-0.1	50	46.8938
11	0	50	39.9116	42	-0.1	50	38.6010
12	0	50	10.7311	43	-0.1	50	49.9588
13	0	50	26.9734	44	-0.1	50	33.6515
14	0	50	36.3145	45	0	50	8.9116
15	0	50	13.9313	46	-0.1	50	40.5991
16	0	50	39.2651	47	-0.1	50	27.2651
17	0	50	12.5489	48	-0.1	50	2.0558
18	0	50	41.2913	49	-0.1	50	27.5948
19	0	50	43.1333	50	-0.1	50	32.0417
20	0	50	20.0425	51	-0.1	50	1.0926
21	0	50	12.3995	52	-0.1	50	20.7121
22	0	50	24.0298	53	0	50	11.3664
23	0	50	9.3262	54	-0.1	50	30.8450
24	0	50	37.4254	55	-0.1	50	5.9976
25	0	50	1.5851	56	-0.1	50	11.1102
26	0	50	19.6865	57	-0.1	50	13.5843
27	0	50	24.2686	58	-0.1	50	42.04315
28	0	50	39.8887	59	-0.1	50	35.2211
29	0	50	48.5984	60	-0.1	50	4.2611
30	0	50	22.8520	61	0	10	0.6991
31	0	50	22.8337	62	0	9	0.8511

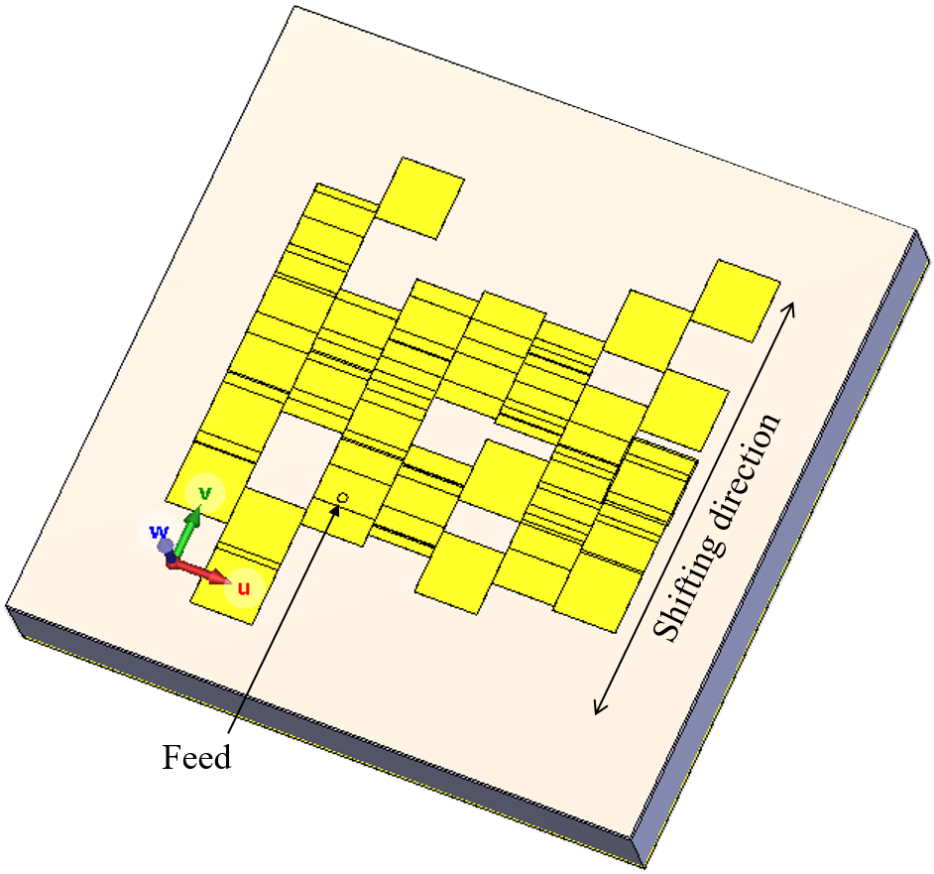


Figure 5.10: An illustrative layout of the quasi-digitally coded microstrip patch antenna. This figure is a modified screenshot from CST-MWS.

Table 5.1, lists the design variables for this case study. Amongst the design variables, No. 1-2 define the coordinate position of the feed, No. 3-4 define the coordinate position of a freely movable microstrips, No. 5-60 define the coordinate positions of vertically shifting microstrips patches, and No. 61-62 define the substrate parameters.

In this case study, the optimisation goal is to maximise the percentage of the available bandwidth within the operational bands. The optimisation problem is expressed mathematically as

$$\min_{AP} \quad -BW \tag{5.4}$$

where AP denotes the coordinate positions of centres of the microstrip patches and feed, and BW is the percentage of the available bandwidth, which is the percentage of bandwidth that has its reflection coefficient values being less or equal to -10 dB. Search ranges and a known optimum are provided in Table 5.9.

5.4.2 Optimisation results and discussion

Figure 5.11 shows 10 independent runs of the qDMPA optimisation using SB-SADEA. All runs converge to less than -55 of fitness value, in approximately 1000 EM simulations. The known best is presented in Table 5.1. According to (5.4), the optimised fitness value means that the optimised MPAs have at least 55% of available bandwidth across the band, which is a satisfying results empirically. As training surrogate

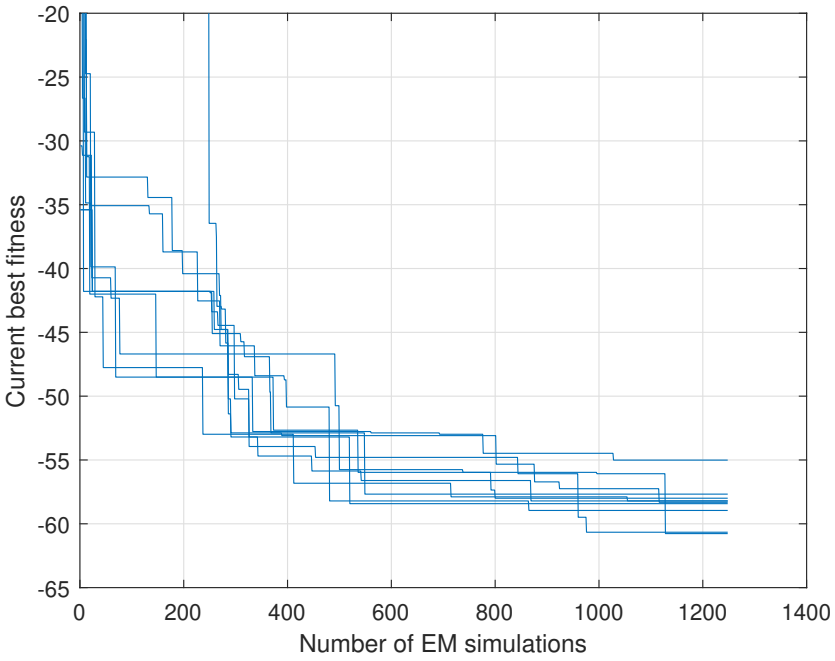


Figure 5.11: Convergence trends of qDMPA optimisation using SB-SADEA.

model for the case study is unaffordable using GPs, the comparison was only conducted between BNNs and NNDO. From Table 3.6, optimisation using SB-SADEA saves more than 8% of the time comparing to using SMAS with NNDO as the surrogate model.

The qDMPA optimisation also serves as benchmark problem F6 of the test problems in Chapter 3. Detailed comparison and discussion is in Subsection 3.5.2.

Figure 5.12 shows the S-parameter response of the typical optimal design mentioned in Table 5.9. In the plot, the available bandwidth is marked in orange line, covering approximately 61% of the band, which is consistent with the optimised fitness value.

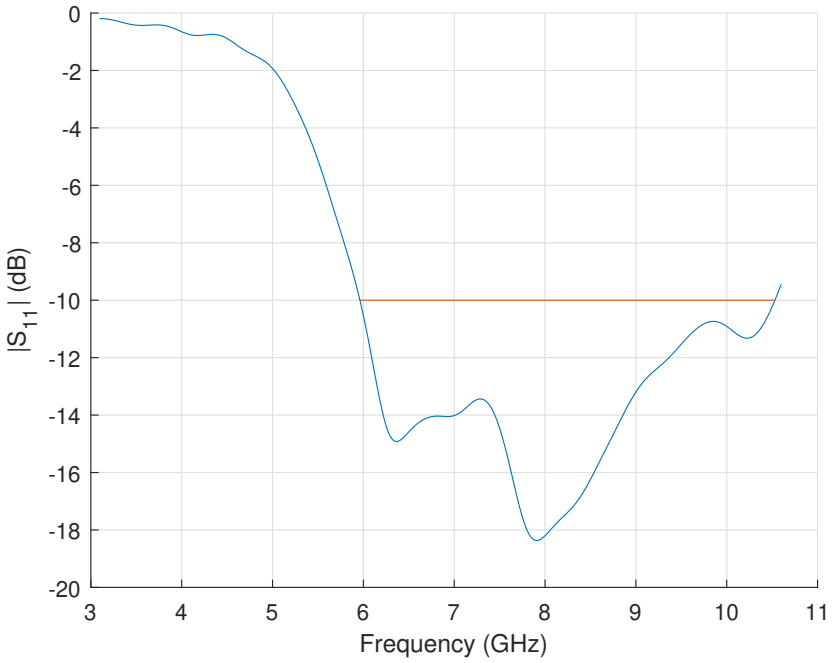


Figure 5.12: S-parameter response of the optimal design obtained by SB-SADEA.

Chapter 6

Applications of SB-SADEA in a wider scope

The success of the proposed SB-SADEA in antenna global optimisation tasks is due to its optimisation capability. Theoretically, the landscapes of optimisation problems in other fields, in essence, are similar to that of the antennas', if not simpler, despite having different engineering backgrounds. Moreover, the proposed SB-SADEA is tested against multiple benchmark functions, making it more likely to be available on general optimisation problems. This is to say, the proposed SB-SADEA should work on antenna optimisation problems and optimisation tasks in a wider scope. This chapter demonstrates two optimisation case studies not from antenna or R/F fields to show the proposed SB-SADEA's optimisation performance on other fields. In Section 6.1, a supercontinuum generation waveguide optimisation is demonstrated using SB-SADEA. In Section 6.2, a holistic radar signal processing and recognition system is optimised using SB-SADEA. Both experiments validate the hypothesis that the proposed SB-SADEA works on general optimisation problems.

6.1 Supercontinuum generation waveguide optimisation

6.1.1 Engineering background

The optimisation of a supercontinuum generation waveguide here means optimising the cross-section of an optical waveguide to produce the broadest supercontinuum possible. In this case study, the waveguide is a structure which guides electromagnetic radiation in or around the visible spectrum. A supercontinuum is produced when a laser is fired into the waveguide, and broad-spectrum light is obtained. The ultimate objective of such waveguides is to make photonic integrated circuits (ICs) similar to silicon ICs but much smaller and more flexible.

Light interferes with itself constructively and destructively as it bounces around (Pfleegor and Mandel 1967), and different intensity patterns are obtained. The intensity pattern is called mode. The higher order the mode is on, the more nodes it has (Caroselli et al. 2017). The transverse-electric (TE) and transverse-magnetic (TM) modes arise because the light has perpendicular electric and magnetic components. For the TE mode, the magnetic component is in the propagation direction; for the TM mode, the electric component is in the propagation direction. TE and TM modes can have different properties.

Group velocity dispersion (GVD) describes how light propagation changes with wavelength. To ensure that the laser pulse disintegrates in the right way, the GVD below 0 is needed (McKay et al. 2023). As the GVD is related to the effective refractive index, the contributing factors are considered design variables relating to GVD. Theoretically, the design variables should include the following:

- Wavelength.
- Material refractive index.
- Waveguide cross-section.
- Mode.

Amongst these factors, the material and its refractive index are fixed. A wavelength of 780 nm is also fixed for the laser and will provide the input wavelength. Therefore, to optimise the supercontinuum generation waveguide, the waveguide cross-section (i.e., waveguide width and waveguide thickness) and mode are considered as the design variables for the optimisation.

ANSYS® Lumerical MODE is used as the supercontinuum generation simulation software.

6.1.2 Optimisation results and discussion

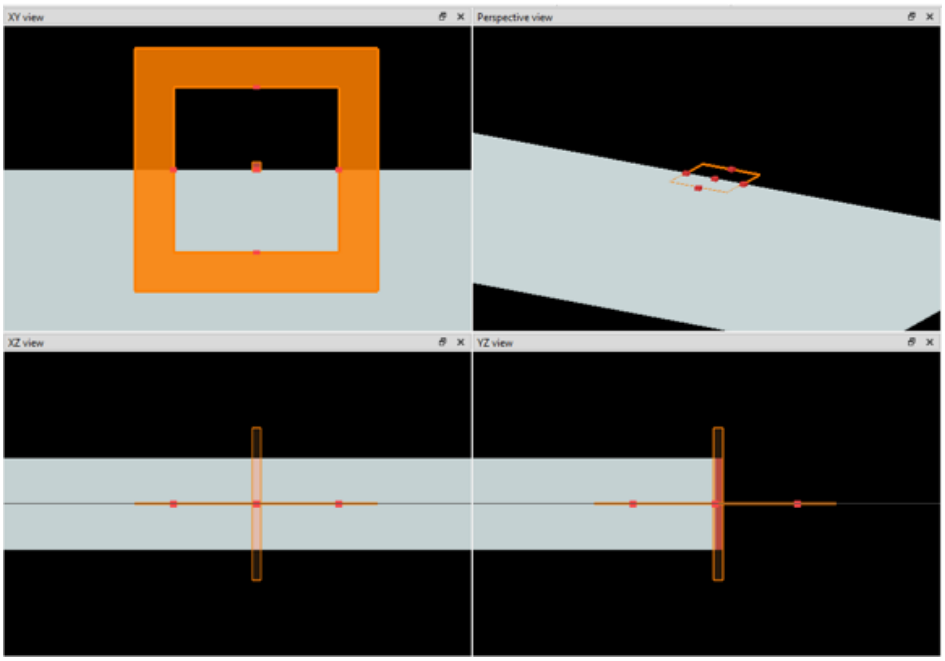


Figure 6.1: The structure of the waveguide. Screenshot of Ansys Lumerical MODE.

Figure 6.1 show a screenshot of Ansys Lumerical MODE, demonstrating the waveguide in XY view, perspective view, XZ view and YZ view, respectively.

Figure 6.2 illustrates the parametric sweep result of the GVD, showing the distribution of positive and negative GVD with respect to waveguide thickness and width. This also implicitly shows that using a parametric sweep to search for the optimal supercontinuum generation waveguide is possible but computationally expensive.

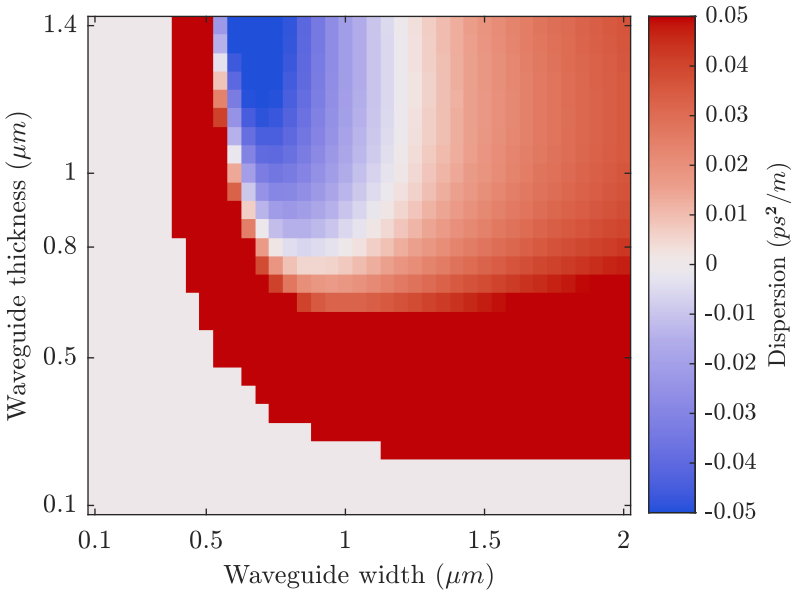


Figure 6.2: Supercontinuum generation waveguides optimisation using parametric sweep. This figure is modified from (McKay 2024)

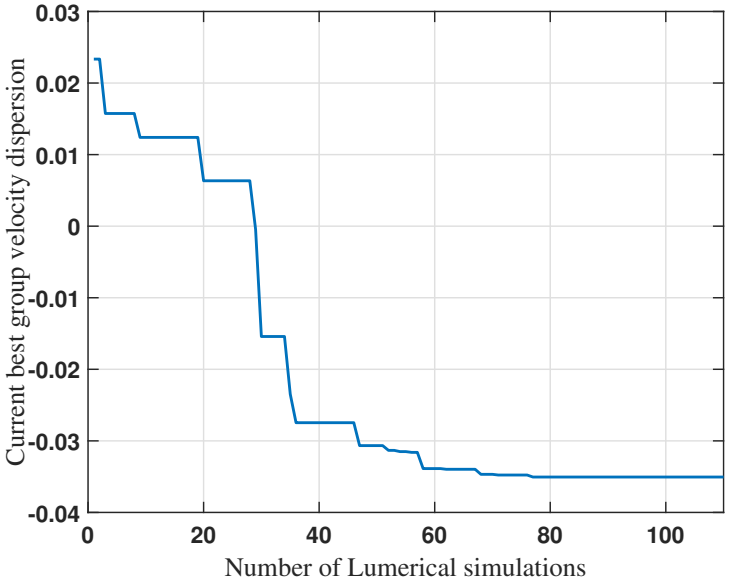


Figure 6.3: The convergence trend of the supercontinuum generation waveguides optimisation.

Figure 6.3 shows the convergence trend of the waveguide optimisation using the proposed method. The current best solution reaches 0 for only approximately 30 Lemerical simulations and converges at approximately -0.04 in less than 80 Lumerical simulations. This is much computationally cheaper than using a parametric sweep.

6.2 Radar signal processing and human activity recognition system optimisation

6.2.1 Engineering background

Recently, human activity recognition (HAR) has drawn much attention in fields like smart homes, assisted living and security (Le Kernec et al. 2019). And radio frequency sensors like radar can effectively collect data for the recognition. Radar-based HAR mainly utilises micro-Doppler signature (MDS) as it provides useful features for distinguishing different activities (Chen et al. 2006). This has been proven available by multiple works (Le Kernec et al. 2019; Gurbuz and Amin 2019; Yang et al. 2023; Li et al. 2019).

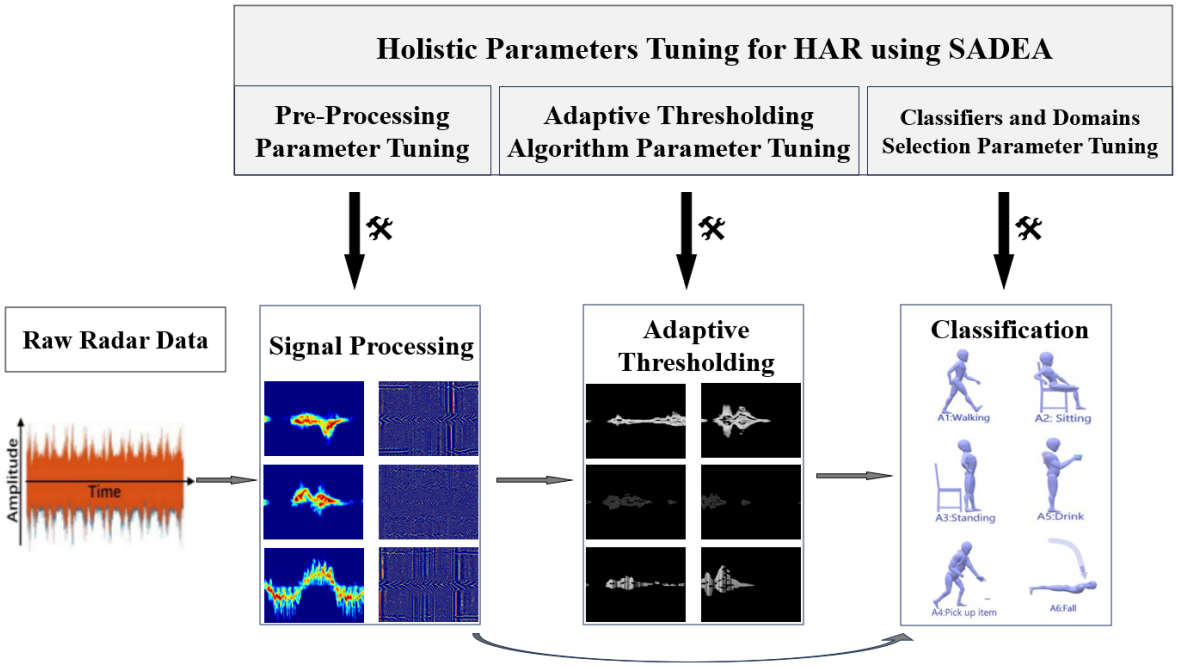


Figure 6.4: Holistic HAR optimisation from signal processing to classification using SADEA. This figure is modified from (Li et al. 2024).

The holistic approach is validated using the University of Glasgow human radar signatures dataset (Fioranelli et al. 2019a; Fioranelli et al. 2019b), and the dataset has over 1754 data samples captured from 72 individuals with aged from 21 to 98 years old. Furthermore, an patented adaptive thresholding method (Romain et al. 2023; Le Kernec et al. 2023; Li et al. 2021) is applied in the HAR and it can binarise the grayscale MDS image data with a threshold T. The method adaptively focuses on the region of interest by setting the threshold and adjust it depending on the data being windowed.

Figure 6.4 shows the holistic HAR system structure optimisation framework from signal pre-processing, to adaptive thresholding, and to domain selection. Using SADEA, all three parts are optimised simultaneously.

On the other hand, With the introduction of feature extractions, the number of tunable parameters increased, making the performance of the HAR system more dependent on the parameter settings. Therefore, a set of optimal system parameters is essential for the system. There are 12 design variables in total in this optimisation. There are seven binary design variables indicating to include or not to include such features, and five ordinary continuous design variables. The design variables include the following:

- Binary
    1. Window shape (rectangular or hamming).
    2. Binary mask.
    3. Masked phase.
    4. Masked unwrapped phase.
    5. Spectrogram.
    6. Masked spectrogram (radar).
    7. Masked spectrogram (patented).
- Continuous
    8. Overlapping factor F (from 0.5 to 0.95).
    9. Time window length (from 100 to 1000).
    10. Difference value V (from 0.01 to 1).
    11. Adaptive thresholding type T_e (from -20 to 20, integer).
    12. Clipping time K (from 1.5 to 5 seconds).

Amongst the system design parameters, window shape (rectangular or hamming window), overlapping factor F, time windowing length and clipping time K are from the data pre-processing. Difference value V and adaptive thresholding type T_e are with the adaptive thresholding. Mask, masked phase, masked unwrapped phase, spectrogram, masked spectrogram (radar data) and masked spectrogram (radar data through pat-

ented adaptive thresholding) are from domain selection. One or multiple domains are selected as training samples for the classification.In the University of Glasgow human radar signatures dataset (Fioranelli et al. 2019a; Fioranelli et al. 2019b) mentioned above, there are six different human activities labelled, and they are:

1. Walking.
2. Sitting.
3. Standing.
4. Drinking.
5. Picking up.
6. Falling.

In this case study, support vector machine (SVM) and AlexNet is applied as the classifier to the HAR system, respectively.

In the following, Subsection 6.2.2 and Subsection 6.2.4 briefly introduce SVM and AlexNet, respectively. The optimisation results and analysis for SVM embedded HAR and AlexNet embedded HAR are presented in Subsection 6.2.3 and Subsection 6.2.5, respectively.

6.2.2 SVM as the classifier

SVM is a frequently used supervised classification algorithm model. It finds the optimal hyperplane that maximise the separation between the data samples of different classes and it is known for its effectiveness in high-dimensional classifications and robustness against overfitting (Cortes and Vapnik 1995). SVM is used in a previous study and it is outperforming deep learning in that specific similar task (Li et al. 2023). In this case study, SVM is chosen to be the classifier in the system illustrated in Figure 6.4 in the first experiment.

6.2.3 Optimisation results and discussion: SVM as the classifier

Figure 6.5 shows the convergence trend for the HAR system optimisation with SVM as the classifier, with a referenced manual design result in which the accuracy is 88.02%. From the trend, the HAR system optimisation converges after 2000 function evaluations using SADEA, in which the averaged accuracy is 89.41% from three independent optimisation runs. The absolute accuracy is improved by 11.39% from using manual design.

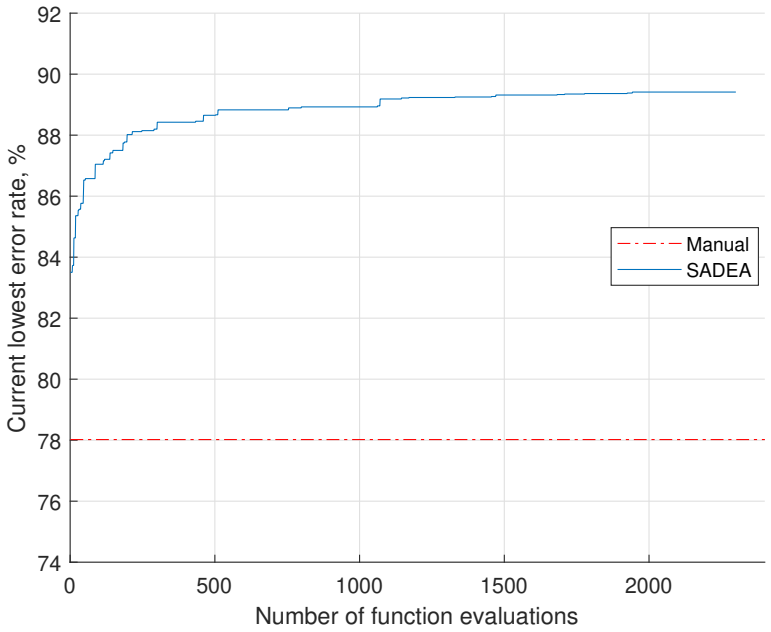


Figure 6.5: Convergence trend of the holistic HAR (with SVM as the classifier) optimisation using SADEA with a referenced result using manual design.

With SVM as the classifier, the optimised HAR system parameters are listed in Table 6.1

6.2.4 AlexNet as the classifier

AlexNet is a widely used convolutional neural network (CNN) architecture proposed specifically for classification tasks. It consists of five convolutional layers followed by three fully connected layers, utilising techniques such as ReLU activation, max pooling, and dropout to improve performance. It is one of the state-of-the-art CNN classification architectures.

Table 6.1: Optimised parameters: SVM as the classifier

Variable names	Optimised values
Window shape	Hamming
Binary mask	Yes
Masked phase	Yes
Masked unwrapped phase	Yes
Spectrogram	No
Masked spectrogram (radar)	Yes
Masked spectrogram (patented)	Yes
Overlapping factor F	0.95
Time window length	154 ms
Difference value V	0.9166
Adaptive thresholding type T_e	3
Clipping time K	4.58 s

In Figure 6.4, AlexNet (Krizhevsky et al. 2012) is used as the classifier for the final activity classification tasks.

6.2.5 Optimisation results and discussion: AlexNet as the classifier

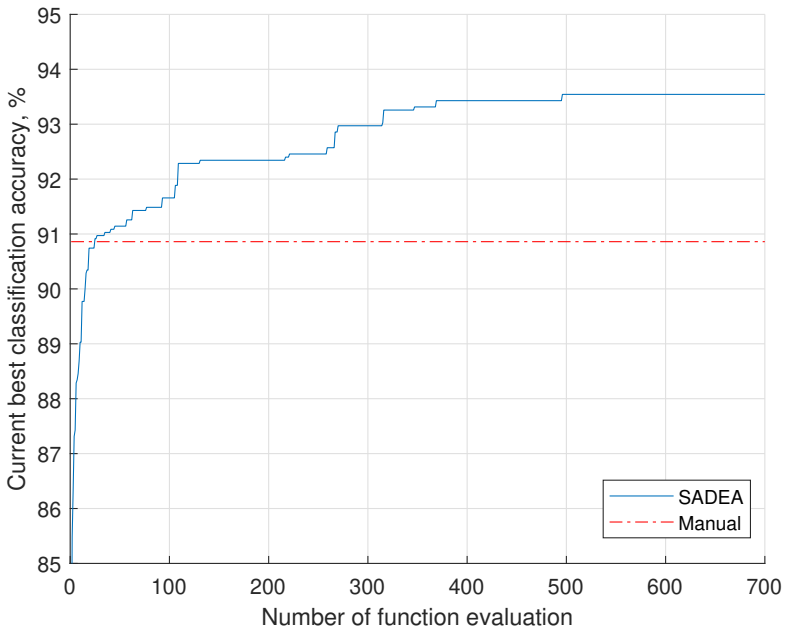


Figure 6.6: Convergence trend of the holistic HAR (with AlexNet as the classifier) optimisation using SADEA with a referenced result using manual design.

Figure 6.6 shows the convergence trend of the holistic optimisation of radar signal processing and human activity recognition systems. From the plot, the optimisation converges at a classification accuracy of approximately 93.5% in about 500 function evaluations. Before the full convergence, the optimisation reaches good results using around 150 function evaluations. The optimised system has a significantly high classification accuracy than using manual tuning and adjustment (Zhang and Cao 2018), which has a classification accuracy of 90.86%.

With AlexNet as the classifier, the optimised HAR system parameters are listed in Table 6.2

Table 6.2: Optimised parameters: AlexNet as the classifier

Variable names	Optimised values
Window shape	Hamming
Binary mask	Yes
Masked phase	Yes
Masked unwrapped phase	Yes
Spectrogram	Yes
Masked spectrogram (radar)	Yes
Masked spectrogram (patented)	No
Overlapping factor F	0.87
Time window length	390 ms
Difference value V	0.8249
Adaptive thresholding type T_e	-13
Clipping time K	5 s

A potential reason that the optimised HAR excludes the patented masked spectrogram is not selected is that the AlexNet-extracted features from this domain duplicates the masked spectrogram (radar) domain. Hence, Including the patented masked spectrogram does not improve the overall accuracy in this specific setting.

Chapter 7

Conclusions and future work

In this thesis, the behaviour of the BNN model under the SAEA background is investigated via empirical study. To the best of my knowledge, this is the first attempt in the SAEA domain. The following potential of BNN is shown:

- Regarding prediction accuracy, BNN is comparable to GP.
- BNN provides prediction uncertainty quantification with a statistical basis, and the magnitude is often smaller than GP but sufficient for SAEA uses.
- BNN has a reasonably low cost of training when under SAEAs. The training cost does not become a challenge when handling problems with about 60 decision variables, while most engineering optimisation problems often have a smaller dimensionality.
- The LCB method is a suitable prescreening method for BNN, although EI and PI can also co-work with BNN.
- BNN has good performance for all three critical factors (i.e., prediction accuracy, prediction uncertainty, and cost of training), while one of them is often a challenge for GP, RBF, and the drop-out method.

Hence, BNN is a possible machine learning alternative for SAEA design, which may inspire many new SAEAs as well as prescreening methods.

With the investigated behaviour of the BNN as surrogate models, the SB-SADEA has been proposed. Its effectiveness and efficiency are presented by multiple real-world challenging antenna design cases. The main contributions regarding the proposed SB-SADEA include:

- Introducing the BNN-based surrogate modelling into online antenna global optimisation area to replace GP modelling.

- Introducing an ad hoc self-adaptive LCB method to work with BNN-based surrogate model.

Hence, significant advantages are obtained regarding convergence efficiency (reflected by the number of EM simulations needed to obtain the optimal design) and the cost of machine learning. Thanks to the above innovations, SB-SADEA transforms the SADEA series from GP model-based to BNN model-based, and it becomes universal for antennas with various numbers of design variables and specifications, with significant performance improvement at the same time. Future works will include behavioural analysis of SB-SADEA and its applications in wider domains.

Appendices

A Test problems

A.1 F1: 10-D Ackley problem

$$\begin{aligned} &\min_x \quad f_1(x) = -20e^{-\frac{1}{5}\sqrt{\frac{1}{10}\sum_{i=1}^{10} x_i^2}} - e^{\frac{1}{10}\sum_{i=1}^{10}\cos(2\pi x_i)} \\ &x \in [-30, 30]^{10} \\ &\text{optimum:} x^* = [0, 0, \cdots, 0] \\ &\text{minimum:} f(x^*) = 0 \end{aligned} \tag{1}$$

A.2 F2: 10-D Griewank problem

$$\begin{aligned} &\min_x \quad f_2(x) = 1 + \sum_{i=1}^{10} \frac{x_i^2}{4000} - \prod_{i=1}^{10} \cos(\frac{x_i}{\sqrt{i}}) \\ &x \in [-600, 600]^{10} \\ &\text{optimum:} x^* = [0, 0, \cdots, 0] \\ &\text{minimum:} f(x^*) = 0 \end{aligned} \tag{2}$$

A.3 F3: Circular antenna array optimization

$$\begin{aligned}\min_{\phi_0} \quad f_3(\phi_0) = &\frac{|AR(\phi_{sll}, \vec{I}, \vec{\beta}, \phi_0)|}{|AR(\phi_{\max}, \vec{I}, \vec{\beta}, \phi_0)|} + \frac{1}{DIR(\phi_0, \vec{I}, \vec{\beta})} \\ &+ |\phi_0 - \phi_{\text{des}}| + \sum_{k=1}^{\text{num}} |AR(\phi_k, \vec{I}, \vec{\beta}, \phi_0)|\end{aligned} \tag{3}$$

where ϕ_0 defines the circular array parameters, AR is the axial ratio, DIR is the directivity, $\vec{I}$ is the current excitation and $\vec{\beta}$ is the phase excitation. The circular array has 12 elements. ϕ_0 has 12 dimensions, the first 6 dimensions are bounded in $[0.2, 1]$ and the last 6 dimensions are bounded in $[-180, 180]$. There are multiple known optimum designs and the known minimum is about -21.8. See details in (Das and Suganthan 2010).

A.4 F4: 12-resonator diplexer coupling matrix optimization

$$\begin{aligned}\min_{x} \quad f_4(x) = &\frac{\max(\max(|S_{11,b_1}(x)|) - (-20), 0)}{20} \\ &+ \frac{\max(\max(|S_{11,b_2}(x)|) - (-20), 0)}{20} \\ &+ \frac{\max(\max(|S_{32,b_1}(x)|) - (-60), 0)}{60} \\ &+ \frac{\max(\max(|S_{32,b_2}(x)|) - (-60), 0)}{60}\end{aligned} \tag{4}$$

where S_{11} and S_{32} are the S-parameters, b_1 and b_2 are the two passbands. The search ranges and known optimum are provided in Table A.1. The known minimum is about 0.054. See details in (Yu et al. 2020).

Table A.1: Search ranges and known optimum of the coupling coefficients

Variable No.	Lower bound	Upper bound	Known optimum
1	-0.5	-0.3	-0.3324
2	0.205	0.305	0.2761
3	0.1844	0.2844	0.2440
4	0.2024	0.3024	0.2595
5	0.3325	0.4325	0.3885
6	0.1614	0.3614	0.1868
7	-0.152	-0.052	-0.1105
8	0.6219	1.1219	0.6723
9	0.3	0.5	0.3088
10	0.1751	0.2751	0.2410
11	0.1601	0.2601	0.2110
12	0.1751	0.2751	0.2233
13	0.2816	0.3816	0.3370
14	0.1343	0.3343	0.1604
15	0.0314	0.1314	0.1002
16	-0.1	0.1	-0.0215
17	-0.6269	-0.5269	-0.5998
18	0.5667	0.6667	0.6334
19	-0.1	0.1	0.0306
20	-0.6209	-0.5209	-0.5840
21	-0.4215	-0.3215	-0.3675
22	-0.6104	-0.5104	-0.5697
23	-0.6167	-0.5167	-0.5571
24	0.5636	0.6636	0.6165
25	0.4048	0.5048	0.4280
26	0.5552	0.6552	0.6111
27	0.5546	0.6546	0.6172

Bibliography

Akinsolu, Mobayode (2019). 'Efficient surrogate model-assisted evolutionary algorithm for electromagnetic design automation with applications'. PhD thesis. University of Chester.

Akinsolu, Mobayode O et al. (2019). 'A parallel surrogate model assisted evolutionary algorithm for electromagnetic design optimization'. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 3.2, pp. 93–105.

Alieldin, Ahmed et al. (2018). 'A triple-band dual-polarized indoor base station antenna for 2G, 3G, 4G and sub-6 GHz 5G applications'. In: *IEEE Access* 6, pp. 49209–49216.

Altshuler, Edward E and Derek S Linden (1997). 'Wire-antenna designs using genetic algorithms'. In: *IEEE Antennas and Propagation magazine* 39.2, pp. 33–43.

ANSYS (Accessed 2024). *ANSYS HFSS*. URL: `https://www.ansys.com/products/electronics/ansys-hfss`.

Ares-Pena, Francisco J et al. (1999). 'Genetic algorithms in the design and optimization of antenna array patterns'. In: *IEEE Transactions on Antennas and Propagation* 47.3, pp. 506–510.

Arora, JS et al. (1995). 'Global optimization methods for engineering applications: a review'. In: *Structural optimization* 9, pp. 137–159.

Audet, Charles and John E Dennis Jr (2002). 'Analysis of generalized pattern searches'. In: *SIAM Journal on optimization* 13.3, pp. 889–903.

Bäck, Thomas and Hans-Paul Schwefel (1993). 'An overview of evolutionary algorithms for parameter optimization'. In: *Evolutionary computation* 1.1, pp. 1–23.

Balanis, Constantine A (2011). *Modern antenna handbook*. John Wiley & Sons.

Bauer, Matthias, Mark van der Wilk and Carl Edward Rasmussen (2016). 'Understanding probabilistic sparse Gaussian process approximations'. In: *Advances in neural information processing systems* 29.

Bergstra, James and Yoshua Bengio (2012). 'Random search for hyper-parameter optimization.' In: *Journal of machine learning research* 13.2.

Blank, Stephen J (1990). 'Array feed/reflector antenna design for intense microwave beams'. In: *Intense Microwave and Particle Beams*. Vol. 1226. SPIE, pp. 252–263.

Blank, Stephen Jon and Michael F Hutt (2005). 'On the empirical optimization of antenna arrays'. In: *IEEE Antennas and Propagation Magazine* 47.2, pp. 58–67.

Blei, David M, Alp Kucukelbir and Jon D McAuliffe (2017). 'Variational inference: A review for statisticians'. In: *Journal of the American statistical Association* 112.518, pp. 859–877.

Boggs, Paul T and Jon W Tolle (1995). 'Sequential quadratic programming'. In: *Acta numerica* 4, pp. 1–51.

Bozorg-Haddad, Omid, Mohammad Solgi and Hugo A Loáiciga (2017). *Meta-heuristic and evolutionary algorithms for engineering optimization*. John Wiley & Sons.

Breiman, Leo (1996). 'Bagging predictors'. In: *Machine learning* 24.2, pp. 123–140.

Briffoteaux, Guillaume et al. (2020). 'Parallel surrogate-assisted optimization: Batched Bayesian Neural Network-assisted GA versus q-EGO'. In: *Swarm and Evolutionary Computation* 57, p. 100717.

Budak, Ahmet Faruk et al. (2021). 'An efficient analog circuit sizing method based on machine learning assisted global optimization'. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41.5, pp. 1209–1221.

Cai, Xiwen, Liang Gao and Xinyu Li (2019). 'Efficient generalized surrogate-assisted evolutionary algorithm for high-dimensional expensive problems'. In: *IEEE Transactions on Evolutionary Computation* 24.2, pp. 365–379.

Cameron, Richard J (2003). 'Advanced coupling matrix synthesis techniques for microwave filters'. In: *IEEE Transactions on Microwave Theory and Techniques* 51.1, pp. 1–10.

Caroselli, Raffaele et al. (2017). 'Experimental study of the sensitivity of a porous silicon ring resonator sensor using continuous in-flow measurements'. In: *Optics Express* 25.25, pp. 31651–31659.

Chakraborty, Avishek et al. (2023). 'Multi-pattern synthesis in fourth-dimensional antenna arrays using BGM-based quasi-Newton memetic optimization method'. In: *International Journal of Microwave and Wireless Technologies*, pp. 1–11.

Chen, Guodong et al. (2021). 'Efficient hierarchical surrogate-assisted differential evolution for high-dimensional expensive optimization'. In: *Information Sciences* 542, pp. 228–246.

Chen, Guodong et al. (2022a). 'A radial basis function surrogate model assisted evolutionary algorithm for high-dimensional expensive optimization problems'. In: *Applied Soft Computing* 116, p. 108353.

Chen, Victor C et al. (2006). 'Micro-Doppler effect in radar: phenomenon, model, and simulation study'. In: *IEEE Transactions on Aerospace and electronic systems* 42.1, pp. 2–21.

Chen, Weiqi et al. (2022b). 'Multibranch machine learning-assisted optimization and its application to antenna design'. In: *IEEE Transactions on Antennas and Propagation* 70.7, pp. 4985–4996.

Chiandussi, Giorgio et al. (2012). 'Comparison of multi-objective optimization methodologies for engineering applications'. In: *Computers & Mathematics with Applications* 63.5, pp. 912–942.

Cortes, Corinna and Vladimir Vapnik (1995). 'Support-vector networks'. In: *Machine learning* 20, pp. 273–297.

Couckuyt, Ivo et al. (2010). 'Surrogate-based infill optimization applied to electromagnetic problems'. In: *International Journal of RF and Microwave Computer-Aided Engineering* 20.5, pp. 492–501.

Crepaldi, Marco et al. (2014). 'A top-down constraint-driven methodology for smart system design'. In: *IEEE Circuits and Systems Magazine* 14.1, pp. 37–57.

CST (Accessed 2024). 'Computer Simulation Technology: Microwave Studio'. In: URL: `https://www.cst.com/products/cstmws`.

Danjuma, Isah Musa et al. (2020). 'Design and optimization of a slotted monopole antenna for ultra-wide band body centric imaging applications'. In: *IEEE Journal of Electromagnetics, RF and Microwaves in Medicine and Biology* 4.2, pp. 140–147.

Das, Swagatam and Ponnuthurai N Suganthan (2010). 'Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems'. In: *Jadavpur University, Nanyang Technological University, Kolkata*, pp. 341–359.

Deb, Arindam, Jibendu Sekhar Roy and Bhaskar Gupta (2017). 'A differential evolution performance comparison: Comparing how various differential evolution algorithms perform in designing microstrip antennas and arrays'. In: *IEEE Antennas and Propagation Magazine* 60.1, pp. 51–61.

Dennis, J.E. and V. Torczon (1997). 'Managing approximation models in optimization'. In: *Multidisciplinary design optimization: State-of-the-art*, pp. 330–347.

Dennis Jr, John E and Jorge J Moré (1977). 'Quasi-Newton methods, motivation and theory'. In: *SIAM review* 19.1, pp. 46–89.

Dieterich, Johannes M and Bernd Hartke (2012). 'Empirical Review of Standard Benchmark Functions Using Evolutionary Global Optimization'. In: *Applied Mathematics* 3, pp. 1552–1564.

Eiben, Agoston E et al. (2015). 'What is an evolutionary algorithm?' In: *Introduction to evolutionary computing*, pp. 25–48.

Emmerich, M.T.M., K.C. Giannakoglou and B. Naujoks (2006). 'Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels'. In: *IEEE Transactions on Evolutionary Computation* 10.4, pp. 421–439.

Fan, Zhun et al. (2008). 'Improved differential evolution based on stochastic ranking for robust layout synthesis of MEMS components'. In: *IEEE Transactions on Industrial Electronics* 56.4, pp. 937–948.

Fioranelli, Dr Francesco et al. (2019a). 'Radar sensing for healthcare: Associate editor francesco fioranelli on the applications of radar in monitoring vital signs and recognising human activity patterns'. In: *Electronics Letters* 55.19, pp. 1022–1024.

Fioranelli, Francesco et al. (2019b). 'Radar signatures of human activities'. In.

Floudas, Christodoulos A and Chrysanthos E Gounaris (2009). 'A review of recent advances in global optimization'. In: *Journal of Global Optimization* 45, pp. 3–38.

Folgoc, Loic Le et al. (2021). 'Is MC Dropout Bayesian?' In: *arXiv preprint arXiv:2110.04286*.

Fu, Kai et al. (2022). 'An efficient surrogate assisted particle swarm optimization for antenna synthesis'. In: *IEEE Transactions on Antennas and Propagation* 70.7, pp. 4977–4984.

Fujimoto, Kyohei and Hisashi Morishita (2013). *Modern small antennas*. Cambridge University Press.

Gal, Yarin and Zoubin Ghahramani (2015). 'Bayesian convolutional neural networks with Bernoulli approximate variational inference'. In: *arXiv preprint arXiv:1506.02158*.

– (2016). 'Dropout as a bayesian approximation: Representing model uncertainty in deep learning'. In: *international conference on machine learning*. PMLR, pp. 1050–1059.

Gao, Steven et al. (2009). 'Antennas for modern small satellites'. In: *IEEE Antennas and Propagation Magazine* 51.4, pp. 40–56.

Gao, Zhengqi et al. (2019). 'Efficient performance trade-off modeling for analog circuit based on Bayesian neural network'. In: *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, pp. 1–8.

Giannakoglou, KC (2002). 'Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence'. In: *Progress in Aerospace sciences* 38.1, pp. 43–76.

Gill, Philip E and Elizabeth Wong (2011). 'Sequential quadratic programming methods'. In: *Mixed integer nonlinear programming*. Springer, pp. 147–224.

Giunta, Anthony and Layne Watson (1998). 'A comparison of approximation modeling techniques-polynomial versus interpolating models'. In: *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, p. 4758.

Goan, Ethan and Clinton Fookes (2020). 'Bayesian neural networks: An introduction and survey'. In: *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018*, pp. 45–87.

Goudos, Sotirios K et al. (2011). 'Self-adaptive differential evolution applied to real-valued antenna and microwave design problems'. In: *IEEE Transactions on Antennas and Propagation* 59.4, pp. 1286–1298.

Gregory, Micah D, Zikri Bayraktar and Douglas H Werner (2011). 'Fast optimization of electromagnetic design problems using the covariance matrix adaptation evolutionary strategy'. In: *IEEE Transactions on Antennas and Propagation* 59.4, pp. 1275–1285.

Guo, Dan et al. (2018). 'Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems'. In: *IEEE transactions on cybernetics* 49.3, pp. 1012–1025.

Guo, Dan et al. (2021). 'Evolutionary optimization of high-dimensional multiobjective and many-objective expensive problems assisted by a dropout neural network'. In: *IEEE transactions on systems, man, and cybernetics: systems* 52.4, pp. 2084–2097.

Gurbuz, Sevgi Zubeyde and Moeness G Amin (2019). 'Radar-based human-motion recognition with deep learning: Promising applications for indoor monitoring'. In: *IEEE Signal Processing Magazine* 36.4, pp. 16–28.

Hammond, P (1954). 'A short modern review of fundamental electromagnetic theory'. In: *Proceedings of the IEE-Part I: General* 101.130, pp. 147–165.

Hara, Kazuyuki, Daisuke Saitoh and Hayaru Shouno (2016). 'Analysis of dropout learning regarded as ensemble learning'. In: *Artificial Neural Networks and Machine Learning–ICANN 2016: 25th International Conference on Artificial Neural Networks, Barcelona, Spain, September 6-9, 2016, Proceedings, Part II 25*. Springer, pp. 72–79.

Haupt, Randy L (1995). 'An introduction to genetic algorithms for electromagnetics'. In: *IEEE Antennas and Propagation Magazine* 37.2, pp. 7–15.

Hawe, Glenn I and Jan K Sykulski (2008). 'A scalarizing one-stage algorithm for efficient multi-objective optimization'. In: *IEEE Transactions on Magnetics* 44.6, pp. 1094–1097.

Holland, John H (1984). 'Genetic algorithms and adaptation'. In: *Adaptive control of ill-defined systems*, pp. 317–333.

Hong, Jia-Shen G and Michael J Lancaster (2004). *Microstrip filters for RF/microwave applications*. John Wiley & Sons.

Hong, Wei et al. (2017). 'Multibeam antenna technologies for 5G wireless communications'. In: *IEEE Transactions on Antennas and Propagation* 65.12, pp. 6231–6249.

Hoorfar, Ahmad (2007). 'Evolutionary programming in electromagnetic optimization: a review'. In: *IEEE transactions on antennas and propagation* 55.3, pp. 523–537.

Hopkins, Brian and John Gordon Skellam (1954). 'A new method for determining the type of distribution of plant individuals'. In: *Annals of Botany* 18.2, pp. 213–227.

Hornby, Gregory et al. (2006). 'Automated antenna design with evolutionary algorithms'. In: *Space 2006*, p. 7242.

Hu, Caie et al. (2021). 'On nonstationary Gaussian process model for solving data-driven optimization problems'. In: *IEEE Transactions on Cybernetics*.

Huang, Qiujun, Jingli Mao and Yong Liu (2012). 'An improved grid search algorithm of SVR parameters optimization'. In: *2012 IEEE 14th International Conference on Communication Technology*. IEEE, pp. 1022–1026.

Hussain, Kashif et al. (2017). 'Common Benchmark Functions for Metaheuristic Evaluation: A Review'. In: *International Journal on Informatics Visualization* 1.4.

Hussine, Umniyyah Ulfa, Yi Huang and Chaoyun Song (2017). 'A new circularly polarized antenna for GNSS applications'. In: *2017 11th European Conference on Antennas and Propagation (EUCAP)*. IEEE, pp. 1954–1956.

Jamil, Momin et al. (2013). 'A literature survey of benchmark functions for global optimisation problems'. In: *International Journal of Mathematical Modelling and Numerical Optimisation* 4.2, pp. 150–194.

Jayasinghe, JM et al. (2015). 'Nonuniform overlapping method in designing microstrip patch antennas using genetic algorithm optimization'. In: *International Journal of Antennas and Propagation* 2015.

Jia, Xiaotao et al. (2020). 'Efficient computation reduction in Bayesian neural networks through feature decomposition and memorization'. In: *IEEE transactions on neural networks and learning systems* 32.4, pp. 1703–1712.

Jin, Ruichen, Wei Chen and Timothy W Simpson (2001). 'Comparative studies of metamodelling techniques under multiple modelling criteria'. In: *Structural and multidisciplinary optimization* 23, pp. 1–13.

Jin, Yaochu (2005). 'A comprehensive survey of fitness approximation in evolutionary computation'. In: *Soft computing* 9.1, pp. 3–12.

– (2011). 'Surrogate-assisted evolutionary computation: Recent advances and future challenges'. In: *Swarm and Evolutionary Computation* 1.2, pp. 61–70.

John, Matthias and Max J Ammann (2009). 'Antenna optimization with a computationally efficient multiobjective evolutionary algorithm'. In: *IEEE transactions on antennas and propagation* 57.1, pp. 260–263.

Jones, Donald R, Matthias Schonlau and William J Welch (1998). 'Efficient global optimization of expensive black-box functions'. In: *Journal of Global optimization* 13.4, pp. 455–492.

Jospin, Laurent Valentin et al. (2022). 'Hands-on Bayesian neural networks - A tutorial for deep learning users'. In: *IEEE Computational Intelligence Magazine* 17.2, pp. 29–48.

Joyce, James M (2011). 'Kullback-leibler divergence'. In: *International encyclopedia of statistical science*. New York: Springer, pp. 720–722.

Kennedy, James (2011). 'Particle swarm optimization'. In: *Encyclopedia of machine learning*. Springer, pp. 760–766.

Kennedy, James and Russell Eberhart (1995). 'Particle swarm optimization'. In: *Proceedings of ICNN'95-international conference on neural networks*. Vol. 4. IEEE, pp. 1942–1948.

Keyes, David E et al. (2013). 'Multiphysics simulations: Challenges and opportunities'. In: *The International Journal of High Performance Computing Applications* 27.1, pp. 4–83.

Khan, Rizwan et al. (2018). 'User influence on mobile terminal antennas: A review of challenges and potential solution for 5G antennas'. In: *IEEE access* 6, pp. 77695–77715.

Khan, Zia Ullah et al. (2020). 'Empty substrate-integrated waveguide-fed patch antenna array for 5G millimeter-wave communication systems'. In: *IEEE Antennas and Wireless Propagation Letters* 19.5, pp. 776–780.

Khodier, Majid M and Christos G Christodoulou (2005). 'Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization'. In: *IEEE Transactions on Antennas and propagation* 53.8, pp. 2674–2679.

King, Ronold Wyeth Percival et al. (1981). 'Antennas in matter: Fundamentals, theory, and applications'. In: *NASA STI/Recon Technical Report A* 81, p. 29690.

Kirkpatrick, Scott, C Daniel Gelatt Jr and Mario P Vecchi (1983). 'Optimization by simulated annealing'. In: *science* 220.4598, pp. 671–680.

Koziel, S and Adrian Bekasiewicz (2016). 'Fast EM-driven optimization using variable-fidelity EM models and adjoint sensitivities'. In: *IEEE Microwave and Wireless Components Letters* 26.2, pp. 80–82.

Koziel, Slawomir and Adrian Bekasiewicz (2018). 'Sequential approximate optimisation for statistical analysis and yield optimisation of circularly polarised antennas'. In: *IET Microwaves, Antennas & Propagation* 12.13, pp. 2060–2064.

Koziel, Slawomir, Qingsha S Cheng and Song Li (2018). 'Optimization-driven antenna design framework with multiple performance constraints'. In: *International Journal of RF and Microwave Computer-Aided Engineering* 28.4, e21208.

Koziel, Slawomir and Anna Pietrenko-Dabrowska (2019a). 'Performance-based nested surrogate modeling of antenna input characteristics'. In: *IEEE Transactions on Antennas and Propagation* 67.5, pp. 2904–2912.

– (2019b). 'Reduced-cost electromagnetic-driven optimisation of antenna structures by means oftrust-region gradient-search with sparse Jacobian updates'. In: *IET Microwaves, Antennas & Propagation* 13.10, pp. 1646–1652.

Koziel, Slawomir, Anna Pietrenko-Dabrowska and Ubaid Ullah (2021). 'Low-cost modeling of microwave components by means of two-stage inverse/forward surrogates and domain confinement'. In: *IEEE Transactions on Microwave Theory and Techniques* 69.12, pp. 5189–5202.

Koziel, Slawomir and Sigmar D Unnsteinsson (2018). 'Expedited design closure of antennas by means of trust-region-based adaptive response scaling'. In: *IEEE Antennas and Wireless Propagation Letters* 17.6, pp. 1099–1103.

Koziel, Slawomir et al. (2014). 'Efficient multi-objective simulation-driven antenna design using co-kriging'. In: *IEEE Transactions on Antennas and Propagation* 62.11, pp. 5900–5905.

Krizhevsky, Alex, Ilya Sutskever and Geoffrey E Hinton (2012). 'Imagenet classification with deep convolutional neural networks'. In: *Advances in neural information processing systems* 25.

Kudela, Jakub and Radomil Matousek (2023). 'Combining Lipschitz and RBF surrogate models for high-dimensional computationally expensive problems'. In: *Information Sciences* 619, pp. 457–477.

Kurup, Dhanesh G, Mohamed Himdi and Anders Rydberg (2003). 'Synthesis of uniform amplitude unequally spaced antenna arrays using the differential evolution algorithm'. In: *IEEE Transactions on Antennas and Propagation* 51.9, pp. 2210–2217.

Lake, Jonathan James, Amy Elizabeth Duwel and Rob N Candler (2013). 'Particle swarm optimization for design of slotted MEMS resonators with low thermoelastic dissipation'. In: *Journal of Microelectromechanical Systems* 23.2, pp. 364–371.

Lazaridis, Pavlos I et al. (2016). 'Comparison of evolutionary algorithms for LPDA antenna optimization'. In: *Radio Science* 51.8, pp. 1377–1384.

Le Kernec, Julien et al. (2019). 'Radar signal processing for sensing in assisted living: The challenges associated with real-time implementation of emerging algorithms'. In: *IEEE Signal Processing Magazine* 36.4, pp. 29–41.

Le Kernec, Julien et al. (2023). *Method and Device for Human Activity Classification Using Radar Micro Doppler and Phase*.

Lee, Chen-Yu et al. (2015). 'Deeply-supervised nets'. In: *Artificial intelligence and statistics*. Pmlr, pp. 562–570.

Lewis, Robert Michael, Virginia Torczon and Michael W Trosset (2000). 'Direct search methods: then and now'. In: *Journal of computational and Applied Mathematics* 124.1-2, pp. 191–207.

Li, Xinyu, Yuan He and Xiaojun Jing (2019). 'A survey of deep learning-based human activity recognition in radar'. In: *Remote Sensing* 11.9, p. 1068.

Li, Zhenghui et al. (2021). 'Human activity classification with adaptive thresholding using radar micro-doppler'. In: *2021 CIE International Conference on Radar (Radar)*. IEEE, pp. 1511–1515.

Li, Zhenghui et al. (2023). 'Radar-based human activity recognition with adaptive thresholding towards resource constrained platforms'. In: *Scientific Reports* 13.1, p. 3473.

Li, Zhenghui et al. (2024). 'A holistic human activity recognition optimisation using AI techniques'. In: *IET Radar, Sonar & Navigation* 18.2, pp. 256–265.

Li, Zhifang, Panos Y Papalambros and John L Volakis (1997). 'Designing broadband patch antennas using the sequential quadratic programming method'. In: *IEEE Transactions on Antennas and Propagation* 45.11, pp. 1689–1692.

Lim, Dudy et al. (2009). 'Generalizing surrogate-assisted evolutionary computation'. In: *IEEE Transactions on Evolutionary Computation* 14.3, pp. 329–355.

Lim, Eng Hock and Kwok Wa Leung (2012). *Compact multifunctional antennas for wireless systems*. Vol. 215. John Wiley & Sons.

Linden, Derek S and Edward E Altshuler (1996). 'Automating wire antenna design using genetic algorithms'. In: *Microwave Journal* 39.3, pp. 74–81.

Liu, Bo, Georges Gielen and Francisco V Fernández (2014a). 'Automated design of analog and high frequency circuits'. In: *A computational intelligence approach. Springer, Berlin, Heidelberg, ISBN*, pp. 978–3.

Liu, Bo, Vic Grout and Anna Nikolaeva (2017a). 'Efficient global optimization of actuator based on a surrogate model assisted hybrid algorithm'. In: *IEEE Transactions on Industrial Electronics* 65.7, pp. 5712–5721.

Liu, Bo, Slawomir Koziel and Nazar Ali (2017b). 'SADEA-II: A generalized method for efficient global optimization of antenna design'. In: *Journal of Computational Design and Engineering* 4.2, pp. 86–97.

Liu, Bo, Hao Yang and Michael J Lancaster (2017c). 'Global Optimization of Microwave Filters Based on a Surrogate Model-Assisted Evolutionary Algorithm'. In: *IEEE Transactions on Microwave Theory and Techniques*.

Liu, Bo, Qingfu Zhang and Georges G. E. Gielen (2014b). 'A Gaussian Process Surrogate Model Assisted Evolutionary Algorith for Medium Scale Expensive Optimization Problems'. In: *IEEE Trans. on Evolutionary Computation* 18.2, pp. 180–192.

Liu, Bo, Qingfu Zhang and Georges GE Gielen (2013). 'A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems'. In: *IEEE Transactions on Evolutionary Computation* 18.2, pp. 180–192.

Liu, Bo et al. (2012). 'Self-adaptive lower confidence bound: A new general and effective prescreening method for gaussian process surrogate model assisted evolutionary algorithms'. In: *2012 IEEE Congress on Evolutionary Computation*. IEEE, pp. 1–6.

Liu, Bo et al. (2014c). 'An Efficient Method for Antenna Design Optimization Based on Evolutionary Computation and Machine Learning Techniques'. In: *IEEE Trans. on Antennas and Propagation* 62.1, pp. 7–18.

Liu, Bo et al. (2014d). 'Behavioral study of the surrogate model-aware evolutionary search framework'. In: *Evolutionary Computation (CEC), 2014 IEEE Congress on*. IEEE, pp. 715–722.

Liu, Bo et al. (2014e). 'Gaspad: A general and efficient mm-wave integrated circuit synthesis method based on surrogate model assisted evolutionary algorithm'. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 33.2, pp. 169–182.

Liu, Bo et al. (2017d). 'GUI design exploration software for microwave antennas'. In: *Journal of Computational Design and Engineering* 4.4, pp. 274–281.

Liu, Bo et al. (2018). 'Efficient global optimisation of microwave antennas based on a parallel surrogate model-assisted evolutionary algorithm'. In: *IET Microwaves, Antennas & Propagation* 13.2, pp. 149–155.

Liu, Bo et al. (2021). 'An efficient method for complex antenna design based on a self adaptive surrogate model-assisted optimization technique'. In: *IEEE Transactions on Antennas and Propagation* 69.4, pp. 2302–2315.

Liu, Wen-Chung (2005). 'Design of a multiband CPW-fed monopole antenna using a particle swarm optimization approach'. In: *IEEE Transactions on Antennas and Propagation* 53.10, pp. 3273–3279.

Liu, Yushi et al. (2022a). 'An Efficient Method for Antenna Design Based on a Self-Adaptive Bayesian Neural Network-Assisted Global Optimization Technique'. In: *IEEE Transactions on Antennas and Propagation* 70.12, pp. 11375–11388.

Liu, Zhening, Handing Wang and Yaochu Jin (2022b). 'Performance Indicator-Based Adaptive Model Selection for Offline Data-Driven Multiobjective Evolutionary Optimization'. In: *IEEE Transactions on Cybernetics*.

Luo, Wenjian et al. (2018). 'Surrogate-assisted evolutionary framework for data-driven dynamic optimization'. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 3.2, pp. 137–150.

Lytle, R Jeffrey and Edwin F Laine (1978). 'Design of a miniature directional antenna for geophysical probing from boreholes'. In: *IEEE Transactions on Geoscience Electronics* 16.4, pp. 304–307.

Marcano, Diogenes and Filinto Durán (2000). 'Synthesis of antenna arrays using genetic algorithms'. In: *IEEE Antennas and Propagation Magazine* 42.3, pp. 12–20.

MathWorks (Accessed 2024). *Matlab Antenna Aoolbox*. URL: `https://uk.mathworks.com/products/antenna.html`.

Maxwell, James Clerk (1864). 'II. A dynamical theory of the electromagnetic field'. In: *Proceedings of the Royal Society of London* 13, pp. 531–536.

– (1865). 'VIII. A dynamical theory of the electromagnetic field'. In: *Philosophical transactions of the Royal Society of London* 155, pp. 459–512.

McKay, Elissa (2024). 'High-Confinement Alumina Waveguides for Ultraviolet-Visible Integrated and Nonlinear Optics'.

McKay, Elissa et al. (2023). 'High-confinement alumina waveguides with sub-dB/cm propagation losses at 450 nm'. In: *Scientific Reports* 13.1, p. 19917.

Michalewicz, Zbigniew (1996). 'Heuristic methods for evolutionary computation techniques'. In: *Journal of Heuristics* 1, pp. 177–206.

Michielssen, Eric et al. (1993). 'Design of lightweight, broad-band microwave absorbers using genetic algorithms'. In: *IEEE Transactions on Microwave Theory and Techniques* 41.6, pp. 1024–1031.

Mockus, Jonas (2005). 'The Bayesian approach to global optimization'. In: *System Modeling and Optimization: Proceedings of the 10th IFIP Conference New York City, USA, August 31–September 4, 1981*. Springer, pp. 473–481.

Myers, Raymond H and Raymond H Myers (1990). *Classical and modern regression with applications*. Vol. 2. Duxbury press Belmont, CA.

Ong, Yew-Soon, Kai Yew Lum and Prasanth B Nair (2008). 'Hybrid evolutionary algorithm with Hermite radial basis function interpolants for computationally expensive adjoint solvers'. In: *Computational Optimization and Applications* 39, pp. 97–119.

Paracha, Kashif Nisar et al. (2019). 'Wearable antennas: A review of materials, structures, and innovative features for autonomous communication and sensing'. In: *IEEE Access* 7, pp. 56694–56712.

Parr, James M et al. (2012). 'Infill sampling criteria for surrogate-based optimization with constraint handling'. In: *Engineering Optimization* 44.10, pp. 1147–1166.

Pfleegor, Robert L and Leonard Mandel (1967). 'Interference of independent photon beams'. In: *Physical Review* 159.5, p. 1084.

Pietrenko-Dabrowska, Anna and Slawomir Koziel (2020a). 'Antenna modeling using variable-fidelity EM simulations and constrained co-kriging'. In: *IEEE Access* 8, pp. 91048–91056.

– (2020b). 'Computationally-efficient design optimisation of antennas by accelerated gradient search with sensitivity and design change monitoring'. In: *IET Microwaves, Antennas & Propagation* 14.2, pp. 165–170.

– (2023). 'Accelerated Parameter Tuning of Antenna Structures by Means of Response Features and Principal Directions'. In: *IEEE Transactions on Antennas and Propagation*.

Pietrenko-Dabrowska, Anna, Slawomir Koziel and Muath Al-Hasan (2020). 'Expedited yield optimization of narrow-and multi-band antennas using performance-driven surrogates'. In: *IEEE Access* 8, pp. 143104–143113.

Pietrenko-Dabrowska, Anna, Slawomir Koziel and Lukasz Golunski (2022). 'Two-stage variable-fidelity modeling of antennas with domain confinement'. In: *Scientific Reports* 12.1, p. 17275.

Powell, Michael JD (1964). 'An efficient method for finding the minimum of a function of several variables without calculating derivatives'. In: *The computer journal* 7.2, pp. 155–162.

– (1992). 'The theory of radial basis function approximation in 1990'. In: *Advances in numerical analysis* 2, pp. 105–210.

– (2007). 'A view of algorithms for optimization without derivatives'. In: *Mathematics Today-Bulletin of the Institute of Mathematics and its Applications* 43.5, pp. 170–174.

Qin, Shufen et al. (2021). 'Multiple infill criterion-assisted hybrid evolutionary optimization for medium-dimensional computationally expensive problems'. In: *Complex & Intelligent Systems*, pp. 1–13.

Queipo, Nestor V et al. (2005). 'Surrogate-based analysis and optimization'. In: *Progress in aerospace sciences* 41.1, pp. 1–28.

Raidl, Günther R and Jens Gottlieb (2005). 'Empirical analysis of locality, heritability and heuristic bias in evolutionary algorithms: A case study for the multidimensional knapsack problem'. In: *Evolutionary computation* 13.4, pp. 441–475.

Rasmussen, Carl Edward and Christopher KI Williams (2006). *Gaussian processes for machine learning*. Vol. 1. Cambridge: MIT press.

Regis, Rommel G (2013). 'Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions'. In: *IEEE Transactions on Evolutionary Computation* 18.3, pp. 326–347.

Rios, Luis Miguel and Nikolaos V Sahinidis (2013). 'Derivative-free optimization: a review of algorithms and comparison of software implementations'. In: *Journal of Global Optimization* 56, pp. 1247–1293.

Rocca, Paolo, Giacomo Oliveri and Andrea Massa (2011). 'Differential evolution as applied to electromagnetics'. In: *IEEE Antennas and Propagation Magazine* 53.1, pp. 38–49.

Romain, Olivier et al. (2023). *Device for characterising the actimetry of a subject in real time*. US Patent App. 17/766,760.

Shah, Amar, Andrew G Wilson and Zoubin Ghahramani (2013). 'Bayesian optimization using Student-t Processes'. In: *NIPS Workshop on Bayesian Optimization*.

Shi, Yuhui and Russell Eberhart (1998). 'A modified particle swarm optimizer'. In: *1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)*. IEEE, pp. 69–73.

Simon, Dan (2013). *Evolutionary optimization algorithms*. John Wiley & Sons.

Singer, Saša and John Nelder (2009). 'Nelder-mead algorithm'. In: *Scholarpedia* 4.7, p. 2928.

Skaik, TF, MJ Lancaster and Frederick Huang (2011). 'Synthesis of multiple output coupled resonator circuits using coupling matrix optimisation'. In: *IET microwaves, antennas & propagation* 5.9, pp. 1081–1088.

Song, Yiran, Qingsha S Cheng and Slawomir Koziel (2019). 'Multi-fidelity local surrogate model for computationally efficient microwave component design optimization'. In: *Sensors* 19.13, p. 3023.

Spears, William M (1993). 'Crossover or mutation?' In: *Foundations of genetic algorithms*. Vol. 2. Elsevier, pp. 221–237.

Srivastava, Nitish et al. (2014). 'Dropout: a simple way to prevent neural networks from overfitting'. In: *The journal of machine learning research* 15.1, pp. 1929–1958.

Stein, M. (1987). 'Large sample properties of simulations using Latin hypercube sampling'. In: *Technometrics*, pp. 143–151.

Storn, Rainer and Kenneth Price (1997). 'Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces'. In: *Journal of global optimization* 11.4, pp. 341–359.

Storn, Rainer, Kenneth V Price and Jouni Lampinen (2005). *Differential Evolution-a practical approach to global optimization*.

Sun, Chaoli et al. (2015). 'A two-layer surrogate-assisted particle swarm optimization algorithm'. In: *Soft computing* 19, pp. 1461–1475.

Swidzinski, Jan F and Kai Chang (2000). 'Nonlinear statistical modeling and yield estimation technique for use in Monte Carlo simulations [microwave devices and ICs]'. In: *IEEE Transactions on Microwave Theory and Techniques* 48.12, pp. 2316–2324.

Tech, Red Cedar (2008). *SHERPA—An Efficient and Robust Optimization/Search Algorithm*.

Tentzeris, Emmanouil M et al. (1998). 'Application of the multiresolution time domain technique (MRTD) to microwave circuit and antennas geometries'. In: *1998 28th European Microwave Conference*. Vol. 2. IEEE, pp. 718–723.

Ulmer, Holger, Felix Streichert and Andreas Zell (2003). 'Evolution strategies assisted by Gaussian processes with improved preselection criterion'. In: *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.* Vol. 1. IEEE, pp. 692–699.

Ur-Rehman, Masood, Michael Adekanye and Hassan Tariq Chattha (2018). 'Tri-band millimetre-wave antenna for body-centric networks'. In: *Nano communication networks* 18, pp. 72–81.

Vikhar, Pradnya A (2016). 'Evolutionary algorithms: A critical review and its future prospects'. In: *2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC)*. IEEE, pp. 261–265.

Villiers, Dirk IL de and Slawomir M Koziel (2018). 'Fast multi-objective optimisation of pencil beam reflector antenna radiation pattern responses using Kriging'. In: *IET Microwaves, Antennas & Propagation* 12.1, pp. 120–126.

Wang, Handing et al. (2018). 'Offline data-driven evolutionary optimization using selective surrogate ensembles'. In: *IEEE Transactions on Evolutionary Computation* 23.2, pp. 203–216.

Wang, Weizhong, Hai-Lin Liu and Kay Chen Tan (2022). 'A surrogate-assisted differential evolution algorithm for high-dimensional expensive optimization problems'. In: *IEEE Transactions on Cybernetics*.

Weiland, Thomas, Martin Timm and Irina Munteanu (2008). 'A practical guide to 3-D simulation'. In: *IEEE microwave magazine* 9.6, pp. 62–75.

Wen, Ding-Liang, Dong-Ze Zheng and Qing-Xin Chu (2017). 'A wideband differentially fed dual-polarized antenna with stable radiation pattern for base stations'. In: *IEEE Transactions on Antennas and Propagation* 65.5, pp. 2248–2255.

Werner, Douglas et al. (2014). 'Optimization Methods in Antenna Engineering'. In: *Handbook of Antenna Technologies*. Ed. by Zhi Ning Chen. Singapore: Springer Singapore, pp. 1–47. ISBN: 978-981-4560-75-7. DOI: 10.1007/978-981-4560-75-7_15-1. URL: https://doi.org/10.1007/978-981-4560-75-7_15-1.

Westermann, Paul and Ralph Evins (2021). 'Using Bayesian deep learning approaches for uncertainty-aware building energy surrogate models'. In: *Energy and AI* 3, p. 100039.

Wolff, Edward A (1966). *Antenna analysis*. New York City: John Wiley & Sons.

Wu, Annie S, Robert K Lindsay and Rick L Riolo (1997). 'Empirical Observations on the Roles of Crossover and Mutation.' In: *ICGA*. Citeseer, pp. 362–369.

Wu, Qi, Haiming Wang and Wei Hong (2020). 'Multistage collaborative machine learning and its application to antenna modeling and optimization'. In: *IEEE Transactions on Antennas and Propagation* 68.5, pp. 3397–3409.

Wu, Qi et al. (2021). 'Multilayer machine learning-assisted optimization-based robust design and its applications to antennas and array'. In: *IEEE Transactions on Antennas and Propagation* 69.9, pp. 6052–6057.

– (2022). 'Knowledge-guided active-base-element modeling in machine-learning-assisted antenna-array design'. In: *IEEE Transactions on Antennas and Propagation* 71.2, pp. 1578–1589.

Wu, Yun, Ruiheng Wu and Yi Wang (2018). 'A compact coupling structure for diplexers and filtering power dividers'. In: *Progress In Electromagnetics Research M* 69, pp. 161–170.

Xue, Liyuan et al. (2022). 'An Unsupervised Microwave Filter Design Optimization Method Based on a Hybrid Surrogate Model-Assisted Evolutionary Algorithm'. In: *IEEE Transactions on Microwave Theory and Techniques*.

Yang, Shufan et al. (2023). 'The human activity radar challenge: benchmarking based on the 'radar signatures of human activities' dataset from Glasgow University'. In: *IEEE Journal of Biomedical and Health Informatics* 27.4, pp. 1813–1824.

Yang, Zan et al. (2019). 'Two-layer adaptive surrogate-assisted evolutionary algorithm for high-dimensional computationally expensive problems'. In: *Journal of Global Optimization* 74, pp. 327–359.

Yeboah-Akowuah, Bright, Panagiotis Kosmas and Yifan Chen (2017). 'A Q-slot monopole for UWB body-centric wireless communications'. In: *IEEE Transactions on Antennas and Propagation* 65.10, pp. 5069–5075.

Yu, Yang et al. (2020). 'A general coupling matrix synthesis method for all-resonator diplexers and multiplexers'. In: *IEEE Transactions on Microwave Theory and Techniques* 68.3, pp. 987–999.

Yuan, Ya-xiang (2000). 'A review of trust region algorithms for optimization'. In: *Iciam*. Vol. 99. 1, pp. 271–282.

Zaharis, Zaharias D and Traianos V Yioultsis (2011). 'A novel adaptive beamforming technique applied on linear antenna arrays using adaptive mutated boolean PSO'. In: *Progress In Electromagnetics Research* 117, pp. 165–179.

Zaharis, Zaharias D et al. (2017). 'Exponential log-periodic antenna design using improved particle swarm optimization with velocity mutation'. In: *IEEE transactions on magnetics* 53.6, pp. 1–4.

Zhan, Dawei and Huanlai Xing (2020). 'Expected improvement for expensive optimization: a review'. In: *Journal of Global Optimization* 78.3, pp. 507–544.

– (2021). 'A fast Kriging-assisted evolutionary algorithm based on incremental learning'. In: *IEEE Transactions on Evolutionary Computation* 25.5, pp. 941–955.

Zhang, Renyuan and Siyang Cao (2018). 'Real-time human motion behavior detection via CNN using mmWave radar'. In: *IEEE Sensors Letters* 3.2, pp. 1–4.

Zhang, Zhen et al. (2022). 'A microwave filter yield optimization method based on off-line surrogate model-assisted evolutionary algorithm'. In: *IEEE Transactions on Microwave Theory and Techniques* 70.6, pp. 2925–2934.

Zhong, Yang et al. (2022). 'A machine learning generative method for automating antenna design and optimization'. In: *IEEE Journal on Multiscale and Multiphysics Computational Techniques* 7, pp. 285–295.

Zhou, Jinzhu et al. (2020). 'A trust-region parallel bayesian optimization method for simulation-driven antenna design'. In: *IEEE Transactions on Antennas and Propagation* 69.7, pp. 3966–3981.

Zhou, Zhi-Hua (2012). *Ensemble methods: foundations and algorithms*. Boca Raton: CRC press.

Zhou, Zongwei et al. (2018). 'Unet++: A nested u-net architecture for medical image segmentation'. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, pp. 3–11.

Zhou, Zongzhao et al. (2006). 'Combining global and local surrogate models to accelerate evolutionary optimization'. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37.1, pp. 66–76.