THE UNIVERSITY of ADELAIDE

The Application of Nature-inspired Metaheuristic Methods for Optimising Renewable Energy Problems and the Design of Water Distribution Networks

by

Mehdi Neshat

A thesis submitted in total fulfillment for the
degree of Doctor of Philosophy

in the
School of Computer Science
Faculty of Engineering, Computer, and Mathematical Sciences
The University of Adelaide

September 2020

Thesis supervisors:
Dr. Bradley Alexander
Dr. Markus Wagner

The University of Adelaide

Abstract

School of Computer Science
Faculty of Engineering, Computer, and Mathematical Sciences

Doctor of Philosophy

by Mehdi Neshat

This work explores the technical challenges that emerge when applying bio-inspired optimisation methods to real-world engineering problems. A number of new heuristic algorithms were proposed and tested to deal with these challenges. The work is divided into three main dimensions:

i) One of the most significant industrial optimisation problems is optimising renewable energy systems. Ocean wave energy is a promising technology for helping to meet future growth in global energy demand. However, the current technologies of wave energy converters (WECs) are not fully developed because of technical engineering and design challenges. This work proposes new hybrid heuristics consisting of cooperative coevolutionary frameworks and neuro-surrogate optimisation methods for optimising WECs problem in three domains, including position, control parameters, and geometric parameters. Our problem-specific algorithms perform better than existing approaches in terms of higher quality results and the speed of convergence.

ii) The second part applies search methods to the optimization of energy output in wind farms. Wind energy has key advantages in terms of technological maturity, cost, and life-cycle greenhouse gas emissions. However, designing an accurate local wind speed and power prediction is challenging. We propose two models for wind speed and power forecasting for two wind farms located in Sweden and the Baltic Sea by a combination of recurrent neural networks and evolutionary search algorithms. The proposed models are superior to other applied machine learning methods.

iii) Finally, we investigate the design of water distribution systems (WDS) as another challenging real-world optimisation problem. WDS optimisation is demanding because it has a high-dimensional discrete search space and complex constraints. A hybrid evolutionary algorithm is suggested for minimising the cost of various water distribution networks and for speeding up the convergence rate of search.

Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I acknowledge that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Signed: *Mehdi Neshat*

Date: September 2020

Statement of Authorship

This thesis is based on ten articles consists of seven conference papers and three journal papers. Four conference papers have already been published, and other three conference papers are accepted for publication. And also, two journal papers are accepted, and one of them is submitted and under review. I have afforded a statement of authorship for each of these papers to certify that I was actively involved in the process of preparing and publishing each article.

- [1] Neshat, M., Alexander, B., Wagner, M., & Xia, Y. (2018, July). A detailed comparison of meta-heuristic methods for optimising wave energy converter placements. Published by Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1318-1325) on 2018.

- [2] Neshat, M., Alexander, B., Sergiienko, N., & Wagner, M. (2020). New insights into position optimization of wave energy converters by a hybrid local search. Published by Swarm and Evolutionary Computation–journal on [26 July 2020].

- [3] Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2019, July). A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters. Published by Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1293-1301).(***received The Best Paper Award in the Real World Application (RWA) Track*** [1] ***of GECCO2019***)

- [4] Neshat, M., Abbasnejad, E., Shi, Q., Alexander, B., & Wagner, M. (2019, December). Adaptive Neuro-Surrogate-Based Optimisation Method for Wave Energy Converters Placement Optimisation. Published by International Conference on Neural Information Processing (pp. 353-366). Springer, Cham on 2019.

- [5] Neshat, M., Alexander, B., & Wagner, M. (2020). A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters. published by Information Sciences–journal on [16 May 2020]

[1] https://gecco-2019.sigevo.org/index.html/Best+Paper+Awards

- [6] Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2020). Optimisation of Large Wave Farms using a Multi-strategy Evolutionary Framework. Published by the Genetic and Evolutionary Computation Conference on 2020. (***received The Best Paper Award in the Real World Application (RWA) Track*** [2] ***of GECCO2020***)

- [7] Sergiienko, N. Y., Neshat, M., da Silva, L. S., Alexander, B., & Wagner, M. (2020). Design optimisation of a multi-mode wave energy converter. Accepted for publication by the 39th International Conference on Ocean, Offshore and Arctic Engineering ASME 2020 on [3 March 2020].

- [8] Neshat, M., Alexander, B., & Simpson, A. (2019). Covariance Matrix Adaptation Greedy Search Applied to Water Distribution System Optimization. arXiv preprint arXiv:1909.04846. Submitted for publication to Engineering Applications of Artificial Intelligence – journal.on [5 April 2020].

- [9] Neshat, M., Nezhad, M. M., Abbasnejad, E., Tjernberg, L. B., Garcia, D. A., Alexander, B., & Wagner, M. (2020). An Evolutionary Deep Learning Method for Short-term Wind Speed Prediction: A Case Study of the Lillgrund Offshore Wind Farm. arXiv preprint arXiv:2002.09106. Accepted for publication by The 1st Asia Pacific Conference on Sustainable Development of Energy, Water and Environment Systems – SDEWES Conference.

- [10] Neshat, M., Nezhad, M.M., Abbasnejad, E., Groppi, D., Heydari, A., Tjernberg, L.B., Garcia, D.A., Alexander, B. and Wagner, M., 2020. Hybrid Neuro-Evolutionary Method for Predicting Wind Turbine Power Output. Accepted for publication by 4th SEE Conference on Sustainable Development of Energy, Water and Environment Systems – SDEWES Conference.

[2] https://gecco-2020.sigevo.org/index.html/Best+Paper+Awards

Acknowledgements

I wish to thank all the people who supported me through the various stages of this PhD.

Foremost, I wish to appreciate my supervisors, Dr. Bradley Alexander and Dr. Markus Wagner, their supports, guidance and encouragements were a milestone in the completion of this PhD. Without their persistent help, the goal of this chapter of my life would not have been achieved.

I would like to express my special gratitude and thanks to Professor Angus Simpson from the School of Civil, Environmental and Mining Engineering for imparting his valuable experience, knowledge and expertise in this study.

My thanks and appreciations also go to Dr. Nataliia Sergiienko and Dr. Ehsan Abbasnejad for contributing and sharing their technical knowledge in making this success possible.

I would like to extend my special thanks to Professor Frank Neumann and all my friends in the optimisation and logistics research group who cooperated with me in this regard.

I also would like to express my deep gratitude to my family, for their valuable supports, and encouragements.

This research is supported by the supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Real-world problems in engineering domains are generally categorised as challenging problems of optimisation, as the majority of them are large-scale, nonlinear, multi-modal, constrained and computationally expensive. Where some of these characteristics emerge together in a real engineering problem, we can see the low efficiency of the traditional numerical methods and existing global optimisation approaches. In this thesis, we study these challenges of technical optimisation problems through several case studies belonging to three different domains. Working with multiple real-world engineering problems in various domains enables us to understand better the practical challenges in employing bio-inspired optimisation methods in an industrial setting and proposing appropriate algorithms with high efficiency. This thesis contains three parts, which we will describe in detail throughout the rest of this chapter.

- Part I: Wave farm Power optimisation, we propose several fast and efficient optimisers to improve the produced total power output of a wave farm with different sizes in various sea sites.
- Part II: Wind farm Power Forecasting, we design two accurate forecasting models to model the wind speed and power output of wind turbines in a wind farm through a combination of evolutionary algorithms and deep neural networks.
- Part III: Water distribution network design optimisation, we develop a new hybrid evolutionary framework that consists of three distinct phases for optimising the water distribution networks with various dimensions.

1.1 Part I: Wave Farm Power Optimisation

The development of renewable energy technologies has increased remarkably throughout many countries in response to the future global energy demand. Among currently available renewable sources, wave energy is one of the most promising forms of renewable energy due to, the high energy density of wave environments and source's minimal environmental impact [18]. However, wave energy technologies are not fully-developed in terms of technical and economical issues, and no device has reached the expected level of reliability for full-scale commercialization [19].

A Wave Energy Converter (WEC) is a device, commonly in the form of a *buoy*, which captures ocean wave energy and then converts it to electricity. One of the popular options for improving the levelized cost of energy (LCoE) of WEC technology is to expand the number of WECs in a wave farm. The use of a farm of converters (rather than one big device) presents the following advantages [19, 20]: it allows for the sharing of mooring points and electrical cables connections, increases power production, allows maintenance to be performed without the need to shut down the whole power output, and facilitates cost-effective deployment. The amount of power generated by a farm or an array of WECs depends on several factors including the number of WECs, their arrangement relative each other, the power take-off (PTO) configurations on each buoy's tethers, geometric parameters (buoy shapes and dimensions) and wave climate [21, 22]. To illustrate the complexity of WECs optimisation problem [19], Figure 1.1 shows a simplified landscape of the most significant WECs system variables. Arrows indicate relationships and the impact of factors on the produced power output. The figure shows that there are complex mutual relations among variables. In this thesis, we focus on three dimensions of the WEC optimisation problem: placement of the converters (layout design), adjusting the PTO parameters (control optimisation) and geometric settings (shape and size of WEC).

Interactions between buoys in a wave farm are very complex to model, and the evaluation of each layout is computationally expensive, sometimes taking several minutes or even hours in a large wave farm. This complexity increases quadratically in proportion to increases in the number of converters [23]. Furthermore, the high number of decision variables in large farms expands in proportion to the size of the search space. These features lead to a very challenging environment for optimisation. In terms of WECs position optimisation, The most pertinent current research question is how to arrange WECs in a size-constrained environment with respect to safety distance constraints, in order to maximise the constructive hydrodynamic interactions and minimise destructive interactions of buoys.

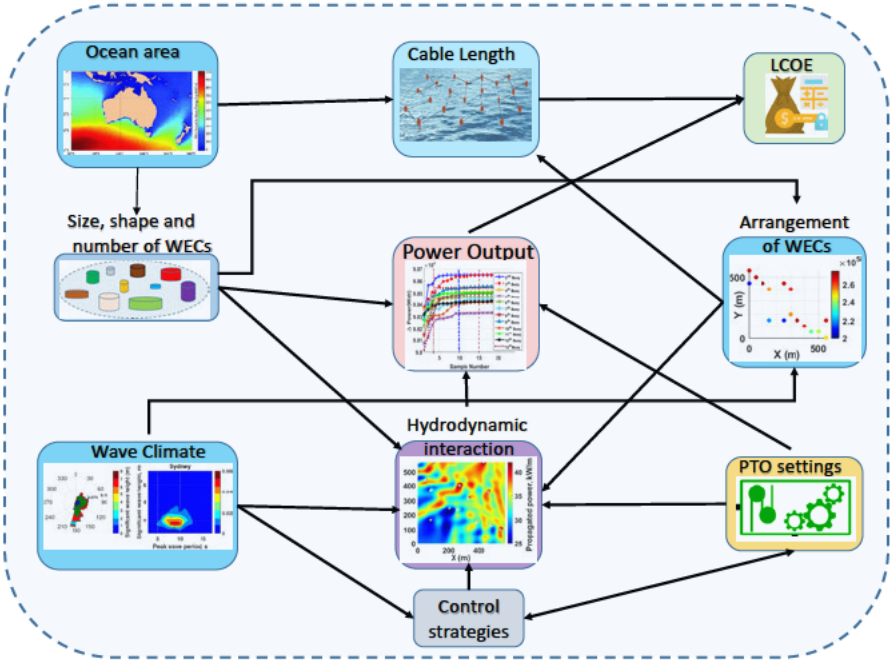

FIGURE 1.1: Several important factors are involved to optimise a wave energy farm. Arrows depict the mutual influence to produce total power output.

In order to maximise the produced total power output of a wave farm, an efficient control system plays an indispensable role. The power take-off (PTO) machine is one example of such a control system. In any WEC design, a control system is an integral component. This system is in charge of providing the best-tuned situation in order to obtain the desired motion for each converter with regard to the incident wave-front. There is a direct relationship between the size, shape and efficiency of a WEC and its control strategy [24]. Recent studies [25–27] represent a developed control mechanism is able to improve the absorbed power of WECs considerably, with direct effects on the capital cost of a WEC by an estimated 20-30% of the total cost [28]. The most important parameters of a typical PTO system are spring and damper coefficients. The spring force can tune the buoy resonance frequency based on the frequency of the incident waves. In addition, the amplitude of the WEC motion should be adjusted by selecting a proper coefficient of damping. If the damping coefficient is set too high, this will lead to an overly high number of motions with regards to the incident wave, as well as a low level of absorbed power. If the damping coefficient is set too low, the motion range of the oscillator will be restricted, causing a low level of power to be extracted [28]. Consequently, fitting damping on the PTO system is crucial.

The third aspect of WEC optimisation involves optimising geometry parameters (shape optimisation) which is considered to be one of the most effective factors in increasing the amount of energy harnessed from waves [29]. Spherical and cylindrical [30] shapes are very popular among engineers because they have easy design and control. The WEC

geometry [26] is generally planned for the prevailing sea conditions in a given location and a control technique is subsequently required to develop the performance of the device for sea states other than the design sea state.

There has been considerable research into the WEC optimisation problem. Various bio-inspired optimisation methods have been proposed to deal with the associated challenges, like Genetic Algorithm (GA) [31–34], improved GA [35, 36], hybrid GA (GA + multiple analytical scattering (MAS) method) [37], hidden genes genetic algorithm (HGGA) [38], CMA-ES [35, 39, 40], Differential Evolution (DE) [41] and 1+1EA [39], glow-worm swarm optimisation (GSO) [35], particle swarm optimisation (PSO) [42, 43], surrogate-model based optimisation method [40]. However, these earlier studies exhibited the following deficiencies:

1. an overly simple environmental wave model with just one or a few wave directions and wave frequencies,
2. most of the studies optimised a wave farm of small size,
3. application of a relatively large number of layout evaluations (costly and inefficient optimisation techniques) in order to optimise the WEC problem.

Our Contributions

In Part I of this thesis, we offer significant contributions to optimise various aspects of wave energy converters problem. Part I consists of three chapters addressing the layout design, PTO settings and geometric parameters optimisation. The main contributions of this Part are summarised as follows:

- In Chapter 2
    1. In Section 2.1, we propose a new heuristic approach to optimising the WEC arrangement, which is a hybrid search consisting of stochastic local search combined with downhill search. This approach is an improvement on previously published methods in terms of performance and convergence speed for a reasonably large wave farm (16-buoy). Furthermore, we deploy a more realistic and practical wave model compared with the previous works with 50 wave frequencies and seven different wave directions. Another most important contribution is that it explores the use of surrogate functions in a partial evaluation [44] framework. The experiments with the partial evaluation method

show that it can be a suitable alternative method for solving expensive optimisation problems. Nevertheless, this method requires more investigations. The outcomes of this research are published in [1].

2. In Section 2.2, we extend on our prior work [1] substantially. We begin by expanding the findings of [1] to include nine new heuristic search methods, including a novel surrogate-based search model. This model consists of a learned model-based local search interleaved with numerical optimisation, all applied to the original irregular wave model. Furthermore, it includes four new real wave regimes from the southern coast of Australia (Adelaide, Perth, Tasmania and Sydney) using a higher granularity of wave-directions than the previous studies [1, 32, 39]. A comprehensive and systematic comparison is made to assess the performance of the new hybrid heuristic method, and it can outperform previous optimisation methods in terms of both convergence rate and higher total power output for 16-buoy layouts. These findings are published in [23].

3. In Section 2.3, we design an adaptive neuro-surrogate optimisation framework that is composed of a surrogate Recurrent Neural Network (RNN) model. This model is trained with a very limited number of layout observations and estimates the power output of the new arrangement. It is also a fast meta-heuristic optimiser which can adjust the model's hyper-parameters in order to deal with the expensive WECs position optimisation problem. Experimental results confirm that the adaptive neuro model is a fast and effective optimisation technique in comparison to previous optimisation methods. The best-found wave farm layout designs and other comparative results have been published in [4].

4. In Section 2.4, we extend previous WECs position optimisation work by using a new hybrid multi-strategy evolutionary framework. This framework combines smart initialisation, a binary population-based evolutionary algorithm, discrete local search and continuous global optimisation. A more detailed energy model is also applied to place buoys in large farms of up to 100 WECs. We compare the performance of our proposed method with a wide variety of state-of-the-art optimisation approaches, including six continuous evolutionary algorithms, four discrete search techniques and three hybrid optimisation methods. Results indicate that this approach significantly outperforms the applied meta-heuristic methods in this work. The details of our approach and its optimisation results have been published in [6].

- In Chapter 3

    1. In Section 3.1, we propose a new hybrid heuristic approach. This approach entails combining a symmetric local search with a Nelder-Mead Simplex direct search, in combination with a back-tracking optimisation strategy. This strategy serves to optimise the position and PTO parameters of a wave farm simultaneously. Furthermore, we explore a comprehensive range of modern optimisation techniques including alternative, cooperative, hybrid and well-known off-the-shelf evolutionary algorithms for evaluating and comparing their performance for optimising this expensive and complex problem. Finally, we show that our hybrid method is able to outperform previously defined search techniques by up to 3% [3].
    2. In Section 3.2, the wave energy converter model is extended using additional details relating to the power take-off settings for each wave frequency. This extension entails a large-scale optimisation problem. Accordingly, we extend the prior our work [3] by introducing a new hybrid cooperative co-evolution algorithm that consists of a symmetric local search plus Nelder-Mead. We also use a cooperative co-evolution algorithm (CC) with a backtracking strategy in order to optimise the positions and PTO settings of WECs, respectively. The experimental study reveals that the hybrid cooperative framework performs best in terms of both runtime and quality of acquired solutions. This approach and all associated experimental results have been published in [45].

- In Chapter 4

    1. In Section 4.1, we present an efficient hybrid approach for the optimisation of geometric parameters and PTO configurations. This approach is intended to maximise the total harnessed power output and minimise the levelised cost of energy (LCoE) of a cylinder-shape WEC. The model of WEC applied in this paper is the CETO 6 technology [46] being under development by Carnegie Clean Energy Limited in Australia. Six different optimisation methods are applied in order to assess and compare the performance of the best-known evolutionary algorithms. The optimisation results represent that our approach performs better than other evaluated evolutionary algorithms [7].

1.2 Part II: Wind farm Power Forecasting

Recently, there has been an acceleration in the extension of power by industries, alongside increasing global demands for energy and restrictions in the reserves of traditional

energy resources. Within this setting, wind power (an economically and environmentally friendly source of renewable energy) has expanded rapidly and attracted much attention [47]. However, because of intermittent and stochastic wind speed characteristics, forecasting wind power is a challenging process. To design a stable and appropriate integrated wind power management system, we need an accurate wind forecasting model. Moreover, the non-stationary attribute of the wind speed can significantly affect the safety and stability of the energy systems [48, 49], wind farm management, inadequate power factor, voltage instability and imbalance influence on the performance of electrical devices. Consequently, improving wind power forecasting models is of numerous interest in power conversion and management fields.

Recently, several studies [50–52] have strongly recommended applying non-parametric models like artificial neural networks (ANN) in order to form accurate predictions and reliable forecasting models. They argue that such models are superior to physical models considerably. Some of the well-known time series forecasting methods for wind power prediction include the autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), fractional version of ARIMA (f-ARIMA) [53], recurrent neural networks (RNNs) [54], Long short-term memory networks (LSTMs) [55], A combination of principal component analysis (PCA) and LSTM (PCA-LSTM) [56], hybrid LSTM and Echo State Network (ESN-LSTM) [57] and LSTM with an enhanced forget-gate network model (LSTM-EFG) [58].

However, the main challenge of applying neural network (NN) and deep NN model to the forecasting of wind power is in tuning the hyper-parameters. Designing an automatic tuning model, involves porting the models to a new setting and makes it possible to compare modelling approaches more rigorously. Some recent works that have applied bio-inspired optimisation method for tuning the neural models hyper-parameters are including the application of Cuckoo Search Optimization (CSO) method to improve performance of a Back Propagation Neural Network (BPNN) by adjusting the connection weights [59], the dragonfly algorithm (DA) to tune RNN hyper-parameters [60], Differential Evolution (DE), Genetic Algorithm and Grey Wolf Optimiser (GWO) to optimise LSTM parameters [4, 61, 62].

Our contribution

In Part II of this thesis, we demonstrate two accurate wind speed and power forecasting frameworks using Deep neural networks and evolutionary algorithms. The principal contributions of this Part are summarised as follows:

1. In Chapter 5, we propose a hybrid evolutionary deep forecasting model combining a recurrent deep learning model (LSTM network), coupled with the CMA-ES algorithm, (called CMAES-LSTM) for predicting the short-term wind speed with high accuracy. Moreover, we investigate a landscape analysis of the hyper-parameters in order to represent the impact of each LSTM hyper-parameters on the forecasting results. Details of this forecasting model and obtained results are published in [9].

2. In Chapter 6, we design a composite Machine Learning approach–namely a hybrid neuro-evolutionary algorithm–for precise forecasting of the power output in wind-turbine farms. We utilise historical data in the supervisory control and data acquisition (SCADA) systems as input to predict the power output from an onshore wind farm in Sweden. At the first stage, to detect and filter noise in the SCADA measurements, a k-means clustering method and an autoencoder are used respectively. Furthermore, we propose the use of a self-adaptive differential evolution (SaDE) algorithm as a hyper-parameter optimiser, as well as a recurrent neural network (RNN) called Long Short-term memory (LSTM) to model the power curve of a wind turbine in a wind farm. The components and performance of this model cab be seen in the published paper [10].

1.3 Part III: Water Distribution Network Design Optimisation

Water distribution networks present an interesting real-world optimisation problem. Water distribution networks are a type of a hydraulic infrastructure that conveys water from reservoirs to consumers, and are composed of various elements such as pipes, valves, pumps, tanks and reservoirs. Water distribution systems (WDSs) are indispensable to the infrastructure of a modern city to satisfy of the water consumption demands posed by households, agriculture and industry. WDS development is a vital part of dealing with population expansion. In the majority of cases, the capital cost of WDS expansion is very high. Furthermore, WDS design presents a challenging optimisation problem with a high number of search dimensions and constraints. With this in mind, a considerable number of optimisation techniques have been studied with the goal of minimising the cost of the designed/extended water distribution systems, including traditional methods, like linear programming and non-linear programming [63, 64], and evolutionary algorithms (EAs), such as genetic algorithms [65–68], simulated annealing [69, 70], tabu search [71], harmony search [72, 73], the shuffled frog leaping algorithm [74] , particle swarm optimisation [75, 76], ant colony optimisation [77–81], memetic algorithm [82] and differential evolution [83–85]. The principal benefit of applying bio-inspired optimisation methods

over the traditional methods is that such optimisation methods can handle the specific characteristics of the WDS design problems. These characteristics include non-linearity of relationships between individual pipe sizes and pressure at nodes, multi-modality with many local minima in the cost-function, large dimensions and discrete search space [86]. However, in spite of the promising performance of bio-inspired methods, they are often time-consuming and require a large number of evaluations.

Our contribution

In Part III of this thesis, we propose the contribution below in order to deal with the complex challenges of WDS optimisation problems:

- In Chapter 7, we deploy a Covariance Matrix Adaptation Evolution Strategy (CMA-ES) in pure and hybrid form to various ranges of small- and large-scale WDS benchmark networks, as well as performing a systematic comparison with previous results. The hybrid search methods combine CMA-ES with novel greedy search heuristics in order to make up the violation of solutions found by CMA-ES. We show that CMA-ES, when combined with a greedy search, performs comparably to the current best search heuristics in terms of runtime and network costs. Furthermore, this is the first research to explore, as a baseline, the performance of simple randomised local search (RLS) and a 1+1 Evolutionary Algorithm (1+1 EA). The experimental results and best-found WDS designs are published in [8].

Part I

Wave Farm Power Optimisation

Chapter 2

Position Optimisation of Wave Energy Converters (WECs)

2.1 A detailed comparison of meta-heuristic methods for optimising wave energy converter placements

2.1.1 Synopsis

The main focus of the article in this section is on position optimisation of Wave Energy Converters (WECs) in a more realistic and practical wave model with 50 wave frequencies and seven different wave directions. The hydrodynamic interactions between WECs in a farm are complex, extensive, and dependent on local conditions. Furthermore, there is no straightforward recipe for WEC arrangement. In this section, we propose a new, fast, effective heuristic consisting of a stochastic local search combined with Nelder-Mead Simplex direct search intended to optimise the placement of converters. In contrast to the other optimisation algorithms, which take an 'all-at-once' approach, this algorithm places and optimises the WECs one-at-a-time. The algorithm does this using a three-sample local search for each buoy placement, followed by a Nelder-Mead search; and makes a sequential array of converters. The experimental results show that the proposed heuristic outperforms previous optimisation methods in terms of efficiency and convergence speed.

Reference

[1] Neshat, M., Alexander, B., Wagner, M., & Xia, Y. (2018, July). A detailed comparison of meta-heuristic methods for optimising wave energy converter placements. In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1318-1325).

Statement of Authorship

Title of Paper	A detailed comparison of meta-heuristic methods for optimising wave energy converter placements
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Alexander, B., Wagner, M., \& Xia, Y. (2018, July). A detailed comparison of meta-heuristic methods for optimising wave energy converter placements. In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1318-1325).

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	70 %		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	05/15/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander (13%)		
Contribution to the Paper	Advised throughout on methodology, experimentation and reporting of results. Revised paper.		
Signature		Date	12/05/20

Name of Co-Author	Markus Wagner (13%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Name of Co-Author	Yuanzhong Xia (4%)		
Contribution to the Paper	Analysed the landscape and plotted a figure.		
Signature		Date	20/05/2020

Please cut and paste additional co-author panels here as required.

2.1.2 Abstract

In order to address environmental concerns and meet growing energy demand the development of green energy technology has expanded tremendously. One of the most promising types of renewable energy is ocean wave energy. While there has been strong research in the development of this technology to date there remain a number of technical hurdles to overcome. This research explores a type of wave energy converter (WEC) called a buoy. This work models a power station as an array of fully submerged three-tether buoys. The target problem of this work is to place buoys in a size-constrained environment to maximise power output. This article improves prior work by using a more detailed model and exploring the search space using a wide variety of search heuristics. We show that a hybrid method of stochastic local search combined with Nelder-Mead Simplex direct search performs better than previous search techniques.

2.1.3 Introduction

Wave Energy Converters (WECs) are of interest to governments and industry as a means of complementing other renewable energy sources such as solar and wind-power. WECs have advantages in terms of high availability of resource (over 90%, depending on the location) [18] and wave energy densities of up to 60kW per square meter of water surface in prime locations. Individual WECs in the form of buoys can also be produced to have a high capacity for each unit with current proposals for units with over 1MW each [46] – providing potential for economies of scale. Finally, WEC's have a low impact on aquatic life [87], comparing favourably with other generation technologies.

This study focuses on WECs in the form of fully-submerged buoys. Submerged buoys are one of the most promising and cost-effective technologies for extraction of energy from waves [39]. The buoys in this study are hollow metallic vessels, floating a few meters below the water surface and tethered to the sea floor. Energy is extracted from changes in tension on the tethers as waves propagate through water. Buoys are usually deployed in farms or arrays consisting of multiple buoys. This is done for the reason of amortizing fixed infrastructure cost but also to take advantage of constructive interference between buoys [21]. To maximise the energy returned by a WEC farm buoys must be placed to exploit prevailing wave conditions, maximise constructive interference between buoys, and minimise destructive interference.

The interactions between buoys in a farm are complex, extensive, and dependent on local conditions. As a consequence there is, as yet, no simple recipe for buoy placement. Research to date on farm design has primarily focused on the placement of semi-submerged

TABLE 2.1: Key parameters for WECs simulated in this work

Buoy number	4, 16
Buoy radius	5m
Submergence depth	3m
Water depth	30m
Buoy mass	376 tonnes
Buoy volume	$523.60m^3$
Tether angle	55°

arrays [31]. Research on placement of fully-submerged arrays [39] has applied two popular evolutionary algorithms, the (1+1)EA [88] and CMA-ES [89]. This found that a (1+1)EA with simple mutation performed better than CMA-ES. However, this earlier work used a greatly simplified environmental model with just one wave direction and few wave frequencies. The current paper improves on prior work substantially in the following ways: deploying a more realistic and practical model with 50 wave frequencies and seven different wave directions; comparing a much broader range of heuristic search techniques adapted to functioning with a small number of function evaluations; exploring the use of surrogate functions in a partial evaluation framework [44]; and conducting a preliminary investigation of the local landscape for buoy placement. As a fair means of comparison, we examine how various frameworks perform within the context of a limited (but realistic) computational budget. Through this comparison we show that a hybrid search consisting of stochastic local search combined with downhill search outperforms previously published methods in terms of performance for 16-buoy array layouts. We also describe layouts resulting from these runs.

The remaining sections of the paper are organised as follows. In the next section we describe the buoy model. The optimisation problem is defined in Section 2.1.5 and the search methods to be compared are briefly described in Section 2.1.4.2. Section 2.1.6 presents experimental results and finally, Section 2.1.7 discusses these results and canvases future work.

2.1.4 Model for wave energy converters (WECs)

This research considers a model for a WEC consisting of a fully submerged three-tether buoy. Each tether is anchored to a generator placed on the sea floor. The anchors are assumed to be placed in a triangular pattern below each buoy in a configuration that optimises the transmission of energy from heave and surge wave motions in the waves, through to the generators [90]. Table 2.1 gives relevant details of the WECs modelled in this work.

2.1.4.1 System dynamics and parameters

The WEC model calculates the energy output of WEC based on a formula of dynamics [91] with three principle force components:

1. The force of wave excitation ($F_{exc,p}(t)$) incorporates the forces of incident and diffracted waves when all converters are in a fixed position.
2. The force of radiation($F_{rad,p}(t)$) describes the force of an oscillating body independent of incident waves.
3. Power take off force($F_{pto,p}(t)$) is the force applied to the buoys through their tethers.

Because oscillating buoys exert a force on the surrounding water they can interact with each other at distance. Buoys can interact not only destructively but also constructively, depending on their relative angles and distances, and depending on the surrounding sea conditions. In a buoy array the power accruing to a buoy number p is characterised by Equation 2.1.

$$M_p\ddot{X}_p(t) = F_{exc,p}(t) + F_{rad,p}(t) + F_{pto,p}(t) \tag{2.1}$$

where M_p is the displacement of the p_{th} buoy, $\ddot{X}_p(t)$ is a vector of body acceleration in heave, sway and surge. The final term, describing the power take-off system, is simulated as a linear damper and spring. For each mooring line two control factors are applied: the coefficient of damping B_{pto} and stiffness K_{pto}. Thus the extended version of Equation (2.1) for all converters is:

$$((M_\Sigma + A_\sigma(\omega))j\omega + B_\sigma(\omega) - \frac{K_{pto,\Sigma}}{\omega}j + B_{pto,\Sigma})\ddot{X}_\Sigma = \hat{F}_{exc,\Sigma} \tag{2.2}$$

where $A_\Sigma(\omega))$ and $B_\Sigma(\omega)$ are hydrodynamic parameters which are derived from the semi-analytical model based on [92]. In addition,$K_{pto,\Sigma}$ and $B_{pto,\Sigma}$ are control coefficients which are tuned to provide the maximum level of isolated buoy power absorption.
In the following, two performance measures are described. To compute the total power output of the layout, we utilise Equation (2.3):

$$P_\Sigma = \frac{1}{4}(\hat{F}^*_{exc,\Sigma}\ddot{X}_\Sigma + \ddot{X}^*{}_\Sigma\hat{F}_{exc,\Sigma}) - \frac{1}{2}\ddot{X}^*{}_\Sigma B\ddot{X}^*{}_\Sigma \tag{2.3}$$

The second important performance measure used here is the the q-factor (q) of the array. q measures the efficiency of an entire array of N as compared power output from each buoy taken in isolation. q is defined in Equation (2.4) as:

$$q = \frac{P_{\sum}}{N \cdot P_0} \tag{2.4}$$

In favorable circumstances $q > 1$ due to constructive interference, even though the buoys extract energy from the waves. In this work we aim to maximise the total power output: P_{Σ} of an array of a given size N within a constrained farm area. Because each buoy in the array is identical the corresponding q-factor is easily derived from the total output.

2.1.4.2 Optimisation Setup

The optimisation problem here can be stated as:

$$P^*_{\Sigma} = argmax_{\mathbf{x},\mathbf{y}} P_{\Sigma}(\mathbf{x}, \mathbf{y})$$

where $P_{\Sigma}(\mathbf{x}, \mathbf{y})$ is the average power obtained by placements of the buoys in a field at x-positions: $\mathbf{x} = [x_1, \ldots, x_N]$ and corresponding y positions: $\mathbf{y} = [y_1, \ldots, y_N]$. In the experiments here $N = 16$.

Constraints All buoy positions (x_i, y_i) are constrained to a square field of dimensions: $l \times w$ where $l = w = \sqrt{N * 20000}\, m$. This gives $20000m^2$ of farm-area per-buoy. In addition buoys are required to maintain a safety distance of at least 50 metres from each other. For any layout $\mathbf{x}, \mathbf{y}$ the sum-total of the inter-buoy distance violations, measured in metres, is:

$$Sum_{dist} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (dist((x_i, y_i), (x_j, y_j)) - 50),$$
$$\text{if } dist((x_i, y_i), (x_j, y_j)) < 50 \text{ else } 0$$

where $dist((x_i, y_i), (x_j, y_j))$ is the L2 (Euclidean) distance between buoys i and j. The penalty applied to the power output (in Watts) is $(Sum_{dist} + 1)^{20}$. This penalty is steep but continuous which allows better handling of constraint violations during search.

Buoy placements which are outside of the farm area are handled by repeating the placement process.

2.1.4.3 Computational Resources

This study aims to compare a diverse set of search methods in a realistic buoy-layout optimisation setting. The setting here assumes a limited computational time budget of three days on a moderately high performance shared-memory parallel platform. In this

study the hardware platform are compute nodes with 2.4GHz Intel 6148 processors and with 128GB of RAM. In terms of software, the meta-heuristic frameworks as well as the evaluative function for $P_{\Sigma}(\mathbf{x}, \mathbf{y})$ were run in MATLAB R2017. The used Matlab license allows us to run 12 worker threads in parallel.

For each heuristic search method, we exploit parallel processing by either evaluating individual layouts in a population in parallel or by evaluating all wave frequencies in parallel. The dimension of parallelism chosen was determined according to which gave the best performance for each search method. In both cases, if there are enough frequencies or individuals to make use of the parallel worker threads, up to ten-fold speedups were achieved.

It should be emphasised that we are not comparing search methods for buoy placement by simply counting evaluations of $P_{\Sigma}(\mathbf{x}, \mathbf{y})$. This is because the computational cost of each evaluation varies greatly between search methods depending on the number of frequencies considered; similarly, run times vary with the number of buoys in the layout. We ran the experiments on dedicated compute resources to minimise the variance of the number of evaluations between runs of the same method and thus avoid bias due to noise or resource contention. In these experiments the standard deviation in the number of evaluations between trials of the same search method is less than 5%. While this deviation might seem substantial at first, we shall later see that the algorithms either tend to converge well before the computation budget is used up, or their performance variance is not significant.

The computation budget for each single optimisation run is three days (72 hours) using 12 worker threads. In practice, this can give engineers two rounds of what-if analyses per week.

2.1.5 Meta-Heuristic Search Methods

We list the search methods compared in this study in Table 2.1.4.2. All methods are run with the computational resources described in the previous section and each method is run for ten times, with the best output produced by each framework measured at the end of the trial. The bulk of experiments were run for $N = 16$ buoys, although we have also conducted experiments with $N = 4$ buoys. The dimension of parallelism used in each is specified in the second column of Table 2.2.

Describing the table in row-order: Random Search (R-S) places buoys at random across the search field; $\mathrm{PE}_{50,\mu}$ and $\mathrm{PE}_{f,\mu}$ are partial-evaluation searches (see Section 2.1.5.1) evaluating solutions (in tournaments) on randomly selected subsets of unique frequencies;

TABLE 2.2: Summary of the search methods used in this paper. All methods are given the same computational budget. Parallelism can be expressed as per-individual or per-frequency depending on the number of individuals in the population (see Section 2.1.4.2).

Abbreviation	Parallelism	Description
R-S	per-frequency	Random Search
$\text{PE}_{50,\mu}$	per-individual	Partial Evaluation[44], all frequencies (PE_{Full}), population $\mu \in \{10, 50, 100\}$
$\text{PE}_{f,\mu}$	per-individual	Partial Evaluation [44], partial frequencies, $f \in \{1, 4, 16\}$, $\mu \in \{10, 50, 100\}$
TDA	per-individual	Algorithm for optimising wind turbine placement from [93]
CMA-ES	per-individual	CMA-ES[89] all dimensions, $\mu =' 4 + int(3 * log(D))$ndim , $\sigma = 0.17 * Area$
CMA-ES (2+2)	per-individual	Setup for CMA-ES from [39], $\sigma = 20m$
$CMA - ES_{PF}$ (2+2)	per-frequency	All settings are based on [39]
DE_{Pcr}	per-individual	Differential evolution [94], $\mu = 50$, $F = 0.5$, $P_{cr} \in \{0.3, 0.5, 0.7, 0.9\}$
1+1EA_{σ}	per-frequency	1+1EA(all dimensions), mutation step size with $\sigma \in 3, 10, 30$(m)
1+1EA_{s}	per-frequency	1+1EA (all dimensions) with uniform mutation in range $[0, s]$ with $s = 30$ from [39]
$\text{1+1EA}_{\text{Linear}}$	per-frequency	1+1EA (all dimensions) with linearly decaying mutation step size [95]
$\text{1+1EA}_{1/5}$	per-frequency	1+1EA (all dimensions) with adaptive step size [95]
Iterative 1+1EA	per-frequency	Iterative local search - buoys are placed in sequence using best of local neighborhood search, $\sigma = 100(m)$ for inserting the new buoy, Mutation step size= $(l/10)$ decreased lineally (Eq.2.5), Stopping Criteria for optimising each buoy based on power and number of mutations
$\text{LS+NM}_{\text{allDims}}$	per-frequency	Local sampling + Nelder-Mead search in all Dimensions
NM_Norm_{2D}	per-frequency	Buoys placed in sequence using Nelder-Mead search, Initial placement normally distributed from last buoy position, MaxFunEvals=30, for inserting the new buoy $\sigma = 100(m)$
NM_Unif_{2D}	per-frequency	Buoys placed randomly and then refined using Nelder-Mead Initial placement uniformly distributed from last buoy position, MaxFunEvals=30.
$LS_1 + NM_{2D}$	per-frequency	Local Sampling + Nelder Mead search. Buoys placed at random offset from previous buoy and placement refined by Nelder-Mead search. [96], Stopping criteria for NM for optimising added buoy (Tolerance=$0.1\% * Power$), $\sigma = 100m$ (inserting buoys) and step size based on Equation 2.5
$LS_3 + NM_{2D}$	per-frequency	Repeated local sampling + Nelder Mead search. Placements sampled at three random offsets from previous location, best placement used as starting point for Nelder-Mead search.

TDA is an algorithm for placing wind-turbines described in [93]. CMA-ES applies CMA-ES to all the dimensions of the search problem with a population determined by the formula in that row; CMA-ES (2+2) is one of the two major buoy placement search methods in [39]. DE is differential evolution with population 50 and three different values for the Pcr parameter ($Pcr \in \{0.3, 0.5, 0.7\}$); (1+1)EA$_\sigma$ and (1+1)EA$_s$ mutate one buoy's location at a time using either a normal distribution ($\sigma = 100m$) or a uniform distribution ($[0, s]$) respectively ; (1+1)EA$_{\text{Linear}}$ uses a mutation step size that decreases linearly [95][1]; (1+1)EA$_{1/5}$ uses a step size that becomes larger if more than 1/5th of the steps are successful in improving fitness and it reduces the step size if less than 1/5th of steps are successful; Iterative-(1+1)EA is an iterative algorithm (see Section 2.1.5.2) for one-at-a-time buoy placement; LS+NM$_{\text{allDims}}$ is a hybrid-search (see Section 2.1.5.3 for all hybrid methods) which follows stochastic buoy placement with optimisation by the Nelder-Mead (NM) simplex direct search [96]; NM_Norm$_{2D}$ and NM_Unif$_{2D}$ are the same as LS+NM$_{\text{allDims}}$ but it uses NM search to refine buoy positions one at a time rather than all-at-once; $LS_1 + NM_{2D}$ alternates stochastic placement and NM search; finally, $LS_3 + NM_{2D}$ conducts a three-sample local search for each buoy placement followed by NM search. Short descriptions of the more specialised search methods listed above follow.

2.1.5.1 Partial Evaluation

PE [44] saves evaluation time by evaluating the fitness of an individual just partially. In our work we applied partial evaluation with randomly selected subsets of frequencies in each generation, where the number of such frequencies is fixed for the duration of the run. We used the non-elitist $\mu + \lambda$EA in [44] as the framework for driving evolution. Note that, because fitness is assessed on partial information it is necessary to include a single generation at the end of the process where each individual layout is evaluated at all frequencies so the best-performing individual can be selected. The cost of this last generation depends on the population. For $\mu = 100$, this time is substantial and 12 hours must be allocated at the end, leaving 2.5 days to run the actual PE search algorithm. Proportionately less time is needed for smaller populations. In the meantime, two kinds of mutations are used. Firstly, the position of buoys are mutated based on uniformly distributed random numbers in a circle ($r = l/16$) with a radius of 18(m) and 35(m) for 4 and 16 buoys respectively. Secondly, a normal distribution is employed for resampling the buoys location with $\sigma = 10(m)$ $(PE - N)$.

[1]

$$Mutation - step_{size} = (Initialstep_{size}) * (1 - 0.92 * iter/Max_{iter}) \quad (2.5)$$

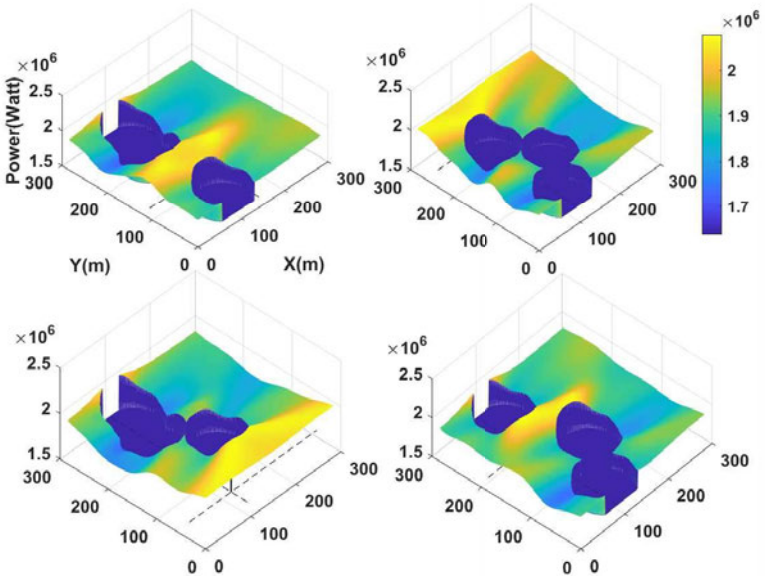

FIGURE 2.1: The wave farm's power landscape for the insertion of the last buoy of 4-buoy layout into locations across the farm area. Dashed lines show the locations of the local optima for adding a fourth buoy.

2.1.5.2 Iterative 1+1EA

In contrast to the other (1+1)EA algorithms described in Table 2.2 the Iterative (1+1)EA method positions buoys one after the other. Each buoy is placed using a (1+1)EA-like search starting from the previously placed buoy. Step size decreases linearly during search (see Equation 2.5). For each buoy the search stops either when the new buoy has a q-factor of ≤ 1.0, or when a preset number of mutation steps is reached. The latter is done in order to limit the time spent in the local search as further buoys remain to be placed.

2.1.5.3 Hybrid Search

In pursuit of a more informed search heuristic, a brief study was conducted to sample the marginal energy gain resulting from adding a new buoy to the neighbourhood of buoys that have already been placed. Figure 2.1 shows the results of this landscape analysis for placing a fourth buoy near three previously placed buoys. Areas of high energy output are shown in yellow, while the blue chasms represent closeness constraint violations.[2]

Two important properties are apparent from these graphs. First, is that the landscape, though multi-modal, is smooth. This means that a search with a local search component

[2]In fact, the actually underlying 4-buoy layout is the result of comprehensive 4-buoy layout optimisations. For each of the four figures, one buoy was removed and then the landscape mapped using a grid search. This figure confirms that the underlying layout was indeed a local optimum with respect to single-buoy mutations.

may be beneficial. The second property is that, due to positive reinforcement effects, peak energy output is often in the neighbourhood of previously placed buoys. This indicates that it might be good to start the search near a previously placed buoy.

These observations have informed the design of the last five search methods in Table 2.2. The first of these is LS+$\text{NM}_{\text{allDims}}$, described in Algorithm 2.1.

Algorithm 2.1 LS+$\text{NM}_{\text{allDims}}$

procedure LOCAL SAMPLING + NELDER-MEAD SEARCH (ALL DIMS)
Initialization
  $size = \sqrt{N * 20000}$ ▷ Farm size
  $\mathbf{x} = [x_1, \ldots, x_N] = \perp$ ▷ x-positions
  $\mathbf{y} = [y_1, \ldots, y_N] = \perp$ ▷ y-positions
  lastx=size/2; lasty=0 ▷ first buoy position
  $bestEnergy = 0$ ▷ Best energy so far
  $bestLayout = [\mathbf{x}, \mathbf{y}]$ ▷ Best layout so far
search
  while *stillTime()* **do** ▷ Iterative search
    for i in $[1, .., N]$ **do**
      while *not valid* $(\mathbf{x}, \mathbf{y})$ **do**
        $x_i = randn(\sigma) + lastx$ ▷ new buoy position
        $y_i = randn(\sigma) + lasty$ ▷ new buoy position
      end while
      $lastx = x_i$; $lasty = y_i$ ▷ Update last buoy position
    end for
    $([\mathbf{x}, \mathbf{y}], energy) = NM\_Search(Eval, [\mathbf{x}, \mathbf{y}])$ ▷ Local search
    if then $energy > bestEnergy$ ▷ If better?
      $bestEnergy = energy$ ▷ Update energy
      $bestLayout = \mathbf{x}, \mathbf{y}]$ ▷ Update layout
    end if
  end while
  return *bestLayout* ▷ Final Layout
end procedure

This algorithm repeatedly adds buoys at random offsets from the previous one followed by a Nelder-Mead local search on all buoy positions. The Nelder-Mead local search is limited to 10 iterations so that the outer **while** loop has time to build and test repeated configurations until the time budget for buoy placement runs out. Inside the **for** loop the buoys are placed one at a time with each successive buoy being placed at a distance, sampled from a normal distribution, from the previous buoy. In this algorithm the normal distribution has $\sigma = 100m$, which is an educated guess informed by the landscape mapping in Figure 2.1. Note that, for this algorithm, the *Eval* function is parallelised on a per-frequency basis.

The next two search methods in Table 2.2 are: NM_Norm_{2D} and NM_Unif_{2D} are greedy algorithms that, like LS+$\text{NM}_{\text{allDims}}$, place buoys one at a time at a random

offset from the previous buoy. However, in these algorithms the *NM_Search* is run to optimise each buoy position before proceeding to the next buoy placement. The time budget for each *NM_Search* phase is: $3days/N$ so that there is equal time devoted to each buoy placement. Note that in this algorithm the call: $Eval_{([x_1,\ldots,x-i],[y_1,\ldots,y-1])}$ is implicitly passed the arguments for the buoys placed to date so that it can evaluate the new buoy position $[x_i, y_i]$ with respect to these. Also note that, due to the shorter evaluation time for smaller numbers of buoys this equal time allocation results in more search iterations for earlier buoys which serves as a good foundation for the rest of the search. The algorithm for NM_Norm_{2D} (normally-distributed offset $\sigma = 100m$) is shown in Algorithm 2.2. NM_Unif_{2D} (uniformly-distributed offset in range $[0, size]$) differs from this only in the sampling approach.

Algorithm 2.2 NM_Norm_{2D}

procedure NELDER-MEAD SEARCH (2 DIMS)
Initialization
  $size = \sqrt{N * 20000}$ ▷ Farm size
  $\mathbf{x} = [x_1, \ldots, x_N] = \perp$ ▷ x-positions
  $\mathbf{y} = [y_1, \ldots, y_N] = \perp$ ▷ y-positions
  lastx=size/2; lasty=0 ▷ first buoy position
search
  for i in $[1, .., N]$ **do**
    while *not valid* $(\mathbf{x}, \mathbf{y})$ **do**
      $x_i = randn(\sigma) + lastx$ ▷ new buoy position
      $y_i = randn(\sigma) + lasty$ ▷ new buoy position
    end while
    $([x_i, y_i], energy) =$
      $NM\_Search(Eval_{([x_1,\ldots,x_{i-1}],[y_1,\ldots,y_{i-1}])}, [x_i, y_i])$
    $lastx = x_i$; $lasty = y_i$ ▷ Update last buoy position
  end for
  return $[\mathbf{x}, \mathbf{y}]$ ▷ Final Layout
end procedure

The last two search methods in Table 2.2 are: $LS_1 + NM_{2D}$ and $LS_3 + NM_{2D}$. The algorithm for $LS_3 + NM_{2D}$ is shown in Algorithm 2.3. This algorithm makes three samples of the neighbourhood surrounding the last buoy and conducts *NM_Search* from the sampled point giving the highest energy. The stopping condition for *NM_Search* is also different from previous algorithms with a stopping tolerance of 0.1% in the energy output. Compared to earlier approaches, this *NM_Search* configuration devotes relatively little time to the search for early buoy placements, which tend to converge fast, and more to the later buoy placements which converge slowly. Note that the stopping tolerance was tuned to make sure the algorithm's running time is close to three days. The $LS_1 + NM_{2D}$ is identical to $LS_3 + NM_{2D}$ but with $iters = 1$.

Algorithm 2.3 $LS_3 + NM_{2D}$

procedure LOCAL SAMPLING + NELDER-MEAD SEARCH (2 DIMS)
Initialization
  $size = \sqrt{N * 20000}$ ▷ Farm size
  $\mathbf{x} = [x_1, \ldots, x_N] = \perp$ ▷ x-positions
  $\mathbf{y} = [y_1, \ldots, y_N] = \perp$ ▷ y-positions
  lastx=size/2; lasty=0 ▷ first buoy position
search
  for i in $[1, .., N]$ **do**
    $iters = 3$ ▷ Number of local samples
    $bestx = 0$; $besty = 0$; $bestEnergy = 0$
    for j in $[1, .., iters]$ **do**
      while *not valid* $(\mathbf{x}, \mathbf{y})$ **do**
        $x_i = randn(\sigma) + lastx$ ▷ new buoy position
        $y_i = randn(\sigma) + lasty$ ▷ new buoy position
      end while
      $energy = Eval([x_1, \ldots, x_{i-1}, x_i, y_1, \ldots, y_{i-1}, y_i])$
      if $energy > bestEnergy$ **then**
        $bestx = x_i$; $besty = y_i$
        $bestEnergy = energy$
      end if
    end for
    $([x_i, y_i], energy) =$
      $NM\_Search(Eval_{([x_1 \ldots x_{i-1}],[y_1 \ldots y_{i-1}])}, [bestx, besty])$
    $lastx = x_i$; $lasty = y_i$ ▷ Update last buoy position
  end for
  return $[\mathbf{x}, \mathbf{y}]$ ▷ Final Layout
end procedure

2.1.6 Experiments

In this section, we report on the results of our experiments. The search methodologies can be divided into single-solution and population-based methods. In the latter group the sizes of populations used vary from 2 to 100 depending on the algorithm. Figures 2.2 and 2.3 show box-and-whiskers plots for the power output of the best individuals resulting from all the configurations of the all the search heuristics shown in Table 2.2 for determining well-performing 16-buoy layout. Note that, Figure 2.3 is a subplot of Figures 2.2 showing the outputs for all the variations of PE. The PE variations shown in Figure 2.2 are full-frequency evaluation variants of the $\mu + \lambda$ algorithm used for PE with uniform and normally distributed mutation, respectively.

The first observation from both figures is that the differences in the mean output attained by all methods is less than 20%. This shows that even the most naive search methods are able to obtain non-trivial power outputs. The second observation is that with the limited number of function evaluations at hand highly adaptive search heuristics such as

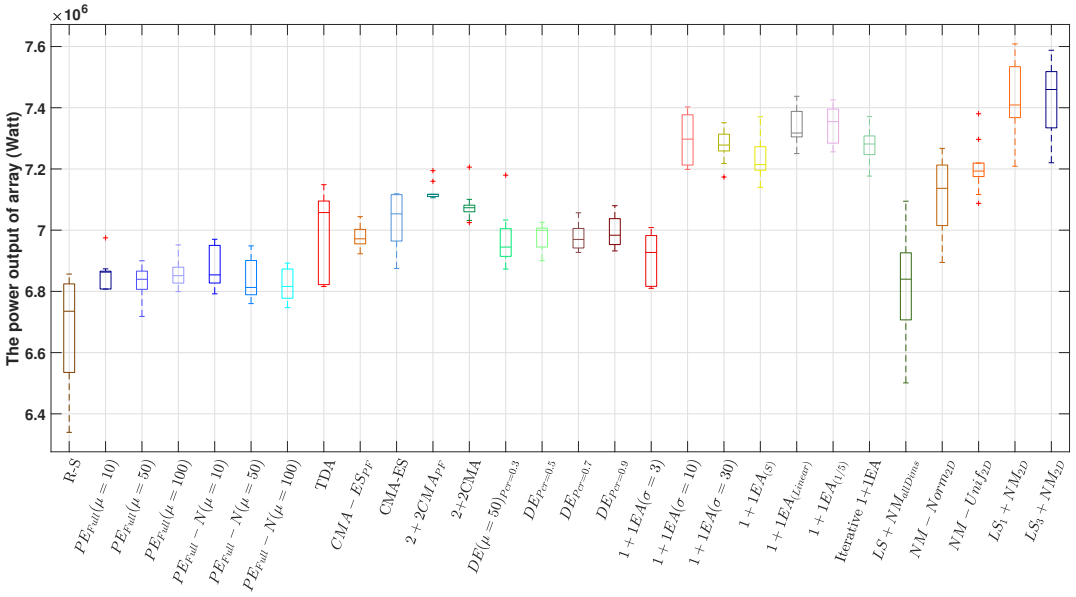

FIGURE 2.2: The comparison of the all proposed ideas results from 16-buoy layout in terms of the best layout per each experiment. With regard to the median performance , $LS_3 + NM_{2D}$ can overcome other methods .

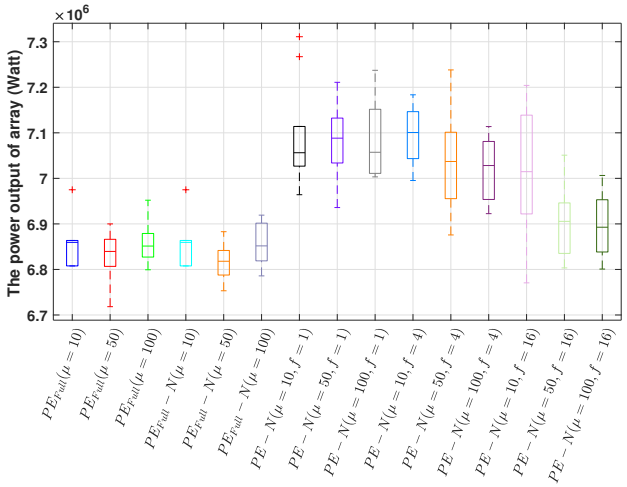

FIGURE 2.3: The optimisation results of Partial Evaluation method with three population sizes:$\mu = 10$, 50, 100 and different wave frequencies are used (1, 4, 16 and 50(f)) for 16-buoy layout.

CMA-ES and DE only perform moderately well. One potential reason for this is that small number of evaluations possible, in the order of 300 full evaluations of 16 buoy layouts in three days, gives little time for these methods to learn the search landscape.[3] Another observation is that the (1+1)EAs and the buoy-at-a-time placement algorithms (with local search) all perform well. The best performing algorithms are the LS_1+NM_{2D} and $LS_3 + NM_{2D}$ which are hybrid searches with settings informed by the landscape. Of these two, the LS_3+NM_{2D}, which does the local sampling appears to have a slightly

[3]Early experiments with four buoy layouts – which allow thousands of evaluations – show CMA-ES performing at least as well as other methods.

higher mean performance but the difference is not significant with this sample size. The best performance overall of 7608600 Watts is given by one of the runs of $LS_1 + NM_{2D}$.

Examining the PE methods in Figure 2.3, it appears that variants with lower number of frequencies sampled seem to perform better. These variants are able to perform many more evaluations than those sampling higher numbers of frequencies, at the cost of having a less informed and more noisy evaluative function. From both figures it appears that there is no clear advantage accruing to methods with larger population sizes. This is likely to be a product of the limited number of evaluations available. Overall there seems to be an advantage in evaluating on fewer frequencies and using a smaller population.

To examine how the various search methods converged the average fitness of the best individuals in each population were recorded for each method. These results are plotted in Figures 2.4 for partial evaluation and 2.5 for all others. Note that, in both sets of plots the averages were obtained by fully evaluating the population at the sampled time and extracting the best performing individual for that run — in case of PE, this happened in post-processing. The top row of Figure 2.4 is ordered by the number of frequencies. As can be seen there is a clear decrease in the speed of optimisation as the number of sampled frequencies increases. Moreover the relative advantage in speed of optimisation for small populations becomes more marked for more evaluated frequencies. In the second row, ordered by population, the speed of evolution is highest for the lowest population but starts off a lower base.

In Figure 2.5 the distinct groups of algorithms are observable. The PE full frequency heuristics start with relatively good performance but have relatively flat fitness curves. Next the CMA-ES variants progress quickly from a low base and then flatten out in performance. The DE and 1+1EA variants, respectively, follow smoother and higher curves. Finally, the $LS_1 + NM_{2D}$ starts off a very low base (below the x-axis) and steps up steeply with initial buoy placements followed by Nelder-Mead search (the shallow-sloping steps). The overall result of this hybrid algorithm is slightly better overall than the other methods. Finally, the layout of wave-buoy's produced by the algorithms offers some interesting insight into the features of these highly productive individuals. Figure 2.6 shows the most productive individual layout found in all the search runs. This layout is built by the algorithm from the x-axis upwards with buoys numbered in the figure in order of placement. It is clear that the initial placement order forms an almost straight diagonal line from the bottom sloping upwards to the right. The buoys then start to slope leftwards toward the front. These placement make sense in terms of placement of adjoining buoys in the peaks of the power landscape. Note that buoy 8 is placed in front of the others which reduces the energy output of the buoys behind before buoy 9 and 10 are placed in the original diagonal pattern. At this point, options that

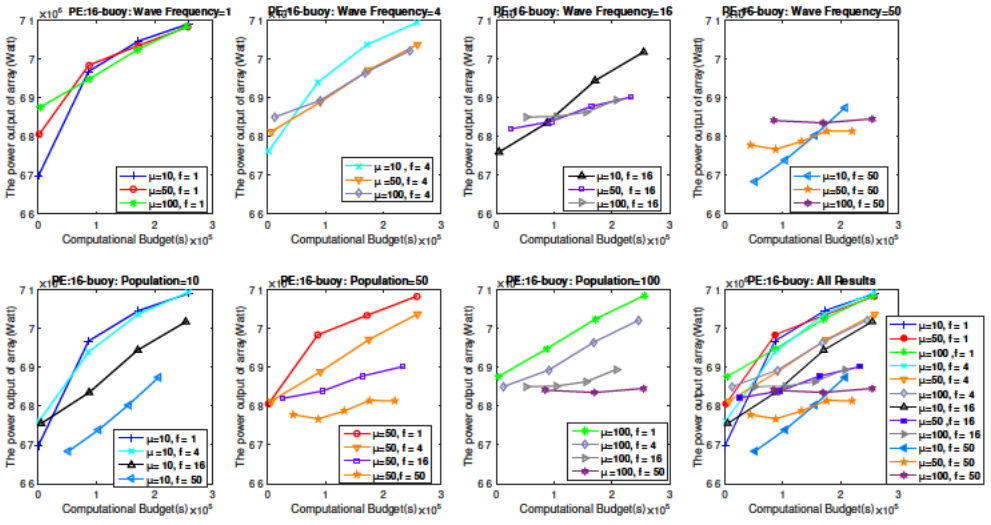

FIGURE 2.4: The PE performance comparison of different number of wave frequency (1, 4, 16 and 50) with three size of populations (μ =10, 50 and 100) results from 16-buoy layout based on the average computational time.

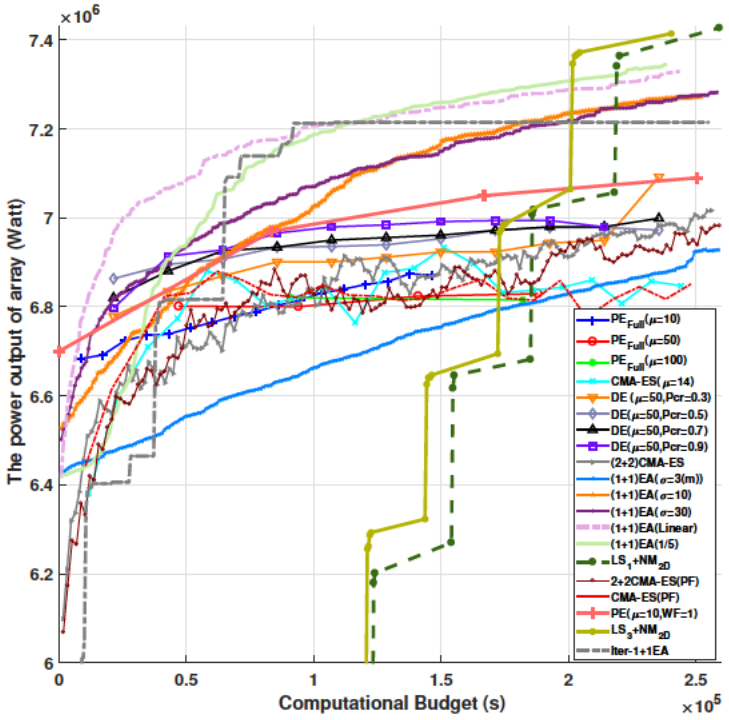

FIGURE 2.5: A comparison of the average computational budget of proposed methods for 16-buoy layout over 72 hours.

do not interfere negatively with other buoys in this layout are exhausted so a second front of buoys has started to form that alternates in the y-dimension with the original front so as to minimise the impact of negative interference. It should be noted that this zig-zag pattern of farm layout is observable in results of many of the high-performing runs. Another feature common to many runs is the formation of the second row of buoys,

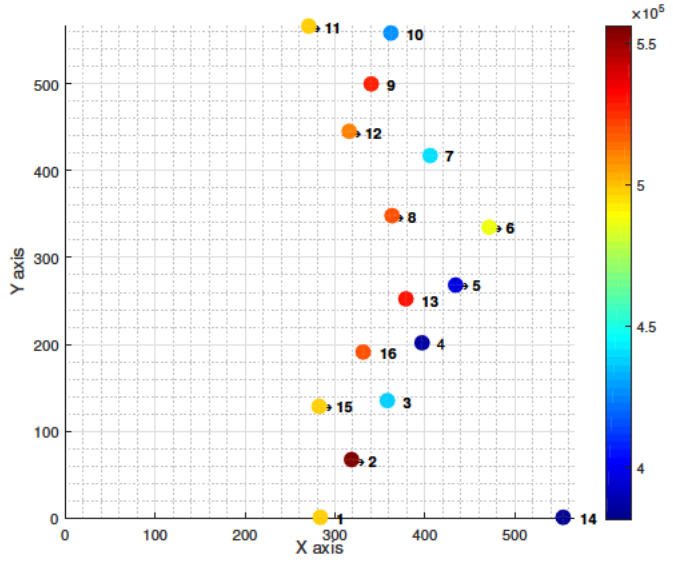

FIGURE 2.6: The best layout of $LS_1 + NM_{2D}$ for 16 buoy research. The area size is $566m^2$, the q-factor=0.956, total power output 7608600 Watts, and energy generated by each converter is shown by a range of colors. The order of inserting a new buoy is numbered.

often started before first row is complete. It is not clear if the early formation of this second row is an artifact of stochastic nature of the hybrid search heuristics or there are fundamental properties of the problem that drive this behaviour, at least in constrained environments.[4]

2.1.7 Conclusions

In this investigation, several evolutionary optimisation algorithms are applied and evaluated for maximising the total captured power of 16 buoy layouts using an improved and detailed evaluative function. The optimisation environment is challenging, with a very limited number of full evaluations possible within the evaluation budget. Because the algorithms explored have diverse behaviour in terms of evaluative costs algorithms were compared in the realistic scenario of searching within a generous time budget on a multi-core machine.

The methods that performed best were hybrids of stochastic buoy placements and uphill local search. One advantage of these search strategies is the one-at-a-time buoy placements reduced the dimensionality of the search space to just the next buoy. A potential disadvantage of this greedy placement approach is that it allows no backtracking to improve the positions of previously placed buoys. However, preliminary experiments with

[4]In experiments with four buoys there is no formation of a second front.

global optimisation of these best buoy layout have yielded very little improvement, indicating that substantial improvement will involve more than simply tuning the discovered layout.

This work also explored partial evaluation by frequency and showed that a small number of frequencies and a small population yielded the best results in terms of search but still less effective overall than other methods.

Finally from many observations of different optimal layouts and analysing the landscapes of the farms, it appears that a positive hydrodynamic interaction can be obtained if buoys are placed at a relative angle of approximately 45 degrees. This observation might be exploited in the initialisation phase.

This work can be carried in several potential directions. First new, more informed hybrid algorithms can be developed. It may be possible to combine smarter initialisation with iterative local search. Variants of partial evaluation can be used that evaluate on the energy from a sample of buoys rather than frequencies. If carefully designed such an algorithm may allow the productive use of crossover as a way of combining individuals with complementary partial fitnesses. There is also scope to apply this work to an even more refined model with more wave directions and non-uniform water depth. Finally, the optimisation can be extended to incorporate a cost model based on sharing tether points, accounting for the different tether angles and tether lengths that this analysis would entail.

Our code, layouts, and auxiliary material are publicly available: https://cs.adelaide.edu.au/~optlog/research/energy.php

Acknowledgements. We would like to thank Boyin Ding, Frank Neumann, and Nataliia Sergiienko for their input. Our work was supported by the Australian Research Council project DE160100850.

2.2 New Insights into Position optimisation of Wave Energy Converters using Hybrid Local Search

2.2.1 Synopsis

As discusses in the previous section, optimising WEC positions is a challenging research problem due to the complex and extensive interactions (both constructive and destructive) between devices. The problem is complicated by the fact that WEC arrangement strongly affects the farm's power output. Meanwhile, we revealed in the previous section that placing and optimising each WEC, one at a time, at a local or neighbourhood position of the previous converter strategy results in better performance than the other global optimisation methods. However, the stochastic local search phase can be a time-consuming method of optimising a large wave farm. This paper focuses on improving our prior work by placing the next converter in a relative position informed by peaks in the power landscape built from sampling positions in a two-buoy model under local wave conditions. This new approach composed of local search, a surrogate power model, and numerical optimisation. In the following section, the proposed hybrid method is compared with other state-of-the-art search methods in five different wave scenarios – one simplified irregular wave model and four real wave regimes. The new hybrid method outperforms well-known previous heuristic methods in terms of both quality of achieved solutions and the convergence rate of searches in all tested wave regimes.

[23] Neshat, M., Alexander, B., Sergiienko, N., & Wagner, M. (2019). New insights into the position optimization of wave energy converters by a hybrid local search. Published by Swarm and Evolutionary Computation–journal on [26 July 2020] .

Statement of Authorship

Title of Paper	New insights into the position optimization of wave energy converters by a hybrid local search
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Alexander, B., Sergiienko, N., & Wagner, M. (2020). New insights into the position optimization of wave energy converters by a hybrid local search. Published by Swarm and Evolutionary Computation--journal on [26 July 2020].

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%) 80%			
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	05/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contr bution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander		
Contribution to the Paper	Helped with some of the design of methodology. Gave feedback on the results. Helped revise the paper.		
Signature		Date	12/05/20

Name of Co-Author	Nataliia Y.Sergiienko		
Contribution to the Paper	Developed and coded an objective function for optimisation, wrote and edited Section 2, generated Figure 14.		
Signature		Date	13/05/2020

Name of Co-Author	Markus Wagner		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

2.2.2 Abstract

Renewable energy will play a pivotal role in meeting future global energy demand. Of current renewable sources, wave energy offers enormous potential for growth. This research investigates the optimisation of the placement of oscillating buoy-type wave energy converters (WECs). This work explores the design of a wave farm consisting of an array of fully submerged three-tether buoys. In a wave farm, buoy positions strongly determine the farm's output. Optimising the buoy positions is a challenging research problem due to complex and extensive interactions (constructive and destructive) between buoys. This research is focused on maximizing the power output of the farm through the placement of buoys in a size-constrained environment. This paper proposes a new hybrid approach mixing local search, using a surrogate power model, and numerical optimisation. The proposed hybrid method is compared with other state-of-the-art search methods in five different wave scenarios – one simplified irregular wave model and four real wave regimes. The new hybrid methods outperform well-known previous heuristic methods in terms of both quality of achieved solutions and the convergence-rate of search in all tested wave regimes. The best performing method in real-wave scenarios uses the active set non-linear optimisation method to tune final placements. The effectiveness of this method seems to stem for its capacity to search over a larger area than other compared tuning methods.

2.2.3 Introduction

Wave energy represents one of the most promising forms of renewable energy due to the high energy density of wave environments and minimal environmental impact [18].

One of the current-best designs for wave energy converters (WEC) consists of a large floating buoy tethered to the seafloor. With this design, energy is produced by the motion of the buoy exerting force on the tether [97]. In some actual deployments, multiple buoys, laid out in a farm, are able to extract power from the waves more than 90% of the time [18]. In addition, WECs are able to take advantage of the high energy-density of marine environments – up to 60 kW per meter of wave front length with a very low impact on aquatic life [87].

The amount of power produced by a farm or an array of WECs depends on their number, their arrangement with respect to each other, and the prevailing wave conditions [21, 22]. Thus, generating the appropriate arrangement of WECs in an array is an important factor in maximizing power absorption.

Currently deployed designs for WECs produce up to 1 MW per buoy [97]. Thus, to be of commercial scale, it is necessary for farms to consist of multiple buoys. However, as the number of converters increases, the optimisation of buoy placement becomes more challenging because of the complex hydrodynamic interactions among converters. These interactions can be constructive or destructive, and the geometry of these interactions depends strongly on the prevailing wave regime in the environment.

In evaluating potential layouts, it is important to use an energy-model that has both high-fidelity and simulates the best available WEC designs. The model used in this study [98] simulates the hydrodynamic behaviour of a fully submerged three-tether WEC in irregular directional waves for several sea sites.

The search space for optimizing array layouts for WECs is multi-modal. Interactions between buoys in an array are complex to model, and the evaluation of each layout is expensive, sometimes taking several minutes. This is because of complex and extensive hydrodynamic interactions between buoys, which in turn depend on the local conditions modelled. These challenges require the use of search meta-heuristics that reliably optimise buoy configurations using a very limited number of layout evaluations.

Work to date on WEC layouts has primarily employed evolutionary algorithms (EAs), which combine stochastic search with selection to progressively improve a population of candidate layouts. In early work [31], Child and Venugopal applied both a simple (and deterministic) Parabolic Intersection (PI) heuristic and a more computationally intensive Genetic Algorithm (GA) to create small (five-buoy) WEC layouts.

Later work by Sharp and DuPont [32] used a GA, coupled with heuristics to ensure minimum separation between buoys, to place a small number of WECs (5 converters, 37000 evaluations) in a discretised space. The same authors later explored a similar problem with an improved GA with a cost model [36]. In both studies, the wave-model used assumed only a single wave direction. The studies also required a relatively large number of layout evaluations, which would limit their application to more detailed wave energy models.

In [99], two meta-heuristic algorithms to optimise the geometry of the wave energy generators were introduced, which combines both particle swarm optimisation [42] and Box's complex optimisation method [100]. An alternative approach was proposed by Ruiz et al. [35], who compared the convergence rate and efficiency of three EAs in a discrete search space with a simple wave energy model. The EAs included: CMA-ES [89], a custom GA, and glow-worm swarm optimisation (GSO) [101]. Their work found that search using CMA-ES converged faster than other methods but was outperformed, in terms of energy production by both the GA and GSO. In another recent study of

WEC position optimisation, Ferri [40] used two global optimisation algorithms, CMA-ES and surrogate-model based optimisation method (MM), and compared the performance of both methods in terms of convergence rate and design quality. The experimental outcomes in [40] realised that the MM algorithm convergence speed is faster than CMA-ES, although the best found array by MM has an estimation error of 10%.

In a recent publication, a Differential Evolution with an adaptive mutation operator (IDE) [41] was applied for optimizing a wave farm with three, five and eight converters. Fang et al. proposed some new arrangements of layouts with higher produced energy; however, IDE was not assessed on large wave farms.

In other studies, Wu et al. [39] investigated two popular EAs: the 1+1EA and CMA-ES for optimizing the locations of fully submerged three-tether buoys. The results show that the 1+1EA performed better than CMA-ES when constrained to a very limited number of layout evaluations. A much more detailed wave scenario was applied in Neshat et al. [1], using an irregular wave model with seven wave directions and fifty sampled wave frequencies, to evaluate a wide variety of EAs and hybrid methods. This work found that a combination of a stochastic local search combined with the Nelder-Mead simplex method (LS-NM) can obtain better 4 and 16-buoy configurations in terms of the total absorbed power. The same real wave scenarios were applied in [4] for optimising the 16-buoy layout position by an adaptive neuro-surrogate optimisation (ANSO) method that constituted of a surrogate Recurrent Neural Network (RNN) model and SLS-NM. However, other dimensions of WECs (PTO and geometric parameters) and the large size of wave farms were not assessed in [4].

Other aspects of wave-farm design have also been considered. For example, Neshat et al. [3] considered the optimisation of power take-off parameters in addition to the layout optimisation simultaneously for farms to maximise power output by comparing several optimisation strategies, and proposing a hybrid method combining a symmetric local search with Nelder-Mead simplex search and a backtracking technique (SLS-NM-B).

In other recent WEC optimisation work [102] compared the performance of the several modern multi-objective optimisers to maximise the total power output with respect to three opposing objectives including the necessary farm area, WEC positions and the length of the cable for connecting all converters. Further, a Particle Swarm optimisation (PSO) algorithm [43] has been used to minimise the technology's Levelized Cost of Energy (LCOE) of the P80 hybrid wind-wave farm by adjusting the parameters of the layout, the offshore substation position, and the size of the export cable option. However, that work did not compare the performance of PSO with modern adaptive and self-adaptive EAs [103–105]. In the following work [38], PTO parameters were tuned by using the hidden genes genetic algorithm (HGGA). While the improved GA boosted the total

energy generated, the performance of HGGA was not compared to other state-of-the-art EAs. Glass et al. [37] proposed a hybrid GA (GA + multiple analytical scattering (MAS) method) to tune the geometric and control parameters of a small wave park (5 and 9-buoy). The layouts produced by that work show considerable constructive interactions in a simple (uni-directional) wave scenario. Furthermore, various dimensions of WECs array optimisation were studied by [33] and [106], where both works optimised WECs position and buoy dimensions by employing GAs. In the mentioned studies, it was achieved that the array performance can be increased if the farm includes different WEC dimensions.

Blanco et al. [107] proposed a Multi-Objective Differential Evolution Algorithm (MODE) to optimise the power output of a particular 2-Body Point absorber by adjusting the arrangement, power take-off (PTO), control strategy and hydrodynamic design. A simple wave model (direction of wave propagation is from left to right) was used by Arbones et al. [108, 109] in a multi-objective optimisation problem. In that study, two methods (MO-CMA-ES and SMS-EMOA) were applied to produce good trade-offs between the converter positions, the farm area and required cable length. One of the shortcomings of these approaches, in terms of real-world applicability, is that these works used only a very simple wave model with just one wave frequency and direction. Furthermore, recently Blanco et al. [110] refined their prior approach by considering PTO constraints and two cost functions, including the mean annual power produced and WEC surface. Bonovas et al. [111] modelled and optimised a special PTO wave energy converter that couples with hydraulic energy storage in a coastal reservoir using Multi-Objective Evolutionary Algorithms (MOEA) with two objective functions: the total cost of the investment and the rate of flow in the reservoir.

This paper improves on prior work in the following ways: augmenting the findings of [1] to include another nine new heuristic search methods, including a novel surrogate-based model, all applied to the original irregular wave model; and including four new real wave regimes from the southern coast of Australia (Adelaide, Perth, Tasmania and Sydney) using a higher granularity of wave-directions.

From the experimental results, it is shown that a hybrid framework consisting of a learned model-based local search interleaved with numerical optimisation outperforms previous heuristic methods in terms of both convergence rate and higher total power output for 16-buoy layouts.

The paper structure is as follows. The next section describes the design of the buoy and the model that is used to simulate the inter-buoy interactions. Section 2.2.5 describes the

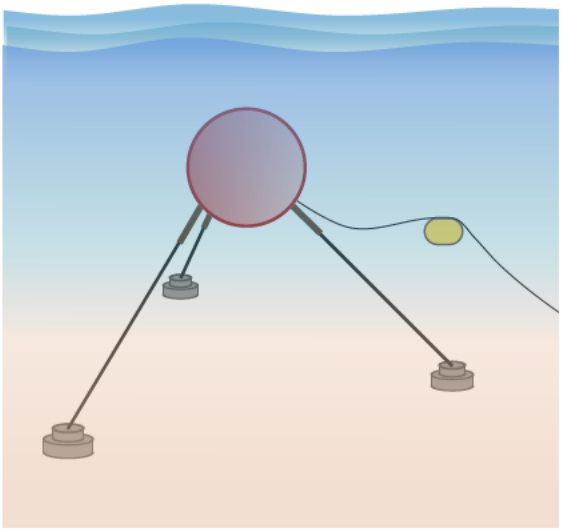

FIGURE 2.7: Wave Energy Converter

TABLE 2.3: WECs parameters

Parameter	Value
Buoy radius	5 m
Water depth	50 m
Submergence depth	8 m
Buoy mass, m	376×10^3 kg
Tether angle	55°
PTO stiffness, K_{pto}	2.7×10^5 N/m
PTO damping, B_{pto}	1.3×10^5 Ns/m

optimisation problem and Section 2.2.8 describes the proposed search methods. The experiments are presented and discussed in Section 2.2.9. Finally, conclusions are presented in Section 2.2.10.

2.2.4 The Numerical Model

2.2.4.1 Wave Energy Converter Description

The wave energy converter considered in this study is shown in Figure 2.7. This converter consists of a fully submerged spherical buoy connected to three independent power take-off units through the inclined taut tethers. This design represents a simplified version of the CETO system that is currently under development by Carnegie Clean Energy [97]. The parameters of the WEC are specified in Table 2.3.

The tripod configuration of the power take-off (PTO) system allows power to be absorbed from three translational degrees of freedom, namely: surge, sway, and heave. When a wave energy converter operates in isolation, its motion and power production depends

only on the incoming wave parameters and settings of the power take-off system. However, in the case of a wave farm, the motion of all converters is coupled, and the strength of this hydrodynamic coupling directly depends on the distances between WECs. Thus, in a regular wave with a frequency ω, the motion of all buoys comprising the wave farm can be described by the following equation:

$$(\mathbf{M}+\mathbf{A})\ddot{\mathbf{x}}+(\mathbf{B}+\mathbf{B}_{pto})\dot{\mathbf{x}}+\mathbf{K}_{pto}\mathbf{x}=\mathbf{F}_{exc}, \tag{2.6}$$

where vector $\mathbf{x}\in\mathbb{R}^{3N\times 1}$ describes the motion of N buoys in three translational degrees of freedom, $\mathbf{M}=m\mathbb{I}_{3N}$ is a diagonal mass matrix of the wave farm ($\mathbb{I}_{3N}$ is the identity matrix of size $3N$), $\mathbf{A}$ and $\mathbf{B}$ are the matrices of hydrodynamic added-mass and radiation damping coefficients respectively, $\mathbf{K}_{pto}=K_{pto}\mathbb{I}_{3N}$ and $\mathbf{B}_{pto}=B_{pto}\mathbb{I}_{3N}$ are the PTO stiffness and damping matrices respectively, and $\mathbf{F}_{exc}$ is the frequency dependent vector of excitation forces acting on all WECs.

Once the parameters of the incoming wave, such as the wave frequency and the travelling direction, and the coordinates of all WECs in a farm are known, hydrodynamic coefficients (matrices $\mathbf{A}$, $\mathbf{B}$ and a vector $\mathbf{F}_{exc}$) can be calculated using a semi-analytical model presented in [92]. As a result, Eq. (2.6) can be solved in the frequency domain with respect to vector $\mathbf{x}$ obtaining the motion of all converters in a regular wave of unit amplitude, wave frequency ω and wave angle β. Thus, knowing $\mathbf{x}(\beta,\omega)$, the averaged power production of a wave farm can be estimated as:

$$P(\beta,\omega)=\frac{\omega^2}{2}\mathbf{x}^{\mathrm{T}}(\beta,\omega)\mathbf{B}_{pto}\mathbf{x}(\beta,\omega). \tag{2.7}$$

2.2.4.2 Wave Resource

Four potential sites on the southern coast of Australia are considered in this study: Adelaide, Perth, Tasmania (southwest coast) and Sydney. The directional wave rose and wave scatter diagram for the Sydney and Tasmania sea sites are shown in Figure 2.8. These underlying wave data were obtained from the Australian Wave Energy Atlas [112].

2.2.4.3 Wave Farm Performance Evaluation:

The total average annual power (P_{AAP}) produced by the wave farm is calculated by summing the contribution of energy absorption from each of the sea states representing

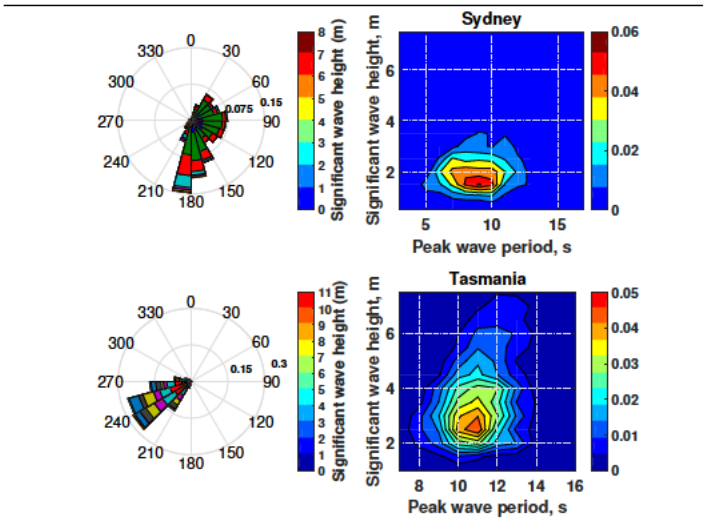

FIGURE 2.8: Wave data for two test sites in Australia: (a) Sydney and (b) Tasmania. The directional wave rose and wave scatter diagram (left to right).

a wave climate [113]:

$$P_{AAP} = \sum_{i=1}^{N_s} O_i(H_s, T_p) P_i(H_s, T_p), \tag{2.8}$$

where N_s is a number of sea states considered, H_s is the significant wave height, T_p is the peak wave period of the sea state, $O_i(H_s, T_p)$ is the probability of occurrence of a sea state (which can be derived from the wave scatter diagram) and $P_i(H_s, T_p)$ is the power generated by the wave farm in the i-th sea state. A sea state refers to the condition of the sea/ocean surface that can be characterized by statistics (significant wave height and peak wave period).

For irregular waves, $P_i(H_s, T_p)$ is the sum of power contribution from each frequency in the spectrum and each wave direction [113]:

$$P_i(H_s, T_p) = \int_0^{2\pi} \int_0^{\infty} 2 S_i(\omega) D(\beta) P(\beta, \omega) \mathrm{d}\omega \mathrm{d}\beta, \tag{2.9}$$

Where $P(\beta, \omega)$ is calculated according to Eq. (2.7), $S_i(\omega)$ is the irregular wave spectrum (the Bretschneider spectrum considered in this study) and $D(\beta)$ is the directional spreading spectrum specific for the site (obtained from the directional wave rose).

The hydrodynamic interaction among converters in the wave array affects its power production and is usually quantified using the q-factor, defined as:

$$q = \frac{P_{AAP}}{N P_{AAP}^{isolated}}, \tag{2.10}$$

where N is a number of WECs forming the array, $P_{AAP}^{isolated}$ is the power generated by an isolated WEC. If the wave interaction has, on average, a constructive effect on the power production of the array, then $q > 1$, and if the effect is destructive then, $q < 1$.

In this work, all optimisation algorithms are, first, evaluated using a simplified synthetic wave model that corresponds to the most frequently occurring sea state at the Sydney site ($H_s = 2$ m, $T_p = 9$ sec) using Eq. (2.9). Subsequently, the algorithms are tested for four real wave scenarios using Eq. (2.8).

2.2.5 Optimisation Setup

Using the wave model, the optimisation problem can be stated in terms of positioning N WECs over a bounded area of a wave farm Ω in order to maximise the total power production P_{AAP}.

$$\begin{aligned} P_{AAP}^{*} &= argmax_{\mathbf{x},\mathbf{y}} P_{AAP}(\mathbf{x_i}, \mathbf{y_i}) \\ \text{Subject to} & \\ & [x_i, y_i] \in \Omega, \quad i = 1, ..., N \\ & dist((x_i, y_i), (x_j, y_j)) \geq R' \quad i \neq j = 1, ..., N \end{aligned} \tag{2.11}$$

where $P_{AAP}(\mathbf{x}, \mathbf{y})$ is the sum of mean power output by buoys positioned in an area at x-positions $\mathbf{x} = [x_1, \ldots, x_N]$ and corresponding y positions $\mathbf{y} = [y_1, \ldots, y_N]$. In this study, the maximum number of buoys is predefined to be $N = 16$. Each buoy i's position is expressed as the coordinate: $[x_i, y_i]$, so the representation of a 16-buoy array is $[(x_1, y_1), (x_2, y_2), \cdots, (x_{16}, y_{16})]$ and the decision variable size is $2 \times N$. This position is constrained to be within the area Ω. Where $\Omega = l \times w$ and $l = w = \sqrt{N * 20000}\, m$. This constraint is given to model the scenario where there are strict limits on the area allotted to a wave farm lease. A second constraint is a minimum separation between buoys ($R' = 50m$), representing a gap required for maintenance vessels to safely pass. For each array, $\mathbf{x}, \mathbf{y}$ the sum-total of the safety distance violations is:

$$\begin{aligned} Sum_{dist} = \textstyle\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} & (dist((x_i, y_i), (x_j, y_j)) - R', \\ & \text{if } dist((x_i, y_i), (x_j, y_j)) < R' \text{ else } 0 \end{aligned}$$

where $dist((x_i, y_i), (x_j, y_j))$ is the Euclidean distance between buoys i and j. To provide a smooth response to such violations, a steep penalty function $(Sum_{dist} + 1)^{20}$ is applied to the total power output (in Watts).

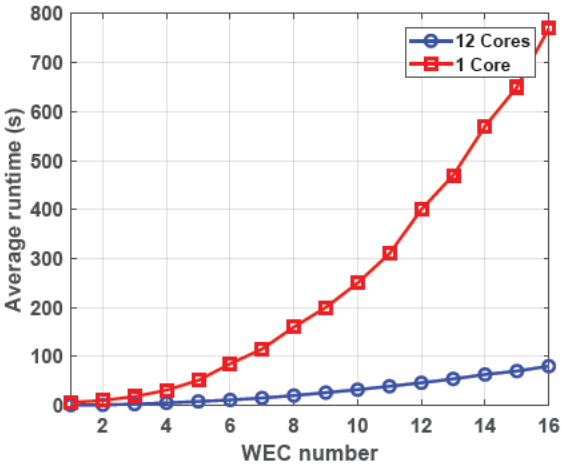

FIGURE 2.9: A runtime comparison between applying one core and running 12 cores in parallel.

2.2.6 Computational Resources and Implementation Details

In this work, to compare methods fairly a uniform time budget is allocated for each optimisation run of three days on the dedicated platform with a 2.4GHz Intel 6148 processor running 12 processes in parallel with 128GB of RAM. Calculations of the objective function in Eq. (2.11), and underlying equations of motion in Eq. (2.7) are done in MATLAB R2018 and the source code is available in [114]. The run time of the objective function with a real wave scenario is relatively high (6 s for a one-buoy layout with one core) and the time complexity increases as $O(N^2)$ with a number of interacting buoys N.

A common practice in comparing heuristic search methods is to run competing methods for the same number of function evaluations. However, such a comparison is not applicable in this paper, because the computational cost of each function evaluation (layout assessment) depends strongly on the number of buoy interactions and wave frequencies modeled – both of which vary with the search methods examined here. Therefore, in order to maximise the use of the time budget the algorithms are, depending on the search heuristic, written to evaluate the energy in parallel either per wave frequency, or per-individual. This means that, for some search methods the parallelism is expressed in terms of parallel evaluations of members of a population of individual layouts. In non-population based (single-solution) methods, the parallelism is expressed through parallel evaluation of the energy extracted from wave-frequencies. Figure 2.9 shows a comparison of the average runtime for evaluating various wave farm sizes from one to 16 buoys by one core and 12 cores in parallel.

This configuration achieves an approximately 10-fold speedup for each algorithm tested. Exceptions to this principle of making maximum use of the time budget are made for

TABLE 2.4: Optimisation Framework availability details

Name of the framework:	ISLS-AS
Version:	1.00
Available from:	Mathworks file-exchange repository [115]
Created with:	R2018b
Supported System:	Windows, MacOS, Linux, Unix
Year first available:	08 Dec 2019

search methods that converge quickly and produce little pay-off for additional evaluations. To allow for a valid statistical comparison all search methods are repeated 10 times.

The implemented optimisation framework code employed to produce the optimisation results in this paper, called ISLS-AS, has been made available. Details of the distribution used to generate the achievements in this paper are shown in Table 7.1. ISLS-AS is written in the MATLAB programming language and has an, easily extensible, portable and modular design. ISLS-AS is set up to run on the four real wave scenarios and one simplified irregular wave model considered in this paper. The primary inputs for running the framework are the wave scenario name, the number of WEC, and the selected optimisation method. Other control parameters are selected based on the optimisation process.

2.2.7 2-buoy layout evaluation

In this section, the effect of a separation distance and a relative angle between WECs on the power production for a 2-buoy wave farm is investigated (refer to Figure 2.10). For this set of experiments, the position of the first buoy ($B1$) is kept fixed at (0,0), while the position of the second buoy ($B2$) is changed iteratively varying the separation distance from 50 m to 238 m with a 1-m increment, and the angle from 0° to 359°.

The power production as a function of the relative position between the two buoys is demonstrated in Figure 2.11 for two wave climates, namely Sydney and Perth. Interestingly, for the Sydney location, the power output from each WEC increases with a separation distance having the highest power generated at a relative angle $\alpha = 90°$. However, the absorbed power from a farm located at Perth reaches its maximum at a relative short separation distance and a specific angle of 135°.

It is important to note that for both wave climates, the separation distance has a different effect on the power generation of each buoy. While increase in the distance leads to higher efficiency of the second buoy regardless the relative angle α, the first buoy experiences a negative interaction effect at angles of 90° and 135°. Therefore, it is important to note,

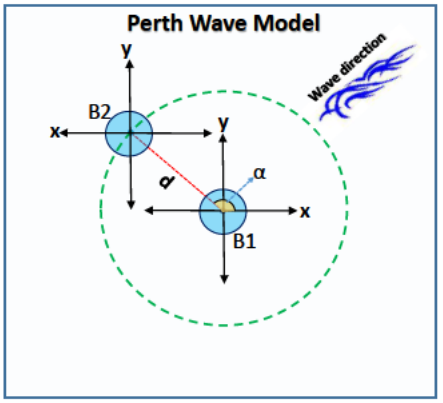

FIGURE 2.10: 2-buoy layout of the considered array. The evaluated parameters are the distance d and the angle α between two WECs (B1 and B2), and the main direction of propagation of the waves in the Perth wave model.

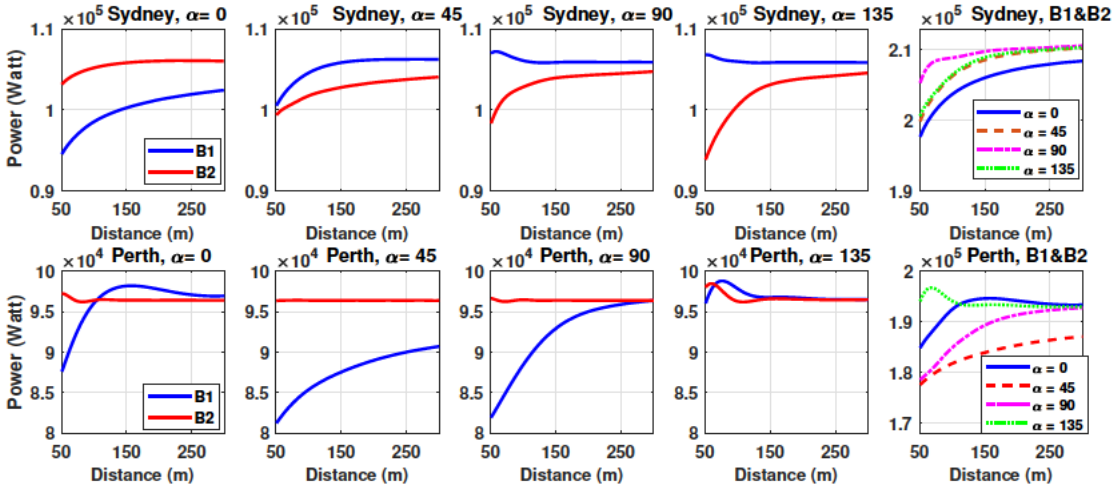

FIGURE 2.11: The impact of the distance and the angle between 2-buoy layout in Perth and Sydney wave model.

that there is no rule of thumb for optimal positioning of two buoys in a farm as their total power output strongly depends on the chosen wave climate. This can be clearly seen in Figure 2.12 where the effect of the separation distance and relative angle between two buoys is compared for two wave scenarios.

2.2.8 Meta-Heuristic Search Techniques

The complex hydrodynamic interactions among WECs in a wave farm make a multi-modal, non-convex, dynamic, constraint and expensive optimisation problem, which is challenging with a very restricted evaluation number within the evaluation resources. These challenges need the advantage of meta-heuristics that reliably optimise the position of WECs that causes to maximise the total power output of the wave farm.

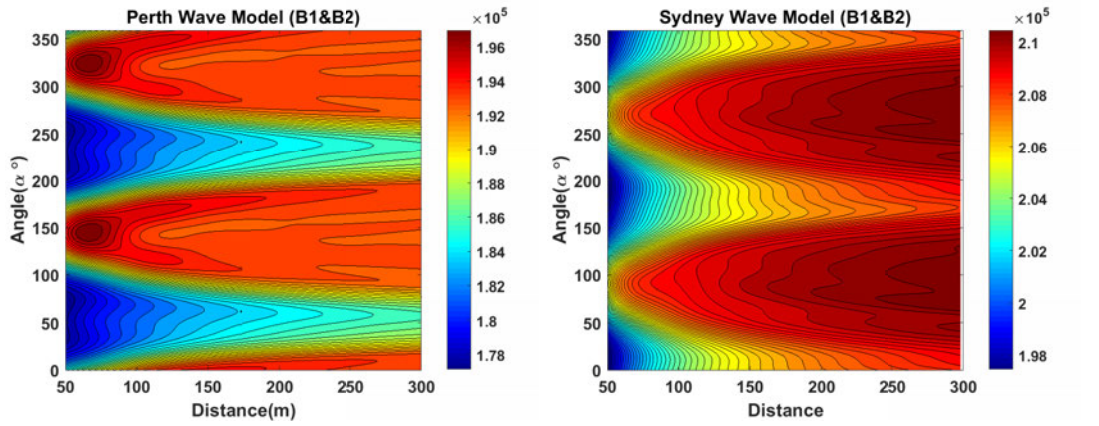

FIGURE 2.12: The overall effect of 2-buoy layout wave interactions with different distances and angles between two buoys in Perth and Sydney wave model.

Meta-heuristic optimisation methods have been applied extensively in fields where the global search is needed, including in the field of renewable energy deployment [116]. This paper compares the methods described in the previous work [1] to nine new algorithm variants derived for this work, and to other recent approaches. This previous work compared the performance of random search (R-S); forms of partial evaluation (PE) [1] where layouts are evaluated on random subsets of frequencies; TDA [93], used for wind-farm layout; CMA-ES [89]; Differential Evolution (DE) [94, 117]; Improved Differential Evolution [41]; binary Genetic Algorithm [36]; 1+1EA's with various mutation settings; local or neighbourhood search (LS); Nelder-Mead downhill search (NM); and three state-of-the-art self-adaptive meta-heuristic methods including LSHADE-EpSin [118], IPOP-CMAES [119] and HCLPSO [120, 121]. In this earlier work, the best performing heuristic, $LS_3 - NM_{2D}$ combined one-at-a-time buoy placement with iterated local search and Nelder-Mead to refine each buoy position. The algorithms described here improve significantly on the performance of this earlier work by exploiting knowledge specific to the target wave scenario. These new (smart) search heuristics are described next. All of the heuristic methods that are compared in this paper are listed in Table 2.5.

2.2.8.1 Smart Local Search (SLS)

In previous work, it was observed that a good candidate position for the next buoy is in the neighborhood of the previous buoy. The SLS method improves upon these earlier searches by placing the next buoy in a relative position informed by peaks in the power landscape built from sampling positions in a two-buoy model under local wave conditions. Examples of such landscapes are given in Figure 2.13, which shows a 3D power landscape of the simplified irregular, Sydney, and Perth wave models. It can be seen from these landscapes that, even for two buoys, there is variation in the shape of the landscape and

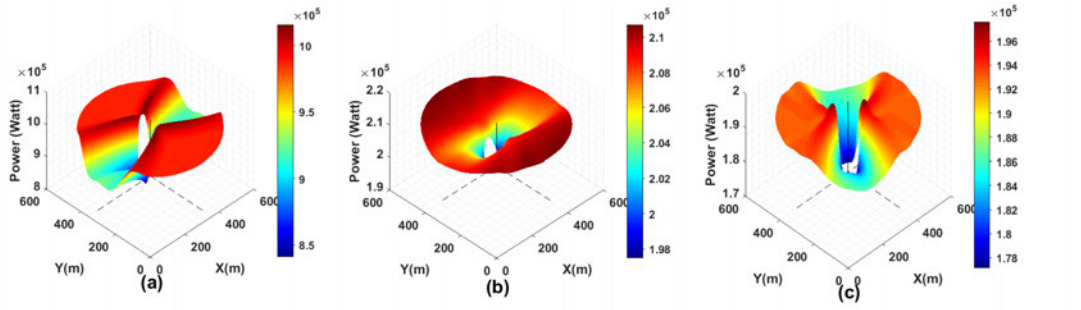

FIGURE 2.13: The 3D power landscape a two-buoy array based on the simplified irregular (a), Sydney (b) and Perth (c) wave scenarios. The first buoy's position is fixed, the second buoys is varied to measure total energy output. The mapped area extends 360◦ and a distance of between 50 m (safety distance) and 283 m (maximum distance between two converters based on the farm size).

the positions of the point at which there are constructive interactions. Note that, for a given wave regime, it is not practical to infer the shape of this power landscape by means other than sampling it. Furthermore, the inter-relationship among the absorbed power, angle and distance between two-buoy layouts for different wave scenarios can be seen in Figure 2.14. In the SLS method, this two-buoy power landscape is sampled. The pattern of sampling into this landscape is shown in Figure 2.15. This sampling landscape has an angular resolution of 45-degree intervals and a distance resolution of 5m intervals. This sampled landscape is computationally cheap to build because it models interactions between only two buoys. Moreover, this sampling exercise only has to be run once for each wave scenario at the beginning of the search process. These samples are then used to define the most promising sectors, called the *search-sectors*, in the search landscape for the placement of the next buoy. These sectors, marked in Figure 2.15 with a dotted line, lie between the best and second-best points in the search landscape on either side of the current buoy.

Algorithm 2.4 Smart Local Search (SLS)

procedure SMART LOCAL SEARCH
First Step
  WMD = Load (wave model data)
  S=Sur-Power(WMD) ▷ generate surrogate power model (Section 2.2.8.1)
Initialization
  $size = \Omega$ ▷ Farm size
  $pos = [(x_1, y_1), \ldots, (x_N, y_N)] = \perp$ ▷ positions
  $pos(1) = (size/2, 0)$
  $N1_S = 15$ ▷ number of Sample in search sector
Second Step:Search Strategy
  for i in $[1, .., N]$ **do**
    update search sectors S
    $bestEnergy = 0$
    $bestPosition = (0, 0)$
    for j in $[1, \cdots, N1_S]$ **do**
      $(x_s, y_s) = U(S)$ ▷ sample sectors
      $pos(i) = (x_s, y_s)$
      $energy = Eval(\mathbf{pos})$
      if $energy > bestEnergy$ **then** ▷ update best-found layout
        $bestEnergy = energy$
        $bestPosition = (x_s, y_s)$
      end if
    end for
    $pos(i) = bestPosition$
  end for
  return pos ▷ Final Layout
end procedure

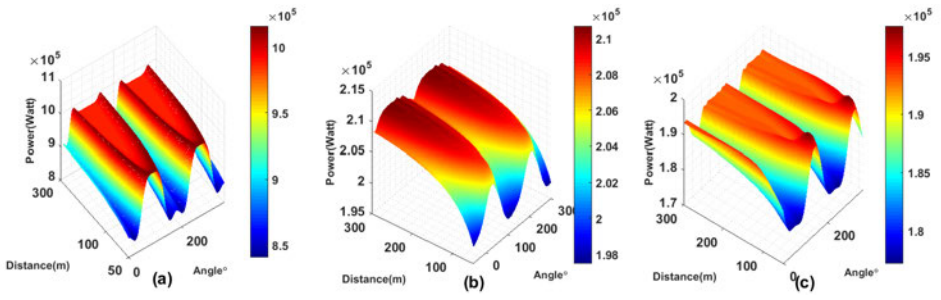

FIGURE 2.14: A 3D power landscape, for relative angle and distance between two buoys based on the simplified irregular wave model (a) and two real wave scenarios: Sydney (b) and Perth (c). Note that there are ridges in the power landscape corresponding to areas of constructive interference. The Improved Smart Local Search algorithm variants, described in this paper, exploit this local landscape when placing buoys.

TABLE 2.5: Summary of the search methods used in this paper. All methods are given the same computational budget. Parallelism can be expressed as per-individual or per-frequency depending on the number of individuals in the population from section.2.2.8.

Abbreviation	Parallelism	Description
R-S	per-frequency	Random Search
$\text{PE}_{50\,\mu}$	per-individual	Partial Evaluation[44], all frequencies (PE-FULL), population $\mu \in \{10, 50, 100\}$
$\text{PE}_{f\,\mu}$	per-individual	Partial Evaluation [44], partial frequencies, $f \in \{1, 4, 16\}$, $\mu \in \{10, 50, 100\}$
CMA-ES_{PF}	per-frequency	All settings are like CMA-ES
CMA-ES_{PF} (2+2)	per-frequency	All settings are based on [39]
TDA	per-individual	Algorithm for optimizing wind turbine placement from [93]
CMA-ES	per-individual	CMA-ES[89] all dimensions, $\mu =' 4 + int(3 \times log(D))$ndim , $\sigma = 0.17 \times size$
CMA-ES (2+2)	per-individual	Setup for CMA-ES from [39], $\sigma = 20m$
$\text{DE}_{P_{cr}}$	per-individual	Differential evolution [94], $\mu = 50$, $F = 0.5$, $P_{cr} \in \{0.3, 0.5, 0.7, 0.9\}$
bGA	per-individual	binary Genetic Algorithm, All settings are based on [36]
Improved DE	per-individual	Improved Differential evolution , All settings are based on [41]
LSHADE-EpSin	per-individual	An ensemble sinusoidal parameter adaptation incorporated with L-SHADE [118, 122]. $\mu = \mu_{Max} = 50$, $\mu_{Min} = 4$, $memory_{size} = 5$, $\mu F = \mu CR = 0.5$
IPOP-CMAES	per-individual	a restart CMA evolution strategy, where the population size is increased for each restart (IPOP) [119, 123]. $\mu = 20$, $\lambda = floor(\frac{\mu}{2})$, $IncPopSize = 1.5$, $\sigma = 0.3 \times size$
HCLPSO	per-individual	Heterogeneous comprehensive learning particle swarm optimisation. $\mu = 50$, $\mu_{G1} = 15$, other parameters are based on [120, 121]
$1{+}1\text{EA}_{\sigma}$	per-frequency	1+1EA(all dimensions), mutation step size with $\sigma \in 3, 10, 30$(m)
$1{+}1\text{EA}_{s}$	per-frequency	1+1EA (all dimensions) with uniform mutation in range $[0, s]$ with $s = 30m$ from [39]
$1{+}1\text{EA}_{\text{Linear}}$	per-frequency	1+1EA (all dimensions) with linearly decaying mutation step size [1]
$1{+}1\text{EA}_{1/5}$	per-frequency	1+1EA (all dimensions) with adaptive step size [124]
Fuzzy-1+1EA	per-frequency	1+1EA (all dimensions) with fuzzy adaptive mutation step size
Iterative-1+1EA	per-frequency	Iterative local search - buoys are placed in sequence using best of local neighborhood search [1]
$\text{LS-NM}_{\text{allDims}}$	per-frequency	Local search + Nelder-Mead search in all Dimensions [1]
NM_Norm_{2D}	per-frequency	Buoys placed in sequence using Nelder-Mead search, Initial placement normally distributed from last buoy position [1]
NM_Unif_{2D}	per-frequency	Buoys placed randomly and then refined using Nelder-Mead Initial placement uniformly distributed from last buoy position [1]
LS_1-NM_{2D}	per-frequency	Local Sampling + Nelder Mead search. Buoys placed at random offset from previous buoy and placement refined by Nelder-Mead search. [1]
LS_3-NM_{2D}	per-frequency	Iterative local search + Nelder-Mead search. Placements sampled at 3 random offsets from previous location, best placement used as starting point for Nelder-Mead search. [1]
SLS	per-frequency	Providing the two-buoy power landscape+ Extracting a proper domain of the distances and the angles +Iterative local search +Smart Mutation;Uniform distribution, $N1_S = 15$ samples, step= $rand(R', \text{Buo}y_{Distance} + \kappa_2(20m))$. 4.5 folds faster than the best method of the prior study [1] for 16-buoy layout.
SLS-NM	per-frequency	Smart Local Search with three samples of the mutation+ Nelder-Mead search, $N_{NM} = 20$
ISLS	per-frequency	Improved Smart Local Search : Creating a more accurate knowledge-based surrogate power model, placing a new buoy: the initial sequential N_{sb}-buoy number $\sigma = R'$ and for next buoys $\sigma = 2 \times R'$: Mutating:$N1_S$ samples for initial sequential N_{sb}-buoy number, and for next buoys $N2_S = 20$ samples, $step = rand(R', \text{Buo}y_{Distance}+10\text{m})$. 60% faster than L$S_3$-N$M_{2D}$in [1] for 16-buoy layout.
ISLS-NM	per-frequency	Improved Smart Local Search ($N1_S$ samples) for the initial sequential N_{sb}-buoy number and for last buoys $N2_S = 3$ samples + Nelder-Mead search, $N_{NM} = 20$
ISL$S_{(II)}$-F	per-frequency	Improved Smart Local Search (for initial sequential N_{sb}-buoy number) ($N_{SF} = 3$ samples)+ Applying SQP (for finding the furthest point of the area based on the layout position). 20 times faster than the best method of the prior work [1] for 16-buoy layout.
ISL$S_{(II)}$-NM	per-frequency	Improved Smart Local Search(II) (for initial sequential N_{sb}-buoy number) 3 samples + Nelder-Mead Search.
ISL$S_{(II)}$-SQP	per-frequency	Improved Smart Local Search(II) (for initial sequential N_{sb}-buoy number) 3 samples + Sequential Quadratic Programming Search.
ISL$S_{(II)}$-AS	per-frequency	Improved Smart Local Search(II) for initial sequential N_{sb}-buoy number) 3 samples + Active-Set Search.
ISL$S_{(II)}$-IP	per-frequency	Improved Smart Local Search(II) for initial sequential N_{sb}-buoy number) 3 samples + Interior-Point Search.

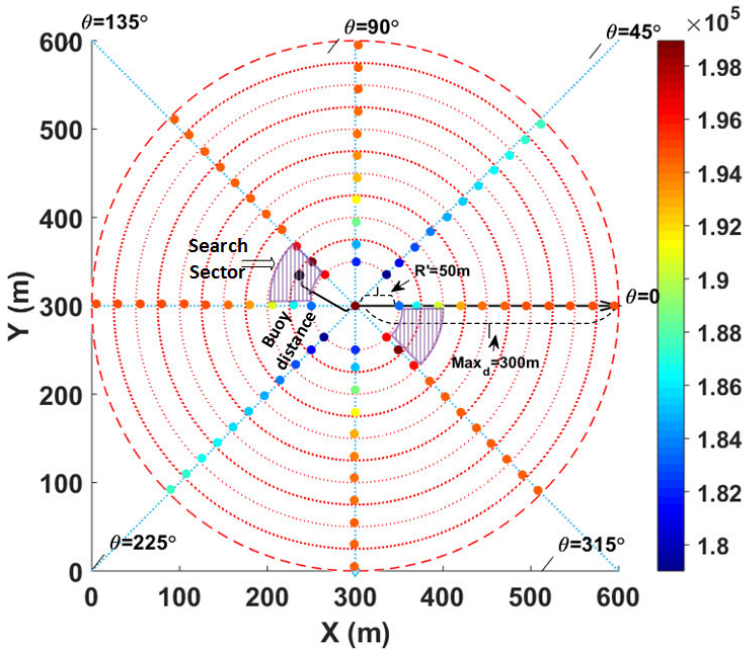

FIGURE 2.15: The 2D power landscape of two-buoy array based on the Adelaide wave scenario.

For the placement of each buoy, a local search makes $N1_S$ samples, uniformly distributed in the search sectors (subject to boundary constraints). These samples are assessed with the full model, which calculates *all* buoy interactions in the current array. From these $N1_S$ samples, the best buoy location is selected. Algorithm 2.4 describes the SLS method.

2.2.8.1.1 Tuning SLS parameter

One of the most important parameters is the sample number ($N1_S$) of the local search. It is recommended that $N1_S$ should be tuned for obtaining a great trade-off between the produced power of the wave farm and the optimisation method runtime. Analysis of the experiments has shown that 15 samples have been sufficient to improve upon the initial placement with a probability of 99% with an expected improvement of power production almost identical to that of a much larger number of samples.

2.2.8.2 Smart Local Search + Nelder-Mead (SLS-NM)

Smart Local Search + Nelder-Mead (SLS-NM) explores the same search sectors as the SLS algorithm defined above. The SLS-NM algorithm differs in that it takes only three random samples from the search sectors and uses the best of these as the start point for a Nelder-Mead (NM) simplex search process. The NM search process can robustly move to a local optimum from its starting point.

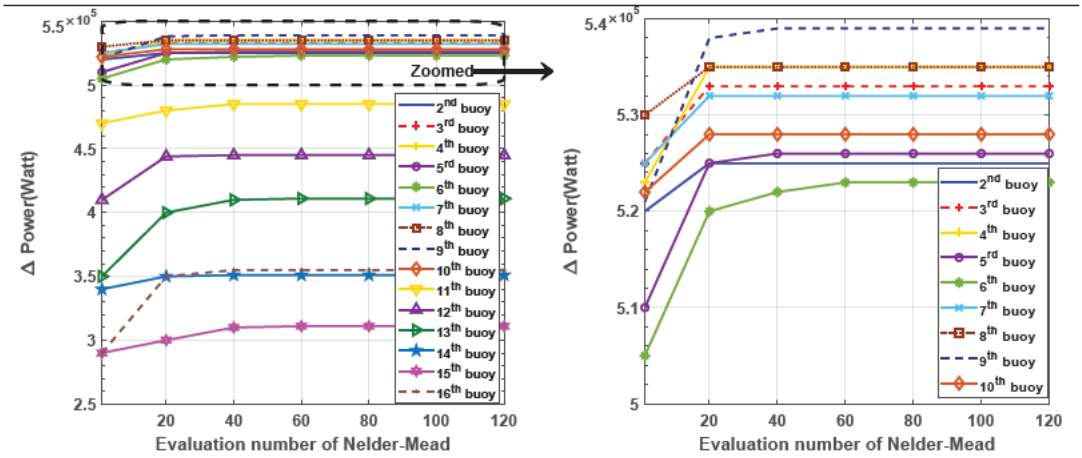

FIGURE 2.16: Tuning the evaluation number of the Nelder-Mean in the SLS-NM performance (16-buoy) for the Perth wave model.

2.2.8.2.1 Tuning the SLS-NM parameter

In order to fit within the computational budget, the number of steps (evaluations) in the NM search is limited to a maximum of 20. To verify that this number of steps is adequate a longer experiment with 120 evaluations was run for each NM search, and we found that, in most cases, the improvement from the extra evaluations was less than 2% of the power output for that buoy. Moreover, when there was a significant improvement, it happened only after a prohibitively large number of evaluations. These experiments can be seen in Figure 2.16.

2.2.8.3 Improved Smart Local Search (ISLS)

From the analysis of experimental runs using the SLS methods, four observations can be made.

1. The search sector containing the best-sampled positions was always in the direction of the opposite side of the farm.

2. The best samples from the SLS-NM method came from a sector which was narrower in angular extent but longer in radial extent.

3. The search space for placing the next buoy becomes much harder after hitting the top boundary of the farm. This is due to occlusion from the front row of buoys for subsequent buoy placement.

4. The placement of the first buoy in the centre of the bottom boundary of the farm can be sub-optimal if the best angle for the alignment of buoys in the surrogate

power landscape causes some buoys to encounter the left or right boundary of the farm.

In response to these observations, a refined search method called Improved Smart Local Search (ISLS) was designed. This search method addresses the first observation above by only sampling the search sector in the direction of the current opposite boundary of the farm (upwards in the implementation). ISLS addresses the second observation by allowing the user to set the angular and radial extent of the search sector for the wave scenario. The third observation is addressed by reducing the number of samples used when placing buoys on the first sweep to the opposite farm boundary (phase 1) and running more samples to place subsequent buoys (phase 2). Lastly, the fourth observation is addressed by placing the first buoy in the left corner of the landscape if the best angle is between zero and 90 degrees and in the right corner otherwise.

2.2.8.3.1 Tuning the ISLS parameter

We base the decision on the number of samples to use in the first sweep of buoy placement (phase 1) and then for the placement of subsequent buoys (phase 2) on empirical studies on the impact of different numbers of samples depending on the different wave scenarios.

To illustrate the findings of this process, Figure 2.17(a) shows the average power gain from sampling for the first 12 buoys for ten runs in the Perth wave scenario. The red and blue vertical lines, respectively, indicate the average power gained after three and ten samples. For the experiments leading to this Figure, the buoys were placed according to the best result obtained after 20 samples. The Figure indicates that for the placement of the first 12 buoys (phase 1) the curves flatten after ten samples, with buoy three showing the largest gain, after ten samples, of 0.018%. This indicates that we gain little from sampling the real power landscape beyond this point. Thus for phase 1, 10 samples were allocated for each buoy placement. For the placement of the last four buoys (phase 2), as illustrated in Figure 2.17(b) the gain from sampling is steeper and, for buoy 16, the improvement between 10 and 20 samples is 0.032%. A second notable feature of Figure 2.17(a) is that the power curves for some buoys are vertically displaced relative to others. This indicates that for some buoy placements the power landscape is more challenging. In general, we have observed that the displacement of these sampling curves for later buoys depends on the placement of previous buoys. In some experiments, we have observed that this dependence on placement history can even lead to some minor anomalies in search behaviour where sampling *less* for earlier buoy placements appears to make the search landscape slightly easier for subsequent buoy placements. However, we have observed for all wave regimes that the best *median* performance for wave farms

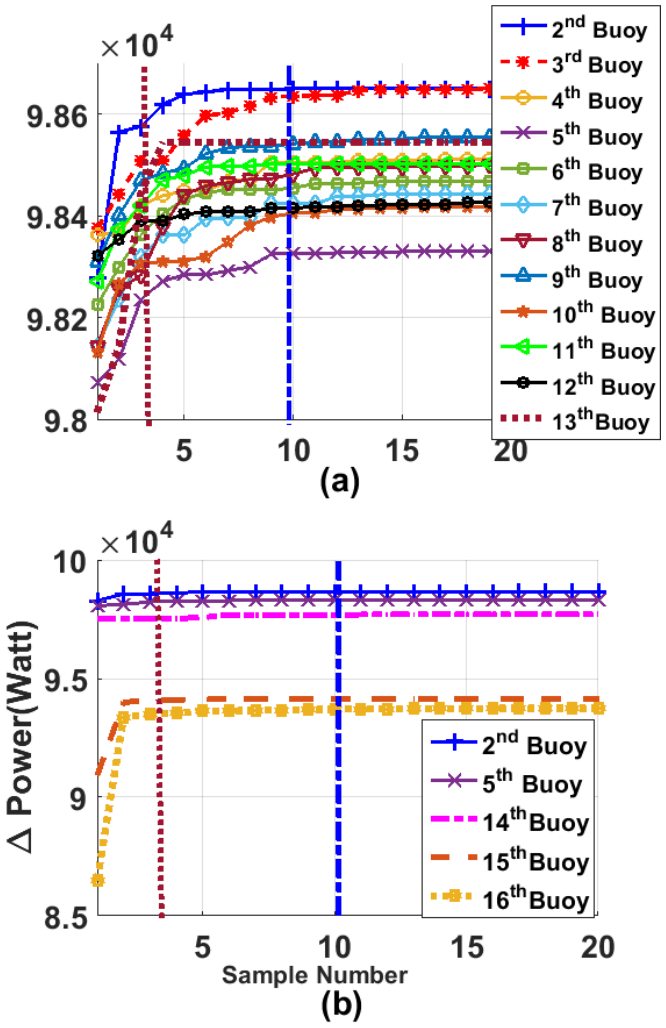

FIGURE 2.17: The impact of sample number on the ISLS performance (16-buoy) for the Perth wave model. The red vertical line shows 3-sample.

is obtained by employing as many samples as the time budget allows in both phases of the search process.

2.2.8.4 Improved Smart Local Search Nelder-Mead (ISLS-NM)

As previously noted, due to occlusion by other buoys, the power landscape for phase 2 of the search is different from that for phase 1. This means that, for phase 2 buoy placements, the search sector from the surrogate power landscape might not contain the best location for the placement of the next buoy. To search more broadly a variant of ISLS, called ISLS-Nelder-Mead (ISLS-NM) was created. This variant has an identical phase 1 search to ISLS, but in phase 2 it performs a local search with three samples followed by 20 samples of Nelder-Mead search starting at the best of these three sampled

locations. In almost all experimental runs, these 20 samples were enough to converge to a point where the step size is less than 1% of the total power output for that buoy.

2.2.8.5 Improved Smart Local Search-II (ISLS-II)

One drawback of ISLS is the need for the user to define the angular and radial extent of the search sector for a wave regime through observation of the surrogate power landscape. In ISLS$_{II}$ this process is automated by, first performing fine-grained sampling of the 2-buoy power landscape. The readings from this sampling are arranged into a table – with columns representing angular increments and rows representing radial distance increments. The maximum power value is then located. The search sector is then defined by the area between this highest and second-highest sample – subject to a maximum radial distance constraint of 283m that comes from the maximum distance between two buoys based on the boundary constraints. As with ISLS, this method tends to produce a longer and narrower search sector than SLS.

We have implemented five variants of ISLS$_{II}$. Apart from the determination of the search sector, each of these variants has an identical first phase to ISLS using the ten-sample randomized local search within the search sector that is defined by a surrogate landscape. Each of these variants, in their second phase, still begins by identifying the best of three random sample positions in the search sector relative to the previously placed buoy. The variants differ, however, in the type of search that proceeds from each sample point. In the following, we describe these search variants in turn.

2.2.8.5.1 ISLS-II + Active Set (ISLS$_{II}$-AS)

Here, we refine the placement of each buoy with N_{NM} evaluations (which is tuned in Section 2.2.8.2.1) of the Active-Set [125] search method. This method identifies the set of boundary constraints that are relevant to the current state of search and concentrates search close to these. This method is able to take large steps through the search space, thus allowing for quick coverage of the search area. This variant is described in Algorithm 2.5.

Algorithm 2.5 *ISLS(II) − AS*

procedure IMPROVED SMART LOCAL SEARCH(II) + ACTIVE-SET
First Step:
  WMD = Load (wave model data)
  S=Sur-Power(WMD) ▷ Generate surrogate power model in Section 2.2.8.1
Initialization
  $size = \Omega$ ▷ Farm size
  $SafeDis = 50$ ▷ safe distance between buoys
  $N1_S = 10$, $N2_S = 3$ ▷ tuned and described in Section 2.2.8.3.1
  $N_{NM} = 20$ ▷ tuned Nelder-Mead evaluation number in Section 2.2.8.2
  $pos = [(x_1, y_1), \ldots, (x_N, y_N)] = \perp$ ▷ WECs positions
  if $0° < bestLocalAngle < 90°$ **then** $pos(1) = (0, 0)$
  else $pos(1) = (size, 0)$
  end if
  $BN_{row} = round(size/\cos(bestLocalAngle))$ ▷ buoy number in first row
  $buoyNum = 2$
Second Step: Search strategy
  Phase 1 ▷ described in Section 2.2.8.3
  update search sector S
  while $bottomYBoundary(S) < size$ **do**
    $bestEnergy = 0$, $bestPosition = (0, 0)$
    for j in $[1, .., N1_S]$ **do** ▷ $N1_S$ random samples
      $(x_s, y_s) = U(S)$ ▷ sample sector
      $pos(i) = (x_s, y_s)$
      $energy = Eval(\mathbf{pos})$
      if $energy > bestEnergy$ **then** ▷ update best-found layout
        $bestEnergy = energy$
        $bestPosition = (x_s, y_s)$
      end if
    end for
    $pos(i) = bestPosition$
    $buoyNum = buoyNum + 1$
    update search sector S
  end while
  Phase 2 ▷ placing and optimising the subsequent buoys (Section 2.2.8.3)
  for i in $[buoyNum, .., N]$ **do**
    $bestEnergy = 0$, $bestPosition = (0, 0)$
    for j in $[1, ..N2_S]$ **do** ▷ $N2_S$ random samples
      $(x_s, y_s) = U(S)$ ▷ sample sector
      $pos(i) = (x_s, y_s)$
      $energy = Eval(\mathbf{pos})$
      if $energy > bestEnergy$ **then** ▷ update best-found layout
        $bestEnergy = energy$
        $bestPosition = (x_s, y_s)$
      end if
    end for
Third Step ▷ enhancing buoys postion by local search (Section 2.2.8.5)
    if $(bouyNum \geq BN_{row}) \vee (bestPosition \geq size - SafeDis)$ **then**
      $bestPosition = ActiveSearch(bestPosition, N_{NM})$
      $pos(i) = bestPosition$
    end if
    update search sector S
  end for
  return pos ▷ Final Layout
end procedure

2.2.8.5.2 ISLS-II + Sequential Quadratic Programming(ISLS$_{II}$-SQP)

This approach refines the placement of each buoy by performing Sequential Quadratic Programming (SQP) [126]. This search method employs Newton's method when the search is away from boundary constraints and reverts to constrained search when boundaries are encountered.

2.2.8.5.3 ISLS-II + Fast placement (ISLS$_{II}$-F)

We observed in the earlier ISLS-NM that the phase 2 buoy placements tended to reside on the lee-side behind the front row of buoys, with these buoys finishing far from each other. Informed by this observation the ISLS$_{II}$-F algorithm uses 20 iterations of SQP search to place each buoy, one at a time, at a position that is the maximum Euclidean distance from the previously placed buoys. Note that using distance as a proxy function makes this method very fast compared to other variants.

2.2.8.5.4 ISLS-II + Nelder-Mead (ISLS$_{II}$-NM)

This approach is similar to the earlier ISLS-NM, but it applies 20 iterations of the Nelder-Mead algorithm to the placement of each buoy.

2.2.8.5.5 ISLS-II + Interior point algorithm (ISLS$_{II}$-IP)

This algorithm refines each buoy position using the interior-point (IP) algorithm [127] for constrained search. This method is similar to other active-set methods above except that boundaries are approximated using barrier functions which allow search near constraint boundaries rather than on constraint boundaries. This concludes the description of the different buoy placement algorithms explored in this paper. The next section presents detailed results comparing the performance of these algorithms.

2.2.9 Experimental Results

This section presents the results of the experiments comparing the performance of the algorithms described above on the placement of buoys under the different wave scenarios.

One challenge for the approaches is that the farm's dimensions do not allow for all 16 buoys to be placed in a single line. Another challenge is that, because of interactions, the cost of the evaluative model scales quadratically with the number of buoys. This

means the number of full evaluations of a 16-buoy layout within a 3-day time budget is limited to a few thousand evaluations. This limited computational budget heavily favours problem-specific search algorithms that place one buoy at a time. This is illustrated for the simplified irregular wave scenario in Figure. 2.18. This box-and-whiskers plot demonstrates the best-achieved power for a 16-buoy layout of each search framework. The new methods described in this paper are given in the last nine columns of the Figure. The improved performance of these new search methods is quantified in Table 2.6. From

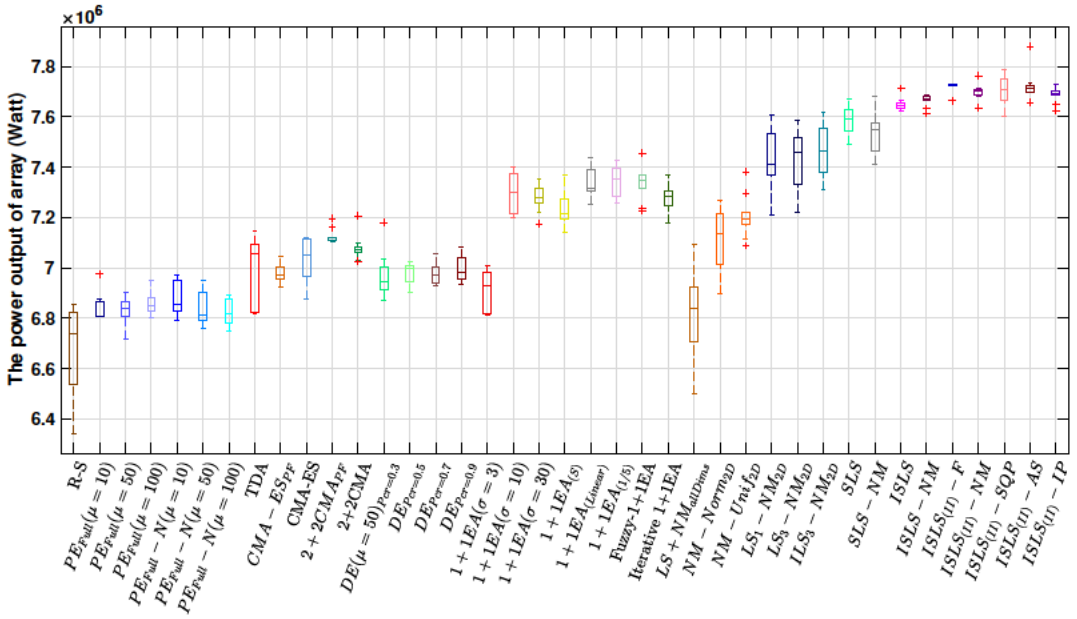

FIGURE 2.18: The performance comparison of the best layout per run for the proposed heuristic approaches for optimizing the position of 16 buoys under the simplified wave scenario. ISLS$_{(II)}$-AS produces 4% more power compared with the best result in [1].

these results, it can be seen that the most reliably performing placement algorithms are the variants of ISLS$_{(II)}$, with a run of ISLS$_{(II)}$-AS producing the best maximum performance with a maximum layout power of 7878917 Watts. In terms of statistical significance, the Active-Set, Nelder-Mead, and Fast-Search variants of ISLS$_{(II)}$ perform significantly better than the non ISLS$_{(II)}$ search methods with $p < 0.025$ over the ten trial runs using a one-tailed Wilcoxon ranked-sum test. Moreover, the other ISLS$_{(II)}$ variants, SQP, and Interior-Point, perform better than all but the ISLS-NM search method.

In Figure 2.19 each curve visualizes the evolution of the average power produced by the best individual layout for each method over three days of computational budget. The search parameters used by most algorithms are tuned to take advantage of most of the search budget. The exceptions to this are the plain SLS and ISLS algorithms where the local sampling converges early and the 3-sample and 10 sample versions of ISLS$_{(II)}$-F which place the last buoys very quickly with a fast distance-based proxy function. From the plots, it can be seen that all the other variants of ISLS$_{(II)}$ converge to a very similar

TABLE 2.6: Performance comparison of various heuristics for 16-buoy layouts for the simplified irregular wave model (10 run each).

Methods	**PE-Full (Uniform)**			**PE-Full(Normal)**			**DE**		
	$\mu = 10$	$\mu = 50$	$\mu = 100$	$\mu = 10$	$\mu = 50$	$\mu = 100$	$P_{cr} = 0.3$	$P_{cr} = 0.5$	$Pcr = 0.9$
Max	6974948	6900024	6952017	6957388	6948746	6892210	7179681	7025873	7079962
Median	6859475	6839557	6851342	6853987	6812866	6816282	6944795	6999356	6983523
Mean	6856337	6821864	6860037	6869586	6837972	6822553	6971231	6981195	6994172
Std	48701	61880	50377	61153	70048	52911	89244	41931	48943
Methods	**1+1EA**						NM_{2D}		**Iter-(1+1)EA**
	Mu-s=3	Mu-s=10	Mu-s=30	Linear	1/5 rule	Fuzzy	Uniform	Normal	
Max	7008380	7402584	7351112	7437481	7425665	7454922	7380318	7267242	7370972
Median	6927230	7297465	7278120	7317408	7354589	7348676	7193110	7136712	7354589
Mean	6908203	7292035	7275118	7330286	7343858	7335624	7205098	7108693	7274989
Std	83157	77794	51745	60803	59690	67061	83944	116380	54380
Methods	**L**S_1**-N**M_{2D}	**TDA**	**CMA-ES**	**R-S**	**1+1E**A_S	**(2+2)CMA-ES**	**L**S_3**-N**M_{2D}	**L**S_1**-N**$M_{allDims}$	**SLS**
Max	7608600	7148655	7118996	6825723	7370389	7205956	7587758	7094642	7669439
Median	7409029	7057564	7053351	6658523	7214263	7073295	7459614	6839911	7590039
Mean	7427027	7005873	7038352	6676831	7236977	7080011	7426742	6823836	7587410
Std	129780	133977	84859	63883	67406	49771	123603	198512	52538
Methods	**SLS+NM**	**ISLS**	**ISLS-NM**	**ISL**S_{II}**-F**	**ISL**S_{II}**-NM**	**ISL**S_{II}**-SQP**	**ISL**S_{II}**-AS**	**IL**S_3**-N**M_{2D}	**ISL**S_{II}**-IP**
Max	7680161	7713744	7685633	7730799	7763249	7790679	**7878917**	7619404	7730844
Median	7548235	7647335	7670401	7723825	7702161	7706788	7714011	7463928	7694381
Mean	7537582	7651360	7666201	7721954	7698639	7701115	**7724370**	7469493	7688422
Std	81727	25635	23437	9773	31790	60012	58341	102760	30453

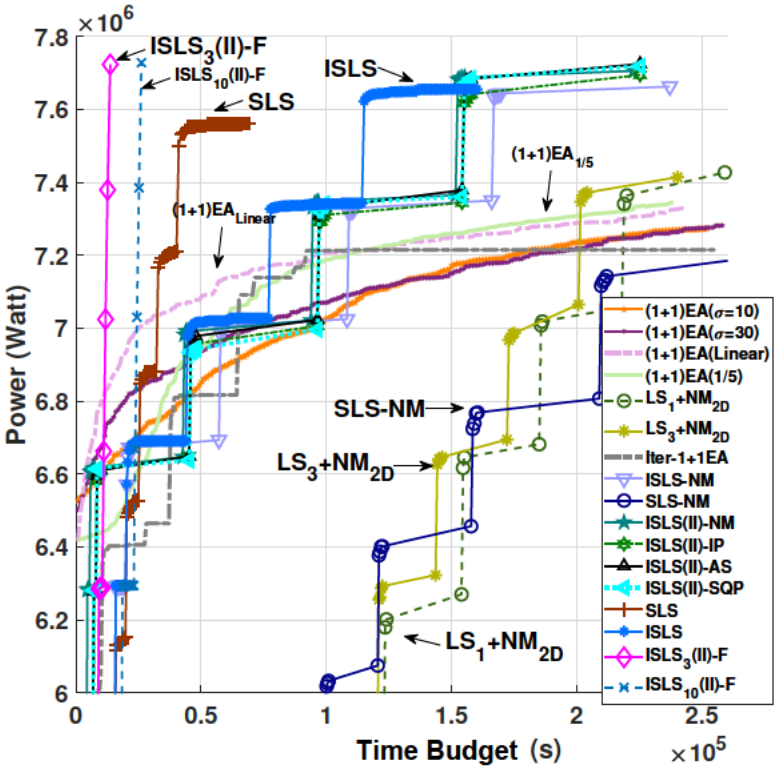

FIGURE 2.19: The comparison of the average convergence rate of the proposed methods with the work [1] for 16-buoy layout over 72 hours (simplified irregular wave model).

level of performance in a very similar time to each other. The average performance of all of $\text{ISLS}_{(II)}$ variants is above all others. Overall, the best performing variant: $\text{ISLS}_{(II)}$-AS extracts 4% more power than the previous best published algorithm: $\text{LS}_1{+}\text{NM}_{2D}$. This small difference in performance equates to approximately $175,000 US dollars in extra annual avenue for the wave farm. [5]

2.2.9.1 Real Wave Scenarios

In this section, we first present a quick summary of the results for 4-buoy layouts before progressing to the much more challenging 16-buoy layouts. As mentioned previously for the 16 buoy-layout problems the full computational budget of 3 days on 12 CPU cores was used wherever possible. In order to make a fair comparison, an equivalent number of evaluations was used for the 4-buoy layout problem. To see how algorithm performance translates to real wave scenarios we ran ten trials each for 11 selected search methods

[5] Based on an annualized average wholesale price of US $65 per MWh average for the Australian market in 2017/18.

TABLE 2.7: Summary of the best 4-buoy layouts per-experiment (Power (Watt)) for the real wave scenarios .

Methods	DE [94]	CMA-ES [89]	LS_3-NM_{2D}[1]	IDE [41]	bGA [36]	2+2CMA-ES [39]	LSHADE-EpSin [118]	IPOP-CMAES [119][120]	HCLPSO	ISL$S_{(II)}$-AS
Sydney wave model										
Max	412667	412705	412294	412683	**413061**	412796	412799	412736	412802	411291
Median	412557	412488	411069	412529	**413028**	412424	412799	412623	412799	410094
Mean	412580	412477	410839	412560	**413004**	412350	412764	412638	412796	409376
Std	63	140	1184	74	54	395	95	52	12	1534
Perth wave model										
Max	398844	**399607**	396759	**399607**	397822	399604	**399607**	399603	**399607**	399476
Median	395898	**399607**	392753	**399607**	397822	399601	**399607**	399601	**399607**	399466
Mean	396615	399117	391361	**399607**	397822	399600	**399607**	399600	399606	399467
Std	1415	1033	5543	0.003	0.00	1.80	0.098	2.90	0.74	3.65
Adelaide wave model										
Max	399431	**402278**	401858	**402278**	402072	402276	**402278**	402275	**402278**	402206
Median	397176	**402278**	398352	**402278**	402072	402274	**402278**	402271	**402278**	402186
Mean	395620	402073	396106	**402278**	402072	402273	402277	402271	402270	402189
Std	4271	709	6685	0.025	0.00	2.42	4.6	3.26	22	6.17
Tasmania wave model										
Max	1093468	**1094611**	1094524	**1094611**	1072416	1094605	**1094611**	1094610	**1094611**	1094530
Median	1090833	**1094611**	1079619	**1094611**	1072379	1094596	**1094611**	1094606	**1094611**	1094524
Mean	1090734	**1094611**	1079429	**1094611**	1072190	1094597	**1094611**	1094605	1094604	1094523
Std	1985	0.0072	10432	0.008	471	4.078	0.097	3.25	29	5.50

(DE [94], CMA-ES [89], LS_3NM_{2D} [1], IDE [41], bGA [36], 2+2CMA-ES [39], LSHADE-EpSin [118, 122], IPOP-CMAES [119], HCLPSO [120], ISL$S_{(II)}$-F and ISL$S_{(II)}$-AS) for 4 and 16-buoy layouts on the Sydney, Perth, Adelaide and Tasmania wave scenarios.

2.2.9.1.1 4-buoy layout results

Table 2.7 summarises the results of the ten best-performing search methods on the 4-buoy layout problem in the four real wave scenarios. The output for the search method with the best maximum performance in each wave scenario is highlighted in bold. This table shows that the best results for each of the ten methods shown are within 1% of each other in terms of raw performance. All methodologies are able to produce layouts with a q-factor more than one except arrays in the Sydney wave model. The q-factor can be interpreted as an optimisation rate, it shows the impact of constructive interference compared with destructive ones. Thus a q-factor is greater than one; it is associated with the wave farm of devices is generating more power than the devices would work individually. It is also clear from these results that the global CMA-ES, IDE, LSHADE-EpSin and HCLPSO methods consistently have the best performance across all wave scenarios (except in the Sydney wave scenario) and the performance variance is quite small.

The 4-buoy layouts produced are shown in Figure 2.20. The buoys are coloured based on their power output. From these layouts, we can observe that, except for the Sydney wave scenario, all buoys form a row with the spacing and orientation determined by the wave environment. The orientation of this row (in the Perth, Adelaide and Tasmania scenarios) is aligned to the norm of the predominate wave direction for each scenario. We can also see that the middle two buoys in these three layouts also produce slightly more energy than the outer buoys. This is due to constructive interactions between

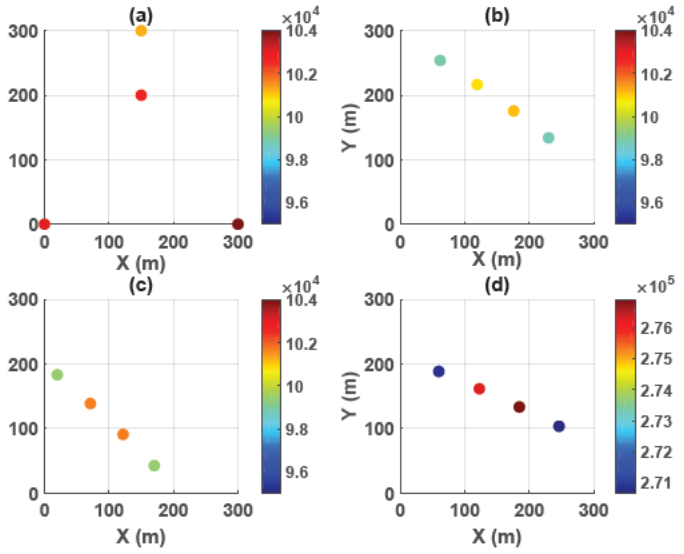

FIGURE 2.20: The best 4-buoy layouts of the real wave scenario by bGA (a) (Sydney:Power=413060 (Watt), q-factor= 0.976), (b) (Perth: Power= 399607 (Watt), q-factor= 1.0366), (c) (Adelaide: Power= 402278 (Watt), q-factor= 1.036) and (d) (Tasmania: Power= 1094611 (Watt), q-factor = 1.0334).

buoys. The layout for the Sydney wave scenario differs markedly in its spacing and orientation. The Sydney wave environment is more varied in terms of wave direction and, thus, opportunities to exploit constructive interactions through a static layout are much reduced. As a result, the buoys in the produced layouts are rather spread out, indicating that the algorithms attempted to minimise destructive interactions.

2.2.9.1.2 16-buoy layout results

Compared to the 4-buoy layout problem, the 16-buoy layout problem is challenging in terms of both problem constraints and computational budget. The farm area for the 16-buoy layout is larger than that for the 4-buoy layout, but not so large as to allow all 16 buoys to be placed in a single line in any wave scenario. The ISLS$_{(II)}$-AS performed significantly better than the other ten methods with $p < 0.025$ over the ten trial runs using a one-tailed Wilcoxon ranked-sum test. Figure 2.21 summarises the results from these runs.

These results are reflected in the clear margin between the performance of ISLS$_{(II)}$-AS and the other methods. It can be seen that ISLS$_{(II)}$-F varies significantly in its performance between scenarios with much poorer performance for Sydney. This perhaps reflects on diminished usefulness for the distance-based proxy function in this complex wave environment. To sum up, in all cases, ISL$S_{(II)}$-AS provides the best mean and maximal power output among the 11 compared algorithms statistically. A comparison of the ISLS$_{(II)}$-AS convergence rate with ten other methods for four real wave models can

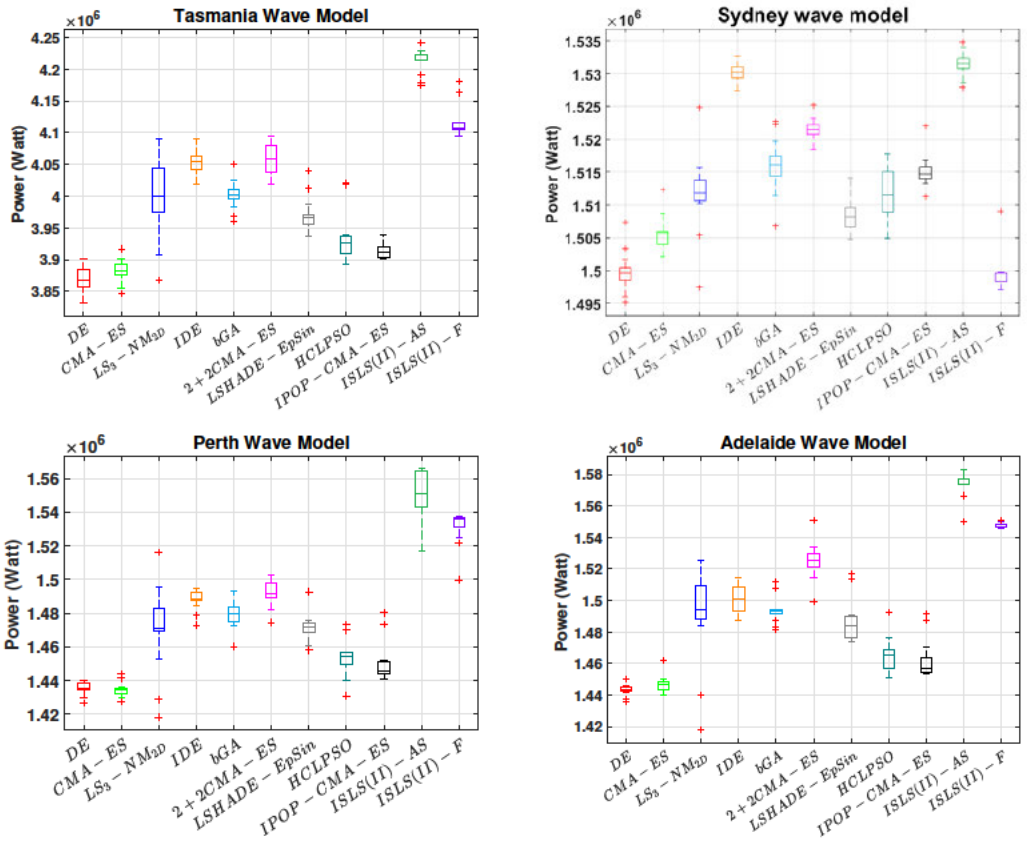

FIGURE 2.21: optimisation results of the proposed algorithms based on the best layout per run for 16 buoys and four real wave scenarios.((a) Tasmania wave scenario, (b) Sydney wave scenario , (c) Perth wave scenario , (d) Adelaide wave scenario)

be seen in Figure 2.22. The figure illustrates that ISLS$_{(II)}$-F has the highest convergence speed; however, the average quality of proposed 16-buoy layouts of ISLS$_{(II)}$-AS can be considerably better.

The best 16-buoy layouts produced by ISLS$_{(II)}$-AS, 2+2CMA-ES, IDE, bGA and $LS_3 - NM_{2D}$ in each of the four wave scenarios are shown in Figure 2.23. In the best-found 16-buoy layouts of ISLS$_{(II)}$-AS for Perth, Adelaide and Tasmania the layouts are similar with the buoys being oriented in a line roughly normal to the prevailing wave direction with buoys placed in phase 1 (lower numbers) forming a bottom row and the buoys placed in phase 2 (higher numbers) behind these. It can be seen that fewer buoys are placed in phase 1 for Tasmania than for Adelaide. The number of buoys placed in this phase appears to depend on how well the first row is aligned with the diagonal of the farm area.

Again, the Sydney layout is very different, with a row for phase 1 oriented to the east and the other buoys being placed at large distances from the others. We have observed this pattern for a number of Sydney runs where the best layouts tended to contain widely dispersed buoys to minimize destructive interference.

TABLE 2.8: Summary of the best achieved 16-buoy layouts power(Watt) per experiment for real wave scenarios

Methods	DE [94]	CMA-ES [89]	LS_3-NM_{2D}[1]	IDE [41]	bGA [36]	2+2CMA-ES [39]	LSHADE-EpSin [118]	IPOP-CMAES [119]	HCLPSO [120]	$ISLS_{(II)}$-F	$ISLS_{(II)}$-AS
Sydney wave model											
Max	1507235	1512337	1524915	1532776	1522817	1525243	1513655	1522088	1517860	1509037	**1534883**
Median	1499663	1505254	1512190	1529766	1516803	1521448	1507975	1514414	1511590	1498963	**1531785**
Mean	1499901	1505705	1511814	1530250	1516061	1521451	1508231	1515484	1511517	1499663	**1531491**
Std	3207	2770	6421	1961	4617	1972	2826	2814	4020	3440	2339
Perth wave model											
Max	1440344	1443893	1516098	1495206	1493394	1502466	1492753	1480307	1473388	1537788	**1565836**
Median	1436441	1433949	1472851	1489437	1479940	1494378	1470401	1445566	1455511	1529076	**1550877**
Mean	1435539	1434333	1470658	1488187	1479265	1491837	1471622	1450389	1453304	1528225	**1549409**
Std	4171	4626	26990	6642	8991	8713	8940	11709	11649	9016	16920
Adelaide wave model											
Max	1449967	1461741	1525144	1514816	1511594	1550701	1516631	1491551	1492807	1551102	**1583052**
Median	1444455	1445241	1497211	1496515	1492807	1527727	1484035	1457667	1462314	1547351	**1578797**
Mean	1443442	1446621	1477342	1500835	1493705	1525274	1487415	1463795	1465523	1548027	**1573476**
Std	3757	5879	57675	10207	10190	13695	14519	13712	12914	1796	11694
Tasmania wave model											
Max	3901664	3916983	4090733	4089215	4050476	4093637	4039171	3939440	4020283	4180781	**4241838**
Median	3868558	3886093	4005319	4052658	4008556	4066558	3964938	3908213	3927356	4105700	**4218894**
Mean	3867923	3882930	3999507	4053854	4002820	4058384	3971415	3912734	3934030	4115758	**4213652**
Std	22588	19705	69758	25983	27647	29421	26003	12368	37715	26696	21775

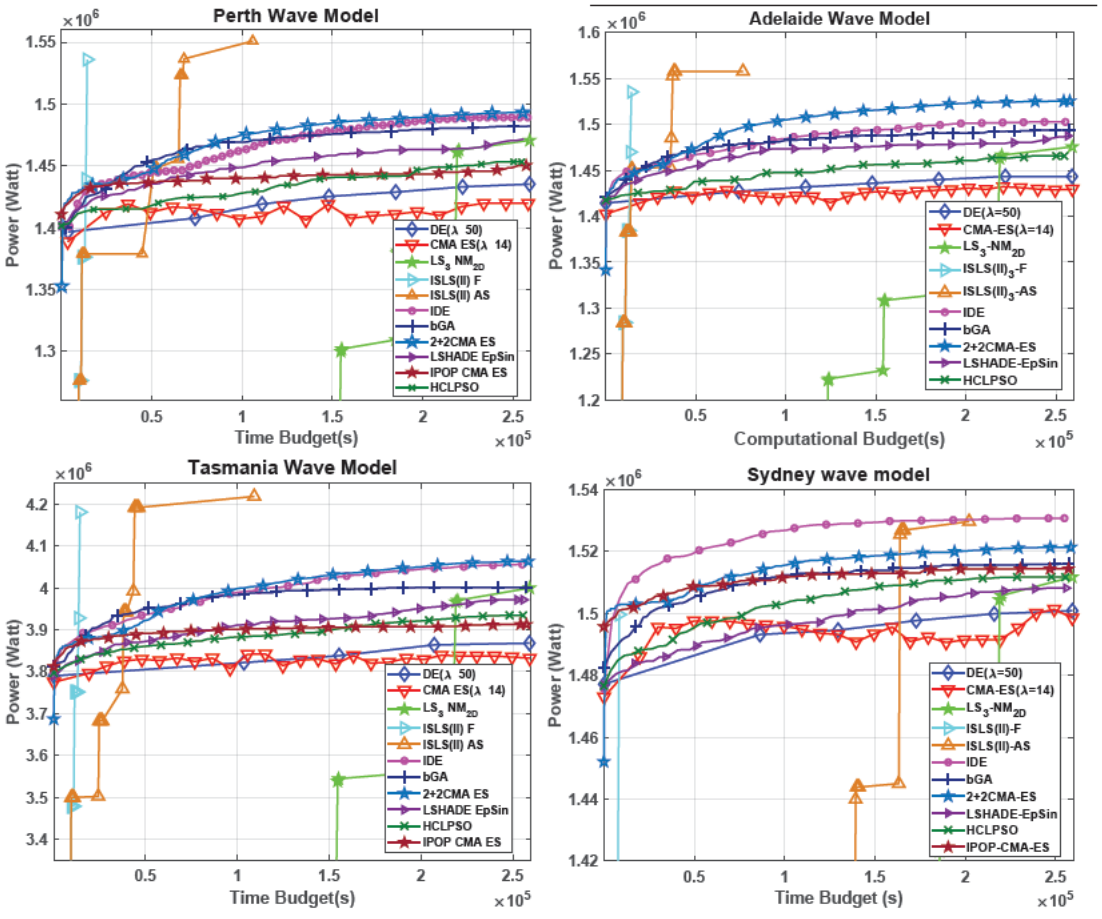

FIGURE 2.22: The average convergence rate comparison of the proposed methods for N=16 in four real wave models.

One interesting observation from the 16-buoy layouts in Figure. 2.23 is that there are buoys on the end of the front row for both Adelaide and Tasmania which produce less energy than the buoys behind them. While this appears to be counter-intuitive at first, it is simply the result of complex interactions between the front row of buoys and the ocean waves.

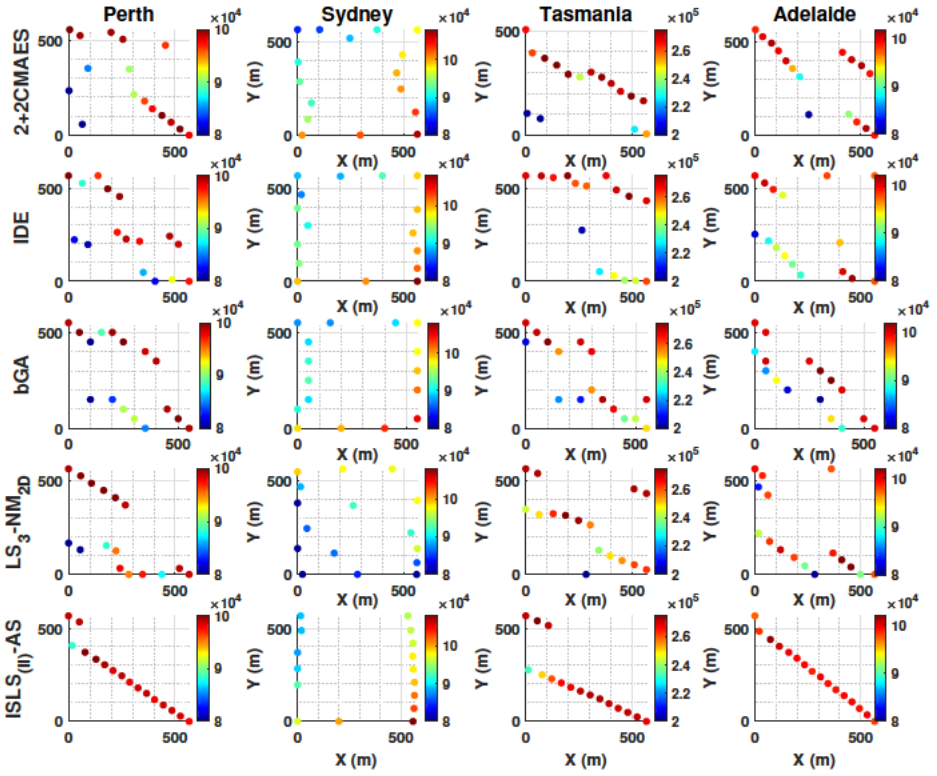

FIGURE 2.23: The best-found 16-buoy layouts of the real wave scenarios by ISL$S_{(II)}$-AS::Sydney: Power=1,534.9 kW, q-factor=0.9068, Perth: Power=1,565.6 kW, q-factor=1.015, Adelaide: Power=1,583.1 kW, q-factor=1.019 and Tasmania: Power=4,241.8 kW, q-factor=1.0012. The absorbed power of other 16-buoy layout can be seen in Table 2.8.

Figure. 2.25 demonstrates the nature of these interactions by showing the power landscape for the Perth wave scenario for four different layouts. In all cases, the wave energy in the area in the lee of the front row of the buoys (top right) is *greater* than the area immediately in front. This phenomenon is due to the fact that buoys interact strongly with the surrounding water on all sides.Modelling over an extended area shows that the wave-damping influence of these layouts stretches more than 500 meters further out to sea than the buoy array. This extended impact helps illustrate the potential for even relatively small buoy arrays to extract energy from a relatively large area of ocean. Another observation is of the relative efficacy of ISLS(II)-AS which places all of its buoys in high-energy locations.

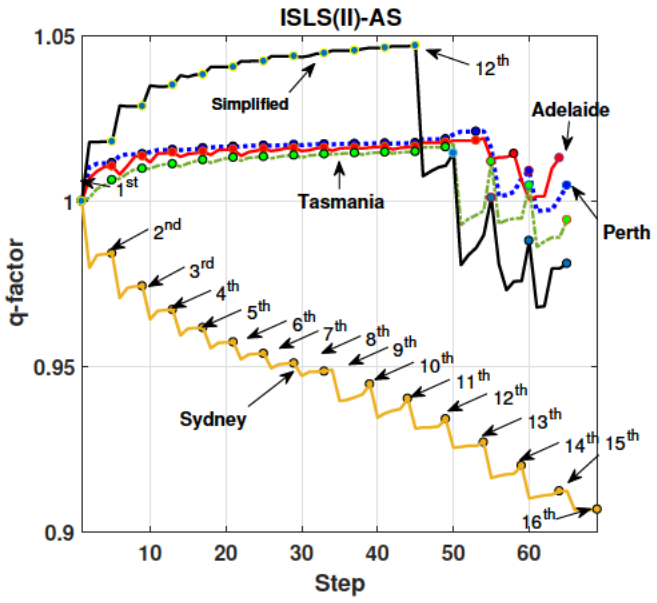

FIGURE 2.24: The average of ISLS(II)-AS q-factor performance (16-buoy) for real wave models and the simplified irregular model.

The impact for constructive and destructive interference in each environment can be visualized in the trajectory of average q-factor as ISLS$_{(II)}$-AS places buoys in each of the wave scenarios. Figure 2.24 shows this trajectory for the four wave scenarios and, as a reference to the simplified irregular wave scenario. It can be seen that Adelaide, Perth, and Tasmania are characterized by constructive interference in phase 1 and both Adelaide and Perth still have a q-factor greater than one even after phase 2. In contrast, the Sydney wave scenario is characterized by destructive interference throughout the search – though still producing net gains in power output for each buoy placed. The implemented codes and auxiliary material are publicly available: https://cs.adelaide.edu.au/~optlog/research/energy.php

2.2.10 Conclusion and future scope

In this work, we have explored algorithmic solutions to the problem of placing wave-energy converter buoys in arrays in order to maximise energy output. We have developed, evaluated, and systematically compared nine new search heuristics to a range of existing standard and domain-specific search techniques. The algorithms were benchmarked on both artificial and real wave scenarios. Producing effective layouts presented interesting challenges in terms of the high cost of full-function evaluations (approximately 700 seconds for one evaluation of a 16-buoy farm), but also complex problem dynamics including multi-modal constructive and destructive interference between buoys and highly varied power landscapes for buoy placement.

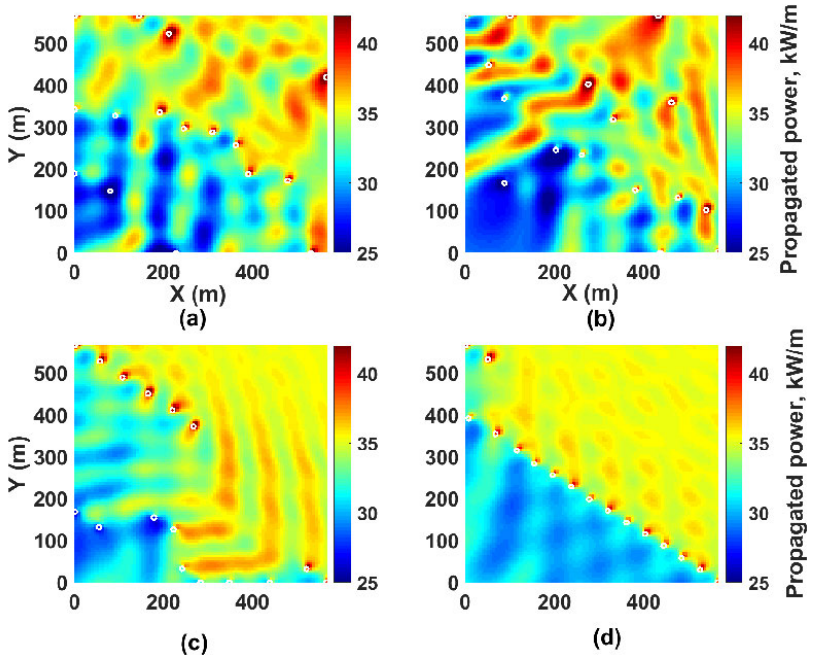

FIGURE 2.25: Interpolated wave energy landscape for the best 16-buoy layouts for Perth wave scenario, a) CMA-ES, b) DE, c) LS-NM and d) ISLS(II)-AS. White circles represent the buoy placement. (the wave angle propagates at 232.5 degrees).

In this research, the algorithms that performed best in experiments were hybrid search heuristics that used local search informed by inexpensive proxy models that were customised for local conditions. These methods further optimised the cost of function evaluations by placing one buoy at a time – thus minimising the number of modelled interactions. The most effective search techniques of all used the proxy model to inform the placement of the first row of buoys and switched to a combination of local search and gradient search techniques once the farm boundary was reached.

One possible limitation of the best approaches described here is that they allow no backtracking to further optimise buoy positions once they have been placed. Preliminary experiments with further global optimisation of four buoy layouts have shown some small potential gains from further global optimisation, though at a much much greater (87-fold) computational cost.

This work can be extended in several ways. First, while the hydrodynamic models employed here are state-of-the-art in terms of fidelity, the wave farm environments are still simplified in terms of farm geometry (assumed to be squared) and seafloor topography (assumed to be uniform in depth). In the future, both of these assumptions can be relaxed with farm geometry allowed to vary to match realistic lease-boundaries and the hydrodynamic model updated to allow varied seafloor depth. It is also possible to increase the complexity of the model for each buoy in terms of size, depth, and tether parameters.

Some of these parameters impact on cost and, thus, produce scope for multi-objective optimisation. Finally, there is scope to learn a robust and accurate proxy function for evaluating energy outputs. Such an approach might use machine learning techniques such as Deep Neural Networks to act as a partial or complete estimator function for the output of a given layout. Such an estimator has the potential to increase the speed of search significantly and open the way for further improvements in search heuristics.

Acknowledgements

We would like to show our gratitude to Prof.Suganthan from the Nanyang Technological University, Singapore, Dr.Hansen from the National Institute for Research in Computer Science and Control, France and Dr.Kalami from the Khaje Nasir Toosi University of Technology, Iran for publishing their valuable source codes. Meanwhile, this research is supported by the supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

2.3 Adaptive Neuro-Surrogate-Based Optimisation Method for Wave Energy Converters Placement Optimisation

2.3.1 Synopsis

To gain a more in-depth insight into the challenges of Wave energy converter (WEC) position optimisation problem, we include an article in this chapter which outline a new adaptive neuro-surrogate optimisation (ANSO) method. This method consists of a surrogate Recurrent Neural Network (RNN) model trained with a very limited number of observations. This neuro model is applied to simulate complicated hydrodynamic interactions and estimate the power output of the layout. As the model's hyper-parameters tuning is challenging, we apply a fast meta-heuristic approach to exploring the hyper-parameters search space and proposing better settings. After training the model, we develop a greedy local search with a backtracking search strategy to repair the position of the converters with low absorbed power in farms. The performance of the proposed method (ANSO) is evaluated by four real wave scenarios (Sydney, Perth, Adelaide and Tasmania) and compared with some of the more popular Evolutionary Algorithms (EAs). Comparison of results shows that the adaptive neuro model performs better than previously reported optimisation methods in terms of computational complexity and the total power output of the farm.

Reference

[4] Neshat, M., Abbasnejad, E., Shi, Q., Alexander, B., & Wagner, M. (2019, December). Adaptive Neuro-Surrogate-Based Optimisation Method for Wave Energy Converters Placement Optimisation. Published by International Conference on Neural Information Processing (pp. 353-366). Springer, Cham on 2019.

Statement of Authorship

Title of Paper	Adaptive Neuro-Surrogate-Based Optimisation Method for Wave Energy Converters Placement Optimisation.
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Abbasnejad, E., Shi, Q., Alexander, B., & Wagner, M. (2019, December). Adaptive Neuro-Surrogate-Based Optimisation Method for Wave Energy Converters Placement Optimisation. In International Conference on Neural Information Processing (pp. 353-366). Springer, Cham.

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	80%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	15/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Ehsan Abbasnejad (5%)		
Contribution to the Paper	Provided advice on the machine learning methodology and recommendations on the implementations and results. In addition, provided feedbacks and revisions on the manuscript.		
Signature		Date	14/05/2020

Name of Co-Author	Qinfeng Shi (5%)		
Contribution to the Paper	Providing advice on machine learning aspects.		
Signature		Date	4/06/2020

Name of Co-Author	Bradley Alexander (5%)		
Contribution to the Paper	Advised on some initial strategies. Gave feedback on results. Revised paper.		
Signature		Date	12/05/20

Name of Co-Author	Markus Wagner (5%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

2.3.2 Abstract

The installed amount of renewable energy has expanded massively in recent years. Wave energy, with its high capacity factors has great potential to complement established sources of solar and wind energy. This study explores the problem of optimising the layout of advanced, three-tether wave energy converters in a size-constrained farm in a numerically modelled ocean environment. Simulating and computing the complicated hydrodynamic interactions in wave farms can be computationally costly, which limits optimisation methods to have just a few thousand evaluations. For dealing with this expensive optimisation problem, an adaptive neuro-surrogate optimisation (ANSO) method is proposed that consists of a surrogate Recurrent Neural Network (RNN) model trained with a very limited number of observations. This model is coupled with a fast meta-heuristic optimiser for adjusting the model's hyper-parameters. The trained model is applied using a greedy local search with a backtracking optimisation strategy. For evaluating the performance of the proposed approach, some of the more popular and successful Evolutionary Algorithms (EAs) are compared in four real wave scenarios (Sydney, Perth, Adelaide and Tasmania). Experimental results show that the adaptive neuro model is competitive with other optimisation methods in terms of total harnessed power output and faster in terms of total computational costs.

2.3.3 Introduction

As the global demand for energy continues to grow, the advancement and deployment of new green energy sources are of paramount significance. Due to high capacity factors and energy densities compared to other renewable energy sources, ocean waves energy has attracted research and industry interest for a number of years [18]. Wave Energy Converters (WEC's) are typically laid out in arrays and, to maximise power absorption, it is important to arrange them carefully with respect to each other [21]. The number of hydrodynamic interactions increases quadratically with the number of WEC's in the array. Modelling these interactions for a single moderately-sized farm layout can take several minutes. Moreover, the optimisation problem for farm-layouts is multi-modal– typically requiring the use of many evaluations to adequately explore the search space. There is scope to improve the efficiency of the search process through the use of a learned surrogate model. The challenge is to train such a model fast enough to allow an overall reduction in optimisation time. This paper proposes a new hybrid adaptive neuro-surrogate model (ANSO) for maximizing the total absorbed power of WECs layouts in detailed models of four real wave regimes from the southern coast of Australia (Sydney, Adelaide, Perth and Tasmania). Our approach utilises a neural network that acts as

a surrogate for estimating the best position for placement of the converters. The key contributions of this paper are:

1. Designing a neuro-surrogate model for predicting total wave farm energy by training of recurrent neural network (RNNs) using data accumulated from evaluations of farm layouts.
2. The use of the Grey Wolf Optimiser [128] to continuously tune hyper-parameters for each surrogate.
3. A new symmetric local search heuristic with greedy WEC position selection combined with a backtracking modification (BO) to improve the layouts further for delicate adjustments.

We demonstrate that the adaptive framework described outperforms previously published results in terms of both optimisation speed (even when total training time is included) and total absorbed power output for 16-WEC layouts.

2.3.4 Related work

In this application domain, neural networks have been utilized for predicting the wave features (height, period and direction) more than other ML techniques [129]. In early work, Alexandre et al. [130] applied a hybrid Genetic Algorithm (GA) and an extreme learning machine (ELM) (GA-ELM) for reconstructing missing parameters from readings from nearby sensor buoys. The same study [131] investigated a combination of the grouping GA and ELM (GGA-ELM) for feature extraction and wave parameter estimation. A later approach [132], combined the GGA–ELM with Bayesian Optimisation (BO) for predicting the ocean wave features. BO improved the model significantly at the cost of increased computation time. Sarkar et al. [133] combined machine learning and optimisation of arrays of, relatively simple, oscillating surge WECs. They were able to use this technique to effectively optimise arrays of up to 40 WEC's – subject to fixed spacing constraints. Recently, James et al. [134] used two different supervised ML methods (MLP and SVM) to estimate WEC layout performance and characterise the wave environment [134]. However, the models produced required a large training data-set and manual tuning of hyper-parameters.

In work optimising WEC control parameters, Li et al. [135] trained a feed-forward neural network (FFNN) to learn key temporal relationships between wave forces. While the model required many samples to train it exhibited high accuracy and was used effectively in parameter optimisation for the WEC controller. Recently, Lu et al. [136] proposed a

hybrid WECs PTO controller which consists of a recurrent wavelet-based Elman neural network (RWENN) with an online back-propagation training method and a modified gravitational search algorithm (MGSA) for tuning the learning rate and improving learning capability. The method was used to control the rotor acceleration of the combined offshore wind and wave power converter arrangements. Finally, recent work by Neshat et al. [1] evaluated a wide variety of EAs and hybrid methods by utilizing an irregular wave model with seven wave directions and found that a mixture of a local search combined with the Nelder-Mead simplex method achieved the best array configurations in terms of the total power output.

2.3.5 Wave Energy Converter Model

We use a WEC hydrodynamic model for a fully submerged three-tether buoy. Each tether is attached to a converter installed on the seafloor [90]. The relevant details of the WECs modelled in this research are: Buoy number=16, Buoy radius=5 m, Submergence depth=3 m, Water depth=30 m, Buoy mass=376 t, Buoy volume=523.60 m^2 and Tether angle=55°.

2.3.5.1 System dynamics and parameters

The total energy produced by each buoy in an array is modelled as the sum of three forces [137]:

1. The power of wave excitation ($F_{exc,p}(t)$) includes the forces of the diffracted and incident ocean waves when all generators locations are fixed.
2. The force of radiation($F_{rad,p}(t)$) is the derived power of an oscillating buoy independent of incident waves.
3. Power take-off force($F_{pto,p}(t)$) is the force exerted on the generators by their tethers.

Interactions between buoys are captured by the $F_{exc,p}(t)$ term. These interactions can be destructive or constructive, depending on buoys' relative angles, distances and surrounding sea conditions. Equation 2.12 shows the power accumulating to a buoy number p In a buoy array.

$$M_p\ddot{X}_p(t) = F_{exc,p}(t) + F_{rad,p}(t) + F_{pto,p}(t) \tag{2.12}$$

Where M_p is the displacement of the p_{th} buoy, $\ddot{X}_p(t)$ is a vector of body acceleration in the surge, heave and sway. The last term, denoting the power take-off system, that can be simulated as a linear spring and damper. Two control factors are involved for each

mooring line: the damping B_{pto} and stiffness K_{pto} coefficients. Therefore the Equation (2.12) can be elaborated as:

$$((M_\Sigma + A_\sigma(\omega))j\omega + B_\sigma(\omega) - \frac{K_{pto,\Sigma}}{\omega}j + B_{pto,\Sigma})\ddot{X}_\Sigma = \hat{F}_{exc,\Sigma} \tag{2.13}$$

where $A_\Sigma(\omega))$ and $B_\Sigma(\omega)$ are hydrodynamic parameters which are derived from the semi-analytical model based on [92]. Hence, the total power output of a buoy array is:

$$P_\Sigma = \frac{1}{4}(\hat{F^*}_{exc,\Sigma}\ddot{X}_\Sigma + \ddot{X^*}_\Sigma\hat{F}_{exc,\Sigma}) - \frac{1}{2}\ddot{X^*}_\Sigma B\ddot{X^*}_\Sigma \tag{2.14}$$

While we can compute the total power in Equation 2.14, it is very computationally demanding and increases exponentially with the number of buoys. With constructive interference the total power output can scale super–linearly with the number of buoys. The detailed wave characteristics including the number, direction and the probability of wave frequencies can be seen in figure 2.26.

2.3.6 Optimisation Setup

The optimisation problem studied in this work can be expressed as:

$$P^*_\Sigma = argmax_{\mathbf{x},\mathbf{y}} P_\Sigma(\mathbf{x}, \mathbf{y})$$

,where $P_\Sigma(\mathbf{x}, \mathbf{y})$ is the average whole-farm power given by the buoys placements in a field at x-positions: $\mathbf{x} = [x_1, \ldots, x_N]$ and corresponding y positions: $\mathbf{y} = [y_1, \ldots, y_N]$. The buoy number is here $N = 16$.

2.3.6.1 Constraints

There is a square-shaped boundary constraint for placing all buoys positions (x_i, y_i): $l \times w$ where $l = w = \sqrt{N * 20000}\, m$. This gives $20\,000\,\mathrm{m}^2$ of the farm-area per-buoy. To maintain a safety distance, buoys must also be at least 50 metres distant from each other. For any layout $\mathbf{x}, \mathbf{y}$ the sum-total of the inter-buoy distance violations, measured in metres, is:

$$Sum_{dist} = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}(dist((x_i, y_i), (x_j, y_j)) - 50),$$
$$\text{if } dist((x_i, y_i), (x_j, y_j)) < 50 \text{ else } 0$$

where $dist((x_i, y_i), (x_j, y_j))$ is the L2 (Euclidean) distance between buoys i and j. The penalty applied to the farm power output (in Watts) is $(Sum_{dist} + 1)^{20}$. This steep

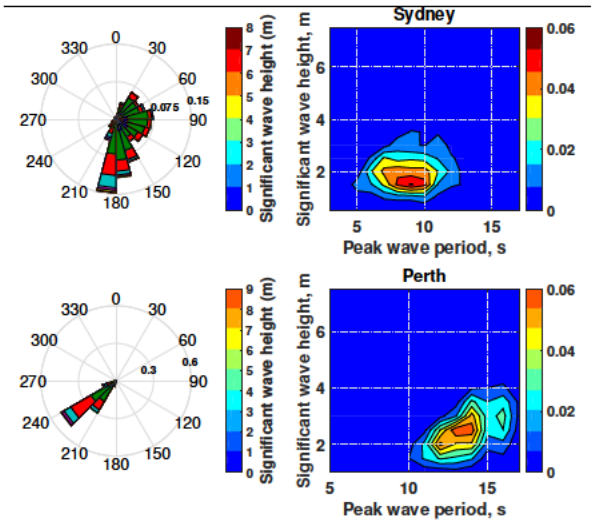

FIGURE 2.26: Wave data for two test sites in Australia: (a) Sydney and (b) Perth. These are: the directional wave rose (left) and wave scatter diagram (right).

penalty allows better handling of constraint violations during the search. Buoy placements which are outside of the farm area are handled by reiterating the positioning process.

2.3.6.2 Computational Resources

In this work, depending on the optimisation method, the average evaluation time for a candidate layout can vary greatly. To ensure a fair comparison of methods the maximum budget for all optimisation methods is three days of elapsed time on a dedicated high-performance shared-memory parallel platform. The compute nodes have 2.4GHz Intel 6148 processors and 128GB of RAM. The meta-heuristic frameworks as well as the hydrodynamic simulator for $P_{\Sigma}(\mathbf{x}, \mathbf{y})$ are run in MATLAB R2018. This MATLAB license enables us to run 12 worker threads in parallel and the methods are optimised to use as many of these threads as the methodology allows.

2.3.7 Methods

In this study, the optimisation approaches employ two strategies. First, optimising all decision variables (buoy placements) at the same time. We compare five population-based EAs that use this strategy. Second, based on [1], we place one buoy at a time sequentially, comparing two hybrid techniques.

2.3.8 Evolutionary Algorithms (EAs)

Five popular off-the-shelf EAs are compared in the first strategy to optimise all problem dimensions. These EAs include: (1) Differential Evolution (DE) [94], with a configuration of $\lambda = 30$ (population size), $F = 0.5$ and $P_{cr} = 0.5$; (2) covariance matrix adaptation evolutionary-strategy (CMA-ES) [89] with the default settings and λ =DE configurations; (3) a $(\mu + \lambda)$EA that mutates buoys' position with a [95] probability of $1/N$ using a normal distribution ($\sigma = 0.1 \times (U_b - L_b)$) when $\mu = 50$ and $\lambda = 25$; and (4) Particle Swarm optimisation (PSO) [42], with λ= DE configurations , $c_1 = 1.5, c_2 = 2, \omega = 1$ (linearly decreased).

2.3.8.1 Hybrid optimisation algorithms

Relevant researches [1, 23, 138] noticed that employing a neighborhood search around the previously placed-buoys could be beneficial for exploiting constructive interactions between buoys. The two following methods utilise this observation by placing and optimising the position of one buoy at a time.

2.3.8.1.1 Local Search + Nelder-Mead(LS-NM)

LS-NM [1] is one of the most effective and fast WEC placement methods. LS-NM positions generators sequentially by sampling at a normally-distributed random deviation (σ =70 m) from the previous buoy location. The best-sampled location is optimised using N_s iterations of the Nelder-Mead search method. This process is repeated until all buoys are placed.

2.3.8.1.2 Adaptive Neuro-Surrogate Optimisation method (ANSO)

Given the complexity of the optimisation problem we devise a novel approach with the intuition that (a) sequential placement of the converters provide a simple, yet effective baseline and (b) we can *learn* a surrogate to mimic the potential power output for an array of buoys. Hence, we provide a three step solution (as detailed in Algorithm 2.6).

2.3.8.1.3 Symmetric Local Search (SLS):

Inspired by LS-NM [1, 138], in the first step we sequentially place buoys by conducting a local search for each placement. SLS starts by placing the first buoy in the recommended

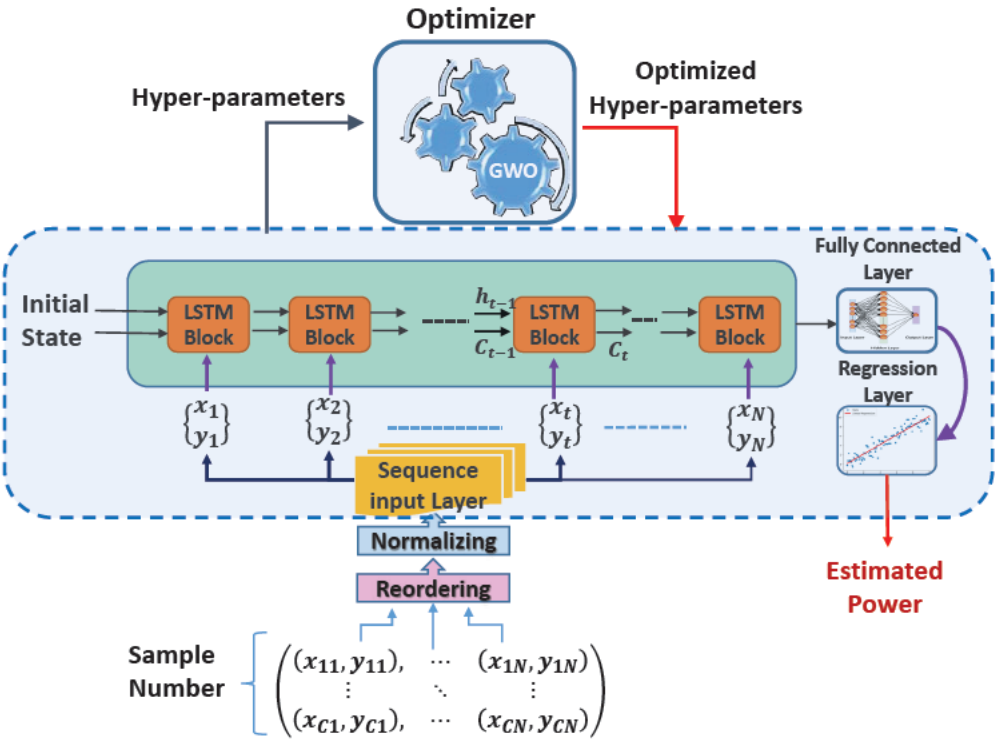

FIGURE 2.27: The Neuro-Surrogate model architecture

position (bottom corner of the field) and then for each subsequent buoy position, uniformly performs N_{iters} of feasible local samples are made in different sectors commencing at angles: $\{angles = [0°, Res°, 2 \times Res°, ..., 360 - Res°]\}$ and bounded by a radial distance of between $50 + R_1 m$ (safe distance+first search radius) and $50 + R_2 m$. The best sample is chosen among the N_{iters} local samples. Next, two extra neighbourhood samples near the best sample ($\pm Res/2$) are made for increasing the exploitation ability of the method. The best of these three samples, based on total absorbed power, is then selected.

Algorithm 2.6 Adaptive Neuro-Surrogate Optimisation (ANSO)

procedure ADAPTIVE NEURO-SURROGATE OPTIMISATION (ANSO)
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size
$Res = 3$ ▷ angle resolution
$angle = \{0, Res, 2 \times Res, \ldots, 360 - Res\}$ ▷ symmetric samples angle
$iters = Size([angle])$ ▷ Number of symmetric samples
$EvalSet = \{2^{nd}, 3^{rd}, 5^{th}, ..., 15^{th}\}$ ▷ Set of evaluated buoys
$EstimSet = \{4^{th}, 6^{th}, 8^{th}, ..., 16^{th}\}$ ▷ Set of estimated buoys
$S = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = \perp$ ▷ Positions
$S_{(1)} = \{\langle size, 0\rangle\}$ ▷ first buoy position
$energy = Eval([S_{(1)}])$
$bestPosition = S_{(1)}$;
  for i in $[2, .., N]$ **do**
    $bestEnergy = 0$;
    if $i \in Evalset$ **then** ▷ layouts should be evaluated by Simulator
      for j in $[1, .., iters]$ **do**
        $(Sample_j, energy_j)$=$SymmetricSampleEval(angle_j, S_{(i-1)})$
        if $Sample_j$ *is feasible* & $energy_j > bestEnergy$ **then**
          $tPos = Sample_j$ ▷ Temporary buoy position
          $bestEnergy = energy_j$
          $bestAngle = j$
        end if
      end for
      if *No feasible solution is found* **then**
        $(Sample_1, energy_1)$=$rand(S_{(i-1)})$
      end if
      (Es_1, Es_2)=$SymmetricSampleEval(bestAngle \pm Res/2, S_{(i-1)})$
      $(S_{(i)}, energy)$=$FindbestS(tPos, Es_1, Es_2)$
      $DataSet_i = UpdateData(Sample, energy)$
    else ▷ layouts should be estimated by the LSTM
      $(HyperParameters_i)$=$Optimise\text{-}Hyper(DataSet_i)$ ▷ Optimising by GWO
      $(Deep_i)$=$reTrain(Deep_i, DataSet_i, HyperParameters_i)$
      for j in $[1, .., iters]$ **do**
        $(Sample_j, energy_j)$=$SymmetricSampleEstim(angle_j, S_{(i-1)}, Deep_i)$
        if $Sample_j$ *is feasible* & $energy_j > bestEnergy$ **then**
          $tPos = Sample_j$ ▷ Temporary buoy position
          $bestEnergy = energy_j$
          $bestAngle = j$
        end if
      end for
    end if
  end for
  $(bestPosition', bestEnergy') = BackTrackingOp(bestPosition)$
  return $bestPosition', bestEnergy'$ ▷ Final Layout
end procedure

2.3.8.1.4 Learning the neuro-surrogate model:

The hydrodynamic simulator is computationally expensive to run. A fast and accurate neuro-surrogate is used here to estimate the power of layout based on the position of the next buoy: (x_i, y_i). Our motivation is that a fast surrogate function can quickly estimate what the simulator takes a long time to compute. The key challenges to overcome in designing a neuro-surrogate are: 1) *function complexity*: a highly nonlinear and complex relationship between buoys position and absorbed farm power, 2) *changing dataset*: as more evaluations of the placements are performed, new data for training is collected that has to be incorporated, and, 3)*efficiency*: training time plus the hyper-parameter tuning has to be included in our computational budget.

For handling these challenges, we use a combination of recurrent networks with LSTM cells [139] (sequential learning strategy), and, an optimiser (GWO) [11] for tuning the network hyper-parameters for estimating the power of the layouts. The overall framework is shown in Figure 2.27. The proposed LSTM network is designed for sequence-to-one regression in which the input layer is from 2D buoy positions (x_i, y_i) and the output of the regression layer is the estimated layout power. The LSTM training process is done using the back-propagation algorithm, in which the gradient of the cost function (in this case the mean squared error between the true ground-truth and the estimated output of the LSTM) at different layers are computed to update the weights.

For tuning the hyper-parameters of the LSTM we use the ranges: MiniBatch size $(5 - 100)$, learning rate $(10^{-4} - 10^{-1})$, the maximum number of training epochs $(50 - 600)$, LSTM layer number (one or two) and hidden node number $(10 - 150)$. At each step of the position optimisation, a fast and effective meta-heuristic algorithm (GWO) [128] is used. This is because the collected data-set is dynamic in terms of input length (increases over time) and the arrangement of buoys. This hybrid idea depicts an adaptive learning process that is fast (is converged by a few evaluations (Figure 2.31)), accurate and easily scalable to larger sizes.

2.3.8.1.5 Backtracking Optimisation:

The third component of ANSO is applying a backtracking optimisation strategy (BO). This is because the initial placements described above are based on greedy selection, the previous buoys' positions are revisited during this phase. Consequently, introducing backtracking can help maximise the power of the layouts. For this part, a 1+1EA[140] is employed. In each iteration, the buoys position (x_i, y_i) is mutated based on a Gaussian normally distributed random variable with a dynamic mutation step size (σ) that

TABLE 2.9: Performance comparison of various heuristics for the 16-buoy case, based on maximum, median and mean power output layout of the best solution per experiment.

	DE	**CMA-ES**	**PSO**	**(μ + λ) EA**	**LS-NM**	**ANSO-S_1**	**ANSO-S_2**	**ANSO-S_3**	**ANSO-S_4**	**ANSO-S_1-B**	**ANSO-S_2-B**	**ANSO-S_3-B**	**ANSO-S_4-B**
Perth wave scenario (16-buoy)													
Max	1474381	1490404	1463608	1506746	1501145	1544546	1533055	1554926	1555446	1552108	1549299	1554833	**1559535**
Min	1455256	1412438	1433776	1486397	1435714	1513894	1489365	1531290	1543637	1535508	1502373	1543384	1549517
Mean	1462331	1476503	1450589	1494311	1479345	1534032	1514147	1543361	1550171	1544832	1525112	1549276	**1556073**
Median	1462697	1482974	1448835	1493109	1490195	1535162	1516162	1544076	1551105	1544733	1523082	1549701	**1556091**
STD	4742.1	23004.6	8897.7	6227.9	23196.4	7991.1	12092.2	7441.2	3333.5	5531.3	12663.7	4006.2	2783.2
Adelaide wave scenario (16-buoy)													
Max	1494124	1501992	1475991	1517424	1523773	1563935	1563249	1583623	1585626	1576713	1571181	**1589830**	1588297
Min	1468335	1478052	1452804	1488276	1496878	1558613	1520681	1565725	1571131	1566240	1527665	1567491	1576009
Mean	1479247	1488783	1461579	1502708	1513070	1561624	1541404	1573125	1575439	1572454	1552201	**1581643**	1578365
Median	1479707	1487430	1460687	1501805	1515266	1562548	1541101	1576658	1575092	1573763	1552663	**1582515**	1577353
STD	7704.9	8167.9	6670.9	8443.2	7434.7	2154.3	12366.9	7572.5	3676.1	3639.8	12373.1	6481.1	3428.9
Sydney wave scenario (16-buoy)													
Max	1520654	1529809	1525938	1528934	1524164	1523552	1523353	1523549	1524974	1531566	**1532200**	1528619	1531155
Min	1515231	1520031	1508729	1516014	1487836	1509677	1493596	1500115	1514248	1517559	1506128	1513182	1520086
Mean	1518047	1524054	1519251	1522625	1507594	1517627	1514384	1514300	1520597	1524357	1523382	1521277	**1526443**
Median	1518014	1523440	1520319	1522234	1507898	1518667	1516523	1518055	1521351	1524767	1524356	1522289	**1527839**
STD	1880.1	2767.8	5818.1	3887.1	10929.2	4871.7	8811.3	8642.02	4021.8	4161.7	6710.5	6393.02	3481.1
Tasmania wave scenario (16-buoy)													
Max	3985498	4063049	3933518	4047620	4082043	4144344	4085915	4121312	4135256	**4162505**	4104237	4143536	4160738
Min	3935990	3935833	3893456	3992362	3904892	4025709	4021772	4071497	4113146	4053715	4043849	4103441	4128702
Mean	3956691	4000087	3914316	4019472	4008228	4072874	4042537	4093453	4122447	4095608	4071852	4123334	**4145569**
Median	3951489	3994739	3914764	4019623	4020515	4066904	4033063	4091620	4121959	4079286	4074154	4124520	**4144359**
STD	17243.1	37701.2	13758.4	18377.5	54771.9	33897.8	19819.9	17367.4	6422.9	34789.9	16516.9	12411.4	10085.3

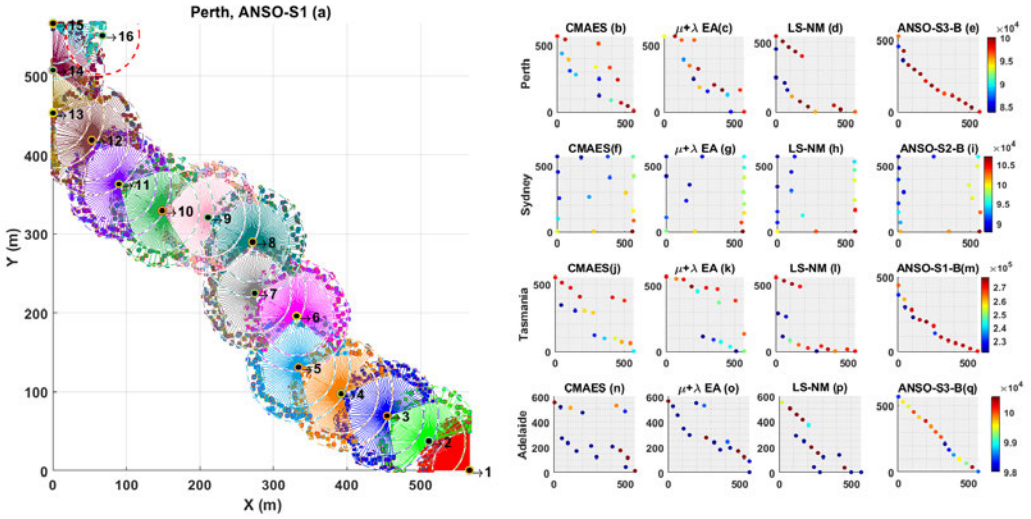

FIGURE 2.28: The best-obtained 16-buoy layouts: figure(a) presents how the proposed hybrid method can optimise buoys position and estimate the power of some buoys ($4^{th}, 6^{th}, ..., 16^{th}$) sequentially. Two rings around each buoy show the exploration space(Res = 3°,ΔR =20 m). Other figures show the best layouts arrangement of the four real wave scenarios based on Table 2.9.

is decreased linearly and an adaptive probability rate (P_m). The mutated position is evaluated by the simulator. Both Equations 2.15 and 2.16 represent the details of these control parameters of the BO method.

$$\sigma_{iter} = \sigma_{Max} \times 0.08 \times iter/iter_{Max} \; \forall \; iter \in \{1, ..., iter_{Max}\} \tag{2.15}$$

$$P_{m_i} = (1/N) \times (1/(Power_{Buoy_i}/Max_{Power})) + \omega_i \; \forall \; i \in \{1, ..., N\} \tag{2.16}$$

Where σ_{Max} is the initial mutation step size at 10 m and P_{m_i} shows the mutation probability of each buoy in the layout. We assume that the buoys with lower absorbed power need more chance of modification, so the highest mutation probability rate should be allocated to that buoy with the lowest power and vice-versa. In addition, ω_i is a weighted linear coefficient from 0.1 (for the lowest power of the buoys) to 0 (highest buoy power). The reserved runtime for the BO method is one hour. Algorithm 2.6 describes this method in detail.

2.3.9 Experiments

The adaptive tuning of hyper-parameters in ANSO makes it compatible with each layout problem. Moreover, no pre-processing time is required for collecting the relevant training data-set, ANSO is able to collect the required training data in real-time during the sampling and optimisation of previous buoy positions.

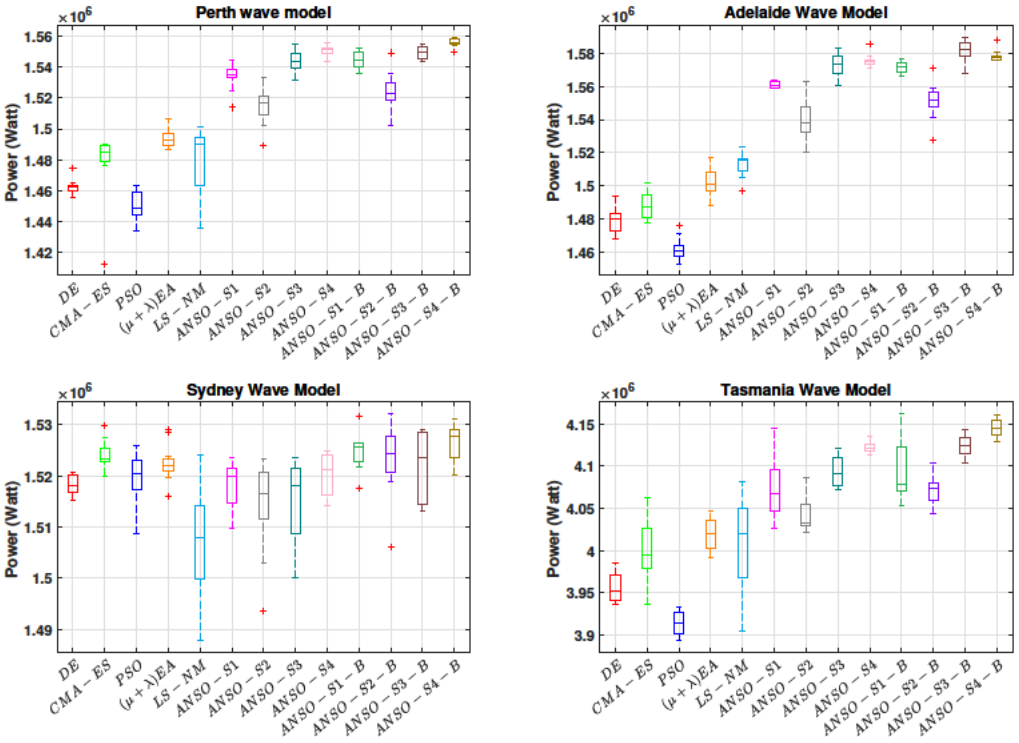

FIGURE 2.29: Comparison of the algorithms' effectiveness for 16-buoy layouts in four real wave scenarios (Perth(a), Adelaide(b), Sydney(c), Tasmania(d)). The optimisation results show the best solutions of 10 independent runs..

In this work, we test four strategies for buoy placement under ANSO these vary in the membership of the *EvalSet* – the set of buoys evaluated by the full simulator and the sample set used to train the neuro-surrogate model. The strategies tested are:

1. With $EvalSet = \{2^{nd}, 3^{rd}, 5^{th}, ..., 15^{th}\}$ so the neuro-surrogate is used to place buoys: $4, 6, 8, \ldots, 16$. The neuro-surrogate is trained prior to each placement using sampled positions used for the previous buoy placement.

2. Use $EvalSet = \{2^{nd}, 3^{rd}, 6^{th}, 9^{th}, ..., 15^{th}\}$ so the neuro-surrogate is used to place buoys: $4, 5, 7, 8, 10, 11, 13, 14, 16$. As with strategy 1 above, the previous sampled positions evaluated by the simulator are used to train the neuro-surrogate model.

3. The same setup as the first strategy but the LSTM is trained by *all* previous simulator samples.

4. All evaluations are done by the simulator.

First, Table 2.9 shows the statistical results of the maximal power output of the 13 compared heuristics for 10 independent runs each for four real wave scenarios. As shown

TABLE 2.10: The average ranking of the proposed methods for ρ by Friedman test.

Rank	**Adelaide**		**Perth**		**Sydney**		**Tasmania**	
1	**ANSO-S_3-B**	**(1.75)**	**ANSO-S_4-B**	**(1.08)**	**ANSO-S_4-B**	**(3.00)**	**ANSO-S_4-B**	**(1.25)**
2	ANSO-S_4-B	(2.08)	ANSO-S_4	(3.08)	ANSO-S_1-B	(4.17)	ANSO-S_3-B	(2.75)
3	ANSO-S_3	(3.67)	ANSO-S_3-B	(3.17)	CMA-ES	(4.33)	ANSO-S_4	(3.00)
4	ANSO-S_4	(3.67)	ANSO-S_1-B	(4.00)	ANSO-S_2-B	(4.50)	ANSO-S_1-B	(4.67)
5	ANSO-S_1-B	(4.00)	ANSO-S_3	(4.42)	ANSO-S_3-B	(5.08)	ANSO-S_3	(5.00)
6	ANSO-S_1	(6.08)	ANSO-S_1	(6.00)	$(\mu + \lambda)$EA	(6.00)	ANSO-S_2-B	(6.00)
7	ANSO-S_2-B	(6.75)	ANSO-S_2-B	(6.50)	ANSO-S_4	(7.08)	ANSO-S_1	(6.33)
8	ANSO-S_2	(8.08)	ANSO-S_2	(8.00)	PSO	(7.92)	ANSO-S_2	(8.33)
9	LS-NM	(9.17)	$(\mu + \lambda)$EA	(9.25)	ANSO-S_1	(8.42)	LS-NM	(9.25)
10	$(\mu + \lambda)$EA	(9.92)	LS-NM	(10.50)	DE	(9.58)	$(\mu + \lambda)$EA	(9.42)
11	CMAES	(11.00)	CMA-ES	(10.67)	ANSO-S_3	(9.67)	CMA-ES	(10.25)
12	DE	(11.83)	DE	(11.83)	ANSO-S_2	(10.00)	DE	(11.75)
13	PSO	(13.00)	PSO	(12.50)	LS-NM	(11.25)	PSO	(13.00)

in Table 2.9, ANSO-S_4-B is best, on average, in Sydney, Perth and Tasmania. ANSO-S_3-B shows the best performance in Adelaide. However, all methodologies using the neuro-surrogate are competitive in terms of performance. The results of applying the Friedman test are shown in Table 2.10. Algorithms are ranked according to their best configuration for each run. Again, ANSO-S_3-B obtained first ranking in the Adelaide wave model and non-neuro-surrogate ANSO-S_4-B algorithm ranks highest in other scenarios. The best 16-buoy layouts of the 4 compared algorithms (CMAES, $(\mu + \lambda)$EA, LS-NM and the best-performing versions of ANSO) are shown in Figure 2.28. The sampling used by the optimisation process of ANSO-S_1 is shown in Figure 2.28(a). It shows how ANSO-S_1 explores each buoy's neighbourhood and modifies positions during backtracking.

Figure 2.29 shows box-and-whiskers plots for the best solutions power output per run for all approaches and all wave scenarios. It can be seen that the best mean performance is given by ANSO-S_4-B in three of four-wave scenarios. In the Adelaide case study, ANSO-S_3-B performs best. Another interesting observation is that, among population-based EAs, $(\mu + \lambda)$EA excels. However, both ANSO and LS-NM outperform all population–based methods.

Figure 2.30 exhibits the convergence diagrams of the average power output of the nine compared algorithms. In all wave scenarios, ANSO-S_2-B has the ability to converge very fast because it estimates two sequentially placed buoys layouts power after each training process instead of evaluating one of these using the expensive simulator. ANSO-S_2-B is able to not only save the runtime for evaluating samples but also save the surrogate training time and is, respectively, 3, 4.5 and 14.6 times faster, on average, than ANSO-S_4-B, LS-NM and $(\mu + \lambda)EA$. Note again, that these timings include training and configuration times.

For our neuro-surrogate model to produce accurate and reliable power estimation, we need to obtain good settings for hyper-parameters. Finding the best configuration for these parameters in such a continuous, multi-modal and complex search space is not a trivial challenge. Figure 2.31 shows the GWO performance for tuning the LSTM hyper-parameters for ANSO-S_1. In addition, the Pearson correlation coefficient testing values (R-value) for all trained LSTMs estimates the performance of the trained LSTM ($\bar{R} >= 0.7$). The most challenging training process is related to the power estimation of the 14^{th} buoy because ANSO is faced with the boundary constraint of the search space, so the arrangement of the layouts changes.

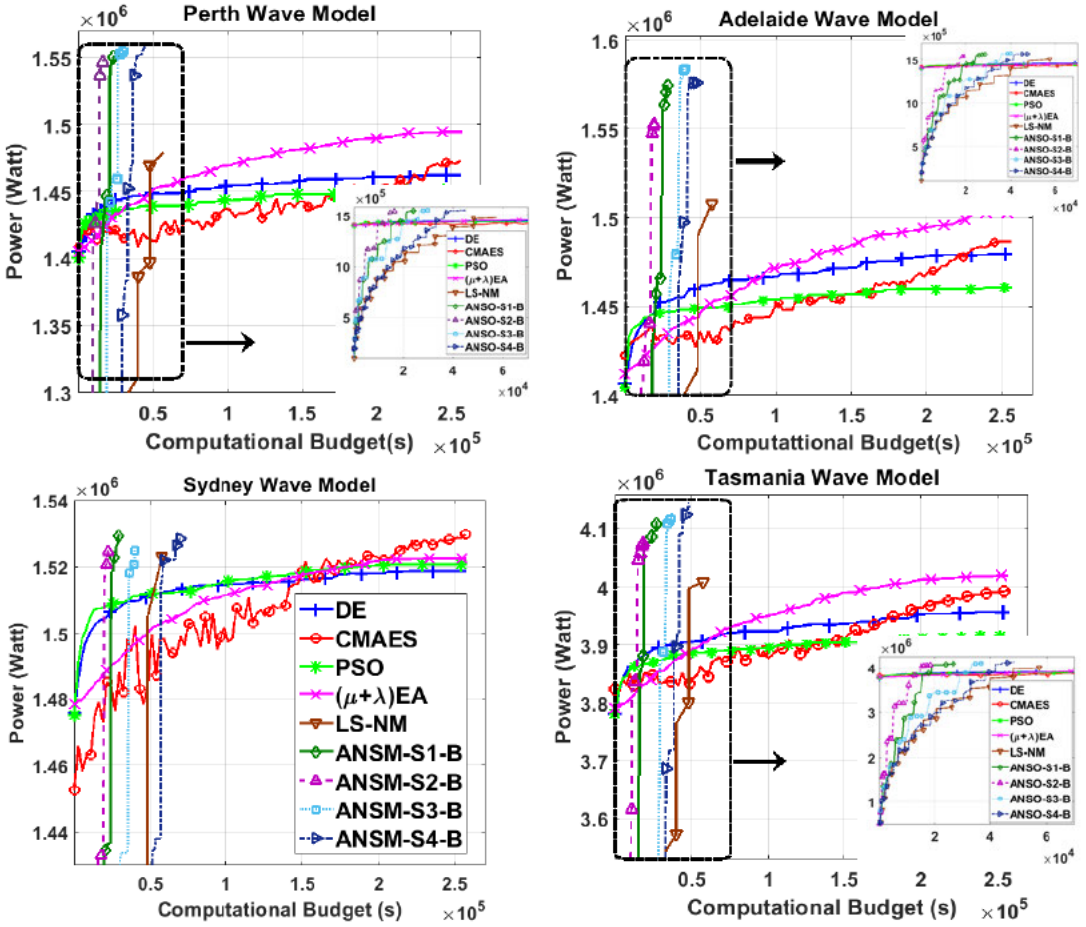

FIGURE 2.30: Evolution and convergence rate of the average power output contributed by the nine algorithms for four real wave models. A zoomed version of the plots is provided to show a better insight into the convergence speed.

2.3.10 Conclusions

Optimising the arrangement of a large WEC farm is computationally intensive, taking days in some cases. Faster and smarter optimisation methods are needed. We have shown that a neuro-surrogate optimisation approach, with online training and hyper-parameter optimisation, is able to outperform previous methods in terms of layout performance – 3.2% to 3.6% better, respectively than $(\mu + \lambda)EA$ and LS-NM. Moreover, even including the time for training and tuning the LSTM network the neuro-surrogate model finishes optimisation faster than previous methods. Thus better results are obtained in less time – up to 14 times faster than the $(\mu + \lambda)EA$. The approach is also highly adaptable with the model's and its hyper-parameters being tuned online for each environment. Future work will include a more detailed analysis of the training setup of the neuro-surrogate to further focus training to the sectors most relevant to buoy placement and to further adapt training for placement of buoys once the farm boundary has been reached.

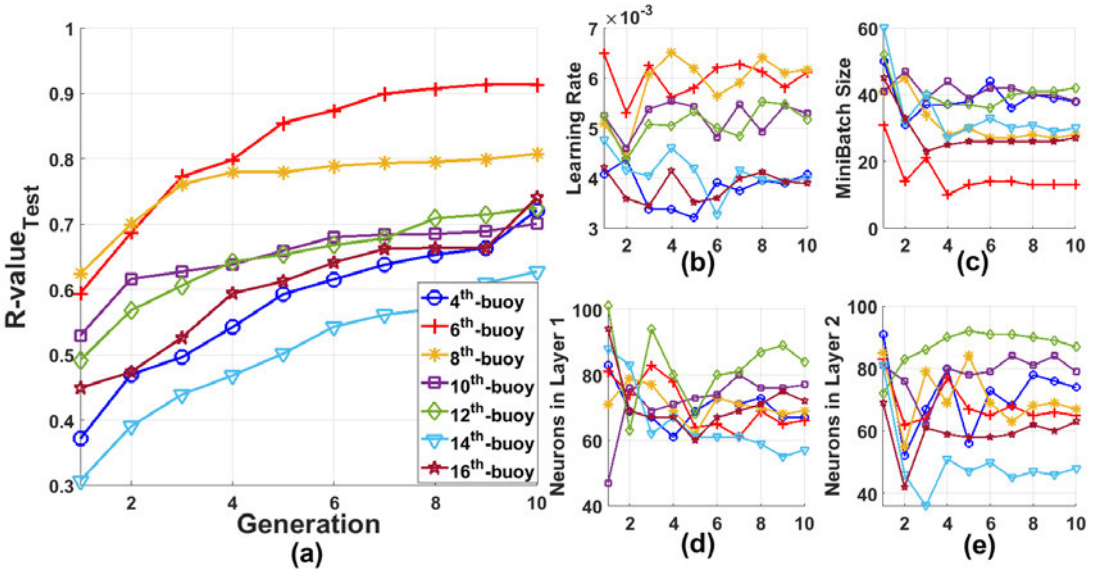

FIGURE 2.31: Evolutionary (GWO) hyper-parameters optimisation: a) The vertical axis is test-set accuracy of the mean best configuration by cross-validation per generation. b) and c) show the optimisation process of the learning rate and minibatch size for estimating the power of the seven buoys. Both d) and e) show the optimised number of neurons in the first and second LSTM layers.

2.4 Optimisation of Large Wave Farms using a Multi-strategy Evolutionary Framework

2.4.1 Synopsis

In the previous sections of this chapter, we discuss a wide range of optimisation algorithms and then propose the most effective heuristic (problem-specific) algorithms in order to optimise the WECs placement. However, the performances of the proposed heuristic approaches are not evaluated by large wave farms. There are some challenges involved in optimising large wave farm in real wave scenarios. These include a difficult and time-consuming initialisation of the first population due to the safety distance constraint, high dimensionality of the search space, very complex hydrodynamic interactions between WECs and the computationally expensive evaluation. In order to deal with the listed challenges, we extend and modify previous works using a new hybrid multi-strategy evolutionary framework which combines smart initialisation, a binary population-based evolutionary algorithm, discrete local search and continuous global optimisation. The statistical results show that, on average, the new multi-strategy evolutionary algorithm outperforms other applied optimisation methods for both 49- and 100-buoy layouts in terms of convergence speed and power production.

Reference

[6] Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2020). Optimisation of Large Wave Farms using a Multi-strategy Evolutionary Framework. Accepted for publication by the Genetic and Evolutionary Computation Conference on 2020.(***this paper has been nominated as a Best Paper Award in the Real World Application (RWA) Track***)

Statement of Authorship

Title of Paper	Optimisation of Large Wave Farms using a Multi-strategy Evolutionary Framework.
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2020). Optimisation of Large Wave Farms using a Multi-strategy Evolutionary Framework. In Proceedings of the Genetic and Evolutionary Computation Conference.

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	70%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	15/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander (10%)		
Contribution to the Paper	Helped advise experimental design and methodology. Gave feedback on preliminary results. Helped revise paper.		
Signature		Date	12/05/20

Name of Co-Author	Nataliia Y.Sergiienko (10%)		
Contribution to the Paper	Developed and coded an objective function for optimisation.		
Signature		Date	13/05/2020

Name of Co-Author	Markus Wagner (10%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

2.4.2 Abstract

Wave energy is a fast-developing and promising renewable energy resource. The primary goal of this research is to maximise the total harnessed power of a large wave farm consisting of fully-submerged three-tether wave energy converters (WECs). Energy maximisation for large farms is a challenging search problem due to the costly calculations of the hydrodynamic interactions between WECs in a large wave farm and the high dimensionality of the search space. To address this problem, we propose a new hybrid multi-strategy evolutionary framework combining smart initialisation, binary population-based evolutionary algorithm, discrete local search and continuous global optimisation. For assessing the performance of the proposed hybrid method, we compare it with a wide variety of state-of-the-art optimisation approaches, including six continuous evolutionary algorithms, four discrete search techniques and three hybrid optimisation methods. The results show that the proposed method performs considerably better in terms of convergence speed and farm output.

2.4.3 Introduction

The use of renewable energy sources continues to exhibit very fast growth of deployment, and it has resulted in savings of more than two gigatonnes of carbon dioxide in 2018 alone [141]. One of the most promising renewable sources is ocean wave energy, which has a high energy density per unit area of ocean, high level of predictability, and potentially high capacity factors [18, 142]. However, compared to wind and solar energy, wave energy is still a nascent field, and research is still very active converter design [7], wave-farm layout, and power-take-off parameters [21, 143].

While there has been significant research on the placement of wave energy converters (WECs) in farms [1, 23, 31, 35, 39], to date, only Wu et al. [39] has considered the design of larger layouts of over 20 converters, using a much-simplified wave energy model.

The research described in this paper extends previous work by using a much more detailed energy model to place buoys in large farms of up to 100 WECs. Due to the much higher number of interactions modelled in such farms this work requires the development of novel, specialised, and highly-efficient search heuristics. Using an improved energy model, we demonstrate the performance of these new algorithms in two contrasting real wave scenarios (Sydney and Perth) and compare their performance to a suite of extant optimisation algorithms.

This paper is organised as follows. In the next section, we survey related work. Section 2.4.5 describes our WEC model. Section 2.4.5.3 formulates the optimisation problem. The proposed optimisation methods are described in Section 2.4.6. The results of the optimisation experiments, including simple landscape analysis, are described in Section 2.4.8. Section 2.4.9 concludes this paper and canvases future work.

2.4.4 Related Work

Placement of WECs in larger farms is a challenging optimisation problem. Hydrodynamic interactions between WECs are complex, which makes evaluation of each potential layout time-consuming [1], ranging from minutes to hours for large farms. Second, due to complex inter-WEC interactions, the search space for this problem is multi-modal – thus requiring global search to be assured of good results. Finally, the high number of decision variables in large farms increases the search space to traverse.

There has been substantial past research into the problem of WEC placement. One of the first studies to optimise WEC layout compared a customised genetic algorithm (GA) with an iterative Parabolic Intersection (PI) method [31] for a small wave farm (five buoys). The GA outperformed PI, but required more evaluations to do so. A more recent position optimisation study [35] compared three search metaheuristics: a custom GA, CMA-ES [89], and glow-worm optimisation [101]), using a simple wave model. The study observed that CMA-ES converges the fastest, while the other models produced slightly better results. Wu et al. [39] considered optimising a large wave farm (25–100 WECs) as an array of fully submerged three-tether buoys using 1+1EA and 2+2CMA-ES. That research found that the 1+1EA with a simple mutation operator performed better than CMA-ES. A limitation of that work was that it was limited to a highly simplified single-wave-direction wave scenario.

In a move toward problem-specific algorithms, Neshat et al. [1] proposed a hybrid optimisation method (LS-NM) combined with a neighbourhood search and Nelder-Mead search. Their study found that LS-NM performed better than generic and custom EAs. However, the wave model applied by that study, though quite detailed, still used an artificial wave scenario and small farm sizes (4 and 16 WECs). More recently, more problem-specific search techniques [3, 23] were, respectively, proposed for optimising WECs positions by utilising a surrogate power model (that is learned on the fly); and hybrid symmetric local search by defining a search sector to speed up the optimisation process. These approaches were also applied to real wave scenarios. For handling this real expensive optimisation problem, a neuro-surrogate optimisation approach was recommended [4] that is composed of a surrogate Recurrent Neural Network (RNN) model and a symmetric

local search. This surrogate model is joined with a metaheuristic (Grey Wolf optimiser) for tuning the model's hyper-parameters. However, these search strategies performance were not evaluated on a large farm.

This article differs from previous work by optimising large layouts using an improved high-fidelity hydrodynamic model to optimise layouts in real wave scenarios. We develop a new hybrid multi-strategy evolutionary algorithm for optimising the positions of buoys in the wave farm to maximise the average total farm power output. For evaluating the new algorithm, we compare its performance to: (1) six continuous off-the-shelf evolutionary methods, (2) four discrete heuristic approaches (3 new), population and individual-based, and (3) three new hybrid EAs (continuous+discrete). We use these methods to optimise wave farms of sizes 49 and 100. We use fine-grained models of contrasting real wave climates, Perth and Sydney, which are located off the southern coast of Australia. The optimisation results demonstrate that the new hybrid multi-strategy search approach produces the best results.

2.4.5 The wave energy converter model

This section describes the energy model for WEC layouts used in this study. The WEC design simulated here is a three-tether spherical buoy based on the highly effective CETO 6 system developed by Carnegie Clean Energy [97].

2.4.5.1 Equation of motion

We model a fully submerged spherical buoy of 5 m radius that is tethered to three power take-off units installed on a seabed. A detailed description of this WEC and its physical parameters can be found in [1].

The motion of each buoy in the farm depends on the forces due to the fluid-structure interaction and the force exerted on the buoy from the PTO system. The generalised equation that describes the motion of all buoys can be written in the frequency domain as:

$$(\mathbf{M}+\mathbf{A})\ddot{\mathbf{X}}+(\mathbf{B}+\mathbf{D}_{pto})\dot{\mathbf{X}}+\mathbf{K}_{pto}\mathbf{X}=\mathbf{F}_{exc}, \tag{2.17}$$

where $\mathbf{X} \in \mathbb{R}^{3N\times 1}$ is a vector of surge, sway and heave displacements of each buoy, $\mathbf{M} = m\mathbb{I}_{3N}$ is a diagonal mass matrix of the wave farm, $\mathbf{A}$ and $\mathbf{B} \in \mathbb{R}^{3N\times 3N}$ are the matrices of hydrodynamic added mass and damping coefficients respectively, $\mathbf{K}_{pto}$ and $\mathbf{D}_{pto} \in \mathbb{R}^{3N\times 3N}$ are the block diagonal matrices of PTO stiffness and damping coefficients respectively, and $\mathbf{F}_{exc} \in \mathbb{R}^{3N\times 1}$ is a vector of excitation forces.

2.4.5.2 Performance assessment

After solving the equation of motion (2.17), we can calculate the power absorbed by the farm in a regular wave of frequency ω that propagates from direction β:

$$p(\omega, \beta) = \frac{1}{2}\dot{\mathbf{X}}^* \mathbf{D}_{pto} \dot{\mathbf{X}} \tag{2.18}$$

where $()^*$ denotes the conjugate transpose of a matrix.

Eq. (2.18) allows us to estimate the power production of a farm assuming that the ocean wave has only one frequency component (like a sinusoidal wave) and propagates only from one direction. In reality, ocean waves travel from different directions and contain multiple frequencies. This behaviour of the wave is usually described by the directional wave spectrum $S(\omega, \beta)$, and power generated by the wave farm in the irregular wave, or sea state (H_s, T_p), can be approximated by:

$$P(H_s, T_p) = \int_0^{2\pi} \int_0^{\infty} S(\omega, \beta) p(\omega, \beta)\, d\omega\, d\beta. \tag{2.19}$$

A potential deployment site (e.g. Perth or Sydney) can be characterised by the wave climate where each sea state has the probability of occurrence $O(H_s, T_p)$. Therefore, using values from Eq. (2.19) and having historical wave climate statistics, it is possible to calculate the annual average power generated by the wave farm at a given location:

$$P_\Sigma = \sum P(H_s, T_p) O(H_s, T_p). \tag{2.20}$$

The Perth and Sydney sites are qualitatively very different: Perth has a small sector from which the prevailing waves arrive, while Sydney's wave directions vary much more. For Perth, this can result in very pronounced constructive and destructive interference, while the same are "smeared" out for Sydney, thus resulting in two very different optimisation scenarios.

Another metric that is widely used to demonstrate the quality of the buoy placement in a farm is called the q-factor. It can be calculated as a ratio of the power generated by the entire farm P_Σ to the sum of power outputs from all WECs if they operate in isolation (not in a farm) P_Σ^i:

$$q = \frac{P_\Sigma}{\sum_i^N P_\Sigma^i} \tag{2.21}$$

Values of $q > 1$ indicate that this particular farm benefits from the constructive interaction between WECs, and more energy can be generated if these WECs operate together.

The MATLAB implementation of this model can be downloaded at [144].

2.4.5.3 Optimisation problem formulation

Based on our WEC model, the problem of positioning N converters on a restricted area of a wave farm $(l \times w)$ in order to maximise the average annual power production P_Σ is:

$$P_\Sigma^* = argmax_{\mathbf{x},\mathbf{y}} P_\Sigma(\mathbf{x}, \mathbf{y})$$

where $P_\Sigma(\mathbf{x}, \mathbf{y})$ is the average power obtained by placements of the buoys in a field at x-positions $\mathbf{x} = [x_1, \ldots, x_N]$ and corresponding y positions $\mathbf{y} = [y_1, \ldots, y_N]$. In our experiments, the number of buoys is $N = 49$ and 100.

Constraints All buoy positions (x_i, y_i) are constrained to a square field of dimensions: $l \times w$ where $l = w = \sqrt{N * 20000}\, m$. This allocates $20000m^2$ of farm-area per-buoy. In addition, the intra-buoy distance must not be less than 50 meters for reasons of safety and maintenance access. For any layout $\mathbf{x}, \mathbf{y}$ the sum-total of the inter-buoy distance violations, measured in metres, is:

$$Sum_{dist} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (dist((x_i, y_i), (x_j, y_j)) - 50),$$

if $dist((x_i, y_i), (x_j, y_j)) < 50$ else 0

where $dist((x_i, y_i), (x_j, y_j))$ is the Euclidean distance between each pair of buoys i and j.

Violations of the inter-buoy distance constraint are handled by applying a steep penalty function: $(\ Sum_{dist}+1)^{20}$ and then applying the Nelder-Mead simplex algorithm over this penalty function to repair the violations in the layout. This approach avoids expensive re-evaluations of the full-wave model that would be required if the penalty function were combined with the full model whilst repairing distance violations. Meanwhile, we handle buoy placements outside of the farm area by moving them back to the farm boundary.

Computational Resources In this paper, we aim to compare several heuristic search methods, for 49 and 100-buoy layouts, in two realistic wave models. Because the search methods apply the interaction model to differing numbers of buoys at a time, it is not feasible to compare methods fairly in terms of a fixed number of model evaluations.

Instead, we use an allocated time budget for each run of three days on dedicated nodes of an HPC platform with 2.4GHz Intel 6148 processors and 128GB of RAM. The software environment running the function evaluations and the search algorithms is MATLAB

R2019. On this platform, 12-fold parallelisation inside of Matlab yields up to 10-fold speedup. All algorithm variants are carefully implemented to make use of the parallelism available.

2.4.6 Optimisation Methods

The algorithms that follow apply three broad strategies. In the first strategy, we optimise in a continuous space using five off-the-shelf evolutionary algorithms. We also use the LS-NM [1] algorithm, which places and fine-tunes one buoy at a time. In the second strategy, we optimise the positions in a discretised grid where the spacing is based on the safety-distance. Here, we consider four different EAs.

Last, we propose a hybrid multi-strategy heuristic that is designed based on our observations that attempts to combine the strengths of the algorithms from the first two strategies.

2.4.6.1 Continuous methods

For the continuous optimisation strategy, we compare six meta-heuristic approaches to optimise all problem dimensions simultaneously:

1. covariance matrix adaptation evolutionary-strategy (CMA-ES) [89, 145] which is an state-of-the-art and self-adaptive EA with the default $\lambda = 12$, and initial $\sigma = 0.25 \times (U_b - L_b)$;
2. (2+2)CMA-ES [39] with the default $\lambda = 2$, and $\sigma = 0.3 \times (U_b - L_b)$;
3. Differential Evolution (DE) [94], a well known global search heuristic using a binomial crossover and a mutation operator of $DE/rand/1/bin$, The population size is adjusted by the $\lambda = 12$ and other control parameters are $F = 0.5$, $P_{cr} = 0.8, 0.9$ respectively for 49 and 100-buoy layouts;
4. Improved Differential Evolution [41], with $\lambda = 12$, and generating mutation vector in the form of $DE/best/1/bin$ with an adaptive mutation operator $F = F_0 \times 2^{e^{1-\frac{G_m}{G_{m+1}-G}}}$, where $F_0 = 0.5$ and G_m is the maximum number of generations and G is the current generation;
5. a simple (1+1)EA as used in [39] that mutates one buoy location in each iteration with a probability of $1/N$ using a normal distribution ($\sigma = 0.1 \times (U_b - L_b)$);

6. Local Search + Nelder-Mead (LS-NM) [1]: which is a fast and effective WEC position optimisation method. Each buoy is placed and optimised one-at-a-time sequentially by sampling at a normally-distributed random offset ($\sigma = 70m$) from the earlier placed buoy position. The sampled position proffering the highest power output, after NM search, is taken.

2.4.6.2 Discrete methods

We test and compare four discrete optimisation methods. All methods place buoys at locations on a grid spaced at the safety distance of 50m. A-priori this discretisation offers advantages in terms of avoiding infeasible layouts and reduced overall search space.

The discrete algorithms used here are:

1. binary Genetic Algorithm (bGA) [36] with $\lambda = 12$, $e_p = 10\%$, $C_r = 80\%$, $M_r = 10\%$, a binary mutation and double point crossover with respect to the number of buoys as a constraint, where e_p, C_r and M_r are the elitism, crossover and mutation rate respectively.

2. Improved binary Differential Evolution (bDE) [146] with the same IDE settings and to construct the mutant vector, formula 2.22 and 2.23 are used;

$$Diff - Vector^j = \begin{cases} 0, & if(X^j_{r1} = X^j_{r2}) \\ X^j_{r1}, & otherwise \end{cases} \tag{2.22}$$

$$Mutant - Vector^j = \begin{cases} 1, & if(Diff - Vector^j = 1) \\ X^j_{G_{best}}, & otherwise \end{cases} \tag{2.23}$$

   where r_1 and $r2$ are the index of two randomly chosen individuals, and G_{best} mentions the best solution number in the current population.

3. Enhanced binary Particle Swarm Optimisation (bPSO) [147, 148] with $\lambda = 12$, and other settings are $C_1 = C_2 = 2$, $\omega_{ini} = 0.9, \omega_{max} = 0.9, \omega_{min} = 0.4$, and ω is linearly decreased to 0.4. the applied transfer function (V-shaped) is represented by Equation 2.24 and the position vector is updated by Equation 2.25.

$$T(v^k_i(t)) = \left| \frac{2}{\pi} \arctan(\frac{\pi}{2} v^k_i(t)) \right| \tag{2.24}$$

$$X^k_i(t+1) = \begin{cases} (X^k_i(t))^{-1} & if\ rand < T(v^k_i(t)) \\ X^k_i(t) & otherwise \end{cases} \tag{2.25}$$

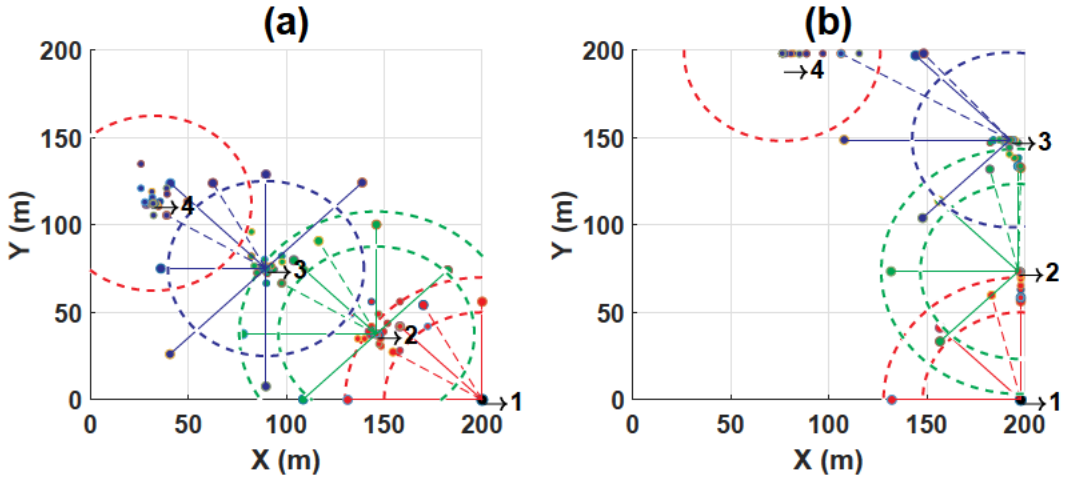

FIGURE 2.32: Symmetric Local Search + Nelder-Mead (SLS-NM) [3] for making the surrogate model. (a) 4-buoy layout in Perth, power=399474 (Watt), (b) Sydney wave model, power=405943 (Watt). The order of the placed and modified buoys position are numbered.

where $v_i^k(t)$ indicates the i^{th} particle velocity at iteration t in the k^{th} dimension.

4. Discrete Local Search (DLS), which is an individual-based evolutionary algorithm similar to a $1+1EA$ with two kinds of mutation step sizes: one discrete interval based on a uniform random distribution that can be vertical, horizontal or diagonal with the same probability, and using a discrete normally-distributed random offset with $\sigma = 3$ (DLS(II)). The mutation probability rate is $\frac{1}{N}$, where N is the number of buoys. In DLS(II), we first generate an initial population, and then the best arrangement is chosen as a start individual.

2.4.7 Hybrid methods

In the third strategy, we proposed a hybrid heuristic framework which consists of five steps:

- **First step**: Applying Symmetric Local Search + Nelder-Mead (SLS-NM) [3]: which places one buoy at a time but offers a more systematic local search. The search starts by placing the first buoy in a pre-determined position of the field based on the dominant wave direction; a symmetric sampling around the neighbourhood of the current buoy is done. Next, Nelder-Mead is applied to optimise the placed buoy arrangement concerning the continuous variables. This process is repeated until the last buoy is placed. In our present case, we use SLS-NM to optimise a local 4-buoy sub-layout which will act as a surrogate layout model. Restricting the

model to interactions between just 4 buoys makes these evaluations very fast and efficient. Figure 2.32 shows the detailed behaviour of this step.

- **Second step**: Discretising the search space (wave farm) based on the size of the surrogate sub-layout model as a smart initialisation method. Thus composing a large wave farm as a mosaic of the small surrogate sub-layouts that produce the most energy.
- **Third step**: Generating the initial population with a sufficient number of well-arranged 4-buoy sub-layouts (smart initialisation) and then encoding to binary representation in preparation for running binary GAs on WEC positions.
- **Fourth step**: Applying discrete optimisation methods on binary representations. We evaluate and compare the performance of three methods (bDE [146], bGA [36] and bPSO [147]).
- **Fifth step**: if the improvement rate of the last populations of the applied optimisation method is low, the rotate procedure is run to perturb sub-layouts and avoid premature convergence. The rotate algorithm mutates a 4-buoy sub-layout by a random clockwise rotation degree with discrete 45 intervals.

  The probability of the applied rotation on each sub-layout is $\frac{1}{N}$.

Using this configurable method, we compare three combinations:

1. SLS-NM + binary GA + Rotate (SLSNM-bGA)
2. SLS-NM + Improved binary DE + Rotate (SLSNM-bDE)
3. SLS-NM + Enhanced binary PSO + Rotate (SLSNM-bPSO)

2.4.7.1 Hybrid Multi-strategy Evolutionary algorithms

The binary-encoded search space in the third hybrid search strategy is discrete. This means that, often, there is still scope to further tune layout locations. To implement this tuning we develop the third hybrid search strategy using a backtracking method for enhancing the buoys position. This backtracking idea is consists of

1. A discrete local search (DLS) for providing a second chance for running a fast neighbourhood exploration of the buoys with a large step size (interval=50m)
2. and a continuous local search (CLS) that uses a 1+1EA for exploring near each buoy using small random normally distributed step size ($\sigma = 20m$, linearly decreased).

Furthermore, the rotation procedure is embedded with the discrete metaheuristic algorithms as a mutation operator which is applied to perturb the best solution after each generation. According to the above descriptions, three Hybrid Multi-strategy Evolutionary algorithms are proposed including

1. SLS-NM + bGA-Rotate + DLS + CLS (MS-bGA)
2. SLS-NM+Improved bDE-Rotate + DLS + CLS (MS-bDE)
3. SLS-NM+Enhanced bPSO-Rotate + DLS + CLS (MS-bPSO)

Algorithm 2.7 describes MS-bDE in detail, where N, N_s, N_b are the buoy numbers, the surrogate model's buoy number (4-buoy layout) and the number of binary decision variables respectively. And also both Tr_1 and Tr_2 are the stopping criteria of 24 (hours) and 48 (hours) respectively.

2.4.8 Experimental study

This section shows detailed optimisation results comparing the 17 variations of search heuristics (six existing methods with and 11 new combinations) described in the previous section. In order to evaluate the performance of the proposed algorithms, we performed a comparative study using two distinct real wave scenarios (Perth and Sydney), and for two different large farm sizes with $N = 49$ and $N = 100$ buoys. For each optimisation method with the configurations above, we execute ten runs. For a set of runs, we tracked performance distributions, and the best layouts were gathered to compare each method.

Table 2.11 shows summary statistics from the experimental runs. The best-obtained results are indicated in bold type. The minimum, maximum, average, median and standard deviation (STD) of the best-produced solutions (power output) for each experiment are reported.

Algorithm 2.7 *MS − bDE*

procedure HYBRID MULTI-STRATEGY EVOLUTIONARY ALGORITHM
Initialisation
$N = 49, 100,\ N_s = 4,\ N_b = N/N_s, N_{Pop} = 12,\ F_0 = 0.5,\ P_{cr} = 0.9, iter = 1$
$size = \sqrt{N * 20000}$ ▷ Farm size
$\vec{S}_s = \{\langle x_1, y_1\rangle, \ldots, \langle x_{N_s}, y_{N_s}\rangle\} = \perp$ ▷ Continuous surrogate position
$\vec{S} = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = \perp$ ▷ Discrete layout position
$\chi_{dis} = \{\langle \vec{S_1}\rangle, \langle \vec{S_2}\rangle \ldots, \langle \vec{S}_{N_{Pop}}\rangle\}$ ▷ Discrete Population
Symmetric Local Search + Nelder-Mead (SLS-NM)
$(energy_s, Array_s) = \mathbf{SLS-NM}([\vec{S_s}])$ ▷ Optimise surrogate model
χ_{dis}^{iter}=**IniFirstPop**$(Array_s, \chi_{dis})$ ▷ Generate initial discrete population
$(Energy, bestEnergy, bestArray) = \mathbf{Eval}(\chi_{dis}^{iter})$ ▷ Evaluate population
χ_b^{iter}=**ConDisBin**(χ_{dis}^{iter}, N_b) ▷ Encode discrete to binary population
Discrete Differential Evolution (bDE)
  while $ImPorate \geq 0.1\%$ & $\sum_{t=1}^{iter} runtime_t \leq Tr_1$ **do**
    for i in $[1, .., N_{Pop}]$ **do** ▷ Mutation
      Generate two rand indexes $r_1, r_2 \in (1, N_{Pop})$, $r_1 \neq r_2 \neq i$
      Compute mutant vector (V_i^{iter}) by Equations. 2.22 and 2.23
      for j in $[1, .., N_b]$ **do** ▷ Crossover
        if $rand \leq P_{cr}$ *or* $j == j_{rand}$ **then** $U_{i,j}^{iter} = V_{i,j}^{iter}$
        else $U_{i,j}^{iter} = \chi_{b_{i,j}}^{iter}$
        end if
      end for
      if $f(U_i^{iter}) \geq f(\chi_{b_i}^{iter})$ **then** $\chi_{b_i}^{iter+1} = U_i^{iter}$ ▷ Selection(Maximisation)
      else $\chi_{b_i}^{iter+1} = \chi_{b_i}^{iter}$
      end if
    end for
    $(bestarray, bestIndex, bestEnergy, ImPorate)$=**Max**$(\chi_b^{iter+1})$
Rotation Operator
    for k in $[1, .., N_b]$ **do**
      if $rand < \frac{1}{N_b}$ **then**
        $(array_{R_k})$=**Rotate**$(bestarray, k)$
      end if
    end for
    $(Energy_R) = \mathbf{Eval}(array_R)$ ▷ Evaluate rotated layout
    $\chi_{b_{bestIndex}}^{iter+1} = \begin{cases} array_R, & \text{if } Energy_R > bestEnergy \\ bestarray, & \text{Otherwise} \end{cases}$
    $iter = iter + 1$, and Update $ImPorate$
  end while
  $(bestarray, bestIndex, bestEnergy, ImPorate)$=**Max**$(\chi_b^{iter})$
Discrete Local Search
  while $ImPorate \geq 0.001\%$ & $\sum_{t=1}^{iter} runtime_t \leq Tr_2$ **do**
    $(array_{dls}, Energy_{dls}) = \mathbf{DLS}(bestarray)$
    if $Energy_{dls} > bestEnergy$ **then**
      $bestarray = array_{dls}$, $bestEnergy = Energy_{dls}$, Update $ImPorate$
    end if
  end while
Continuous Local Search
  while $\sum_{t=1}^{iter} runtime_t \leq 72(hour)$ **do**
    $(array_{cls}, Energy_{cls}) = \mathbf{CLS}(bestarray)$
    $bestarray = \begin{cases} array_{cls}, & \text{if } Energy_{cls} > bestEnergy \\ bestarray, & \text{Otherwise} \end{cases}$
    Update $bestEnergy$
  end while
  return $bestarray, bestEnergy$ ▷ Final Layout and Energy
end procedure

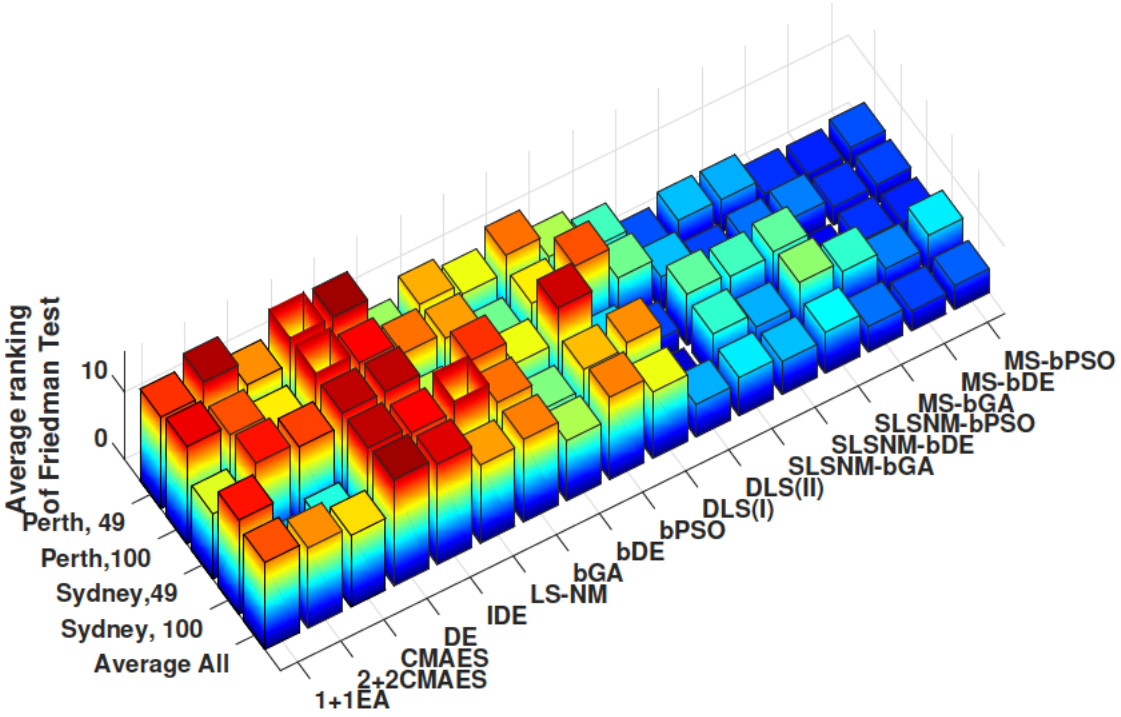

FIGURE 2.33: Average ranking of the Friedman test for performance of the proposed optimisation methods. Among all applied heuristic methods, MS-bDE achieves the best average rank in both wave scenarios and wave farm sizes (2.96).

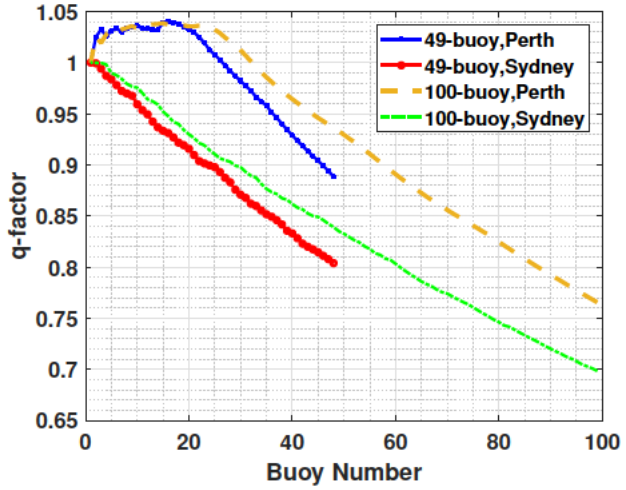

FIGURE 2.34: Evaluation of the q-factor performance of the best 49 and 100-buoy layouts by iteratively removing the buoy with the lowest produced power.

In the Perth wave scenario, the best 49 and 100-buoy layouts are found by MS-bDE. However, we can see the MS-bGA and DLS perform better than other optimisation methods in the Sydney wave regime for 49 and 100-buoy layouts, respectively.

In addition, Figure 2.33 depicts a broad comparison of all proposed optimisation methods by the average ranking of the non-parametric Friedman's test [149] including both real wave scenarios with two different farm sizes and the total average rank of each method

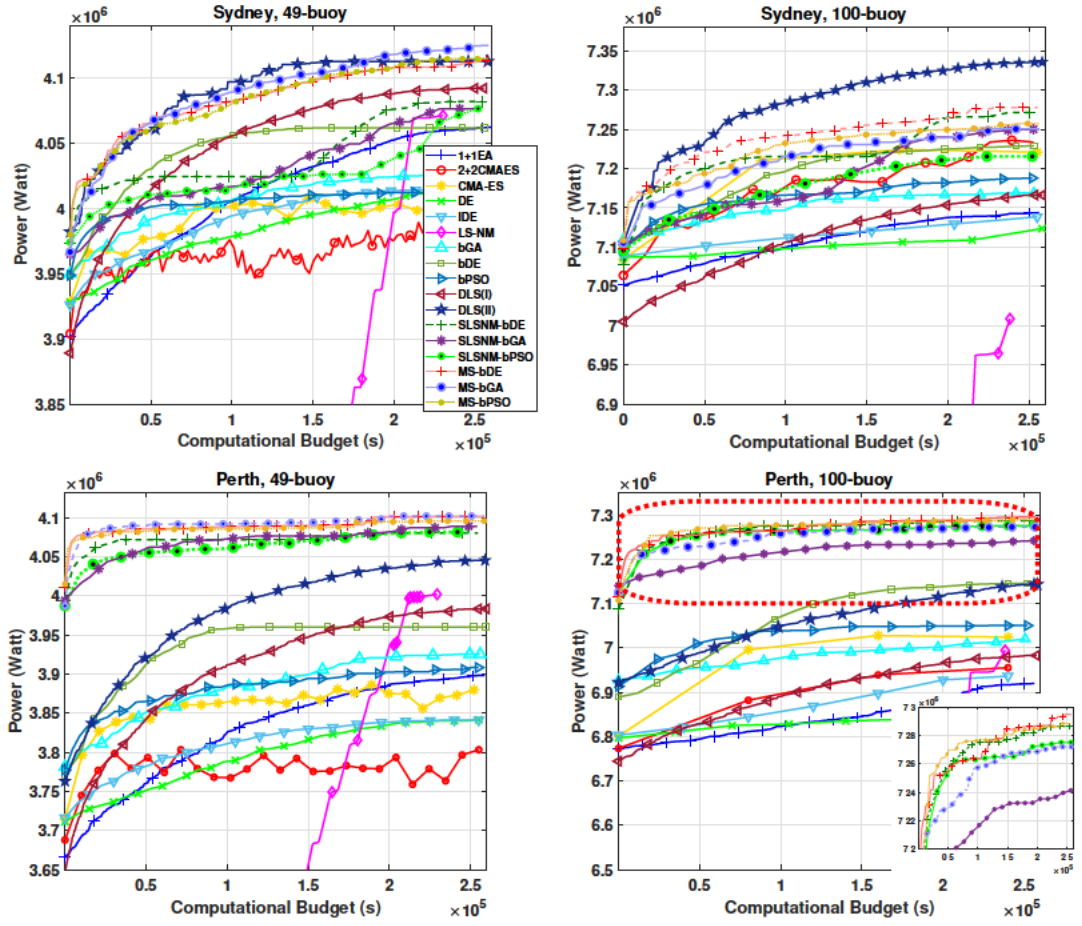

FIGURE 2.35: Evolution and convergence rate of the average power output of the 17 algorithms for two real wave models. A zoomed version of the plots is provided to give a better view of convergence speed of the new proposed algorithms.

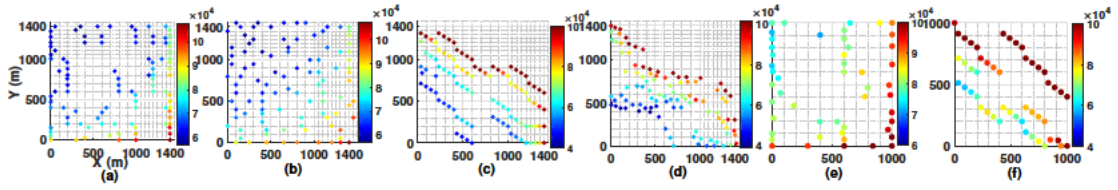

FIGURE 2.36: The best 49 and 100-buoy layouts: (a) Power= 7337922 (Watt), q-factor=0.7 by MS-bDE for 100-buoy in Sydney wave scenario; (b) Power= 7362279 (Watt), q-factor=0.70 for 100-buoy in Sydney wave farm ; (c) Power= 7347403 (Watt), q-factor=0.76 by MS-bDE for 100-buoy in Perth wave farm; (d) Power= 7235425 (Watt), q-factor=0.75 by LS-NM for 100-buoy in Perth; (e) Power= 4145252 (Watt), q-factor=0.80 by MS-bGA for 49-buoy in Sydney ; (f) Power= 4177658 (Watt), q-factor=0.88 by MS-bDE for 49-buoy in Perth wave farm (2.4% more power than LS-NM best 49-buoy layout).

TABLE 2.11: A performance comparison of the tested heuristics for the 49 and 100-buoy cases, based on maximum, minimum, median and mean power output layout of the best solution per experiment.

Perth wave scenario (49-buoy)																	
	1+1EA	**2+2CMAES**	**CMAES**	**DE**	**IDE**	**LS-NM**	**bGA**	**bDE**	**bPSO**	**DLS(I)**	**DLS(II)**	**SLSNM-bGA**	**SLSNM-bDE**	**SLSNM-bPSO**	**MS-bGA**	**MS-bDE**	**MS-bPSO**
Max	3933532	3863159	3946189	3860444	3875783	4080796	3984477	4010788	3951612	4029731	4075129	4112549	4097756	4112602	4143453	**4177659**	4153355
Min	3822270	3819043	3900198	3817360	3774422	3870593	3884613	3914666	3889598	3916594	4034768	4071398	4055033	4065949	4080411	4108205	4081240
Mean	3897954	3847996	3920466	3841577	3841125	4001900	3925487	3960060	3912453	3984165	4045675	4092000	4080173	4082473	4101887	4101260	4095598
Median	3904627	3853485	3923422	3842720	3848962	4008882	3919409	3964299	3912960	3987689	4044025	4090929	4081414	4082183	4099826	4095057	4089500
Std	32160	14659	17904	13118	26134	57673	30027	27647	17177	33047	10639	10535	14291	12556	15447	33245	19794
Perth wave scenario (100-buoy)																	
	1+1EA	**2+2CMAES**	**CMAES**	**DE**	**IDE**	**LS-NM**	**bGA**	**bDE**	**bPSO**	**DLS(I)**	**DLS(II)**	**SLSNM-bGA**	**SLSNM-bDE**	**SLSNM-bPSO**	**MS-bGA**	**MS-bDE**	**MS-bPSO**
Max	6949622	7159987	7106268	6884148	7071418	7235426	7069638	7205581	7115011	7147428	7192402	7293928	7334975	7317723	7337150	**7347403**	7323919
Min	6869548	6893283	6840548	6822275	6830504	6522058	6974571	7031121	7001017	6881014	7096030	7198081	7185982	7201029	7209405	7240886	7233752
Mean	6909165	6976394	7038973	6841196	6936048	6926550	7019850	7145057	7050040	6982706	7144278	7252075	7286727	7275411	7272215	7293645	7287221
Median	6908770	6931793	7058938	6834645	6912788	6888912	7017830	7160194	7036276	6978218	7143539	7250198	7294572	7276700	7259815	7287454	7286829
Std	23522	101719	68757	19143	80329	213896	28773	46510	33182	74593	29736	26291	40569	30945	48338	30656	23286
Sydney wave scenario (49-buoy)																	
	1+1EA	**2+2CMAES**	**CMAES**	**DE**	**IDE**	**LS-NM**	**bGA**	**bDE**	**bPSO**	**DLS(I)**	**DLS(II)**	**SLSNM-bGA**	**SLSNM-bDE**	**SLSNM-bPSO**	**MS-bGA**	**MS-bDE**	**MS-bPSO**
Max	4082524	4036152	4061528	4028337	4062984	4089731	4052984	4078389	4036728	4108751	4134622	4105401	4100089	4084955	**4145252**	4132385	4125968
Min	4046311	4013471	4015919	4002550	3976301	4039525	4006283	4050349	3992467	4070321	4098276	4031291	4047364	4066979	4106192	4097087	4107365
Mean	4062828	4026230	4029509	4014852	4014611	4063963	4028394	4062048	4015029	4092373	4116149	4075606	4082176	4076665	4125113	4113961	4117115
Median	4060505	4029230	4030342	4015482	4011081	4064273	4025310	4059954	4015661	4095077	4113040	4072725	4083311	4076621	4127949	4113672	4117134
Std	9580	7808	12500	8591	20268	18106	13725	7924	11643	10140	10814	19897	13306	4946	12113	9324	5856
Sydney wave scenario (100-buoy)																	
	1+1EA	**2+2CMAES**	**CMAES**	**DE**	**IDE**	**LS-NM**	**bGA**	**bDE**	**bPSO**	**DLS(I)**	**DLS(II)**	**SLSNM-bGA**	**SLSNM-bDE**	**SLSNM-bPSO**	**MS-bGA**	**MS-bDE**	**MS-bPSO**
Max	7143849	7325364	7292118	7179529	7195009	7182442	7209740	7285926	7229868	7246878	**7362279**	7288882	7338191	7290958	7300092	7337922	7309508
Min	7117313	7225156	7088371	7094498	7114283	6770261	7128255	7132653	7159246	7048150	7307167	7210217	7240062	7167618	7209436	7247446	7203600
Mean	7143849	7268166	7242833	7123588	7138670	7008764	7168951	7228751	7187912	7166332	7335497	7246161	7271553	7228570	7250348	7277918	7257887
Median	7140541	7263653	7266642	7121216	7132516	7025996	7164768	7236813	7183483	7172586	7339777	7245065	7267610	7237676	7233001	7274685	7255276
Std	18069	34055	64716	24211	25094	131454	25026	41309	20879	57189	18211	22690	31592	42732	32945	26909	33889

in all case studies. It can be seen that, overall, MS-bDE produces the best optimisation performance.

In Figure 2.35, in all configurations of the Perth wave model, three hybrid and three multi-strategy methods converge very fast and still outperform the other methods. It is notable that these six proposed methods start the optimisation process with a high power output solution due to the smart initialisation technique described in Section 2.4.7. Looking more closely at Figure 2.35, we can see that all discrete optimisation approaches converge faster than the continuous algorithms on average. Furthermore, because of the embedding of the rotation operator with the binary EAs, the multi-strategy techniques are able to converge faster than the hybrid methods, especially in the initial iterations. In terms of other algorithms, in the Sydney wave model, the performance of the DLS is strong ($N = 100$) and outperforms other methods in terms of the convergence rate and the produced power. However, we can see that in the smaller farm, the performance of multi-strategy EAs are competitive, and MS-bGA performs better than other optimisation methods in the final iterations.

Some of the most productive 49 and 100-buoy layouts are presented by Figure 2.36 from all the runs in the two scenarios. The absorbed power of buoys is characterised by their colour. It can be seen that the best layouts in Perth are multi-row diagonal arrangements; however, this trend is different in the Sydney wave site where the optimisation method pushes some buoys to the farm boundaries.

Lastly, to further investigate the hydrodynamic interactions between buoys in the best layouts, we perform two different analyses.

In the first analysis, we iteratively remove the buoy with the lowest absorbed power and evaluate the performance of the layout. While this experiment focuses on the least-performing buoy, the interactions of these buoys might be beneficial for the wave farms nevertheless. Figure 2.34 shows that a lot of constructive interference is exploited in both the 49 and 100 buoy Perth scenario (up to the 26th buoy), while the marginal improvement from adding buoy's declines after that. For Sydney, there is an almost uniform decline in marginal performance from the start.

The second analysis of the best layouts selects the buoy with the highest power, removes it, and then maps the landscape using a 25-meter grid. We record both the absorbed power of the buoy and the total wave farm power output per each sample. Figure 2.37 shows the power landscape analysis of this experiment.

Note that the gaps are the infeasible areas around the already-placed buoys. The subplots (b) and (d) indicate a multimodal and complex power landscape, for the placement of the last of the 49 buoys, especially for Sydney.

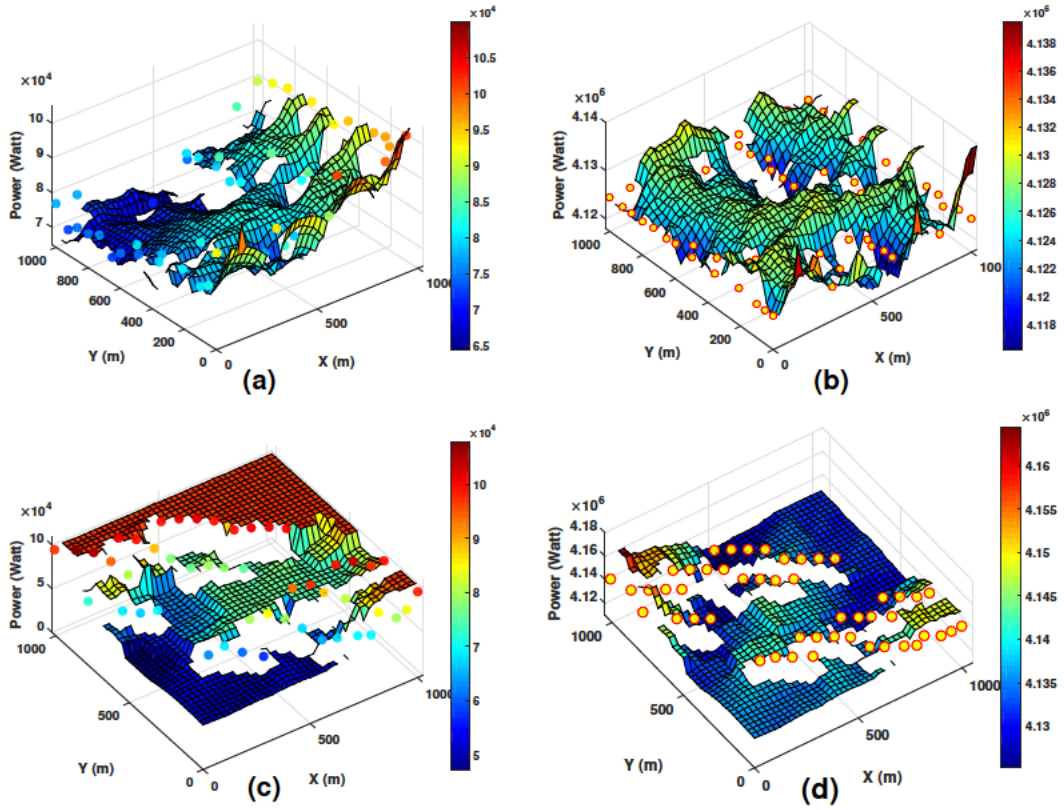

FIGURE 2.37: Power landscape analysis of the best 49-buoy layouts. (a) Sydney - wave state - grid sampling the power extracted by the final buoy. (b) Sydney, *total* energy extracted with grid sampling of the final buoy position (c) Perth-wave state - grid sampling of last buoy's power. (d) grid sampling of total power w.r.t last buoy's position. No samples are made within the safe distance of already-placed buoys. The power of each buoy is characterised by a specific colour in both (a) and (c).

In order to report on the distribution of the performance of the different approaches across 10 independent runs, the box-plots (Figure 2.38) is represented. Figure 2.38 shows and highlights the considerable performance of the proposed multi-strategy optimisation framework compared with other optimisation techniques in the large wave farms problem.

Meanwhile, All implemented codes and auxiliary materials are publicly available: `https://cs.adelaide.edu.au/~optlog/research/energy.php`.

2.4.9 Conclusions

In this paper, we have proposed, assessed, and systematically compared 17 different optimisation approaches for optimising the arrangement of large wave farms with 49 and 100 generators in two real wave regimes (Sydney and Perth). This study comprised three new hybrid algorithms, each with three variants as a multi-strategy EA framework customised to this field.

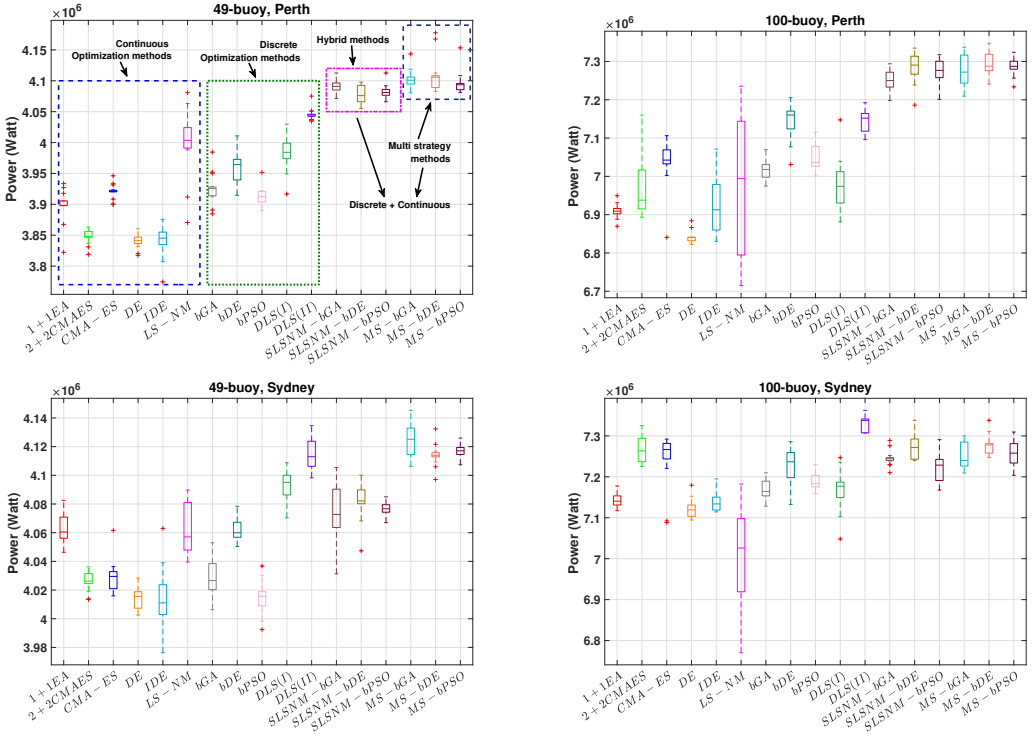

FIGURE 2.38: The comparison of the optimisation algorithms performance for 49 and 100-buoy layouts in Sydney and Perth wave models. The optimisation results present the best solution per each experiment. (10 independent runs per each method)

This optimisation problem is challenging in terms of the cost of its evaluation model and the large multimodal search landscape. Our new framework addresses this problem through careful problem decomposition into sub-farms, the use of discrete search spaces and a customised mutation operator (rotation).

The statistical results indicate that the new multi-strategy evolutionary algorithm consisting of symmetric local search and Nelder Mead search, combined with an embedded rotation operator, plus an improved binary DE and a hybrid backtracking strategy (DLS+CLS) performs better than other applied optimisation methods on average. In our experiments, this method overcomes other state-of-the-art algorithms, for both 49 and 100-buoy layouts, in terms of convergence speed and power production.

Future work could explore other optimisation dimensions, including considering other effective buoy designs and power take-off system settings.

Acknowledgements

We would like to offer our special thanks to Dr.Hansen, Dr.kalami and Dr.Mirjalili for sharing and publishing their valuable source codes and also thank Mr.Roostapour for his

good comments. Additionally, This research is supported with supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

Chapter 3

Control Optimisation of Wave Energy Converters

3.1 A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters

3.1.1 Synopsis

In the previous chapter, we considered and proposed several new heuristic (problem-specific) methods for optimising the arrangement of wave energy converters in order to harness the maximum produced power. However, the performance of a wave farm can be significantly increased when we adjust adequate power take-off (PTO) parameters rather than assigning random PTO values. Consequently, our main goal in this chapter is to explore, investigate and propose new insights into optimisation of a wave farm by configuring WEC locations and power-take-off (PTO) settings for each WEC. In this paper, our optimisation methods apply three comprehensive strategies. The first strategy is to optimise all decision variables at once. 'All-at-once' means that both placement and PTO settings of all WECs in the farm should be optimised simultaneously. Here, we evaluate and compare the performance of five well-known off-the-shelf optimisation methods that apply this strategy. In the second optimisation strategy, we focus on optimising the placements and PTO parameters of all WECs in an alternating cooperative algorithm. We propose four new methods that apply this strategy. Finally, the third strategy, (adopted in [1]) is to place and optimise each WEC in a sequential layout. Here, we apply this strategy in order to propose three new hybrid heuristics. We assess the effectiveness of these approaches in two real wave scenarios (Sydney and Perth) with farms of two different scales. Results indicate that the combination of symmetric local search, a Nelder-Mead Simplex direct search combined and a back-tracking optimisation strategy can outperform previously specified search techniques by up to 3%.

Reference

[3] Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2019, July). A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters. In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1293-1301).(***Received The Best Paper Award in the Real World Application (RWA) Track*** [150])

Statement of Authorship

Title of Paper	A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Alexander, B., Sergiienko, N. Y., & Wagner, M. (2019, July). A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters. In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1293-1301).

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	70%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	15/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander (10%)		
Contribution to the Paper	Helped advice experimental design and methodology. Gave feedback on process and initial reporting of results. Helped revised paper.		
Signature		Date	12/05/20

Name of Co-Author	Nataliia Y.Sergiienko (10%)		
Contribution to the Paper	Developed and coded an objective function for optimisation.		
Signature		Date	13/05/2020

Name of Co-Author	Markus Wagner (10%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

3.1.2 Abstract

Ocean wave energy is a source of renewable energy that has gained much attention for its potential to contribute significantly to meeting the global energy demand. In this research, we investigate the problem of maximising the energy delivered by farms of wave energy converters (WEC's). We consider state-of-the-art fully submerged three-tether converters deployed in arrays. The goal of this work is to use heuristic search to optimise the power output of arrays in a size-constrained environment by configuring WEC locations and the power-take-off (PTO) settings for each WEC. Modelling the complex hydrodynamic interactions in wave farms is expensive, which constrains search to only a few thousand model evaluations. We explore a variety of heuristic approaches including cooperative and hybrid methods. The effectiveness of these approaches is assessed in two real wave scenarios (Sydney and Perth) with farms of two different scales. We find that a combination of symmetric local search with Nelder-Mead Simplex direct search combined with a back-tracking optimisation strategy is able to outperform previously defined search techniques by up to 3%.

3.1.3 Introduction

Environmental concerns and declining costs are favouring the widespread deployment of renewable electricity generation. Wave energy converters (WECs), in particular, offer strong potential for growth because of their high capacity factors and energy densities compared to other renewable energy technologies [18]. However, WECs are relatively new technology, which presents design challenges in the development of individual converters and in the configuration of farms consisting of arrays of WECs. The WEC model considered in this research is similar to a new generation of CETO systems that were introduced and developed by the Carnegie Clean Energy company [97]. The CETO system is composed of an array of fully submerged three-tether converters (buoys) [151]. The aim of this research is to maximise the absorbed power of an array (farm) of these buoys. In maximising the power produced by such an array the key factors are [21]: (1) the layout of WECs in the sea, (2) the power-takeoff (PTO) parameters for each WEC, (3) wave climate (wave frequencies and directions) of a specific test site, and (4) the number of WECs.

The combined search space for optimising WECs placements and PTO settings is non-linear and multi-modal. Furthermore, because of complicated and extensive hydrodynamic interactions among generators, the evaluation of each farm configuration is expensive, taking several minutes in larger farms. These factors make the use of smart and specialised meta-heuristics attractive for this problem.

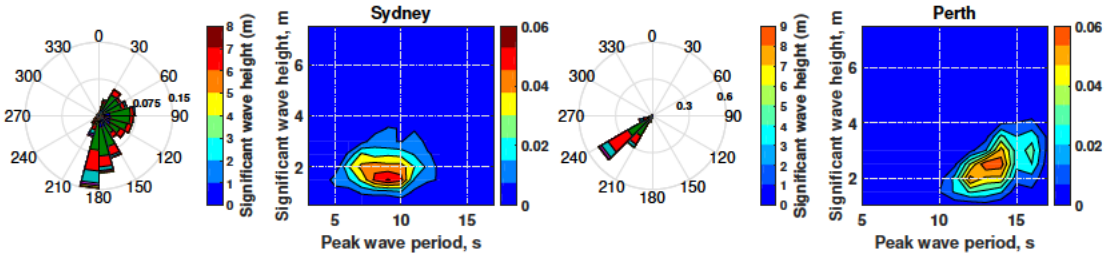

FIGURE 3.1: Wave data for two test sites in Australia: (a) Sydney and (b) Perth. These are: the directional wave rose (left) and wave scatter diagram (right).

One early work [31] used a simple uni-directional wave model to compare a custom GA with an iterative Parabolic Intersection (PI) method for placing 5 buoys. Both of these search methods deployed a high number of evaluations (37000). A recent study by Ruiz et al. [35] used another simple wave model to compare a specialised GA, CMA-ES [89], and glow-worm optimisation [101] in placing buoys at positions in a discrete grid. The study found that CMA-ES converged faster than the other two methods, but ultimately produced poorer-performing layouts. In other recent work, Wu et al. [39] studied two EAs: a 1+1EA and CMA-ES for optimising buoy's positions in an array of fully submerged three-tether WECs using a simplified uni-directional irregular wave model. That work found that the 1+1EA with a simple mutation operator performed better than CMA-ES. More recently, Neshat et al. [1] applied a more detailed wave scenario (seven wave directions and 50 wave frequencies) to evaluate a wide range of generic and custom EAs for the buoy placement. This study found that a hybrid approach (local search + Nelder-Mead) achieved better 4 and 16-buoy arrangements in terms of power produced. However, the model used by that work still embedded an artificial wave scenario. Moreover, the optimisation did not attempt to tune buoy PTO parameters to maximise the power produced by each buoy. The optimisation of PTO parameters presents another dimension for WEC farm optimisation. PTO parameters control how WECs oscillate with the frequency of incoming waves. Maximum efficiency is achieved when converters resonate with the sea waves. However, maintaining a resonant condition is not easy because real sea waves consist of multiple different frequencies [152]. In work optimising the PTO damping of one converter (CETO 6), Ding et al. [153] applied the maximum power point tracking (MPPT) control method which is a simple gradient-ascent algorithm for the online-optimisation of the deployed WEC. The results show that the MPPT damping controller can be more effective and robust than a fixed-damping system. However, when the buoy number is increased the optimisation process becomes more complicated because of the hydrodynamic interactions between buoys. In later work Abdelkhalik et al. [38] used a version of the hidden genes genetic algorithm (HGGA) to control PTO parameters. While this work raised the effective energy harvested the algorithm was not compared to other methods.

In this paper, we develop a new hybrid Evolutionary framework for simultaneously optimising both placement and PTO parameters of a wave farm. We study a broad range of meta-heuristic approaches: (1) five well-known off-the-shelf EAs, (2) four alternating optimisation ideas, and (3) three hybrid optimisation algorithms. Additionally, two new real wave scenarios from the southern coast of Australia (Perth and Sydney) with a high granularity of wave direction is used (Figure 3.1) to evaluate and compare the performance of the proposed methods. According to our optimisation results, a new hybrid search heuristic combining symmetric local search with Nelder-Mead simplex direct search, coupled with a backtracking strategy outperforms other proposed optimisation methods in terms of the power output and computational time.

The rest of this paper is arranged as follows. Section 3.1.4 formulates the WEC model. Section 3.1.5 gives the details of the optimisation problem. The search methods are explained in Section 3.1.6 and a brief characterisation of the fitness landscape is given. We present our comparative studies and experimental results in Section 3.1.7. Finally, Section 3.1.9 concludes this paper.

3.1.4 Model for wave energy converters

In this paper, we consider a fully submerged three-tether buoy model with each tether fastened to a converter installed on the seabed. We assume an optimal tether angle of 55 degrees, which was previously observed to maximise the extraction of energy from heave and surge motions [90]. Other features of the wave energy converters (WECs) used in this investigation, such as physical dimensions and submergence depth, can be found in [1].

3.1.4.1 Power Model

In the WEC model used here, linear wave theory is used to calculate the system dynamics [91]. This model includes three different key forces:

1. The wave excitation force ($F_{exc,p}(t)$) combines the incident and diffracted waves forces from generators in a fixed location.

2. The radiation force ($F_{rad,p}(t)$), derived by the oscillating body due to their motion independent of incident waves.

3. Power take-off (PTO) force ($F_{pto,p}(t)$) is the control force applied to the buoy from the PTO machinery.

Through these forces, the buoys can affect each other's output through hydrodynamic interactions. The complex nature of these interactions, which can either be constructive or destructive, makes the calculation of farm layout and PTO parameter settings a challenging optimisation problem. The dynamic equation that describes a buoy motion in ocean waves has the form:

$$M_p\ddot{X}_p(t) = F_{exc,p}(t) + F_{rad,p}(t) + F_{pto,p}(t) \tag{3.1}$$

where M_p is the mass matrix of a p_{th} buoy, $X_p(t)$ is the buoy displacement expressed as surge, heave and sway. Finally, the power take-off system is modeled as a linear spring-damper system. For each mooring line two control factors are involved: the damping D_{pto} and stiffness K_{pto} coefficients. Therefore, Equation (3.1) can be written in a frequency domain for all WECs in a farm as:

$$\hat{F}_{exc,\Sigma} = ((M_\Sigma + A_\sigma(\omega))j\omega + B_\sigma(\omega) - \frac{K_{pto,\Sigma}}{\omega}j + D_{pto,\Sigma})\ddot{X}_\Sigma \tag{3.2}$$

The hydrodynamic parameters $(A_\Sigma(\omega))$ and $B_\Sigma(\omega)$) are calculated from the semi-analytical model described in [92]. In addition, $K_{pto,\Sigma}$ and $D_{pto,\Sigma}$ are control factors, described above, which can be adjusted to maximise the power output of each buoy. The total power output of the layout is computed by Equation (3.3):

$$P_\Sigma = \frac{1}{4}(\hat{F^*}_{exc,\Sigma}\ddot{X}_\Sigma + \ddot{X^*}_\Sigma\hat{F}_{exc,\Sigma}) - \frac{1}{2}\ddot{X^*}_\Sigma B\ddot{X^*}_\Sigma \tag{3.3}$$

Additionally, the q-factor (q) of the array measures the efficiency of a entire wave farm as compared to the power output from N isolated WECs. For a given layout, the q-factor can be calculated as:

$$q = \frac{P_\Sigma}{\sum_{i=1}^{N} P_i}. \tag{3.4}$$

$q > 1$ indicates constructive interference between WECs. The main purpose of this study is maximising the total power output: P_Σ for N buoys within a constrained farm area.

3.1.5 Optimisation problem formulation

The formulation of the optimisation problem in this paper can be declared as:

$$P_\Sigma^* = argmax_{X,Y,K_{pto},D_{pto}} P_\Sigma(X, Y, K_{pto}, D_{pto})$$

where $P_\Sigma(X, Y, K_{pto}, D_{pto})$ is the mean power obtained by placements and PTO parameters of the buoys in a 2-D coordinate system at x-positions: $X = [x_1, \ldots, x_N]$,

y-positions: $Y = [y_1, \ldots, y_N]$ and corresponding Power Take-off parameters including $K_{pto} = [k_1, \ldots, k_N]$ and $D_{pto} = [d_1, \ldots, d_N]$. In the experiments here $N \in \{4, 16\}$.

Constraints All buoy locations (x_i, y_i) are constrained to a square search space $S = [x_l, x_u] \times [y_l, y_u]$: where $x_l = y_l = 0$ *and* $x_u = y_u = \sqrt{N * 20000}\, m$. This allocates $20000m^2$ of farm-area per-buoy. Moreover, a safety distance for maintenance vessels must be maintained between buoys of at least 50 meters. For spring and damper coefficients the boundary constraints are $d_l = 5 \times 10^4, d_u = 4 \times 10^5$ and $k_l = 1, k_u = 5.5 \times 10^5$. For any array X, Y the sum-total violations of the inter-buoy distance calculated in meters, is:

$$Sum_{dist} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (dist((x_i, y_i), (x_j, y_j)) - 50),$$
$$\text{if } dist((x_i, y_i), (x_j, y_j)) < 50 \text{ else } 0$$

where $dist((x_i, y_i), (x_j, y_j))$ is the Euclidean distance between buoys i and j. The penalty function of the power output (in Watts) is computed by $(Sum_{dist} + 1)^{20}$. The penalty strongly encourages feasible buoy placements. This penalty is also used to handle farm-boundary constraints. For the D_{pto} and K_{pto} parameters, we handle constraint violations by setting the parameter to the nearest valid value.

Computational Resources In this paper, we aim to compare a various heuristic search methods, for 4 and 16 buoy arrays, in two realistic wave scenarios. We allocate a time budget for each optimisation run of three days on dedicated platform with a 2.4GHz Intel 6148 processor running 12 processes in parallel with 128GB of RAM. Note, that where the search heuristic allows, we tune algorithm settings to utilise this time budget. The software environment running the function evaluations and the search algorithm is MATLAB R2017. On this platform, parallelisation provides up to 10 times speedup.

3.1.6 Optimisation Methods

In this research, our search methods employ three broad strategies. The first strategy is to optimise all decision variables at once. This means that for a 16-buoy farm we search in 16×4 dimensions simultaneously. Here, we test five heuristics that apply this strategy. The second strategy is to optimise the positions and PTO parameters of all buoys in an alternating cooperative algorithm [154]. We test four different methods that apply this strategy. Finally, the third strategy, used in [1] is to place and optimise each buoy in sequence. Here, we deploy this strategy for three hybrid EAs. Details of the algorithms tested for each strategy follow.

3.1.6.1 Evolutionary Algorithms (All-at-once)

For the first strategy, five well-known off-the-shelf EAs are deployed to simultaneously optimise all problem dimensions (Positions+PTOs). These EAs are: (1) covariance matrix adaptation evolutionary-strategy (CMA-ES) [89] with the default $\lambda = 12$, for 4-buoy layouts and and $\lambda = 16$ for 16-buoy layouts; (2) Differential Evolution (DE) [94], with parameter settings of $\lambda = 50, 30$, respectively for 4 and 16-buoy layouts, and $F = 0.5$, $P_{cr} = 0.5$; (3) a (1+1)EA [95] that mutates buoys' location and PTO parameters with a probability of $1/N$ using a normal distribution ($\sigma = 0.1 \times (U_b - L_b)$); (4) Particle Swarm optimisation (PSO) [42], with λ= DE settings, $c_1 = 1.5, c_2 = 2, \omega = 1$ (linearly decreased); (5) Nelder-Mead simplex direct search (NM) [96] is combined with a mutation operator (Nelder-Mead+Mutation or NM-M). The mutation operation is applied when the NM has converged to a solution before exhausting its computational budget, so that it can explore other parts of the solution-space (Algorithm 3.1).

Algorithm 3.1 NM+Mutation

procedure NELDER-MEAD + MUTATION (ALL DIMS)
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size
$S = \{\langle x_1, y_1, k_1, d_1\rangle, \ldots, \langle x_N, y_N, k_N, d_N\rangle\}$ ▷ Positions&PTOs
$bestEnergy = 0$ ▷ Best energy so far
$bestLayout = [\mathbf{S}]$ ▷ Best layout so far
$EIRate = 0$ ▷ Energy Improvement rate
Iterative search
  while *stillTime()* **do**
    $(S', energy) = NM\_Search(\mathbf{S}, MaxEval)$ ▷ Local search
    $EIRate = ComputeEIRate(energy, bestEnergy)$
    if $energy > bestEnergy$ ***then***
      $bestEnergy = energy$ ▷ *Update energy*
      $bestLayout = S'$ ▷ *Update layout*
      $S = S'$
    end if
    if $EIRate = 0$ ***then***
      while $(EIRate = 0)$ ***do***
        $S' = randn(\sigma) + S$ ▷ *new buoys Position&PTO*
        $energy = Eval(S')$
        $EIRate =$ ComputeEIRate$(energy, bestEnergy)$
      end while
      if $energy > bestEnergy$ **then**
        $bestEnergy = energy$ ▷ Update energy
        $bestLayout = S'$ ▷ Update layout
        $\mathbf{S} = S'$
      end if
    end if
  end while
  return $bestLayout, bestEnergy$ ▷ Final Layout
end procedure

Algorithm 3.2 CMAES+NM

procedure (2+2)CMA-ES + NELDER-MEAD (ALL DIMS)
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size
$NPop = 2$ ▷ Population size
$S = \{\langle x_1, y_1, k_1, d_1\rangle, \ldots, \langle x_N, y_N, k_N, d_N\rangle\}$ ▷ Positions&PTOs
$\langle S_1, S_2\rangle = Decompose(S)$ ▷ Decomposing
$S_1 = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = \perp$ ▷ Positions
$S_2 = \{\langle k_1, d_1\rangle, \ldots, \langle k_N, d_N\rangle\} = \perp$ ▷ PTO parameters
$\mathbf{Pop} = initPopulation(\{\mathbf{S_1}, \mathbf{S_2}\}, NPop)$
$bestEnergy = 0$ ▷ Best energy so far
$bestPosition = [S_1]$ ▷ Best Position so far
$bestPTO = [S_2]$ ▷ Best PTO parameters so far
$MaxEval = MaxIterC \times NPop$
Cooperative search
  while *stillTime()* **do**
    Position Optimisation
    $(\mathbf{Pop_{S_1}}, energies) = 2{+}2CMA\text{-}ES(\mathbf{Pop}, MaxIterC)$
    $\langle bestPosition, bestIndex\rangle = FindBest(\mathbf{Pop_{S_1}}, energies)$
    PTO Optimisation
    $(bestEnergy, bestPTO) = \text{NM}(\mathbf{Pop}(bestIndex), MaxEval)$
    $\mathbf{Pop_{S_2}}(bestIndex) = bestPTO$ ▷ *Update best solution*
  end while
  return $bestPosition, bestPTO, bestEnergy$
end procedure

3.1.6.2 Alternating optimisation methods (Cooperative ideas)

Optimising both positions and PTO parameters of a WEC array simultaneously can be challenging because of the high number of dimensions and heterogeneous kinds of variables. There is a natural division of variables into two subsets which might, at least in part, be optimised separately. In this section, we describe a set of alternating optimisation techniques which combine one evolutionary algorithm idea such as CMA-ES, DE, and 1+1EA, with Nelder-Mead. In addition, a cooperative, Dual-DE (DE+DE), algorithm is also described. The details of each are given next.

(2+2)CMA-ES + Nelder-Mead This alternating strategy applies CMA-ES with $\mu = \lambda = 2$ for $iter = 25$ iterations to optimise buoy positions. Then the best solution is selected and NM is applied to PTO settings for $iter * \lambda$ iterations. This improved setting is then given to the CMA-ES population for another round of optimisation. The CMA-ES and NM optimisation processes are alternated until the time budget expires. Algorithm 3.2 shows the process of the CMAES-NM approach.

DE + Nelder-Mead (DE-NM) This method alternates DE, for buoy-positions, and NM for PTO parameters, using the same iteration settings as above until the time budget runs out.

1+1EA + Nelder-Mead (1+1EA-NM) This method alternates a 1+1 EA, for buoy positions, and NM, for PTO parameters until the time budget runs out. The iteration settings for the 1+1EA are, respectively, 200 and 50 times, for 4 and 16-buoy layouts. The same limits are also used for the NM optimisation rounds.

Dual-DE This method uses the same parameter settings as described for DE in subsection 3.1.6.1 to optimise both buoy positions and PTO parameters in parallel. After *iter* iterations the improved values from the positional and PTO optimisations are exchanged. This iterative pattern continues until the time budget runs out.

3.1.6.3 Hybrid optimisation algorithms

In other WEC-related research [1], it was found that applying local search around the neighborhood of previously placed buoys could help exploit constructive interactions between buoys. The following methods exploit this observation by placing and optimising the position and PTO parameters of one buoy at a time.

Local Search + Nelder-Mead(LS-NM) This method places buoys sequentially. The position of each buoy placement is optimised by sampling at a normally-distributed random offset ($\sigma = 70m$) from the previous buoy position. The sampled location giving the highest output is chosen. In our experiments we try three different numbers of samples: ($N_s = 2^4, 2^5$ *and* 2^6). After the best position is selected, we optimise the PTO parameters of the last placed buoy using N_s iterations of Nelder-Mead search. This process is repeated until all buoys are placed. Note that, the *Eval* function of LS-NM is parallelised on a per-wave-frequency basis. An example of 16-buoy layout that is built by LS-NM(16s) and the sampling process used to build it, is shown in Figure 3.4(a). The details of the proposed method can be seen in Algorithm 3.3.

Algorithm 3.3 $LS + NM$

procedure LOCAL SEARCH + NELDER-MEAD (2 DIMS)
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size
$S = \{\langle x_1, y_1, k_1, d_1\rangle, \ldots, \langle x_N, y_N, k_N, d_N\rangle\}$ ▷ Positions&PTOs
$\langle S1, S2\rangle = Decompose(S)$ ▷ Decomposing
$S1 = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = \perp$ ▷ Positions
$S2 = \{\langle k_1, d_1\rangle, \ldots, \langle k_N, d_N\rangle\} = \perp$ ▷ PTO parameters
$S1_{(1)} = \{\langle size/2, 0\rangle\}$ ▷ first buoy position
$S2_{(1)} = \{\langle rand \times Max_k, rand \times Max_d\rangle\}$ ▷ first buoy k and d
$(S2_{(1)})$=$NM(S1_{(1)}, S2_{(1)}, MaxEN)$ ▷ Optimise first buoy PTO
$bestPosition = S1_{(1)}$; $bestPTO = S2_{(1)}$
  for i in $[2, .., N]$ **do**
    $iters = MaxSN$ ▷ Number of local samples
    $bestEnergy = 0$;
    Position Optimisation
    for j in $[1, .., iters]$ **do**
      while *not feasible position* **do**
        $tPos = randn(\sigma) + S1_{(i-1)}$ ▷ new buoy position
      end while
      $energy = Eval([S1_{(1)}, \ldots, S1_{(i-1)}, tPos])$
      if $energy > bestEnergy$ **then**
        $S1_{(i)} = tPos$ ▷ Update last buoy position
        $bestPosition = [S1_{(1)}, \ldots, S1_{(i-1)}, S1_{(i)}]$
        $bestEnergy = energy$
      end if
    end for
    PTO Optimisation
    $(S2_{(i)}, energy)$=$NM(bestPosition, S2_{(i-1)}, MaxEN)$
    if $energy > bestEnergy$ **then**
      $bestPTO = [S2_{(1)}, \ldots, S2_{(i-1)}, S2_{(i)}]$
      $bestEnergy = energy$
    end if
  end for
  return $bestPosition, bestPTO, bestEnergy$ ▷ Final Layout
end procedure

Symmetric Local Search + Nelder-Mead (SLS+NM(2D)) This method also places one buoy at a time, but performs a more systematic local search. The search starts by placing the first buoy in the middle of the bottom of the field and then uses NM to optimise the PTO parameters for 25 iterations.

For each subsequent buoy placement, eight local samples are made in different sectors starting at angles: $\{angles = [0, 45, 90, ..., 315]\}$ and bounded by a radial distance of between 50 (safe distance) and $50 + R'$. Within each sector a buoy position is sampled uniformly. Our strategy for handling infeasible solutions is that we refuse them and if all

symmetric solutions are infeasible, a feasible layout is produced using uniform random sampling.

After finding the best sample among the eight local samples, two extra samples are done for increasing the resolution of the search direction. The angles of these two samples are $\pm$ 15^o plus the best angle sample. The candidate position is then selected from the 8 original samples plus these two extra samples based on the buoy's energy output.

In the next step a check is done to see if the PTO optimisation process for the previously placed buoy (using NM) had a high percentage improvement in its last step. A large improvement indicates that there is scope to improve energy production, in this environment, by giving priority to PTO optimisation. Thus, if the last PTO search step for the last buoy is greater than 0.01% then we optimise PTO parameters for 25 iterations using NM. Otherwise we check to see if the last *position* optimisation converged to within 0.01% and if so, we optimise position instead. Otherwise we choose between optimising PTO or position parameters for this buoy at random.

Note that this design assigns optimisation resources to PTO parameters as a first priority because we have observed stronger gains in output from tuning PTO parameters. Position parameters are given priority only when the PTO parameters for the last buoy were observed to be close to a local optimum. Algorithm 3.4 describes this method in detail. In addition, experiments were run with different starting buoy positions of were run with bottom center (C), bottom right (BR) and a uniform random position (r).

Algorithm 3.4 Symmetric Local Search + Nelder-Mead ($SLS + NM(2D)$)

procedure SYMMETRIC LOCAL SEARCH + NELDER-MEAD
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size
$angle = \{0, 45, 90, \ldots, 315\}$ ▷ symmetric samples angle
$iters = Size([angle])$ ▷ Number of symmetric samples
$S = \{\langle x_1, y_1, k_1, d_1\rangle, \ldots, \langle x_N, y_N, k_N, d_N\rangle\}$ ▷ Positions&PTOs
$\langle S1, S2\rangle = Decompose(S)$ ▷ Decomposing
$S1 = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = \perp$ ▷ Positions
$S2 = \{\langle k_1, d_1\rangle, \ldots, \langle k_N, d_N\rangle\} = \perp$ ▷ PTO parameters
$S1_{(1)} = \{\langle size/2, 0\rangle\}$ ▷ first buoy position
$S2_{(1)} = \{\langle rand \times Max_k, rand \times Max_d\rangle\}$ ▷ first buoy k and d
$energy = Eval([S1_{(1)}, S2_{(1)}])$
$(S2_{(1)}, bestEnergy) = NM(S1_{(1)}, S2_{(1)})$ ▷ Optimise first buoy PTOs
$(ImPTOrate) = ComputeImrate(bestEnergy, energy)$
$bestPosition = S1_{(1)}$; $bestPTO = S2_{(1)}$
$ImPorate = 1$ ▷ optimisation improvement rate Position
  for i in $[2, .., N]$ **do** $bestEnergy = 0$;
    for j in $[1, .., iters]$ **do**
      $(Sample_j, energy_j) = SymmetricSample(angle_j, S1_{(i-1)})$
      if $Sample_j$ *is feasible &* $energy_j > bestEnergy$ **then**
        $tPos = Sample_j$ $bestEnergy = energy_j$ $bestAngle = j$ ▷ Temporary position
      end if
    end for
    if *No feasible solution is found* **then** $(Sample_1, energy_1) = rand(S1_{(i-1)})$
    end if
    $(Es_1, Es_2) = SymmetricSample(bestAngle \pm 15, S1_{(i-1)})$
    $(S1_{(i)}, energy) = FindbestS(tPos, Es_1, Es_2)$
    if $ImPTOrate \geq 0.01\%$ **then**
      PTO optimisation
      $(S2_{(i)}, energy) = NM(bestPosition, S2_{(i-1)}, MaxEN)$
      $(ImPTOrate) = ComputeImrate(bestEnergy, energy)$
      if $energy > bestEnergy$ **then**
        $bestPTO = [S2_{(1)}, \ldots, S2_{(i-1)}, S2_{(i)}]$
        $bestEnergy = energy$
      end if
    else if $ImPorate \geq 0.01\%$ **then**
      Position optimisation
      $(S1_{(i)}, energy) = NM(S1_{(i)}, bestPTO, MaxEN)$
      $(ImPorate) = ComputeImrate(bestEnergy, energy)$
      if $energy > bestEnergy$ **then**
        $bestPosition = [S1_{(1)}, \ldots, S1_{(i-1)}, S1_{(i)}]$
        $bestEnergy = energy$
      end if
    else
      Optimise one of buoy Position or PTO randomly
    end if
  end for
  return $bestPosition, bestPTO, bestEnergy$ ▷ *Final Layout*
end procedure

Algorithm 3.5 Backtracking optimisation Algorithm (BOA)

procedure BOA $(Position, PTOs, Energy\ )$
Initialization
$S1 = \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\} = Position$ ▷ Positions
$S2 = \{\langle k_1, d_1\rangle, \ldots, \langle k_N, d_N\rangle\} = PTOs$ ▷ PTO parameters
$energy = ([E_1, E_2, \ldots, E_N]) = Energy$ ▷ Buoys energy
$N_w = N/4$
$(WIndex)$=$FindWorst(energy, N_w)$ ▷ Find worst buoys power
    for i in $[1, .., N_w]$ **do**
        PTO optimisation
        $(S2_{WIndex(i)}, energy_{WIndex(i)})$=$NM(S2_{WIndex(i)}, MaxEN)$
        Position optimisation
        $(S1_{WIndex(i)}, energy_{WIndex(i)})$=$NM(S1_{WIndex(i)}, MaxEN)$
    end for
    return $S1, S2, energy$ ▷ Final Layout
end procedure

Symmetric Local Search + Nelder-Mead + Backtracking (SLS-NM-B) The general idea of SLS-NM-B is like SLS-NM but with two differences. The first difference is optimising the initial buoy PTO settings by Nelder-Mead and then to share this configuration with the next placed buoys for speeding up the search process and saving computational time. Therefore, after applying symmetric local sampling and finding the best position, Nelder-Mead search tries to improve just the position (2D) of the new buoy.

The second contribution is applying a backtracking optimisation idea (described in Algorithm 3.5). As the search process of SLS is based on the greedy selection, we never come back to enhance previous buoys' attributes, so introducing backtracking can be effective for maximising total power output. Among all placed buoys in the array, the worst $round(N \times 0.25)$ buoys in terms of power are chosen and Nelder-Mead search is then used to optimise the position (2D) and PTO settings (2D) of these buoys in a bi-level optimisation process. This procedure is called SLS-NM-B1. We can observe the performance of SLS-NM-B1 in Figure 3.4(b,c). This shows how the eight symmetric samples are done and the effect of the later backtracking process which refines buoy placements. A second version of this algorithm is proposed (SLS-NM-B2) to evaluate the effectiveness of optimising both position and PTOs of each buoy (4D) simultaneously instead of in a bi-level search. Other details of the backtracking algorithm are the same.

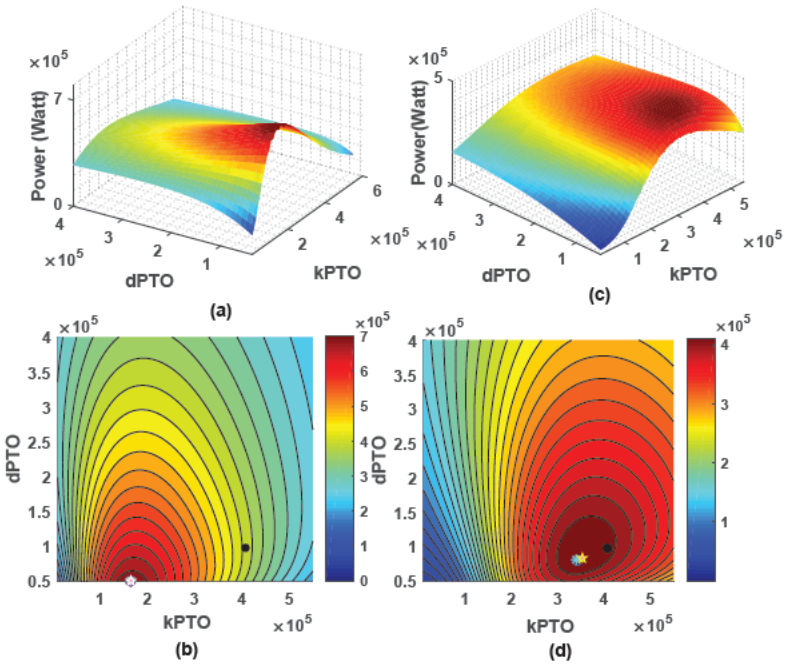

FIGURE 3.2: Power landscape analysis of both real wave scenarios ((a,b) Perth, (c,d) Sydney) for the best discovered 4-buoy layouts. The spring-damping PTO configuration step size is 10000. The black circle shows the manufacturer's PTO defaults for the predominant wave frequency and the star, cross, circle, and Pentagon markers present the k and dPTO settings of the best-discovered 4 buoys layout. Note that the search space for buoy positions is multi-modal [1], and that we only visualise a 2D slice of the 8D PTO optimisation space here without considering interactions with buoys' positions.

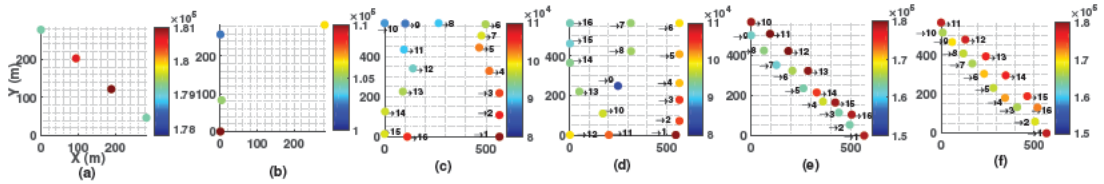

FIGURE 3.3: The best-obtained 4 and 16-buoy layouts: (a) 4-buoy, Perth wave model, Power=719978.29(Watt), q-factor=1.013 by DE; (b) 4-buoy, Sydney wave model, Power=423898.52(Watt), q-factor=0.98 by DE; (c) 16-buoy, Sydney, Power= 1559605, q-factor=0.903 by SLS-NM-B1; (d) 16-buoy, Sydney, Power=1564334.59, q-factor=0.916 by SLS-NM-B2; (e) 16-buoy, Perth, Power=2739657.74, q-factor=0.966 by SLS-NM-B1; (f) 16-buoy, Perth, Power=2741489.18, q-factor=0.972 by SLS-NM-B2 (2.26% more power than CMA-ES best layout).

3.1.7 Experiments

This section first presents a brief landscape analysis for PTO parameters for two wave scenarios (Perth and Sydney). We then present detailed results comparing the different search heuristics outlined in the previous section.

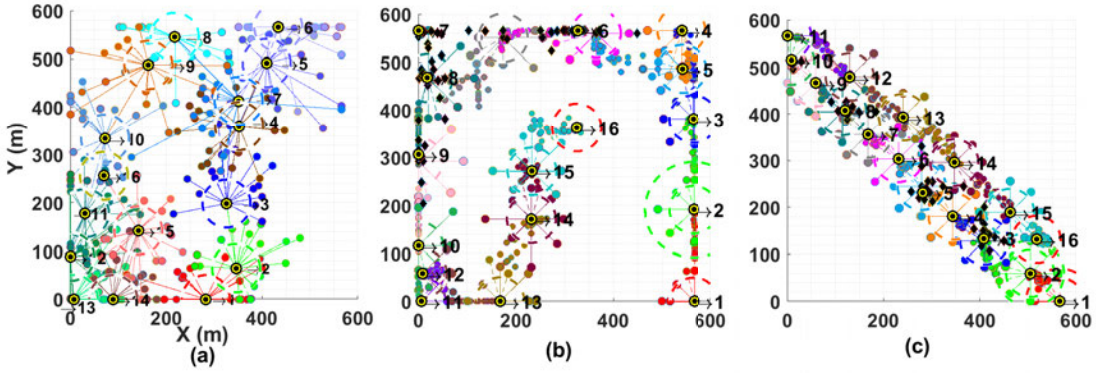

FIGURE 3.4: Three illustrations of the local search process for the placement of 16 buoys using LS-NM (part (a)) and SLS-NM-B2 (parts (b) and (c)). Small yellow circles represent the final buoy positions. The coloured radial lines represent the neighbourhood sampling process. The black diamonds in parts (b) and (c) represent the positions sampled by the backtracking algorithm. The internal circles show the safety distance and the external ones demonstrate the local search space. Part (a) (Power=1525780W, q-factor=0.89) and (b) (Power= 1562138W, q-factor=0.91), optimise for the Sydney wave model; and part (c)(Power=2741489W, q-factor=0.972) is for Perth.

3.1.7.1 Landscape analysis

For visualising the impact of PTO parameter optimisation, a simple experiment was done. First of all, we optimised the buoy positions for a 4-buoy layout using a manufacturer's PTOs defaults ($k = 407510$ and $d = 97412$) for all converters for both the Perth and Sydney test sites. The black circle in Figure 3.2 marks this default PTO configuration. The energy produced by this layout is 402 kW and 703 kW, respectively, for the Sydney and Perth wave climates. Next, this obtained layout is evaluated where the buoy positions are fixed and we grid-sample the energy produced when all four buoys are assigned the same PTO parameters. This process produces the contoured backgrounds shown in Figure 3.2. Finally, we optimise the PTO parameters for each buoy independently and plot a marker for each of the four buoys. These markers are roughly, but not completely, coincident with the peak in the background power landscape produced by optimising buoys' PTO parameters in unison. These markers are also at a different point to that produced by the default setting. The best energy produced after optimisation has improved to 420 kW and 720 kW respectively for Sydney and Perth. Another observation from Figure 3.2 is that the best PTO configurations of the 4-buoy layouts are relatively alike in both wave scenarios.

3.1.7.2 Layout evaluations

In order to evaluate the effectiveness the proposed algorithms in Sections 3.1.6.1, 3.1.6.2, and 3.1.6.3, we performed a systematic comparison of the best layouts produced by each

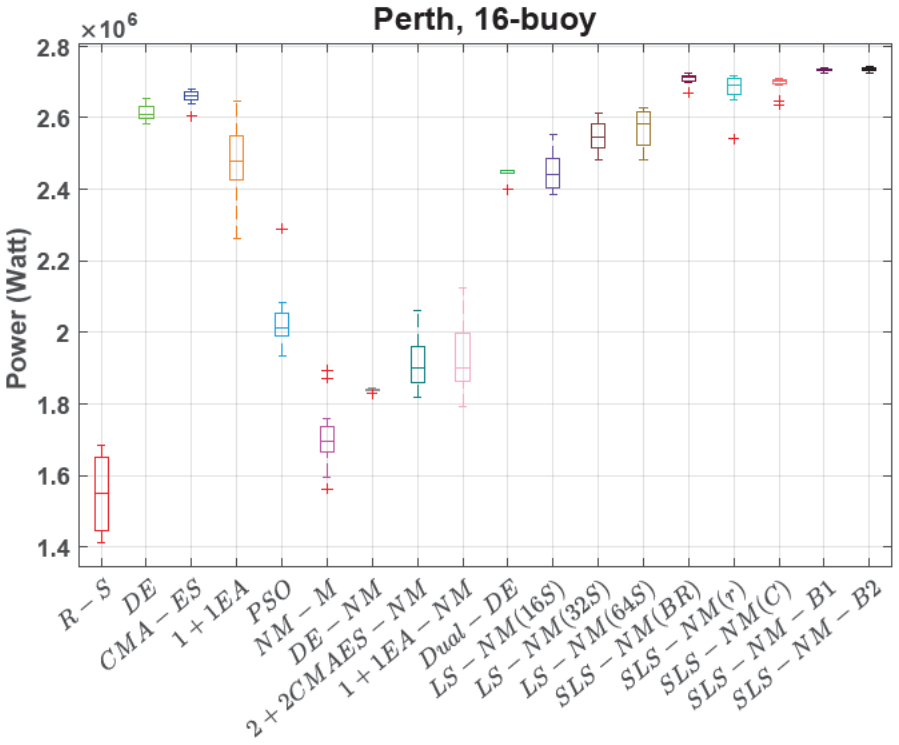

FIGURE 3.5: The comparison of the proposed algorithms performances for 16-buoy layout in Perth wave model. The optimisation results present the best solution per experiment. (10 independent runs per each method)

in two different real wave scenarios (Perth and Sydney), and for two different numbers of buoys ($N = 4$ and $N = 16$). Ten runs were performed for each optimisation method and the best solutions were collected for each.

Figure 3.5 shows the box-and-whiskers plot for the power output of the best solution per run for all search heuristics, for 16-buoy layouts for the Perth wave scenario. The corresponding summary statistics are presented in Table 3.1 and Table 3.2 for 16 and 4 converters respectively, and we illustrate the search process for three cases in Figure 3.4.

It can be seen that the best mean layout performance is produced by both SLS-NM-B1 and SLS-NM-B2. Additionally, the average optimisation results of SLS-NM with various first buoy locations are also competitive. Among these, the best results arise from placing the first buoy in the bottom right corner of the search space. This results in more total power output because the farm layout this placement enables a greater number of constructive buoy interactions. Of the standard EAs, CMA-ES performs best. Interestingly, the performance of the alternating approaches is not competitive compared with other methods.

Looking more closely at Table 3.1, in both wave scenarios, the SLS-NM-B2 method significantly outperforms all but the SLS-NM-B1 method using the Wilcoxon rank-sum test ($p < 0.01$). The SLS-NM performs better than CMA-ES for the Perth wave model,

Perth wave scenario (16-buoy)																
Methods	**DE**	**CMA-ES**	**1+1EA**	**PSO**	**NM-M**	**DE-NM**	**CMAES-NM**	**1+1EA-NM**	**Dual-DE**	**LS-NM_{16s}**	**LS-NM_{32s}**	**LS-NM_{64s}**	**SLS-NM(BR)**	**SLS-NM(r)**	**SLS-NM(C)**	**SLS-NM-B1**	**SLS-NM-B2**
Max	2652393	2680843	2644987	2289764	1893411	1845065	2059607	2125726	2453857	2554865	2613619	2626506	2723676	2716463	2709385	2739658	**2741489**
Min	2582793	2603920	2263180	1935340	1561609	1829109	1816940	1790521	2399372	2384981	2481663	2482512	2669097	2540090	2635628	2723886	2723470
Mean	2613938	2657924	2476649	2034625	1709664	1839680	1917947	1930481	2442276	2449269	2547633	2570651	2708267	2677821	2691542	2733105	**2735345**
Median	2609441	2661285	2476649	2011311	1696728	1840299	1902074	1902254	2453857	2442901	2545870	2584010	2711875	2692056	2701771	2733962	**2736453**
Std	21601 36	20844 29	109986 19	90666 26	96667 21	4261 50	76927 84	96648 77	20511 38	53689 15	40651 08	49948 44	14434 14	48718 95	24252 10	4426 12	4986 80
Sydney wave scenario (16-buoy)																	
Methods	**DE**	**CMA-ES**	**1+1EA**	**PSO**	**NM-M**	**DE-NM**	**CMAES-NM**	**1+1EA-NM**	**Dual-DE**	**LS-NM_{16s}**	**LS-NM_{32s}**	**LS-NM_{64s}**	**SLS-NM(BR)**	**SLS-NM(r)**	**SLS-NM(C)**	**SLS-NM-B1**	**SLS-NM-B2**
Max	1544911	1551852	1550820	1498996	1393383	1372431	1524002	1541064	1488451	1525789	1542636	1551640	1556956	1550054	1534157	1559578	**1564334**
Min	1525043	1533453	1461996	1396223	1256857	1363834	1392057	1414872	1420995	1507479	1523444	1518276	1526266	1489493	1465638	1546369	1529929
Mean	1536324	1547951	1526867	1438377	1337175	1367502	1454505	1467659	1462382	1514404	1532215	1535923	1544706	1525152	1512476	1553629	**1556447**
Median	1538708	1549616	1531683	1435726	1338054	1367767	1441785	1467420	1465419	1513593	1528728	1535516	1548100	1523762	1518423	1553779	**1558319**
Std	6559 22	4996 61	25962 37	31262	41794 00	2508 76	47091 11	32623 75	14999 60	5125 37	7224 27	12944 20	10965 95	17681 23	18379 27	3293 68	8931 08

TABLE 3.1: The performance comparison of various heuristics for the 16-buoy case, based on maximum, median and mean power output layout of the best solution per experiment.

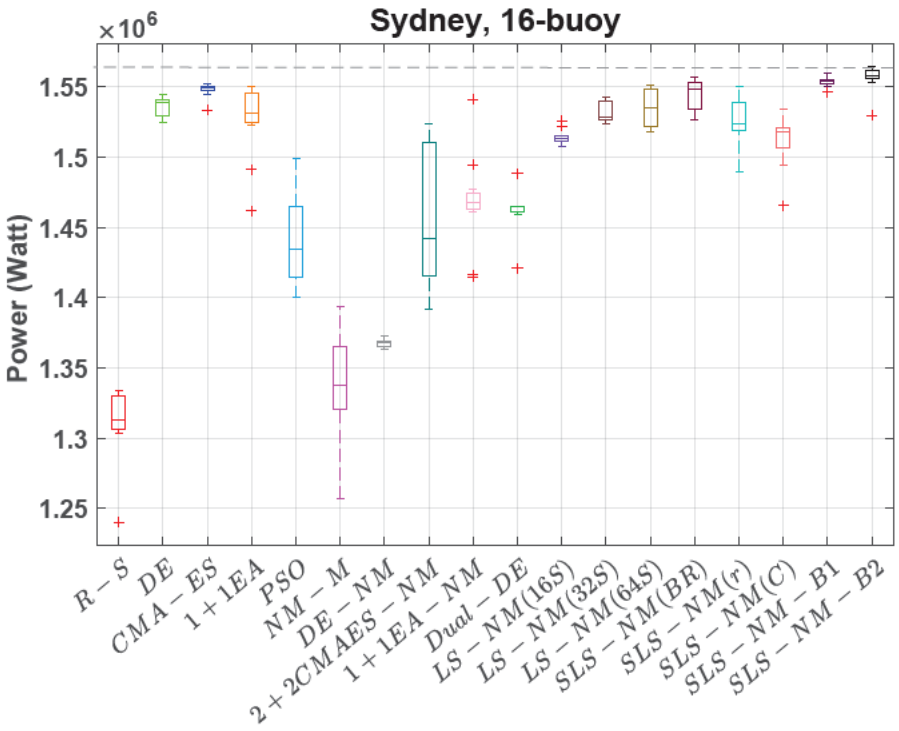

FIGURE 3.6: The comparison of the proposed algorithms' performance for 16-buoy layouts in Sydney wave model. The optimisation results present the best solution per experiment. (10 independent runs per each method)

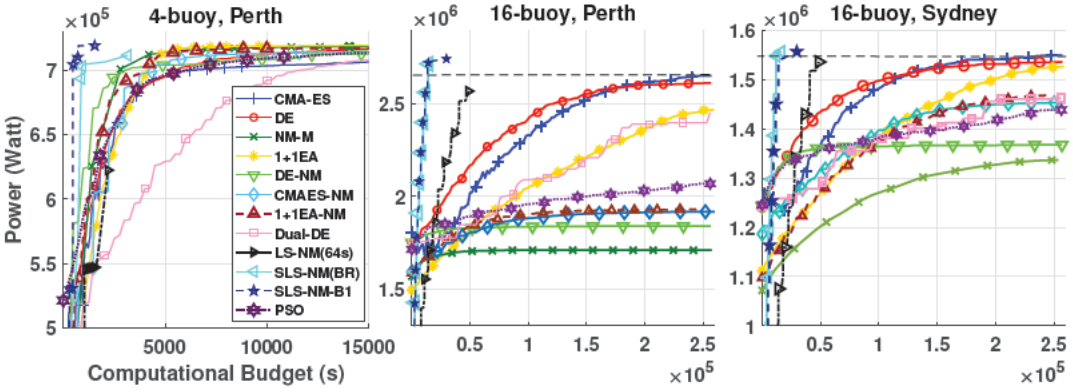

FIGURE 3.7: The convergence rate comparison for all proposed algorithms in both real wave scenarios(mean best layouts per generation). Both SLS-NM(BR) and SLS-NM(B1) methods are able to place and optimise the position and PTO configurations of 4 and 16-buoy layouts faster than other proposed approaches. The horizontal dashed lines show the improvement rate difference of both SLS-NM(BR) and SLS-NM(B1) with CMA-ES.

but is no better than CMA-ES or DE for the, more challenging, Sydney scenario. This can be seen in the box-plots for the Sydney scenario shown in Figure 3.6. As a last observation, there appears to be some positive impact from increasing the number of samples in the LS-NM heuristic from 32 samples to 64.

Figure 3.7 shows the convergence of average fitness of the best layout over time for all of the heuristics. Part (a) shows this convergence for N =4 for the Perth model, part (b)

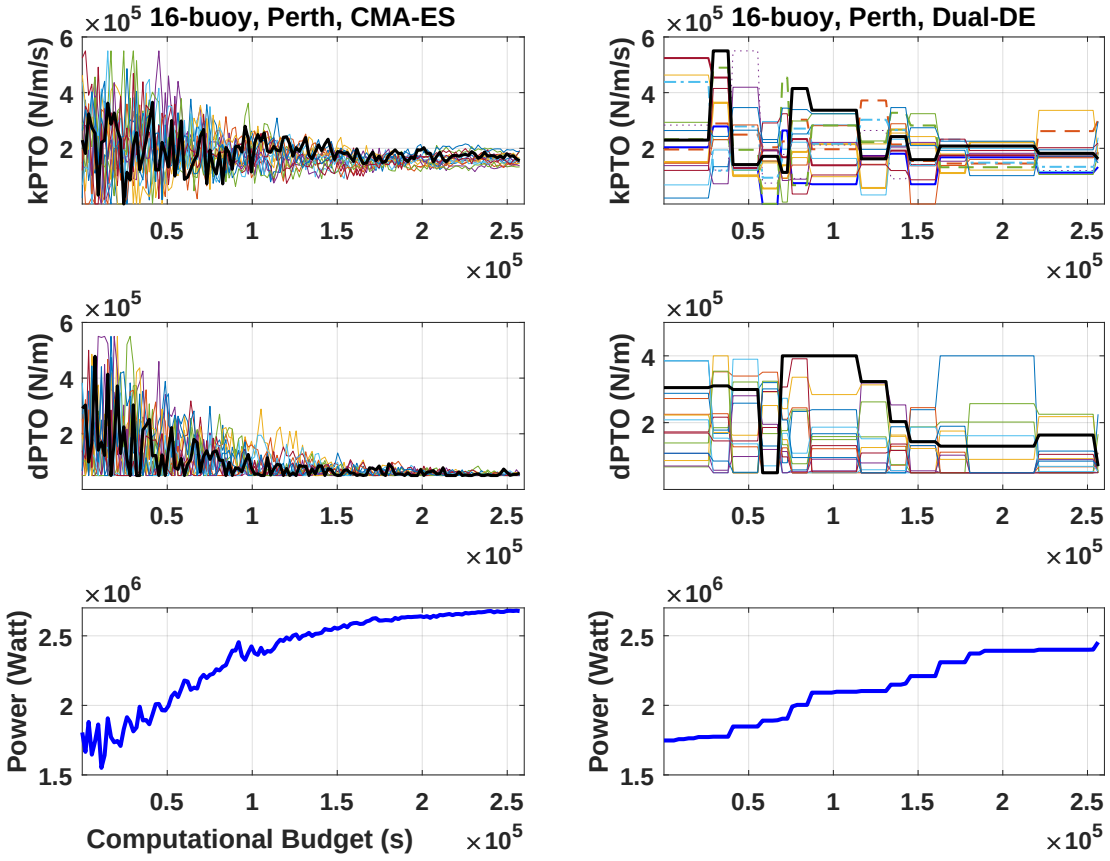

FIGURE 3.8: The convergence of spring-damping PTOs of 16 buoys by CMA-ES (All-in-one) and Dual-DE (alternating style) methods in Perth wave scenario. The black line shows the 16^{th}buoy PTO settings.

is for N = 16 for Perth, and part (c) is for N = 16 for Sydney.

In all configurations, SLS-NM-B converges very fast and still outperforms the other methods. To sum up, the experimental results in Table 3.1 and Figure 3.7 reveal that SLS-NM-B succeeds in attaining higher absorbed power as well as faster convergence speed. A second important remark about Figure 3.7 is that the alternating optimisation methods perform worse than the standard EAs, where both positions and PTO settings are mixed as an all-in-one problem. One possible path to improving these alternating methods in the future could be to shift some of the budget for PTO optimisation to positional optimisation, which appears to be more challenging.

Figure 3.8 tracks the convergence of just the PTO parameters for each buoy during a run for CMA-ES (graphs on the left) and Dual-DE optimisation (graphs on the right). It can be seen that both methods are able to optimise power output over time and the phased nature of the search in Dual-DE is visible in the graphs of the parameter values. It can also be observed that the parameter values for each buoy change non-monotonically as the best PTO settings interact with buoy positions over the course

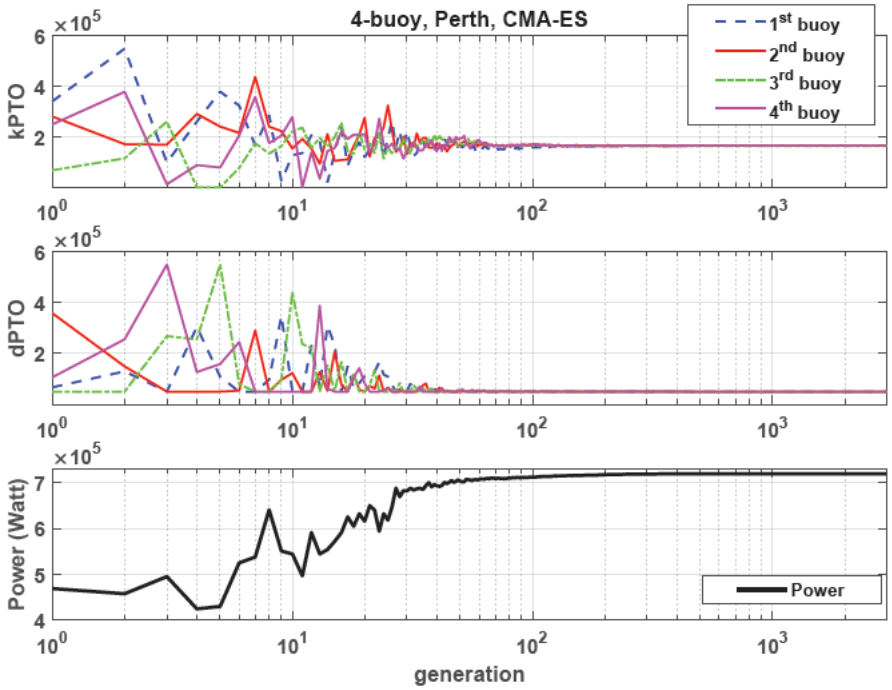

FIGURE 3.9: The convergence of spring-damping PTOs of 4 buoys by CMA-ES (All-in-one) in Perth wave scenario.

of optimisation. Meanwhile, Figure 3.9 shows the PTO optimisation process of 4-buoy layouts by CMA-ES method.

Figure 3.3 presents the most productive 4 and 16-buoy layouts attained from all the runs in the two scenarios. The best 16-buoy layouts are built by SLS-NM-B2 from the x-axis upwards with buoys labelled, in the figure, by order of placement. In all layouts, the first buoy is placed at the bottom right.

The best 4-buoy layout of the Perth wave model slopes diagonally upwards from right to left. This layout was found by DE. For 16-buoys, the best SLS-NM-B2 configuration produces a maximum power output that is 2.26% higher than the best CMA-ES configuration. Another observation is that the layouts for Sydney place buoys far from each other. This is likely to be due to the fact that the more diverse wave directions in Sydney make it harder to consistently exploit constructive interactions from having buoys in closer proximity.

3.1.8 Hydrodynamic interpretation

Figure 3.10 demonstrates how the ocean wave power propagates through the farm for each best-discovered solutions (4 and 16 buoy layouts) for the Sydney and Perth sites. These landscapes model interactions at the single dominant wave direction and frequency.

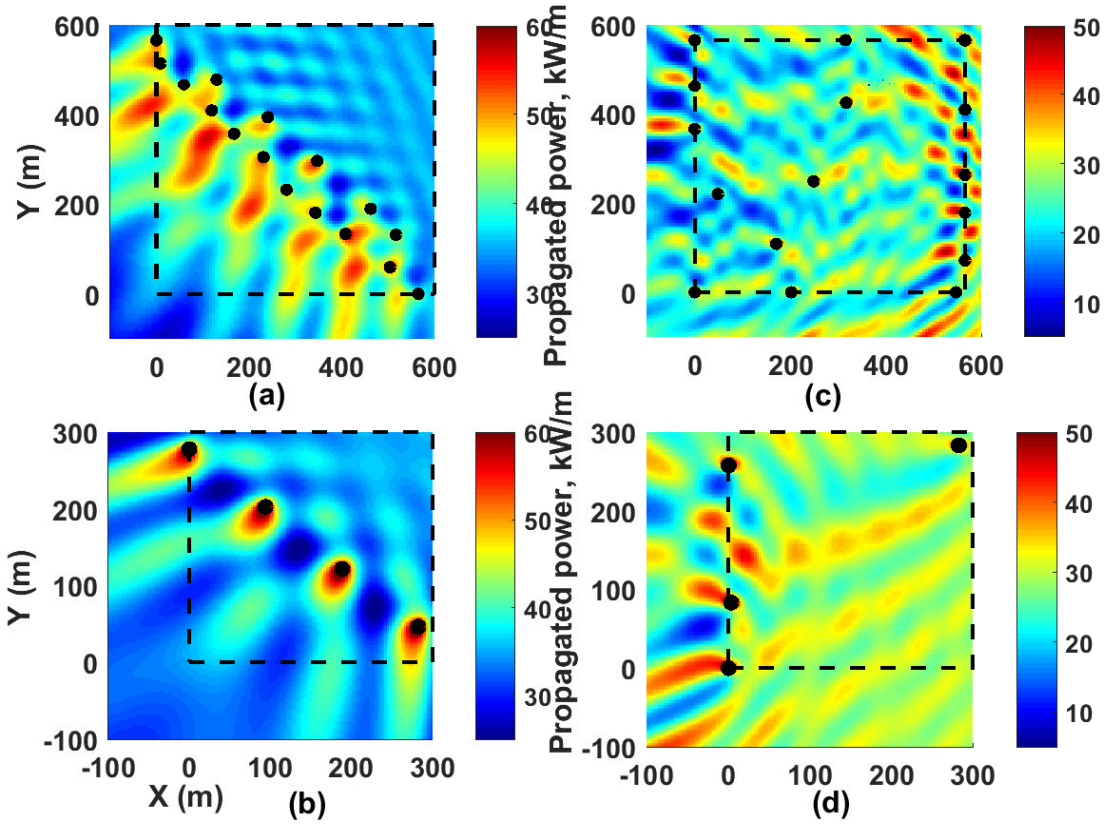

FIGURE 3.10: The wave power around the best-founded 4 and 16-buoy layouts by SLS-NM-B2; (a) 16 buoys, Perth wave scenario; (b) 4 buoys, Perth; (c) 16 buoys, Sydney, and (d) 4 buoys, Sydney wave scenario. Black circles and squares show the buoys placement and the search space.

The wave resource at the Sydney and Perth sites is 30 and 35 kW/m, respectively. While these waves propagate through the farm, the wave field is modified by the buoys and we can see that the wave energy across the farm varies between 10 and 60 kW/m. It can be seen that, in both sites, the best layout succeeds in extracting much of the energy from the surrounding environment and, in the case of Perth, the impact of extraction extends far out to sea beyond the farm. The red areas near buoys are produced by interactions of buoys with their local environment. It should be noted that, though these areas might appear to be good candidate positions for further buoy placements, destructive interference with other buoys would produce sub-optimal results from such a placement. Another observation is that at both sites at least one row of buoys is perpendicular to the dominant wave direction (232.5 deg for the Perth site, and 172.5 deg for the Sydney site). This indicates that this wave direction can inform the initialisation of buoy positions in optimising wave farm settings.

3.1.9 Conclusions

In this paper, we have described, evaluated, and systematically compared twelve different heuristic methods for optimising layout and PTO parameters for wave energy converter arrays. This study included four alternating hybrid algorithms and three new methods that are specialised to this domain. The results in this study indicate that the search problem is challenging, with buoys inducing changes in the local power landscape and hydro-dynamic interactions occurring between buoys. The PTO optimisation results, also, indicate at least some interaction between buoy placement and optimal PTO settings for each buoy. Moreover, the hydrodynamic modelling required for larger buoy layouts is expensive, which constrains optimisation to take place with a limited number of evaluations.

The best performing method is a new hybrid of a symmetric local search combined with Nelder-Mead search and a backtracking strategy. In our experiments, this method out-performed other state-of-the-art algorithms, for 16-buoy layouts, in terms of power production and in terms of speed-of-convergence. Future work can further improve the fidelity of the environment including considering a mix of buoy designs, tethering configurations, farm-boundaries and sea-floor shapes. These additional factors also create a more complex cost landscape, which opens the way for multi-objective optimisation.

Our code, layouts, and auxiliary material are publicly available: https://cs.adelaide.edu.au/~optlog/research/energy.php

Acknowledgements

We would like to offer our special thanks to Dr. Andrew Sutton and Sagari Vatchavayi (University of Minnesota Duluth) for their valuable and constructive suggestions. Our work was supported by the Australian Research Council project DE160100850.

Perth wave scenario (4-buoy)																
Methods	**DE**	**CMA-ES**	**1+1EA**	**PSO**	**NM-M**	**DE-NM**	**CMAES-NM**	**1+1EA-NM**	**Dual-DE**	**LS-NM_{16s}**	**LS-NM_{32s}**	**LS-NM_{64s}**	**SLS-NM(BR)**	**SLS-NM(r)**	**SLS-NM(C)**	**SLS-NM-B1**
Max	719978	719879	719851	719913	719845	718321	718418	719049	719915	629667	633448	635676	713573	714041	703908	719663
Min	719878	708731	708731	708445	708690	713598	706583	717363	719851	546821	600825	615328	710449	694667	701964	719143
Mean	719921	718005	718491	715730	718914	717041	715364	718500	719882	599239	617694	622393	711976	704714	702821	719495
Median	719914	719851	719850	719107	719844	717380	716988	718653	719879	599921	617716	621512	711877	705196	702835	719554
Std	27 78	4331 96	3170 29	5078 80	3219 83	1509 80	3925 23	478 99	28 92	24069 76	9739 71	5585 69	835 78	6707 32	563 52	172 24
Sydney wave scenario (4-buoy)																
Methods	**DE**	**CMA-ES**	**1+1EA**	**PSO**	**NM-M**	**DE-NM**	**CMAES-NM**	**1+1EA-NM**	**Dual-DE**	**LS-NM_{16s}**	**LS-NM_{32s}**	**LS-NM_{64s}**	**SLS-NM(BR)**	**SLS-NM(r)**	**SLS-NM(C)**	**SLS-NM-B1**
Max	423898	423878	423847	423872	423806	423628	423485	423775	423899	419504	420549	420850	422619	422906	422878	422866
Min	423489	422046	422784	420883	423392	423255	422464	423397	423789	386137	415848	413949	420667	401907	420125	420724
Mean	423767	423516	423579	423218	423703	423406	423006	423602	423844	411155	418305	418210	421665	414943	421368	422335
Median	423808	423646	423636	423564	423710	423352	422988	423625	423840	415909	418262	418607	421798	416878	421421	422660
Std	140 96	492 19	285 52	859 18	119 82	125 47	294 73	132 68	37 9	11436 02	1523 88	2104 40	624 80	7100 91	771 31	638 80

TABLE 3.2: The performance comparison of various heuristics for the 4-buoy case, based on maximum, median and mean power output layout of the best solution per experiment (Std = standard deviation). In SLS-NM, the first buoy location in the search space is investigated and three options are evaluated including: Bottom right (BR), Bottom Center (C) and random (r).

3.2 A Hybrid Cooperative Co-evolution Algorithm Framework for Optimising Power Take Off and Placements of Wave Energy Converters

3.2.1 Synopsis

In this chapter which is based on our paper published in [5], we aim to optimise a wave farm power output by finding appropriate configurations of WEC positions and power take-off parameters. In the first step, we develop the previous WEC simulator in [3] to allow for the PTO configurations to be adjusted for each wave frequency. This allows for a more realistic consideration of the real sea states. As this extension leads to a large number of decision variables and a multimodal power landscape, it is also computationally costly. With this in mind, we propose a new hybrid cooperative co-evolution algorithm (HCCA) in order to optimise WEC positions in arrays, as well as their PTO parameters. This hybrid approach consists of a new adaptive grey wolf optimiser, a Social Learning Particle Swarm Optimisation [155] (SLPSO) and a Self-adaptive Differential Evolution with Neighbourhood search [156] (SaNSDE). Finally, in the last step, we propose a backtracking optimisation strategy (BOA) to further optimise both the position and PTO settings of the obtained array. We compare this method to a broad range of meta-heuristics in order to optimise the total power output of a wave farm. These comparison heuristics include five popular off-the-shelf Evolutionary Algorithms (EAs), four cooperative optimisation ideas, and, lastly three hybrid optimisation algorithms. The real wave scenarios, which are evaluated in this research, are from the Southern coast of Australia (Perth, Sydney, Adelaide and Tasmania). According to the experimental results, HCCA performs better than other heuristic search methods in terms of convergence speed (5 times faster than the most effective previous algorithm) and the quality of layouts (80% development of the sustained energy output in 16-buoy experiment).

Reference

[5] Neshat, M., Alexander, B., & Wagner, M. (2020). A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters. *Information Sciences*, 534:218–244, 2020. Published by Information Sciences–journal on [15 May 2020].

Statement of Authorship

Title of Paper	A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted work written in manuscript style
Publication Details	Neshat, M., Alexander, B., & Wagner, M. (2019). A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters. Published by Information Sciences--journal on [13 May 2020]

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%) 75			
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	05/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander		
Contribution to the Paper	Advised on some aspects of experiments. Gave feedback on experiments and drafts. Helped with some visualisations and helped revise paper.		
Signature		Date	12/05/20

Name of Co-Author	Markus Wagner		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	14/05/2020

Please cut and paste additional co-author panels here as required.

3.2.2 Abstract

Wave energy technologies have the potential to play a significant role in the supply of renewable energy on a world scale. One of the most promising designs for wave energy converters (WECs) are fully submerged buoys. In this work, we explore the optimisation of WEC arrays consisting of three-tether buoys. Such arrays can be optimised for total energy output by adjusting both the relative positions of buoys and also the power-take-off (PTO) parameters for each buoy. The search space for these parameters is complex and multi-modal. Moreover, the evaluation of each parameter setting is computationally expensive and thus limits the number of full model evaluations that can be made. To handle this problem, we propose a new hybrid cooperative co-evolution algorithm (HCCA). HCCA consists of a symmetric local search plus Nelder-Mead and a cooperative co-evolution algorithm (CC) with a backtracking strategy for optimising the positions and PTO settings of WECs, respectively. For assessing the effectiveness of the proposed approach five popular Evolutionary Algorithms (EAs), four alternating optimisation methods and two recent hybrid ideas (LS-NM and SLS-NM-B) are compared in four real wave situations (Adelaide, Tasmania, Sydney and Perth) with two wave farm sizes (4 and 16). The experimental study shows that the hybrid cooperative framework performs best in terms of both runtime and quality of obtained solutions.

3.2.3 Introduction

Renewable energy technologies make up an increasing proportion of new-build electricity generating worldwide [157]. Ocean wave energy is one very promising technology for contributing to the growth in energy demand from renewable sources due to the high-energy densities of ocean environments, and high capacity factors of wave energy converter (WEC) technology. Wave energy converters are mechanical systems that are designed to generate electricity by harnessing the power of ocean waves motions. The waves' energy is converted into electricity by the power take-off system of the WECs [18].

It is envisaged that ocean wave energy could supply more than 70% of the world's whole energy demand [158]; however, the current WECs technologies are not fully developed due to the technical engineering challenges of harnessing ocean wave power in harsh ocean environments.

In this research, we apply a WEC simulator for evaluating the absorbed power of a wave farm consisting of CETO 6 model WEC converters. CETO converters are spherical submerged three-tether buoys. These converters were first designed in 2007 by the Carnegie Clean Energy company [151]. The energy output of WECs of this design depends on a

number of factors including the relative positions of WECs in an array (position optimisation), the power-take-off (PTO) settings on each buoy's tethers (control optimisation), and the long-term sea conditions of the wave-farm site.

In position optimisation, the main aim is finding the best location of each converter to maximise the constructive hydro-dynamic interactions and minimise the destructive interaction between converters.

In this work, we optimise both the position and PTO parameters of simulated wave farms consisting of both 4 and 16 buoys in 4 real wave environments. Since computing the complex hydrodynamic interactions among converters is computationally costly, the evaluation of each WEC's arrangement can take several minutes. Moreover, the combination of the search spaces of both WEC positions and PTO settings creates a multimodal and large-scale optimisation problem. These challenges require the use of robust, low-cost global search heuristics customised to this problem domain.

To date, the best performing heuristics for this problem [138] have been hybrid optimisation methods that placed and refined buoys parameters one at a time.

The main contributions of this article are:

1. Extending the simulator to allow for the PTO configurations to be set for each wave frequency, which allows for a more realistic consideration of the real sea states.
2. Proposing a new hybrid cooperative co-evolution algorithm (HCCA) for optimising WEC positions in arrays and their PTO parameters, which builds on these previous approaches.
3. Extending the Grey Wolf Optimiser (GWO) by a new adaptive mechanism (AGWO) for balancing exploration and exploitation.
4. Embedding the AGWO within a Cooperative Co-evolution method including (social learning particle swarm optimisation [155] (SLPSO) + self-adaptive differential evolution with neighborhood search [156] (SaNSDE)) for optimising the PTO configuration of the wave farm.
5. Employing a backtracking optimisation strategy (BOA) to further optimise both position and PTO settings of the obtained array.

To evaluate this new algorithm, we compare HCCA to a comprehensive range of metaheuristics for optimising the total power output of a wave farm, including (1) five popular off-the-shelf Evolutionary Algorithms (EAs), (2) four cooperative optimisation ideas, and (3) three hybrid optimisation algorithms. We evaluate these using four real wave

scenarios from the Southern coast of Australia (Perth, Sydney, Adelaide and Tasmania). Each scenario embeds a detailed model of a wave environment including time-integrated distributions of wave-heights, periods and directions. The experimental results show that HCCA is able to significantly outperform other optimisation approaches with regard to both the convergence speed and the total absorbed power output.

The remainder of the paper is organised as follows. We provide an overview of the related works in Section 3.2.4, and introduce the mathematical model for the WECs being studied in Section 3.2.5. The optimisation setup and the proposed optimisation methods are described in Sections 3.2.6 and 3.2.7, respectively. In Section 3.2.10, the experimental results are presented. We conclude with a summary and outline potential future works.

3.2.4 Related Work

There have been a number of studies in optimising the power output a variety of WEC models. One initial study [31] optimised WEC positions for five buoys using both the Parabolic Intersection (PI) method and a GA. The study required a high number of function evaluations (37000). The wave environment modelled was highly simplified, with just one wave direction.

A recent study by Ruiz et al. [35] employed another simple wave scenario to compare a custom Genetic Algorithm (GA), Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [89] and glow-worm optimisation (GSO) [101] for optimising the position of the buoys in a discrete grid. The investigation found that while the convergence rate of CMA-ES is faster than of the other two methods, it did not outperform both the GA and GSO in terms of total power production.

The Multi-Objective Differential Evolution Algorithm (MODE) [107] was applied to optimise the fundamental dimensions of a particular WEC concept including location, type of WEC, control strategy, hydrodynamic design and evaluated characteristics of the power take-off (PTO). However, that investigation evaluated the performance of one and 2-Body Point absorber. In other recent WEC position optimisation research, Wu et al. [39] compared a 1+1EA and population-based evolutionary algorithm (CMA-ES) to optimise both 25 and 50 buoys using a simplified uni-directional irregular wave model. That paper revealed that the 1+1EA with a simple mutation operator is able to outperform CMA-ES. However, the performance achieved for both the 25 and 50-buoy layouts was low. In another WEC array optimisation study, Ferri [40] compared the performance of two global optimisation algorithms, CMA-ES and surrogate-model based optimisation method (MM). The experimental results in [40] represented the convergence

speed of the surrogate-model optimisation algorithm is faster than CMA-ES considerably. Nevertheless, the introduced best array by MM has an estimation error of 10%. Fang et al. [41] proposed an Improved Differential Evolution (IDE) with an adaptive mutation factor for optimising an array of 3, 5 and 8-buoys, where the regular wave direction is just left to right (one direction). The research represented that IDE performance is superior to the generic DE.

Neshat et al. [1, 159] characterised a more complicated wave scenario (seven wave directions and 50 wave frequencies) with the intra-buoy effects and employed this knowledge to make a customised, single-objective hybrid heuristic (local search + Nelder-Mead). However, the wave model still used an artificial wave scenario, and the proposed method did not tune PTO parameters. An adaptive neuro-surrogate optimisation (ANSO) method [4] was proposed for the 16-buoy layout location optimisation that composed of a surrogate Recurrent Neural Network (RNN) model, a Symmetric Local Search and Nelder-Mead (SLS-NM). However, other dimensions of WECs were not evaluated in [4], such as control and geometric parameters.

Another challenging aspect of maximising the total power output of the wave farm is controlling the WECs' oscillations with respect to the incoming waves' frequency. This is because maximum efficiency will be achieved at resonance. However, maintaining a resonant condition can be challenging in the real sea states with the multiple different frequencies [152].

One way of achieving resonance is by configuring the power take-off (PTO) system of the WECs, either in online or offline settings. For then online setting, Ding et al. [153] implemented the maximum power point tracking (MPPT) control method for optimising the damping rate (dPTO) of one converter (CETO 6). The MPPT is a type of online-optimisation based on the gradient-ascent algorithm. The outcomes reported a high efficiency of the MPPT damping controller compared with a fixed-damping system, but the performance of MPPT was not assessed for layouts with more than one buoy. In later work Abdelkhalik et al. [38] utilised the hidden genes genetic algorithm (HGGA) to tune PTO parameters. While the proposed optimiser boosted the total energy produced, HGGA's efficiency was not compared to other modern EAs. A collaborative control strategy which is a hybrid model of an artificial neural network (ANN) and latching control was used [160] to find optimal damping control parameters, and the results found that the hybrid method performance is better than the uncontrolled WEC PTOs and with constant latching over 2-fold and 30% respectively. Glass et al. [37] used a combination of a generic GA with an analytical multiple scattering method to optimise WECs parameters including buoy radius, draft, and converter damping. That work produced some

5 and 9-buoy layouts with constructive interactions in a simple (uni-directional) wave scenario.

Silva et al. [161] compared the performance of a GA and COBYLA [162] (Constrained optimisation by Linear Approximations) for maximising the produced annual electrical output of one WEC with a U-shaped design inside an oscillating water column (UGEN) by adjusting PTO settings and the buoy's geometrical characteristics. The obtained results showed that COBYLA method converged to design with fewer evaluations. However, the GA produced a better solution overall, because COBYLA had converged to a local optimum. In another recent study, hybridization of a customized local search with a Nelder-Mead algorithm and a refinement strategy (SLS-NM-B) was introduced [138] for optimising both arrangement and PTO parameters of WECs in a farm. While the optimisation results represented a considerable power improvement of SLS-NM-B compared with other popular EAs, the optimisation of the PTO settings would evolve only one global PTO configuration for all WECs in the farm. It is our expectation that WEC-specific PTO settings can significantly improve the overall farm output.

3.2.5 Mathematical modelling for wave energy converters

3.2.5.1 Wave Resource

Based on the 2016 real wave data set from Australian Wave Energy Atlas [112], we investigate four different sites on the Southern coast of Australia in this paper: Perth, Adelaide, Tasmania (Southwest coast) and Sydney. Figure 3.11 shows the directional wave rose and wave scatter diagram of these four wave scenarios.

For example, Sydney is characterised by its dominant sea state with a peak wave period of $T_p = 9(s)$ and a significant wave height of $H_s = 2(m)$; and the probability distribution of the wave frequency f_w can be inferred associated with the Pierson-Moskowitz wave spectrum. The area of possible incident wave angles (β) for these particular sea sites vary considerably, e.g., covering approximately 180 ° at Sydney. Hence, it is expected that the distribution of the wave angle is normal driving to $f(\beta|\mu, \sigma) = N(\beta|0, \pi/12))$

We can further observe that the wave regimes differ with regard to the directional distributions and cumulative energy. This diversity provides four different search spaces for evaluating the performance of the optimisation methods. Our model of the ocean uses irregular directional waves together with the Bretschneider spectrum.

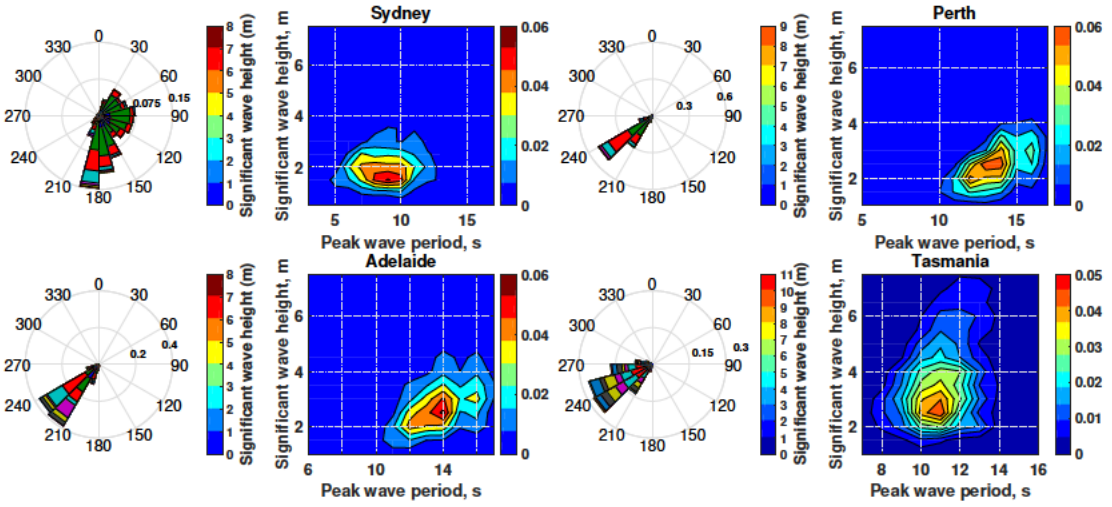

FIGURE 3.11: Scatter and wave rose diagrams for four wave energy sites in Australia: (a) Sydney, (b) Perth, (c) Adelaide and (d) Tasmania. These are: the directional wave rose (left) and wave scatter diagram (right).

3.2.5.2 Power Absorption Modelling

The WEC type in this research is a symmetric, spherical and fully submerged buoy (minimum depth of 8m at the top of the buoy) with three tethers. Each tether is attached to a mooring system installed on the seabed. The principal goal of the mooring system is to keep the floating structure in position within specific tolerances in both under normal load conditions and severe storm load conditions. The assumed optimal angle of each tether is 55 degrees [90]. For further technical details on the WECs, including submergence and ocean depth, we refer the interested reader to [1].

In this research we apply a frequency-domain simulation [163]. This type of simulation provides a faster approximation than time-domain models which makes them suitable for initial studies and especially useful in design scenarios where many model evaluations need to be run [164]. In the frequency domain, a linear PTO model is used. Viscous forces and the end stop forces are not modelled and kinematics are linearised. All bodies have 6 degrees of freedom (3 translational and 3 rotational). The buoy we use is a sphere, so its rotation (pitch, roll, or yaw angles) does not affect power production, which requires modelling of only translational degrees of freedom – surge, sway, and heave. Furthermore, linear potential theory is used here because it is able to capture main hydrodynamic effects, and it is much faster than any nonlinear solvers. In a design setting where arrays are modelled with multiple buoys, it is not feasible to model nonlinear hydrodynamic effects. Most published design work models only one translational degree of freedom, e.g. heave, but for these cases buoy is only attached to one tether and can harness power only from vertical motion. In our design with three tethers, the buoy can extract power from all motion modes [165].

The applied hypotheses are the following:

$$\begin{aligned} \hat{F}_{exc,\Sigma} &= \left(\left(M_\Sigma + A_\Sigma(\omega) \right) j\omega + B_\Sigma(\omega) - \frac{K_{pto,\Sigma}}{\omega} j + D_{pto,\Sigma} \right) \ddot{X}_\Sigma \\ M &= mI_{3N} \\ K_{pto} &= K_{pto} I_{3N} \\ D_{pto} &= D_{pto} I_{3N} \end{aligned} \tag{3.5}$$

where F_{exc} is the frequency-dependent vector of excitation forces and M is known as a mass matrix (I_{3N} is the identity matrix with size $3N$). N is the number of WECs in the layout, and the constant 3 denotes the number of degrees of freedom. $\ddot{X}_\Sigma$ is a vector of body acceleration in the surge, heave and sway directions. The matrices A and B define the hydrodynamic added-mass and radiation damping coefficients, respectively. The PTO mechanism is modelled on an oscillating spring. The K_{pto} and D_{pto} are, respectively, the stiffness of spring and damping PTO matrices. Each row of these matrices represents the settings for each buoy. For each buoy, there are 50 individual K_{pto} and D_{pto} parameter settings representing a tuned response to 50 different ocean wave frequencies. For modelling the hydrodynamic interaction between submerged buoys, a semi-analytical solution is given in [92].

For calculating the power output of an entire WEC array, Equation 3.6 computes the total power harnessed in a regular wave frequency environment:

$$P_\Sigma = \frac{1}{4} \left(\hat{F}^*_{exc,\Sigma} \ddot{X}_\Sigma + \ddot{X}^*_\Sigma \hat{F}_{exc,\Sigma} \right) - \frac{1}{2} \ddot{X}^*_\Sigma B \ddot{X}^*_\Sigma \tag{3.6}$$

While we are able to calculate the total power of the wave farm in Equation 3.6, it is very computationally expensive, and the computational cost rises quadratically with the number of buoys. Note that, where there is constructive interaction between converters, the total power output can grow super–linearly with the number of buoys.

Table 3.3 describes all applied symbols in the modelling of the wave energy converters in this research.

3.2.6 Optimisation Setup

The formulation of the optimisation problem to maximise the power output of a WEC array is represented by Equation 3.7:

$$P^*_\Sigma = argmax_{X,Y,K_{pto},D_{pto}} P_\Sigma(X, Y, K_{pto}, D_{pto}) \tag{3.7}$$

#	Definition	Symbol
1	Peak wave period	T_p(s)
2	Significant wave height	H_s(m)
3	Wave frequency	f_w
4	Wave angles	$0° \leq \beta \leq 360°$
5	Excitation forces vector	F_{exc}
6	Hydrodynamic added-mass coefficient	$\mathbf{A}$
7	Radiation damping coefficient	$\mathbf{B}$
8	Spring stiffness PTO matrices	$5 \times 10^4 \leq K_{pto} \leq 5.5 \times 10^5$
9	Damping stiffness PTO matrices	$4 \times 10^4 \leq D_{pto} \leq 5.5 \times 10^5$
10	Lower and upper bound of Damping stiffness	d_l, d_u
11	Lower and upper bound of Spring stiffness	k_l, k_u
12	Mass matrix	$\mathbf{M}$
13	WECs number	N

TABLE 3.3: Symbol definitions of the WEC model

where $P_\Sigma(X, Y, K_{pto}, D_{pto})$ shows the annual average power produced for given WEC locations and PTO settings in a 2-D coordinate system at x-positions: $\vec{X} = [x_1, \ldots, x_N]$, y-positions: $\vec{Y} = [y_1, \ldots, y_N]$ and Power Take-off configurations including $\vec{K_{pto}} = \{[B^1_{k_1}, \ldots, B^1_{k_{50}}], \ldots, [B^N_{k_1}, \ldots, B^N_{k_{50}}]\}$ and $\vec{D_{pto}} = \{[B^1_{d_1}, \ldots, B^1_{d_{50}}], \ldots, [B^N_{d_1}, \ldots, B^N_{d_{50}}]\}$. where B is the ith buoy here and $N \in \{4, 16\}$. it is assumed that all WECs are placed the same depth (5 metres) in ocean with the uniform depth of 30 metres.

Constraints In this work, there are three types of constraints, including constraints of farm boundaries, safe distance constraints between generators and constraints on PTOs variables. In terms of farm boundaries, the positions of each buoy (x_i, y_i) in the wave farm is restricted to a square search space $S = [x_l, x_u] \times [y_l, y_u]$: where $x_l = y_l = 0$ *and* $x_u = y_u = \sqrt{N * 20000}\, m^2$. The minimum safety distance constraint, to allow for shipping egress, is set to 50 meters. The PTO constraints are on spring damping PTO coefficients of $d_l = 5 \times 10^4, d_u = 4 \times 10^5$ and $k_l = 1, k_u = 5.5 \times 10^5$. Where a candidate solution satisfies all the constraint functions, it is marked as a feasible layout. For handing both boundary constraints (position and PTOs), infeasible solutions are forced to the most adjacent feasible design. For the safety distance constraint, a steep penalty function is used (Equation 3.8):

$$Sum_{dist} = \begin{array}{l} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (dist((x_i, y_i), (x_j, y_j)) - 50), \\ \text{if } dist((x_i, y_i), (x_j, y_j)) < 50 \text{ else } 0 \end{array} \tag{3.8}$$

where sum_{dist} is the sum of violations of the safe distance between buoys. The Euclidean distance between both buoys ith and jth is denoted by $dist((x_i, y_i), (x_j, y_j))$. The penalty

value to the total power absorbed of the wave farm is calculated by $(Sum_{dist}+1)^{20}$. The penalty strongly encourages selecting feasible layouts during the optimisation process.

Computational Resources The optimisation approaches studies here are evaluated and compared for both 4 and 16 WEC arrays for four real wave scenarios. For comparing all proposed methods in a realistic design setting, a time budget criterion of three days is set for optimisation method trial on an HPC supercomputer with a 2.4GHz Intel 6148 processor running 12 processes in parallel with 128GB of RAM. On this platform, this mode of parallelisation usually accommodates more than ten times speedup. Note that the implementations of the proposed optimisation methods are written to so as to exploit the parallel processing capabilities of the platform maximally. The software platform used for running the function evaluations and the optimisation algorithms is MATLAB R2018.

3.2.7 Optimisation Methods

In this research, we employ three different broad optimisation strategies for maximising the total absorbed power output of 4 and 16-buoy layouts in this research. The first approach applies search algorithms to all decision variables simultaneously. These variables include all the x and y buoy position and all of the PTO parameters. For 16 buoys, this approach requires that over 1632 variables are optimised all at once. The second broad approach is to apply cooperative methods [154], which alternate between the optimisation of position and PTO parameters. The third strategy, used in [1, 138], places buoys sequentially (one-at-a-time). Under this strategy, the performance of three hybrid methods are evaluated and compared: $LS-NM$ [1], $SLS-NM-B$ [138], and a new hybrid cooperative EA (HCCA). The details of the algorithms evaluated for each strategy are summarised in Table 3.4.

3.2.7.1 Evolutionary Algorithms (All-at-once)

In these experiments, five popular EAs and a new adaptive variant of GWO are used to optimise all dimensions simultaneously. These EAs are: (1) covariance matrix adaptation evolutionary-strategy (CMA-ES) [89], (2) Differential Evolution (DE) [94], (3) Particle Swarm optimisation (PSO) [42], (4) Grey Wolf optimiser (GWO) [11] and (5) Nelder-Mead simplex direct search (NM) [96] is combined with a mutation operator (Nelder-Mead+Mutation). Furthermore, we introduce a new variant of GWO called the adaptive grey wolf optimiser (AGWO).

TABLE 3.4: A review of the proposed framework methods employed in this paper. All approaches are restricted to the same computational budget constraint. Parallelism can be categorised into two groups as per-individual or per-frequency according to the individual's number in the population.

Abbreviation	parallelism	Description
All-at-once methods (Positions+PTO parameters)		
$CMA-ES$	per-individual	CMA-ES [89] all dimensions, $\mu = 4 + int(3 * log(N_{var}))$, $\sigma = 0.3 * Area$
DE	per-individual	Differential evolution [94], $\mu = 50$, $F = 0.5$, $P_{cr} = 0.5$
PSO	per-individual	Particle Swarm optimisation [42]. with μ= 50, $c_1 = 1.5, c_2 = 2, \omega = 1$ (linearly decreased)
GWO	per-individual	Grey Wolf Optimiser [11]. with μ= 50, $\alpha = 2$ (linearly decreased to zero)
$AGWO$	per-individual	Adaptive Grey Wolf optimiser, where $\mu = 50$, $\alpha = 2$ will be adaptively updated, CN_m^{Max}=0.3, $CN_m^{Min} = 10^{-6}$, $C_f = 0.7$
NM	per-frequency	Nelder-Mead search [96] is run in all dimensions iteratively $MaxFunEvals = 100$
Cooperative Evolutionary Ideas		
$(2+2)CMA-ES+NM$	per-individual & frequency	CMA-ES ($\mu = \lambda = 2$) cooperates with Nelder-mead where the position optimisation is done by CMA-ES and Nelder-Mead adjusts the spring-damping coefficients of all buoys in the round robin fashion.
$(1+1)EA+NM$	per-frequency	Cooperative strategy of 1+1EA (all position dimensions, $P_{Mu} = \frac{1}{N}$) with linearly decreasing mutation step size (σ) per generation at 100 iterations and then Nelder-Mead tries to optimise the PTO parameters in all dimensions.
$AGWO+NM$	per-individual & frequency	Adaptive GWO is in charge of optimising the PTO settings of the buoys. Afterward, the position configuration of the best candidate is optimised by Nelder-Mead search. This cooperative process is repeated until the time budget runs out.
$CCOS$	per-individual	Cooperative Co-evolution of $SLPSO$ and $SaNSDE$ with Online Optimiser Selection [166]. Where $\mu = 50$, $\mathbb{A} = 2$, $\mathbb{C} = N \times 2$.
$SLPSO_{II}$	per-individual	Double Social Learning Particle Swarm optimisation, Setup for $SLPSO_{II}$ from [166]. $\mu = 50$, $\mathbb{A} = 2$, $\mathbb{C} = N \times 2$
$SaNSDE_{II}$	per-individual	Double Self-adaptive Neighborhood Search Differential Evolution [166]. $\mu = 50$, F, P_{cr} and ρ are initialised at 0.5, but updated adaptively. $\mathbb{A} = 2$, $\mathbb{C} = N \times 2$
Hybrid optimisation methods (one-at-a- time)		
$LS+NM$	per-frequency	Repeated Local Sampling + Nelder Mead search [1] buoys are placed at normally distributed random offset ($\sigma = 100m$) from previous buoy and then ($MaxSam = 512$) the best candidate sample is chosen. Next, the PTO parameters of chosen sample are enhanced by Nelder-Mead search.
$SLS+NM+B$	per-frequency	Symmetric Local Sampling + Nelder-Mead + Backtracking [138]. The new buoy is locally placed by a symmetric search approach. Next, both configurations (positions and PTOs) are adjusted by Nelder-Mead iteratively. Finally, the backtracking strategy modifies least-well performing buoy's locations and PTOs.
$HCCA$	per-individual & frequency	Hybrid Cooperative Evolution Algorithm. SLS sets an initial location for each new buoy; Nelder-Mead optimises the buoy's position; then adjusts the PTOs. The process iterates until all buoys are placed. Backtracking is then applied to improves the positions and PTO settings for some buoys (25%) which have the lowest absorbed power.

#	Definition	Symbol
1	Parent and offspring population size	μ, λ
2	Mutation step size	$\sigma(m)$
3	Cognitive and social weight	c_1, c_2
4	Inertia weight	ω
5	Exploration probability rate	α
6	Upper and lower bound of normalisation rate	CN^{Max}, CN^{Min}
7	Chaotic factor	C_f
8	Probability mutation rate	P_{Mu}
9	Candidate optimisers	$\mathbb{A}$
10	Number of decision variables	$\mathbb{C}$
11	Mutation factor	F
12	Probability crossover rate	P_{cr}
13	pre-determined period	ρ
14	Differential vector of prey and members	$\vec{D}$
15	Random coefficient vector	$\vec{C}$
16	Exploration probability coefficient vector	$\vec{A}$
17	Prey location vector	$\vec{X_p}$
18	Three best-sampled candidates vector	$\vec{X_\alpha}, \vec{X_\beta}.\vec{X_\delta}$
19	Normalisation factor	$N_{m_{iter}}$
20	Maximum iterations in each chaotic sequence	Max_{iter_N}
21	Normalized chaotic values	CC_{iter}
22	Population of solutions	$\mathbb{S}$
23	Buoy PTOs values	B_k, B_d
24	Accumulated contributions of optimisers	$U_{(\mathbb{A}_{ii}, \mathbb{S}_i)}$
25	Contribution improvement rate	$I_{(\mathbb{A}_{ii}, \mathbb{S}_i)}$

TABLE 3.5: Symbol definitions of the optimisation methods

3.2.7.2 Adaptive Grey Wolf optimiser (AGWO)

The adaptive grey wolf optimiser is a new variant of the grey wolf optimiser that tunes hyper-parameter settings to improve performance in this search domain.

3.2.7.2.1 Overview of grey wolf optimiser (GWO)

The GWO algorithm [11] is categorised as a bio-inspired stochastic method that mimics grey wolves hunting behaviours in a pack. In the population, there are four classes of responsibility: the alpha wolf is responsible for leading the pack members, beta and delta wolves assist the alpha in decision making, and the remainder of the pack (of omega wolves) helps sample search space. GWO simulates aspects of the hunting process including 1) searching for the prey (optimum), 2) encircling the prey, 3) hunting and 4) attacking the prey.

3.2.7.2.2 Encircling the prey

$$\vec{D} = |\vec{C}.\vec{X_p}(t) - \vec{X}(t)| \tag{3.9}$$

$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A}.\vec{D} \tag{3.10}$$

where $\vec{D}$ describes the interval among the prey location $\vec{X_p}$ and a member of the pack $\vec{X}$ in the current iteration (t). Additionally, There are two coefficient vectors ($\vec{A}$ and $\vec{C}$) which have a substantial effect for adapting the behaviours of the exploration and exploitation processes, which can be computed by Equations 3.11 and 3.13:

$$\vec{A} = 2.\vec{a}.\vec{r_1} - \vec{a} \rightarrow 0 \leq a \leq 2 \tag{3.11}$$

$$a = 2 - iter.(\frac{2}{Max_{iter}}) \tag{3.12}$$

$$\vec{C} = 2.\vec{r_2} \tag{3.13}$$

where a is linearly decreased from 2 to 0 during the optimisation process. r_1 and r_2 are two random numbers between 0 and 1.

3.2.7.2.3 Hunting

For having a successful exploration of the search space, the solutions are updated based on the knowledge of three best-sampled candidates (alpha, beta and delta). This is because we assume a prior that a nearby optimum can be found among these best. The position update formulas (Equations 3.14, 3.15 and 3.16) are as follows.

$$\vec{X}(t+1) = \frac{\vec{X_1} + \vec{X_2} + \vec{X_3}}{3} \tag{3.14}$$

$$\begin{aligned} \vec{X_1} &= \vec{X_\alpha}(t) - \vec{A_1}.\vec{D_\alpha} \\ \vec{X_2} &= \vec{X_\beta}(t) - \vec{A_2}.\vec{D_\beta} \\ \vec{X_3} &= \vec{X_\delta}(t) - \vec{A_3}.\vec{D_\delta} \end{aligned} \tag{3.15}$$

$$\begin{aligned} \vec{D_\alpha} &= |\vec{C_1}.\vec{X_\alpha} - \vec{X}| \\ \vec{D_\beta} &= |\vec{C_2}.\vec{X_\beta} - \vec{X}| \\ \vec{D_\delta} &= |\vec{C_3}.\vec{X_\delta} - \vec{X}| \end{aligned} \tag{3.16}$$

3.2.7.2.4 Attacking the prey (exploitation)

The hunting manner is followed by attacking the prey and converging to the optimum positions. This can be achieved mathematically by reducing the a variable from 2 to 0 gradually. It is observed that when $|\vec{A}| < 1$ search agents are forced to attack the prey that is like a local search (exploitation process). Inversely, where $|\vec{A}| > 1$ leads to a global search (divergence) or exploration process.

Adaptive Grey Wolf optimiser (AGWO) One of the most critical parameters of GWO is $\vec{A}$ because it can adjust both diversification ($|\vec{A}| > 1$) and intensification ($|\vec{A}| < 1$) of the search process. According to Equation 3.11, the vector of $\vec{A}$ values can be between $-a$ and a ($\vec{A} \in [-2a, 2a]$), where a is reduced linearly during the optimisation from 2 to 0. It means that the probability of exploration ($|\vec{A}| > 1$) at the initial iteration is 0.5 and will be linearly decreased until 0 in the middle of the search process. On the other hand, the exploitation probability in the first iteration is 0.5 that is similar to exploration probability (giving a balanced-heuristic setting at the start); however, the exploitation probability is gone up to 1 where half of the iterations are devoted ($iter = Max_{iter}/2$). Significantly, in the remaining iterations ($Max_{iter}/2$), exploitation probability is 1, but exploration probability is 0 without any change. This issue is one reason why GWO is faced with premature convergence in some cases. Figure 3.12 (a and c) show this unbalanced search behavior. To overcome this shortcoming, a number of different mechanisms have been suggested.

Mittal et al. [12] proposed an improved version for updating $\vec{a}$ in (mGWO), which decayed more slowly to improve exploration. Figure 3.13(a) represents this slower decay function. However, in this static mechanism after 70% of the iterations, the value of a has still decayed below 1. A similar modification was introduced by Long et al. [167] in their Improved Grey Wolf optimiser (IGWO). More recently, Saxena et al. [168] scaled the decay function using a β-chaotic sequence to allow for faster oscillation between exploration and exploitation phases during the parameter decay process.

In previous research, various approaches were recommended for adjusting the a parameter, but these ideas did not pay attention to the GWO performance during the optimisation. In this paper, we propose an adaptive mechanism for updating the control variable a of GWO (AGWO). In this way, the optimisation performance is observed, and after a pre-determined period of ρ iterations, where the best-found solution does not overcome the alpha particle, the control parameter should be incremented back to 2. Moreover, a chaotic distribution is implanted with mapping by a normalise function for obtaining

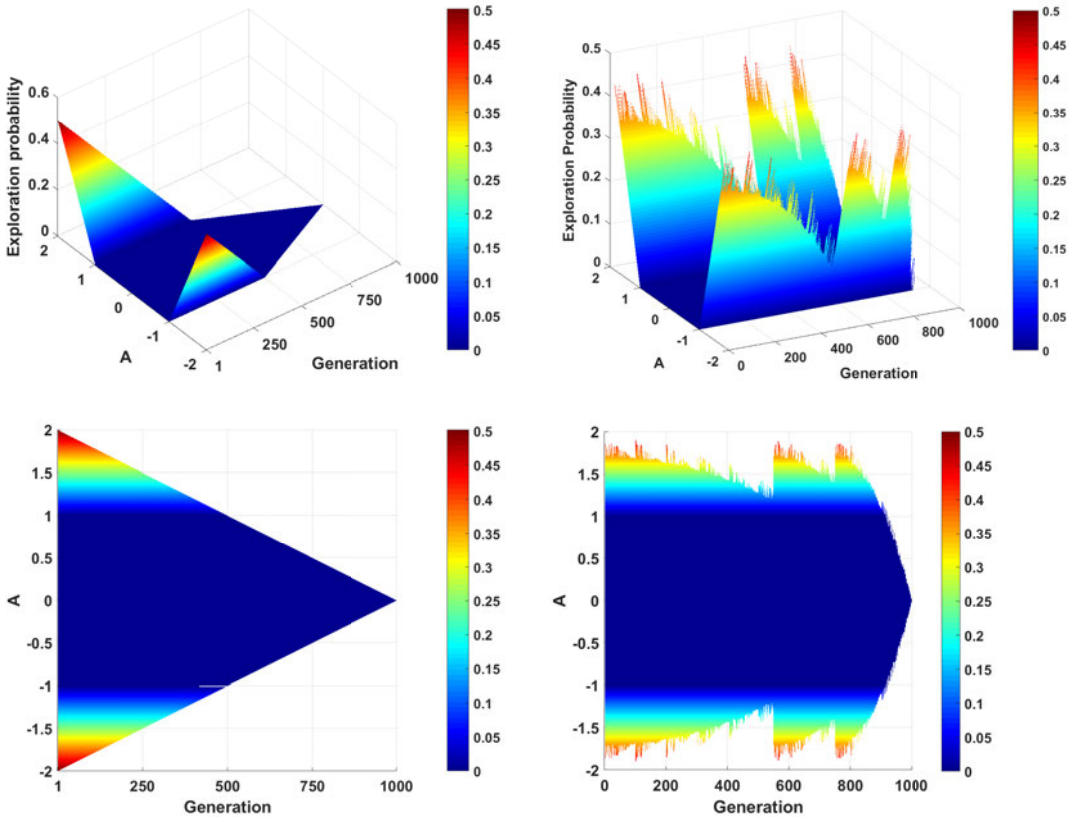

FIGURE 3.12: (a and c) The probability of original GWO exploration per generation (3D and 2D). (b and d) one example of the proposed adaptation mechanism for the control vector (a). These figures show the AGWO exploration probability per generation (3D and 2D).

a great balance between exploration and exploitation. The main AGWO contributions are:

1. Generating and combining a chaotic sequence with the control parameter (a) in each iteration. For achieving the best performance, ten various chaotic maps are applied and compared. Table 3.7 shows the applied these chaotic maps in the adaptive idea.

2. Using the normalisation function periodically for distributing the chaotic sequence between upper and lower bias. The mathematical formulation of the function can be represented by Equation 3.17.

$$N_{m_{iter}} = CN_m^{Max} - (\frac{CN_m^{Max} - CN_m^{Min}}{Max_{iter_N}}) \times iter_N \tag{3.17}$$

where the maximum and minimum values of the normalisation function are $CN_m^{Max} = 0.3$ and $CN_m^{Min} = 10^{-6}$, respectively. Both CN_m^{Max} and CN_m^{Min} control the chaotic behaviour of the applied chaotic map and let to the optimisation

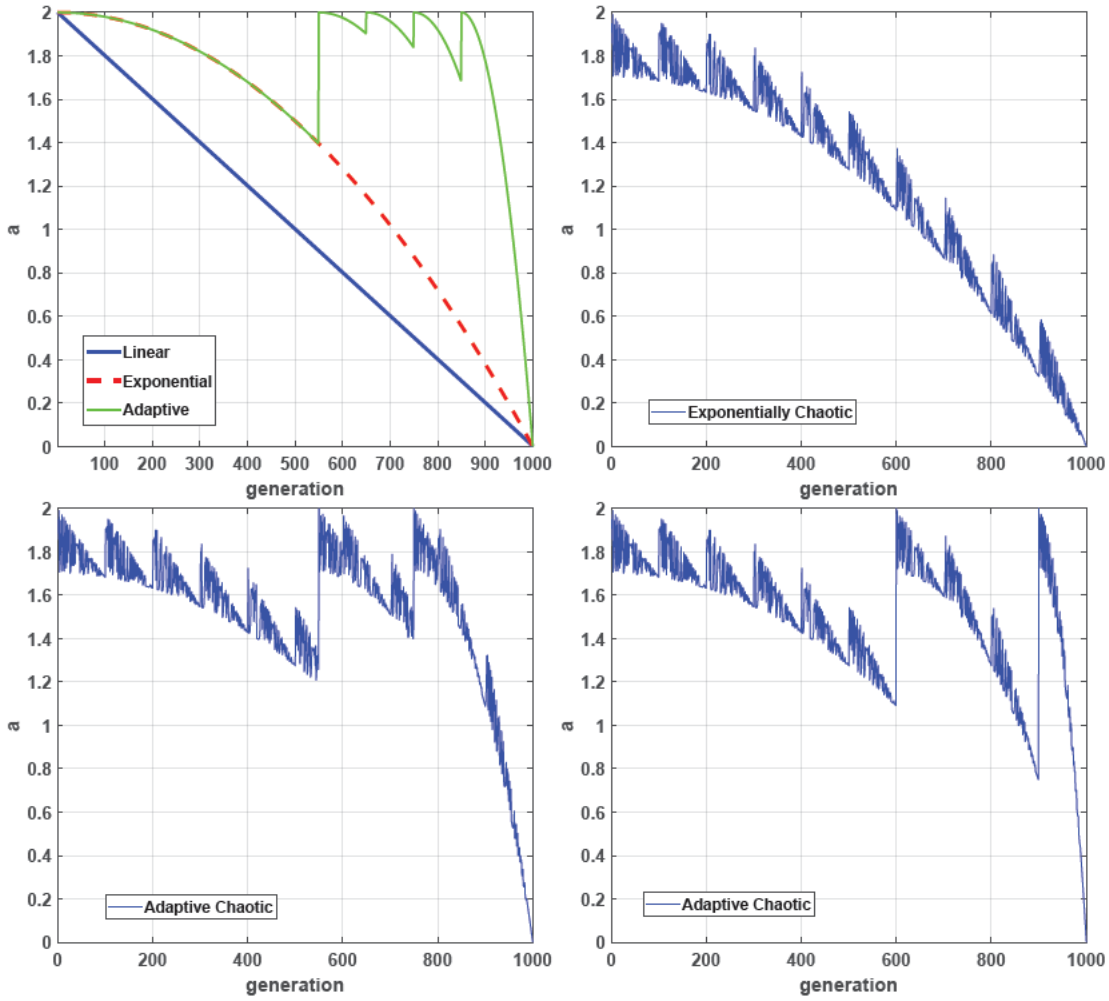

FIGURE 3.13: (a) the original strategy of control variable a decreases linearly [11](blue line) and in [12] it polynomially decreases (red line). This decay process can be adapted based on the optimisation process achievements (green line), which is introduced by this research. (b) shows a combination of polynomial and chaotic behaviour of the control variable. This chaotic behaviour is normalised and periodic. Graphs (c) and (d) show decay plots that combine both adaptive and chaotic decay. The use of adaptation and its combination with chaotic decay is new to the AGWO algorithm described here.

method switches between the exploration and exploitation phases gradually.

Max_{iter_N} is the maximum iterations in each period. Therefore, the normalised chaotic values (CC_{iter}) can be produced by Equation 3.18:

$$CC_{iter} = N_{m_{iter}} * Cf \tag{3.18}$$

where Cf is generated by the applied chaotic map. One example of the adaptive chaotic mechanism is presented in Figure 3.12(b,d). The generated value of the chaotic sequence is embedded in the control vector $\vec{a}$ and is presented in following equation:

$$a = (2 - CN_m^{Max}) - (iter_c^2 \times \frac{2 - CN_m^{Max}}{Max_{iter_c}^2}) + CC_{iter} \tag{3.19}$$

3. Introducing an adaptive mechanism for updating the control vector when the optimisation results are not satisfied for ρ iterations. When search stagnates in this way, the control vector is reset to 2, and then the decay slope of the control vector is adjusted to a sharper gradient. This results in a switch from exploitation ($|\vec{a}| < 1$) to exploration when search stagnates.

Figure 3.13 demonstrates various mechanisms for updating the control vector include linear and polynomial ideas (a), and three samples of the new adaptive chaotic method (b, c and d).

To sum up, AGWO (see Algorithm 3.6) is a combination of two ideas. First, an adaptive updating mechanism for tuning $\vec{a}$ that depends on the current search performance, in order to balance exploration and exploitation throughout the entire search. Second, a chaotic sequence coefficient scales the normalisation function and further helps prevent premature convergence.

In order to measure and test the impact of various chaotic maps on the AGWO performance, a set of well-known chaotic maps [17, 169] are applied. Table 3.7 shows the details of these maps ($M_1, M_2, ..., M_{10}$). Table 3.4 summarises the AGWO parameters.

To make use of modern computing hardware, we employ parallel computations, which we do per individual.

To show the effect of the different maps, we conduct a brief case study on 16-buoy farms for Perth. The results of Figure 3.14 and Table 3.6 are the averages of ten independent runs. It can be seen that applying the chaotic maps with the adaptive strategy results in improved performance for GWO. In comparison to GWO results, the best performance is produced by the $M_8(Singer)$ map with better convergence speed and the average total power outputs improved by 3.83% and 7.95%, respectively. Based on this performance, we use $M_8(Singer)$ for the chaotic map in the subsequent experiments.

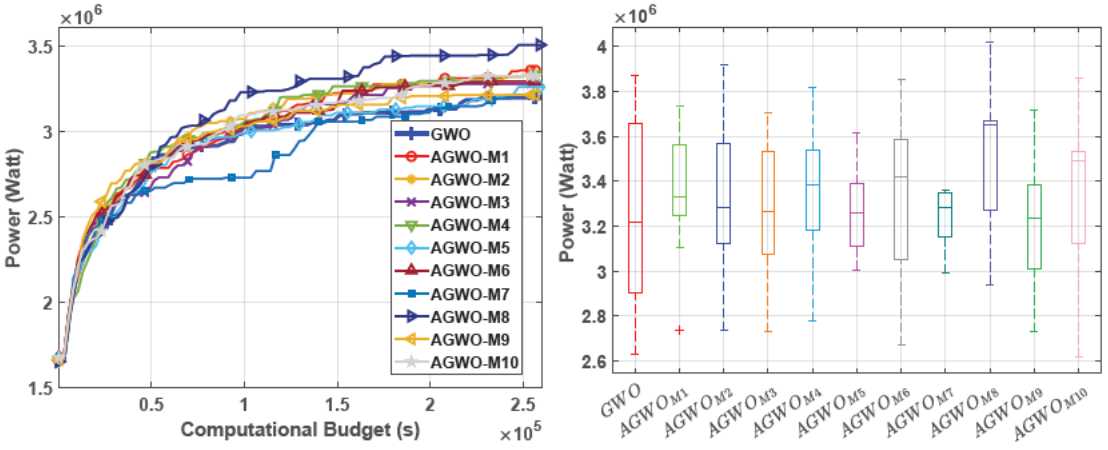

FIGURE 3.14: The convergence (a) and quality (b) comparison of the 10 different chaotic maps (M1-M10 Table 3.7) combined with AGWO performance for 16-buoy layouts in Perth wave model. (10 independent runs for each configuration.)

Algorithm 3.6 Adaptive Grey Wolf optimiser (AGWO)

procedure AGWO
  $size = \sqrt{N * 20000}$ ▷ Farm size
  $Np = 50$ ▷ Population size
  $\mathbb{S} = \{\langle x_1, y_1, B^1_{k_1}, ..., B^1_{k_{50}}, B^1_{d_1}, ..., B^1_{d_{50}}\rangle, \ldots$
, $\langle x_{Np}, y_{Np}, B^{Np}_{k_1}, ..., B^{Np}_{k_{50}}, B^{Np}_{d_1}, ..., B^{Np}_{d_{50}}\rangle\}$ ▷ Initial Population
  $energy = Eval([S_1, S_2, \ldots, S_{Np}])$ ▷ Evaluate Layouts
  Initialize parameters $a, A, C, CN^{Max}_m, CN^{Min}_m, \rho$ *and* Max_{iter_N}
  X_α=*The best layout from* $\langle S_1, \ldots, S_{Np}\rangle$ ▷ *Find three best layouts*
  X_β=The second best layout from $\langle S_1, \ldots, S_{Np-1}\rangle$
  X_δ=The third best layout from $\langle S_1, \ldots, S_{Np-2}\rangle$
  while stillTime() ***do***
    for *i in* $[1, .., Np]$ ***do***
      Update S_i *by Equation 3.14*
      if S_i is not feasible ***then***
        $S_i = Repair(S_i)$ ▷ *replacing by nearby feasible solution*
      end if
    end for
    $energy = Eval([S_1, S_2, \ldots, S_{Np}])$ ▷ *Evaluate Layouts*
    $BestEnergy_{iter}$=Max($energy$)
    if rem($iter, \rho$)=0 & $BestEnergy_{iter} < f(X_\alpha)$ **then**
      $a = 2, iter_N = 1$ *and* $iter_c = 1$ ▷ Reset control variables
      ▷ Reset iteration of normalization and chaotic sequence
    else
      Update $N_{m_{iter}}, CC_{iter}$ *and* Cf by Equation 3.17,3.18
      Update a, A *and* C by Equation 3.19,3.13,3.12
    end if
    Update X_α, X_β *and* X_δ
  end while
  return S, $energy$ ▷ *Final Layout*
end procedure

3.2.8 Cooperative optimisation methods

A standard method of employing a Cooperative Coevolution EA (CCEA) to an optimisation problem begins by decomposing the problem into components. Each component instantiates an independent population worked on by a separate EA instance. These instances run simultaneously or in an interleaved fashion and periodically share information from the best sub-solutions of each instance to improve the overall performance of search [166].

Advantages of CCEA include:

1. CCEAs provide a convenient way to decompose a high dimensional search domain into a number of lower-dimensional search domains (instantiated in each population) whilst still retaining the capacity for global search through periodic sharing of sub-solutions from each domain [170].
2. By exchanging whole or partial sub-solutions between populations, CCEAs create a dynamic search landscape [170]. This changing landscape prevents premature convergence of search.
3. The global search performance of CCEAs can be faster than canonical evolutionary algorithms due to the maintenance of a functional diversity of solutions [171].

The main disadvantages of CCEAs are:

1. CCEAs are not naturally suited to non-separable problems where interactions between problem components mean that many dimensions need to be considered at once for the global search to progress [172]. Strategies can be employed to better handle specific problems [173] but there is, to date, no single approach for applying CCEAs to inseparable problems.
2. CCEAs do not always work well when only one optimiser is applied to all sub-problems [166].
3. CCEAs also introduce the challenge of ensuring that sub-problems are computationally load-balanced to ensure that progress in sub-populations is matched [174].

Wave farm parameter (Position+PTO settings) optimisation has a very high dimensionality which makes it a challenging search problem. One natural option for dealing with this issue is to divide the decision variables into two subsets: WEC positions and PTO settings. This decreases the problem dimension and provides a more homogeneous search

space. Four cooperative optimisation techniques are proposed and compared including a new combination of AGWO and the Nelder-Mead, hybrid of (2+2)CMAES and Nelder-Mead (NM) [138], and a combination of a 1+1EA and Nelder-Mead [138], and, finally, the CCOS algorithm introduced in [166]. Details of these algorithms are as follows:

3.2.8.1 AGWO + Nelder-Mead

(AGWO-NM) As GWO is designed as an unconstrained meta-heuristic idea, it is not able to handle the constraint of WECs distances (safe distance) easily. However, GWO can be a fast and effective unconstrained optimisation method. In this way, a combination of AGWO and Nelder-Mead is proposed that AGWO adjusts the PTO configurations of WECs to achieve the highest power output and then NM is used for optimising the arrangement of buoys. This optimisation process is run iteratively using the same computational budget until the runtime (three days) runs out.

3.2.8.2 Cooperative Co-evolution with Online optimiser Selection: CCOS

The Cooperative Co-evolution with an online mechanism for selecting the suitable optimiser (CCOS) introduced by Sun et al. [166]. The CCOS consists of two general parts: decomposition and optimisation. In the first stage, a robust recursive algorithm [175] is used to group parameters into subsets based on how they correlated during optimisation. These subsets are recursively decomposed according to the strength of parameters interactions.

The algorithm is able to decompose an n-dimensional problem using $\mathcal{O}(n \log n)$ steps. During the optimisation phase, two state-of-the-art adaptive optimisers are employed; the social learning particle swarm optimiser (SLPSO [155]), and self-adaptive differential evolution with neighbourhood search (SaNSDE [156]). The main contributions of SaNSDE are 1) incorporating the search biases of distributions, Cauchy and Gaussian operators. SaNSDE takes into account the trade-off between small and large mutation step sizes; 2) all control parameters of SaNSDE are self-adapted based on statistical performance tracking during the optimisation process. Moreover, to assess the CCOS algorithm (SLPSO+SaNSDE) thoroughly, we compared CCOS's performance against the performances of a double SLPSO ($SLPSO_{II}$) and SaNSDE ($SaNSDE_{II}$). This evaluation helps isolate the impact of using these optimisers cooperatively.

3.2.9 Hybrid optimisation algorithms

In the earlier work, a practical WECs optimisation idea was developed [1] called Local Search + Nelder Mead (LS-NM); that showed using a local sampling by a normal distribution in the previous buoy's neighbourhood (outside of the safe distance) combined with greedy selection could produce high-performing layouts. Such one-at-a-time placement is a fast optimisation strategy. However, tuning the position of the placed buoys required a considerable computational budget. This work also did not consider other WEC optimisation parameters such as PTO settings.

More recently, Neshat et al. [138] proposed an improved heuristic (SLS-NM-B) for placing the new WEC one-at-a-time and tuning PTOs settings. As local sampling in LS-NM is done without strong regard to useful priors of direction and distance, a repaired step is needed to modify the current position. SLS-NM-B represented a symmetric local search with the deterministic directions and bounded search space. Moreover, a backtracking strategy was introduced for improving the WEC parameters, including both position and PTO settings (with the latter being tuned in unison for each buoy). Nevertheless, SLS-NM-B was not designed to handle the high dimensional search problem that arises when all PTO frequency response settings are allowed to move independently. This is because the Nelder-Mead optimiser that is applied for tuning the PTO parameters converges extremely slowly in the high-dimensional search space. Moreover, it has not been proven that the PTO parameter space in this problem is uni-modal and so the downhill search heuristics such as Nelder-Mead may not be suitable for global optimisation.

Algorithm 3.7 Hybrid Cooperative Co-evolution Algorithm (HCCA)

procedure HCCA
Initialization
$size = \sqrt{N * 20000}$ ▷ Farm size and N is buoy number
$\mathbb{A}_{ii} = \{1 \leq ii \leq |\mathbb{A}|\}$ ▷ candidate optimisers
$angle = \{0, 45, 90, \ldots, 315\}$ ▷ symmetric samples angle
$iters = Size([angle])$ ▷ Number of symmetric samples
$\mathbb{S} = \{\langle x_1, y_1, B^1_{k_1}, ..., B^1_{k_{50}}, B^1_{d_1}, ..., B^1_{d_{50}}\rangle, \ldots$
$\ldots, \langle x_N, y_N, B^N_{k_1}, ..., B^N_{k_{50}}, B^N_{d_1}, ..., B^N_{d_{50}}\rangle\}$ ▷ Positions&PTOs
Decomposition
$\langle \mathbb{S}_1, \mathbb{S}_2, ..., \mathbb{S}_N\rangle = Decompose(\mathbb{S})$ ▷ Decomposing $\mathbb{S}$ per buoy
$$\begin{cases} \mathbb{S}_1 = \{\langle x_1, y_1\rangle, \langle B^1_{k_1}, \ldots, B^1_{k_{50}}, B^1_{d_1}, \ldots, B^1_{d_{50}}\rangle\} = \perp \\ \mathbb{S}_2 = \{\langle x_2, y_2\rangle, \langle B^1_{k_1}, \ldots, B^2_{k_{50}}, B^2_{d_1}, \ldots, B^2_{d_{50}}\rangle\} = \perp \\ \ldots \\ \mathbb{S}_N = \{\langle x_N, y_N\rangle, \langle B^N_{k_1}, \ldots, B^N_{k_{50}}, B^N_{d_1}, \ldots, B^N_{d_{50}}\rangle\} = \perp \end{cases}$$
$U_{\mathbb{A}_{ii},\mathbb{S}_i} = 0$ ▷ Initialize the accumulated contributions of optimisers
$S_1 = \{\langle size, 0\rangle, \langle \vec{r_1} \times Max_k, \vec{r_2} \times Max_d\rangle\}$ ▷ initialize first buoy
Hybrid Cooperative Co-Evolution Framework
if $i = 1$ **then** ▷ optimise first buoy PTOs by the optimisers
First Step
for ii in $|\mathbb{A}|$ **do** ▷ Calculate contribution
$\langle I_{(\mathbb{A}_{ii},\mathbb{S}_i)}, Energy\rangle$=$optimise$ $(\mathbb{S}_{i_{PTOs}}, \mathbb{A}_{ii})$
$U_{(\mathbb{A}_{ii},\mathbb{S}_i)}$=$(\hat{U}_{(\mathbb{A}_{ii},\mathbb{S}_i)} + I_{(\mathbb{A}_{ii},\mathbb{S}_i)})$ / 2 ▷ Accumulate contribution
end for
$\mathbb{S}_{i+1\,PTOs} = \mathbb{S}_{i_{PTOs}}$
end if
$BestIndex = Max(U_{(\mathbb{A}_{ii},\mathbb{S}_i)} \rightarrow 1 \leq ii \leq |\mathbb{A}|)$
for i in $[2, .., N]$ **do** $bestEnergy = 0$;
Second Step: Symmetric Sampling
for j in $[1, .., iters]$ **do**
$(Sample_j, energy_j)$=$SymmetricSample(angle_j, \mathbb{S}_{(i-1)})$
if $Sample_j$ *is feasible* & $energy_j > bestEnergy$ **then**
$tPos = Sample_j$ $bestEnergy = energy_j$ $bestAngle = j$ ▷ Temporary position
end if
end for
(Es_1, Es_2)=$SymmetricSample(bestAngle \pm 15, \mathbb{S}_{(i-1)})$
$(\mathbb{S}_{(i)}, energy)$=$FindbestS(tPos, Es_1, Es_2)$
Third Step: PTO settings Optimisation by CCEA
$\langle I_{(\mathbb{A}_{BestIndex},\mathbb{S}_i)}, Energy\rangle$=$optimise$ $(\mathbb{S}_{i_{PTOs}}, \mathbb{A}_{BestIndex})$
$U_{(\mathbb{A}_{BestIndex},\mathbb{S}_i)}$=$(\hat{U}_{(\mathbb{A}_{BestIndex},\mathbb{S}_i)} + I_{(\mathbb{A}_{BestIndex},\mathbb{S}_i)})$ / 2
$BestIndex = Max(U_{(\mathbb{A}_{ii},\mathbb{S}_i)} \rightarrow 1 \leq ii \leq |\mathbb{A}|)$
$\mathbb{S}_{i+1\,PTOs} = \mathbb{S}_{i_{PTOs}}$
Fourth Step: Position Optimisation by Nelder-Mead
$(\mathbb{S}_i, Energy)$=$Nelder\text{-}Mead(\mathbb{S}_{i_{Position}})$
end for
Final Step: BackTracking Optimisation
$\langle \mathbb{S}, Energy\rangle$=$BackTracking$ $(\mathbb{S}, \mathbb{A}_{BestIndex})$
end procedure

3.2.9.1 Hybrid Cooperative Co-evolution algorithm (HCCA)

One of the most effective strategies for solving the large-scale optimisation problems is Cooperative Co-evolution (CC) framework [176]. In CC, the general idea is dividing the decision variables into some components (decomposition) and employing one or more optimisers in a round-robin fashion (in biased or unbiased mode) for optimising the sub-problems. In this paper, as a combination of WECs placements and PTOs settings forms a large number of decision variables ($N \times 102$) with a complex search space, we propose a new hybrid Cooperative Co-evolution (HCCA) method. The steps of the proposed hybrid algorithm are described in more details as follows.

Decomposition: In the decomposition phase, we apply a knowledge-based approach according to the significant WECs hydrodynamic rule [177]. The rule is that both PTO parameters (damping coefficient ($dPTO$) and spring stiffness ($kPTO$)) of each converter should be optimised together. Therefore, the problem is decomposed into two sub-problems for each WEC, including PTO settings ($\langle B^i_{k_1}, ..., B^i_{k_{50}}, B^i_{d_1}, ..., B^i_{d_{50}} \rangle, 100D$) and position ($\langle x_i, y_i \rangle, 2D$).

Optimisation: The HCCA optimisation phase is comprised of optimisation phases for the two-parameter groups listed above. For the buoy position parameter-group, the hybrid systematic neighbourhood search from (SLS-NM) [138] is applied by first uniformly sampling in search sectors whose boundaries are informed by an initial 2-buoy power landscape analysis. After this, a Nelder-Mead search is used to improve the best-sampled positions.

In the symmetric local search (SLS) method, the angle interval ($angle$) is selected to trade-off between the number of neighbourhood samples and the computational cost of their evaluation. A similar trade-off was explored for setting the parameters representing the number of allowed evaluations and the step size (σ) of the Nelder-Mead phases of the search.

In the second group of the optimisation, we propose a Cooperative Co-evolution idea for adjusting the PTOs settings that is a large-scale optimisation problem ($N \times 100$). This CC framework is composed of three modern and efficient optimisers, SLPSO [155], SaNSDE [156] and the new proposed adaptive grey wolf optimiser (AGWO). The SLPSO is a competitive optimiser [166] for working in the context of CC because 1) it is computationally efficient, 2) needs no complicated fine-tuning of the control parameters, 3) has a high exploitation ability and convergence speed, and 4) has shown to perform well on other high-dimensional optimisation problems. However, converging to a local

optimum can be a problem encountered with SLPSO. Consequently, for developing the CC framework, combine this with another optimiser with a high capability of the exploration. The SaNSDE optimiser has considerable capacity for exploration and has been broadly applied in the CC domain [166]. The third optimiser used here in CC framework is AGWO which is the new GWO variant described earlier. During optimisation these three optimisers share the same population and and solve the components collaboratively.

The pseudo-code of the proposed HCCA algorithm for solving the WEC optimisation problem is shown in Algorithm 3.7.

Backtracking: After initial placement and PTO optimisation by the CC framework above a customised backtracking optimisation algorithm (BOA) refines both buoy positions and PTO parameters. For positions, the buoys with the lowest power output are selected and then Nelder-Mead (NM) is applied for optimising the positions one at a time. For PTO parameters, an optimiser is selected as the best prior optimiser performance during the first search phase. The selected optimiser is then used to tune all of the PTOs settings of the layout in all-at-once global search.

The pseudo-code of the backtracking approach is given in Algorithm 3.8. Figure 3.15 provides a graphical view of the proposed hybrid optimisation framework. In the first cycle, after placing the first buoy in a predefined location (recommended by [138]), the three optimisers ($\mathbb{A}$) are employed to resolve PTOs settings ($\mathbb{S}_{j_{\text{PTOs}}}$). Each optimiser is given the same computational budget.

Next, each optimiser's contribution is computed as a fitness improvement ($I_{(\mathbb{A}_i,\mathbb{S}_j)}$):

$$I_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})} = \frac{f(\acute{\mathbb{S}}_{j_{\text{PTOs}}}) - f(\mathbb{S}_{j_{\text{PTOs}}})}{f(\acute{\mathbb{S}}_{j_{\text{PTOs}}})} \quad i \in \{1 \leq i \leq |\mathbb{A}|\}, \quad j \in \{1 \leq j \leq N\} \tag{3.20}$$

where $f(\acute{\mathbb{S}}_{j_{\text{PTOs}}})$ and $f(\mathbb{S}_{j_{\text{PTOs}}})$ show the power of the layout obtained before and after employing i^{th} optimiser in one cycle. The fitness improvement is a measure of the optimiser's ability to adjust the j^{th} buoys PTOs settings. For updating the fitness improvement for each optimiser during the whole optimisation process, an accumulated contribution variable is used [166]. This performance tracking is encoded in Equation 3.21.

$$U_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})} = \frac{\acute{U}_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})} + I_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})}}{2} \tag{3.21}$$

where $\acute{U}_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})}$ tracks each optimiser's ($\mathbb{A}_i$) accumulated contributions. In the first cycle, the $\acute{U}_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})}$ is initialised to 0. The accumulated contribution $U_{(\mathbb{A}_i,\mathbb{S}_{j_{\text{PTOs}}})}$ is the average of all fitness profits for each optimiser from previous cycles. In the next iteration, the

Perth wave scenario (16-buoy)											
	GWO	$AGWO_{M_1}$	$AGWO_{M_2}$	$AGWO_{M_3}$	$AGWO_{M_4}$	$AGWO_{M_5}$	$AGWO_{M_6}$	$AGWO_{M_7}$	$AGWO_{M_8}$	$AGWO_{M_9}$	$AGWO_{M_{10}}$
Max	3869188	3736690	3921497	3704994	3815293	3618343	3855858	3361619	**4017436**	3717341	3858006
Min	2631382	2737420	2735143	2728993	2778637	3006472	2671713	2993289	2940571	2731852	2616268
Mean	3258013	3361803	3324371	3276833	3335201	3258500	3324113	3216207	**3517184**	3212008	3331664
Median	3218467	3328703	3285224	3265160	3382717	3235225	3417449	3230982	**3664481**	3234115	3493016
STD	428448	278820	328343	280856	288771	199893	385533	142654	322678	305389	350371

TABLE 3.6: Results of 10 chaotic maps on the case study of Perth wave model on AGWO

best optimiser accumulated contribution ($BestIndex = Max(U_{(\mathbb{A}_i, \mathbb{S}_{jPTOs})} \rightarrow 1 \leq i \leq |\mathbb{A}|)$) will be selected for optimising the PTOs settings of the next placed generator. Figure 3.16 presents the average contribution of each applied optimiser for tuning the PTOs configuration of 16-buoy layouts in the Perth wave model. It can be seen that after the initial cycle the percentage contribution of SLPSO and SaNSDE is more than AGWO; however, for the last generations, AGWO's contribution is larger than that of the other optimisers. The used symbols are listed and introduced in Table 3.5.

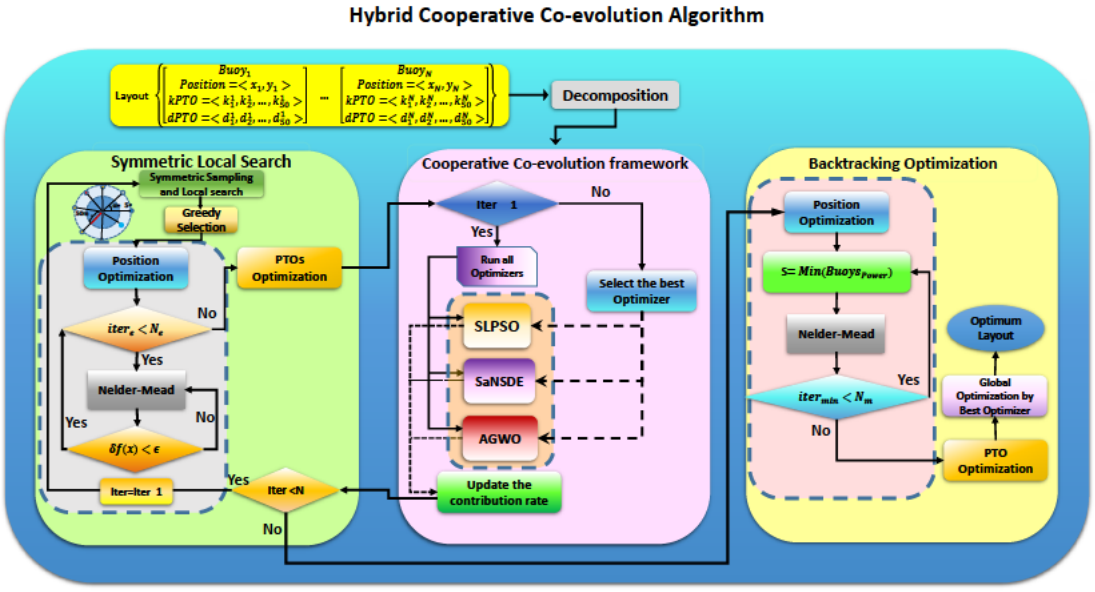

FIGURE 3.15: Outline of the Hybrid Cooperative Co-evolution Algorithm (HCCA). N, N_e and N_m are the maximum buoy number in the layout, the maximum evaluation number of Nelder-Mead and the buoy numbers for refining their positions. ϵ is the Nelder-Mead function tolerance and stopping criterion.

3.2.10 Experiments

This section starts with a small landscape study of the PTO parameter settings for a single buoy for the Perth wave scenario followed by a sensitivity analysis of the best-obtained 16-buoy layouts position. After that, we present the optimisation outcomes of the experiments comparing the effectiveness of the proposed algorithms explained above

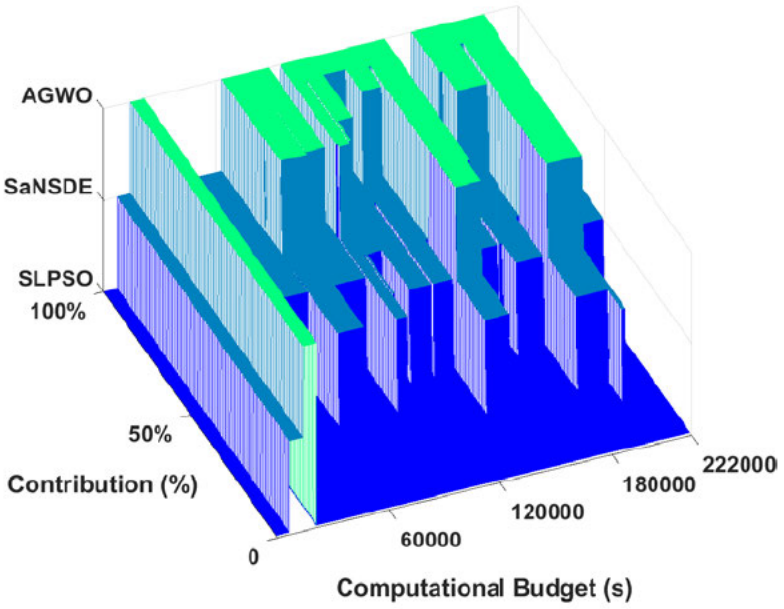

FIGURE 3.16: The contribution percentage of HCCA optimisers (SLPSO, SaNSDE and AGWO) in the optimisation process when used to optimise the PTOs settings of 16-buoy layout in Perth wave Scenario.

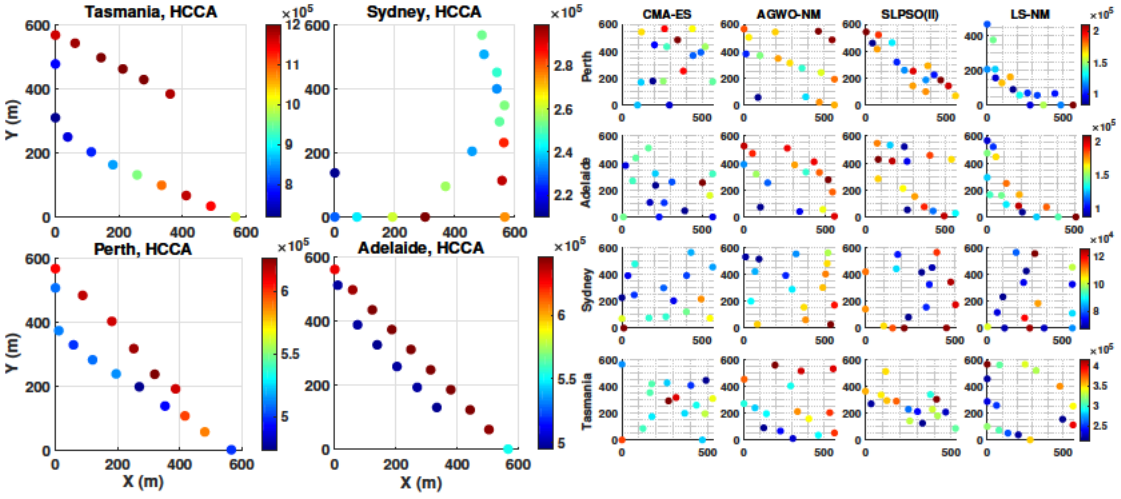

FIGURE 3.17: The best-found 16-buoy layouts arrangement of the four real wave scenarios based on Table 3.9.

applied to WEC positions and PTOs settings under four real wave scenarios. In order to characterise the scalability of the frameworks, we apply them to both the 4-buoy ($= 4 \times 102D$), and 16-buoy ($= 16 \times 102D$) have been evaluated in the experiments. Finally, the algorithm convergence rates and the quality of the obtained solutions are also compared.

NO.	Name	Chaotic Map	Range
1	Chebyshev	$x_{i+1} = cos(icos^{-1}(x_i))$	(-1,1)
2	Circle	$x_{i+1} = mod(x_i + b - (\frac{a}{2\pi})sin(2\pi x_i), 1), a = 0.5 \; and \; b = 0.2$	(0,1)
3	Gauss/mouse	$x_{i+1} = \begin{cases} 1 & x_i = 0 \\ \frac{1}{mod(x_i,1)} & \text{otherwise} \end{cases}$	(0,1)
4	Iterative	$x_{i+1} = sin(\frac{a\pi}{x_i}), \; a = 0.7$	(-1,1)
5	Logistic	$x_{i+1} = ax_i(1 - x_i), \; a = 4$	(0,1)
6	Piecewise	$x_{i+1} = \begin{cases} \frac{x_i}{P} & 0 \le x_i < P \\ \frac{x_i - P}{0.5 - P} & P \le x_i < 0.5 \\ \frac{1 - P - x_i}{0.5 - P} & 0.5 \le x_i < 1 - P \\ \frac{1 - x_i}{P} & 1 - P \le x_i < 1 \end{cases}, P = 0.4$	(0,1)
7	Sine	$x_{i+1} = \frac{a}{4}sin(\pi x_i), \; a = 4$	(0,1)
8	Singer	$x_{i+1} = \mu(7.86x_i - 23.31x_i^2 + 28.75x_i^3 - 13.302875x_i^4) \;\; , \mu = 1.07$	(0,1)
9	Sinusoidal	$x_{i+1} = ax_i^2 sin(\pi x_i) \;\; , a = 2.3$	(0,1)
10	Tent	$x_{i+1} = \begin{cases} \frac{x_i}{0.7} & x_i < 0.7 \\ \frac{10}{3}(1 - x_i) & x_i \ge 0.7 \end{cases}$	(0,1)

TABLE 3.7: The applied chaotic maps from [17].

3.2.10.1 Landscape analysis

3.2.10.1.1 PTOs settings analysis

In recent work [138], the impact of PTO parameter optimisation where these control parameters are kept the same for all wave frequencies for each buoy, was investigated and presented. This work found that tuning the PTOs parameters can be effective in optimising the total absorbed power of WECs (CETO model) in both Perth and Sydney wave climates by 4.48% and 2.42%, respectively. Figure 3.18 illustrates the PTO power landscape of one buoy with a simple grid search for tuning the damping-spring variables, where settings are kept the same for all wave frequencies.

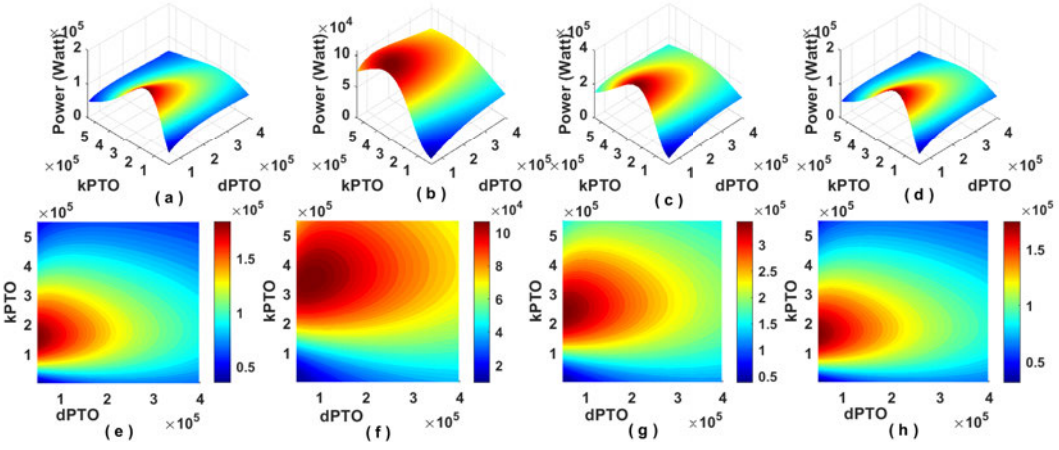

FIGURE 3.18: PTOs settings power landscape analysis of four real wave scenarios (Adelaide(a,e), Sydney(b,f), Tasmania (c,g) and Perth (d,h)) for one buoy layout. We assume the most straightforward experiment of PTOs settings with the same value for all 50 wave frequencies. The spring-damping PTO configuration step size is 2500. Note that the real PTOs configurations search space is multi-modal and different values can be assigned to each wave frequency.

However, in the real sea states, WECs control parameters (PTOs) should be tuned for each wave frequency. By tuning these parameters independently for each buoy, it is possible to extract more power. We allow the PTO settings for all frequencies to be used for all proposed optimisation methods in this paper. For visualising the potential impact of PTO parameter optimisation for each wave frequency, we conduct a simple experiment. Since the dimensionality of this problem is high (2×50 for a single buoy), we divide the 50 frequencies into 10 groups. Each group includes five sequential frequencies, and we constrain them to have the same PTOs parameters. The 45 wave frequencies in each group are assigned the manufacturer's PTOs defaults ($k = 407510$ and $d = 97412$) [1]. Figure 3.19 shows the modified PTOs optimisation power landscape of 10 groups for one buoy in Perth wave model; for mixing all ten surfaces at one 3D figure, we use a normalised version of all landscapes (left figure) as a multi-layer 3D plot. We can see that this simplified search space of just one buoy's PTO configuration is multi-modal and complex to search[1]. It is also of note that, even in this constrained search environment, there is a 3-fold improvement in extracted energy compared to previous studies [1, 138].

Moreover, to provide an alternative visualisation of this experiment, a 4D power landscape is plotted. Figure 3.20 presents a trade-off of damping ($dPTO$), spring ($kPTO$), wave frequency and absorbed power. We can see that a specific range of frequencies with tuned values of PTO settings can produce more power. Note however the figure is plotted for one fixed buoy without the complex details of hydrodynamic interactions between buoys in the wave farm. A-priori, it is expected that introducing more buoys will produce interactions that will increase the complexity of this landscape further.

3.2.10.1.2 Position analysis

For evaluating the position sensitivity of the best-found 16-buoy arrangement in four real wave models, a practical experiment is done. In the first step, we perturb each generator's position by a random variable with a normal distribution ($\mu = (x_i, y_i)$ and $\sigma = 1m$) 100 times. Secondly, we perturb all buoys position by this strategy. Figure 3.21 demonstrates the results of both perturbation experiments and the best 16-buoy layout power. We can see that this practical analysis is able to improve the total power output of Adelaide wave site by 0.04%. This minor improvement shows that the proposed optimisation method (HCCA) can converge very close to a local optimum within the limited computational budget.

[1] Because this diagram is a low-dimensional projection from a higher-dimensional landscape it cannot be automatically assumed that the higher-dimensional landscape for PTO optimisation at least, is also multi-modal. Note that previous work has shown that the buoy-positioning landscape is multi-modal, but exploring the multi-modality or otherwise of the entire higher-dimensional search landscape for PTO settings is future work.

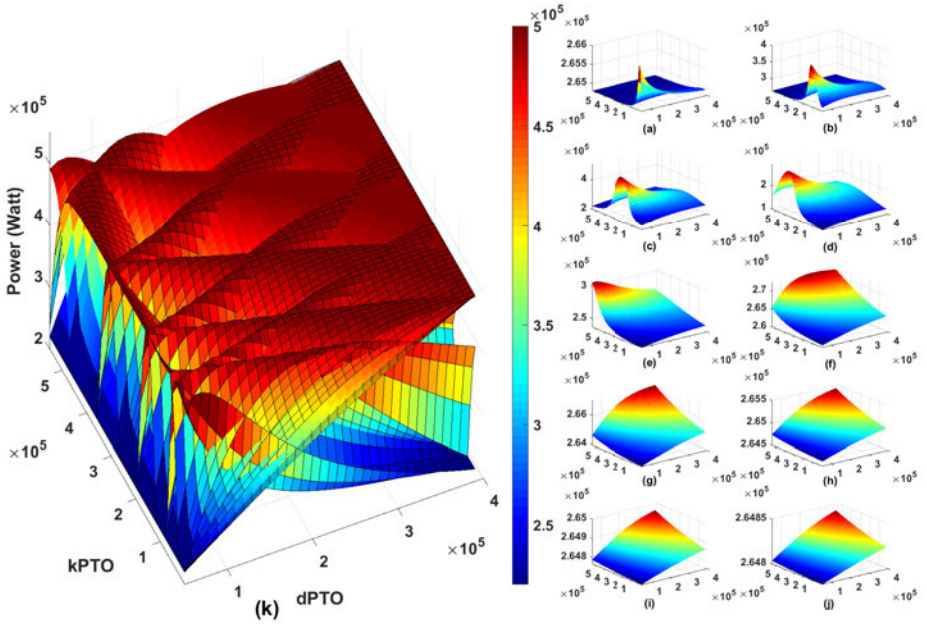

FIGURE 3.19: The simplified power landscape of one buoy where PTO parameters are evaluated in ten sequential five-wave frequency groups. Figure (a) demonstrates the PTO (damping-spring parameters) power landscape of one buoy when we assume k and d parameters for $f_1, f_2, ..., f_5$ are the same and other 45 wave frequencies are set by the predefined value ($dPTO = 97412, kPTO = 407510$). Other figures follow the same pattern for instance Figure (b) represents the performance of a simple grid search ($Step = 10000$) for plotting the power landscape of $f_6, f_7, ..., f_{10}$ of tuned PTO parameters. In the left side, Figure (k) shows a normalised overlapping of the surfaces of all ten landscapes in one graph for depicting the complexity level of the search space.

For the other three wave scenarios (Sydney, Tasmania, and Perth), the position analysis experiments cannot find a better configuration than the HCCA optimisation results. The perturbation loss, respectively, for the best 16-buoy layouts power in Sydney, Tasmania and Perth wave farms are $0.67\%, 0.49\%$ and 0.12% on average. According to the results, the power outputs for the best-found layouts are relatively insensitive to small perturbations in buoy position – this is a good outcome in that small errors in buoy placement in a real environment are unlikely to have a major impact on power output.

3.2.10.2 optimisation Experiments

In this part, we summarise the experimental results from optimising the layout and PTO parameters of 4-buoy arrays and then 16-buoy arrays. The 16-buoy experiments are expected to be more challenging due to the larger number of parameters and a much larger number of buoy interactions.

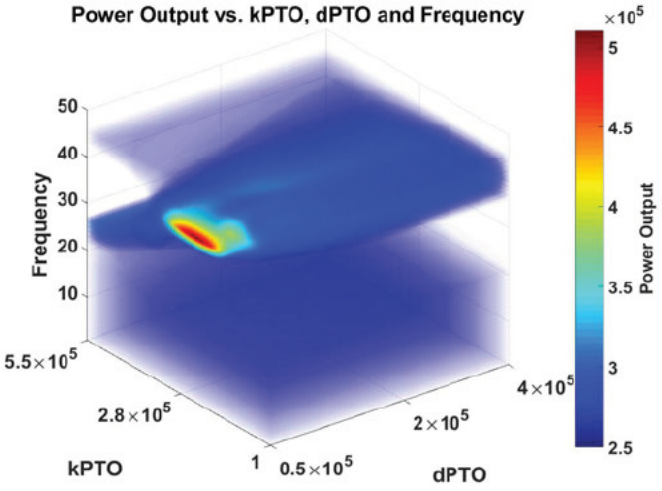

FIGURE 3.20: A 4D view PTO power landscape for one buoy in the Perth wave model.

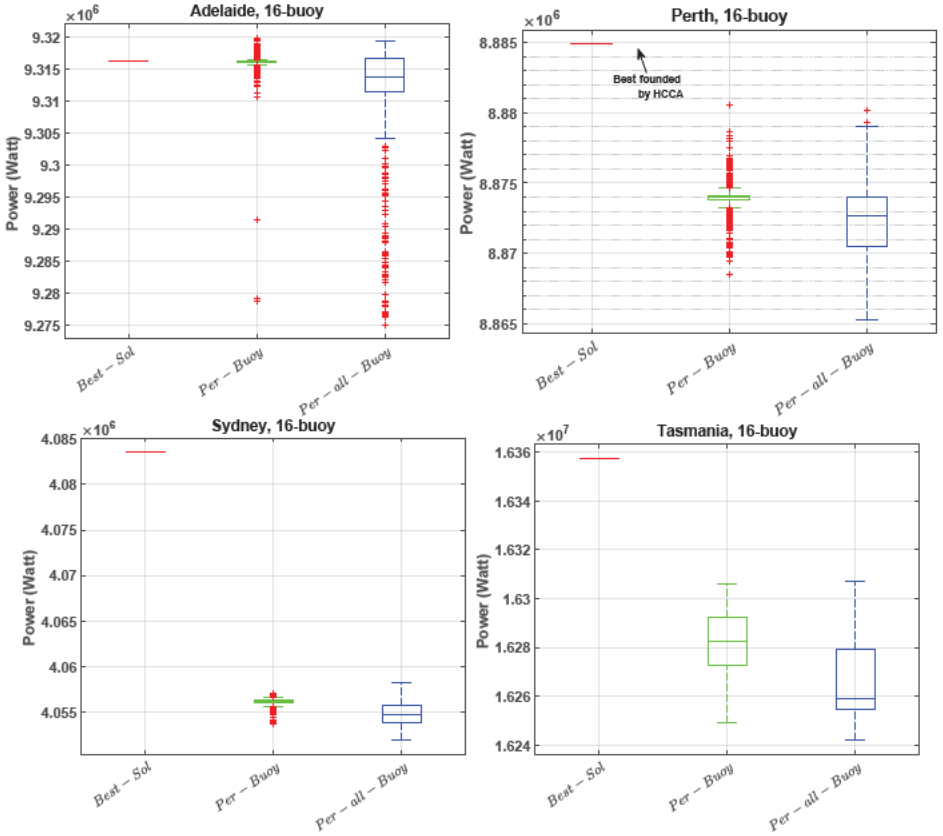

FIGURE 3.21: The position perturbation experimental results 16-buoy in 4 real wave models per each buoy and all buoys.

3.2.10.2.1 4-buoy layout results

In the experiments on 4-buoy layouts (medium-scale optimisation problem), we compare 15 representative meta-heuristic algorithms with our HCCA, including five well-know EAs plus a new adaptive version of GWO, six cooperative EAs and two-hybrid heuristic

approaches in four real wave scenarios. The parameter settings for these meta-heuristic variants are summarised in Table 3.4. The computational budget is three days on 12 CPU cores in parallel.

Figure 3.23 presents the box-and-whiskers plot for the best-found 4-buoy configurations which produce the maximum power output for each run for 16 search heuristics for four real wave models. It can be seen that the performance of cooperative co-evolution strategies (CCOS, SLPSO$_{II}$ and SaNSDE$_{II}$ and the new hybrid method are considerably better than other applied meta-heuristic algorithms. The next best performances are exhibited by both SLS-NM and SLS-NM-B; however, the average absorbed power by these methods is less than the CC and HCCA approaches by 25%.

Looking more closely at Table 3.10, we can observe the highest absorbed power of 4-buoy layouts are found by SLPSO$_{II}$, CCOS and HCCA, respectively. These optimisation results are closely followed by the SaNSDE$_{II}$ algorithm. These competitive performances are supported by the statistical test results, ranked using the non-parametric Friedman test, shown in Table 3.8. It is noteworthy that among the five optimisation methods in the all-at-once strategy, the GWO and AGWO performances are substantially better than the others.

Viewing the convergence curves (Figure 3.24) from this experiment (4-buoy) in four real wave models, it is clear that the HCCA framework converges faster for the 4-buoy layout than other search methods. Furthermore, HCCA improves beyond the power outputs achieved by other methods when it has consumed just 20% of its 3-day computational budget.

3.2.10.2.2 16-buoy layout results

As evaluating one 16-buoy layout is ten times more expensive than a 4-buoy layout, optimising such large wave farms is a challenging problem. According to the statistical results of Table 3.9, we see that the average 16-buoy layouts power output which are found by HCCA is increased substantially to 80% more than previous research outcomes (SLS-NM-B [138]) in all wave scenarios.

Table 3.8 presents the average rank of all heuristic methods for 16-buoy experiments, that HCCA, SLS-NM-B and SLS-NM have the highest rank, respectively.

The best-found 16-buoy layouts power output for each run of all heuristic methods are plotted as a box plot by Figure 3.22. Figure 3.22 shows that HCCA performs much better than other optimisation algorithms, as mentioned before. After SLS-NM and SLS-NM-B, The efficiency of the AGWO-NM, AGWO and GWO are competitive compared with

other cooperative and generic EAs. The primary reason is derived from the robust exploitation and exploration capability of GWO for PTO parameter optimisation and having good performance for high-dimensional problems. It is noteworthy that the CC frameworks (CCOS, SLPSO$_{II}$ and SaNSDE$_{II}$) are not shown to be highly effective. This may be because the CC framework is not equipped with the systematic position optimisation mechanism (SLS) used by some of the one-at-a-time placement algorithms.

Figure 3.25 illustrates the convergence rate of the proposed methods experiments during the three-day runtime budget for 16-buoy layouts. As we can see, GWO and their modified versions rapidly converge to effective configurations; however, they could not keep this upward trend and converge toward locally optimal settings. HCCA, clearly, has the fastest convergence speed in the four-wave models. SLS-NM-B is able to converge to a reasonable configuration, but it does not outperform HCCA because of the low efficiency of Nelder-Mean optimising the large-scale PTO parameter part of the problem. Another important observation is that the CC approaches seem to suffer from premature convergence. In addition, they also appear to be not fast enough for such expensive optimisation problems which allow just a few thousand full evaluations (3×10^3). The best 16-buoy layouts of the nominated five methods among all heuristics can be shown in Figure 3.17, including HCCA, CMA-ES, AGWO-NM, SLPSO$_{II}$ and LS-NM in four real wave scenarios. In terms of position optimisation, it is clearly observed that HCCA is able to adjust the position of each generator successfully, which leads to a distinctive pattern of one or more rows, roughly aligned with the norm of the dominant wave direction, for placing the 16 buoys. This pattern is associated with a fast position optimisation mechanism of SLS-NM [138] that plays the role of one component in HCCA. In contrast to this, the other best layouts do not seem to exhibit any obvious structure.

Algorithm 3.8 Backtracking Search Algorithm (BSA)

procedure BSA $(\mathbb{S}, \mathbb{A}\ )$
Initialisation
$energy = ([E_1, E_2, \ldots, E_N]) = Eval(\mathbb{S})$ ▷ Evaluate layout
$N_w = round(N/4)$ ▷ Buoy number need to be improved
$\langle WIndex \rangle = FindWorst(energy, N_w)$ ▷ Find worst buoys power
  for i in $[1, .., N_w]$ **do**
    Position optimisation
    $(\mathbb{S}^{Position}_{WIndex(i)}, energy_{WIndex(i)}) = Nelder\text{-}Mead(\mathbb{S}^{Position}_{WIndex(i)})$
  end for
  PTO global optimisation
  $(\mathbb{S}_{PTOs}, energy) = optimise(\mathbb{S}_{PTOs}, \mathbb{A}_{BestIndex})$
  return $\mathbb{S}, energy$ ▷ *Final Layout*
end procedure

TABLE 3.8: The average ranking of the proposed methods by non-parametric statistical test (Friedman test).

	4-buoy					16-buoy			
Rank	**Perth**	**Adelaide**	**Sydney**	**Tasmania**	**Rank**	**Perth**	**Adelaide**	**Sydney**	**Tasmania**
1	CCOS (1 50)	SLPSO (1 33)	CCOS (1 58)	SLPSO (1 00)	1	HCCA (1 00)	HCCA (1 00)	HCCA (1 00)	HCCA (1 00)
2	SLPSO (1 83)	CCOS (1 66)	SLPSO (1 83)	SaNSDE (2 25)	2	SLS-NM-B (2 16)	SLS-NM-B (2 58)	SLS-NM-B (2 33)	SLS-NM-B (2 50)
3	HCCA (2 83)	SaNSDE (3 41)	HCCA (3 00)	HCCA (2 83)	3	SLS-NM (3 16)	SLS-NM (2 66)	AGWO (3 75)	AGWO-NM (3 33)
4	SaNSDE (3 83)	HCCA (3 58)	SaNSDE (3 58)	CCOS (4 33)	4	AGWO-NM (4 58)	AGWO-NM (4 66)	SLS-NM (4 33)	SLS-NM (3 58)
5	SLS-NM-B (5 42)	SLS-NM-B (5 25)	SLS-NM-B (5 58)	SLS-NM-B (5 08)	5	AGWO (4 75)	AGWO (4 75)	AGWO-NM (4 58)	GWO (5 25)
6	SLS-NM (5 58)	SLS-NM (5 75)	SLS-NM (5 75)	SLS-NM (5 5)	6	GWO (5 41)	GWO (5 33)	GWO (5 16)	AGWO (5 41)
7	1+1EA-NM (7 25)	1+1EA-NM (7 50)	1+1EA-NM (6 75)	1+1EA-NM (7 00)	7	CCOS (7 00)	CCOS (7 66)	CCOS (7 00)	CCOS (6 91)
8	AGWO-NM (9 08)	AGWO (8 75)	CMAES-NM (8 50)	CMAES-NM (8 66)	8	SLPSO (8 08)	SLPSO (8 16)	SLPSO (8 00)	SLPSO (8 08)
9	AGWO (9 33)	AGWO-NM (9 50)	AGWO-NM (9 91)	AGWO-NM (9 00)	9	SaNSDE (9 08)	SaNSDE (8 50)	SaNSDE (8 83)	SaNSDE (9 16)
10	GWO (10 33)	GWO (10 81)	AGWO (9 91)	AGWO (10 58)	10	LS-NM (10 16)	LS-NM (9 91)	LS-NM (10 58)	LS-NM (10 50)
11	CMAES-NM (10 75)	CMAES-NM (9 83)	GWO (10 08)	GWO (10 75)	11	CMAES-NM (11 66)	CMAES-NM (11 25)	PSO (11 16)	CMAES-NM (11 16)
12	PSO (11 91)	PSO (11 00)	LS-NM (12 25)	LS-NM (11 25)	12	DE (12 83)	CMA-ES (12 58)	CMA-ES (13 16)	PSO (11 91)
13	LS-NM (11 91)	LS-NM (12 41)	PSO (12 25)	PSO (12 75)	13	CMA-ES (13 00)	DE (13 33)	CMAES-NM (13 91)	1+1EA-NM (13 66)
14	DE (14 25)	DE (14 08)	DE (14 16)	DE (14 25)	14	1+1EA-NM (13 41)	PSO (13 58)	1+1EA-NM (14 00)	DE (13 66)
15	CMA-ES (14 91)	CMA-ES (14 91)	NM (15 33)	CMA-ES (14 91)	15	PSO (13 75)	1+1EA-NM (14 00)	DE (14 08)	CMA-ES (13 83)
16	NM (15 25)	NM (16 00)	CMA-ES (15 50)	NM (15 83)	16	NM (15 91)	NM (16 00)	NM (14 08)	NM (16 00)

Perth wave scenario (16-buoy)																
	DE	**CMA-ES**	**NM**	**GWO**	**AGWO**	**PSO**	**CMAES-NM**	**1+1EA-NM**	**AGWO-NM**	**CCOS**	**SLPSO$_{II}$**	**SaNSDE$_{II}$**	**LS-NM**	**SLS-NM**	**SLS-NM-B**	**HCCA**
Max	1978004	1972042	1707140	3869189	4017436	1874697	2205390	1985315	4071870	2926335	2666886	2460983	2316240	4613064	5355093	**8884930**
Min	1813302	1764271	1393953	2631383	2940571	1798449	1527216	1508531	3048410	2395674	2294175	2092220	1887327	3530473	3949997	8000897
Mean	1876041	1855243	1585395	3258014	3517185	1830755	1956295	1830649	3608593	2683725	2507576	2279899	2106215	4010725	4497538	**8561839**
Median	1876824	1847380	1578621	3218468	3664482	1830980	1981432	1889295	3693526	2706138	2528358	2258676	2078183	4010689	4413129	**8571208**
STD	46761	57806	91733	428449	322679	21004	172404	81722	312569	145881	115616	106092	146586	320417	481974	229971
Sydney wave scenario (16-buoy)																
	DE	**CMA-ES**	**NM**	**GWO**	**AGWO**	**PSO**	**CMAES-NM**	**1+1EA-NM**	**AGWO-NM**	**CCOS**	**SLPSO$_{II}$**	**SaNSDE$_{II}$**	**LS-NM**	**SLS-NM**	**SLS-NM-B**	**HCCA**
Max	1288869	1290891	1285570	2160341	2357618	1345119	1359025	1334937	2427610	1808910	1617776	1573388	1476492	2266424	2441157	**4170868**
Min	1218526	1222920	1210160	1643183	1942160	1290986	1206080	1175279	1938793	1578082	1500364	1436004	1228500	1779821	1983922	4011595
Mean	1250086	1261968	124989	1961163	2116037	1320898	1259204	1254215	2101718	1719665	1552703	1499720	1374478	2089419	2345455	**4075718**
Median	1249281	1261820	1249830	1965235	2141568	1326489	1234540	1244844	2116880	1743946	1538400	1487439	1389960	2136654	2381567	**4079520**
STD	18793	18397	22330	149244	110808	14393	53006	52187	150286	78724	42453	45235	69691	158182	118228	61356
Adelaide wave scenario (16-buoy)																
	DE	**CMA-ES**	**NM**	**GWO**	**AGWO**	**PSO**	**CMAES-NM**	**1+1EA-NM**	**AGWO-NM**	**CCOS**	**SLPSO$_{II}$**	**SaNSDE$_{II}$**	**LS-NM**	**SLS-NM**	**SLS-NM-B**	**HCCA**
Max	2073818	2066628	1841800	4055518	4198980	2013721	2182902	1996107	4331090	2890630	2724026	2600827	2353122	5389570	6305950	**9470521**
Min	1847975	1925787	1442820	2615454	2828922	1852867	1918926	1782396	3129079	2027935	2324849	2177553	1998174	3886630	3731321	8482713
Mean	1942373	1980279	1629929	3386471	3475527	1921268	2076939	1884884	3588090	2629506	2521642	2417450	2203759	4424488	4400128	**9001224**
Median	1947228	1976812	1637145	3397133	3385324	1915633	2082066	1863392	3521503	2725506	2512711	2476026	2225386	4271271	4233962	**9124786**
STD	57795	40254	146374	425179	393443	42856	96119	71047	379587	282542	158706	130958	102117	501941	696882	343378
Tasmania wave scenario (16-buoy)																
	DE	**CMA-ES**	**NM**	**GWO**	**AGWO**	**PSO**	**CMAES-NM**	**1+1EA-NM**	**AGWO-NM**	**CCOS**	**SLPSO$_{II}$**	**SaNSDE$_{II}$**	**LS-NM**	**SLS-NM**	**SLS-NM-B**	**HCCA**
Max	4102931	4010201	3636190	6752582	7117766	4202333	4555196	4130826	8143490	5852075	5351325	4874586	4807895	8891329	11771018	**16357582**
Min	3815606	3505586	3165160	5297795	5982637	4031614	3966642	3713873	7093660	5054946	4615768	4331868	3842791	5830263	7054758	15503720
Mean	3889232	3870418	3388250	6344121	6364880	4105837	4237851	3915490	7456337	5477239	4970179	4552102	4332727	7549235	8606078	**15952780**
Median	3860710	3904263	3393264	6411575	6249592	4097199	4230567	3896565	7374770	5499668	4978008	4532614	4295434	7733837	8458468	**16070584**
STD	76482	136384	146906	414505	362093	48965	178946	141545	287090	232319	199092	173880	284993	964014	1279960	341515

TABLE 3.9: Performance comparison of various heuristics for the 16-buoy case, based on maximum, median and mean power output layout of the best solution per experiment.

Perth wave scenario (4-buoy)

	DE	CMA-ES	NM	GWO	AGWO	PSO	CMAES-NM	1+1EA-NM	AGWO-NM	CCOS	SLPSO$_{II}$	SaNSDE$_{II}$	LS-NM	SLS-NM	SLS-NM-B	HCCA
Max	757152	747703	937666	1203583	1301909	1096161	1181345	1547929	1490604	**2589217**	2585665	2496828	1056778	1955382	2081568	2571781
Min	679514	684384	522987	862094	980777	898365	664390	1213837	952285	2544542	2505748	2340314	748581	1498860	1513747	2369251
Mean	728521	710025	659399	1055699	1165472	993932	1021743	1376987	1207704	**2568664**	2556791	2410511	954236	1701684	1780217	2462307
Median	729269	709265	612187	1089776	1166276	986457	1092366	1369274	1196508	**2572645**	2563001	2408779	970925	1677414	1779828	2460435
STD	23057	18345	123361	108562	97251	66499	186353	81722	166147	15795	25069	45826	86612	116545	164896	62600

Sydney wave scenario (4-buoy)

	DE	CMA-ES	NM	GWO	AGWO	PSO	CMAES-NM	1+1EA-NM	AGWO-NM	CCOS	SLPSO$_{II}$	SaNSDE$_{II}$	LS-NM	SLS-NM	SLS-NM-B	HCCA
Max	473972	434564	462284	760479	784675	696357	802578	928381	787006	1268449	**1272316**	1252661	672968	995424	1017464	1262763
Min	434121	406658	352144	616066	645036	542108	692273	768977	620102	1251253	1246321	1182527	493190	785335	847721	1198794
Mean	450864	419080	404325	686588	689951	612739	730352	835518	681253	**1260117**	1259924	1205699	595972	910660	933515	1228633
Median	453002	416255	409289	683622	681652	601032	726847	819626	679740	**1261359**	1260302	1200709	604652	933633	911955	1225774
STD	12739	8043	34621	44746	41462	45143	29601	49786	44509	5192	7322	18938	61123	62389	58217	27489

Adelaide wave scenario (4-buoy)

	DE	CMA-ES	NM	GWO	AGWO	PSO	CMAES-NM	1+1EA-NM	AGWO-NM	CCOS	SLPSO$_{II}$	SaNSDE$_{II}$	LS-NM	SLS-NM	SLS-NM-B	HCCA
Max	828677	764148	722935	1367177	1383420	1217127	1147835	1497037	1439554	2720283	**2732298**	2679951	1161869	2036738	2205885	2711811
Min	706780	730629	520054	984928	1150197	986746	1058355	1215677	978806	2652191	2657184	2514665	903350	1530568	1576432	2303003
Mean	784906	749347	629736	1119675	1218747	1088023	1147835	1403663	1211560	2692888	**2699379**	2603358	1001791	1724642	1818722	2541981
Median	786566	751533	635767	1103790	1205228	1098536	1151581	1436140	1193950	**2700573**	2699965	2617516	970319	1714104	1844716	2574643
STD	33305	11744	78249	123038	73511	72345	55729	80538	157434	21741	23531	56901	82646	156221	178678	124799

Tasmania wave scenario (4-buoy)

	DE	CMA-ES	NM	GWO	AGWO	PSO	CMAES-NM	1+1EA-NM	AGWO-NM	CCOS	SLPSO$_{II}$	SaNSDE$_{II}$	LS-NM	SLS-NM	SLS-NM-B	HCCA
Max	1520332	1514709	1470010	2274781	2384846	2069514	2369299	3221716	2533510	3775336	**4808902**	4596171	2454579	3842230	3991016	4691217
Min	1411485	1383235	1125670	1963594	2044584	1564277	2122243	2612539	2098842	3312687	4743330	4511478	1875692	3028176	3104568	4314911
Mean	1472476	1426742	1272643	2106575	2173009	1834935	2369299	2863883	2310503	3775336	**4786660**	4557535	2094040	3427671	3494835	4512121
Median	1473481	1430254	1236877	2098971	2150044	1838107	2398747	2864481	2362061	3836696	**4792621**	4563350	2040533	3357045	3450918	4507503
STD	46831	37946	104560	111927	89905	162102	100318	185941	147406	351782	20583	33529	185895	271382	299053	95361

TABLE 3.10: Performance comparison of various heuristics for the 4-buoy case, based on maximum, median and mean power output layout of the best solution per experiment.

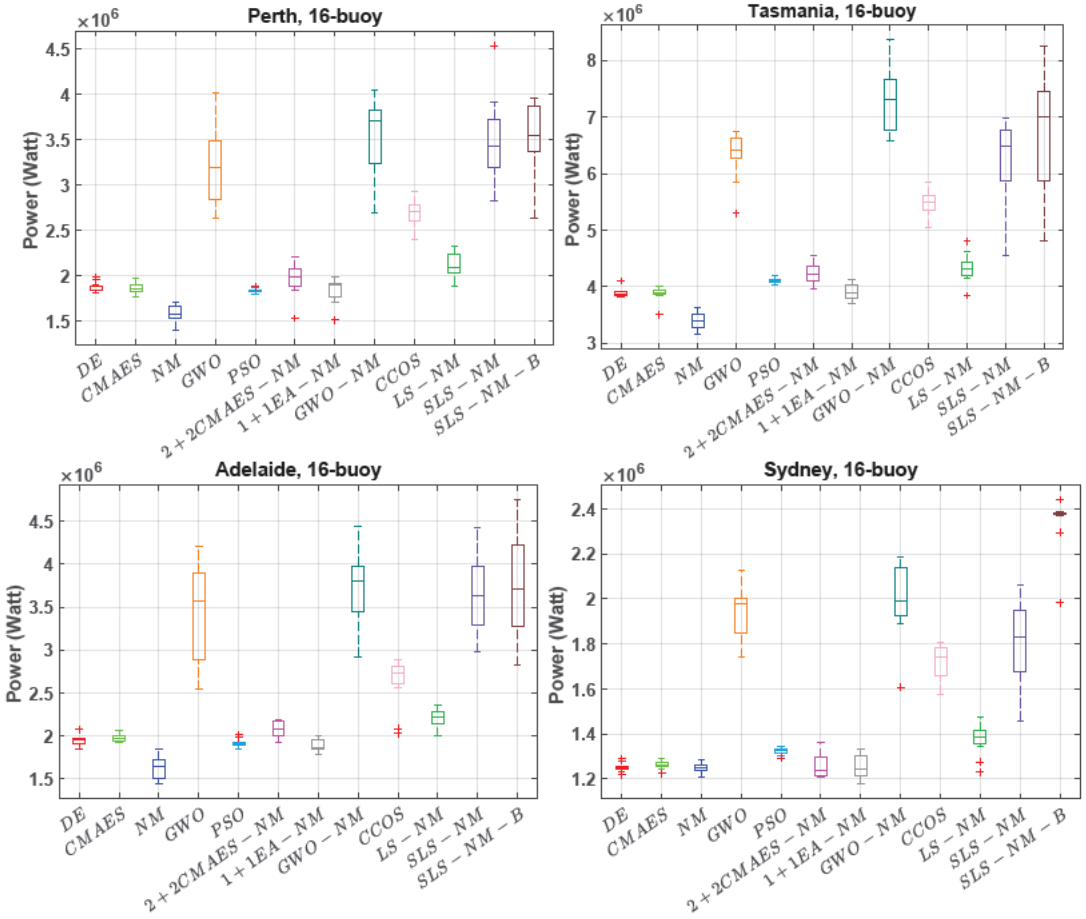

FIGURE 3.22: The comparison of the proposed algorithms' performance for 16 and 4-buoy layouts in four real wave model. The optimisation results present the best solution per experiment. (10 independent runs per each method)

3.2.11 Conclusions

The optimisation of a combination of positions and power take-off parameters of large wave farms is a computationally expensive, multi-modal, large-scale and complicated problem. These challenges are the foremost motivation for discovering faster and smarter optimisation techniques. In this article, we propose a new hybrid cooperative co-evolution method (HCCA) which is composed of a fast strategy for optimising the WEC positions and an effective cooperative strategy (three optimisers) for tuning the PTOs configurations in four real wave scenarios. Moreover, we propose a new adaptive mechanism for improving GWO and the idea is evaluated using ten variants of chaotic maps. To systematically compare the performance of the new search frameworks, we discuss and apply 15 state-of-the-art evolutionary, swarm, alternating (cooperative) and hybrid optimisation algorithms. According to the experimental results, HCCA is able to outperform other heuristic search methods in terms of convergence speed (5 times faster than the best previous algorithm) and the quality of layouts (80% improvement of the sustained energy output in 16-buoy experiments).

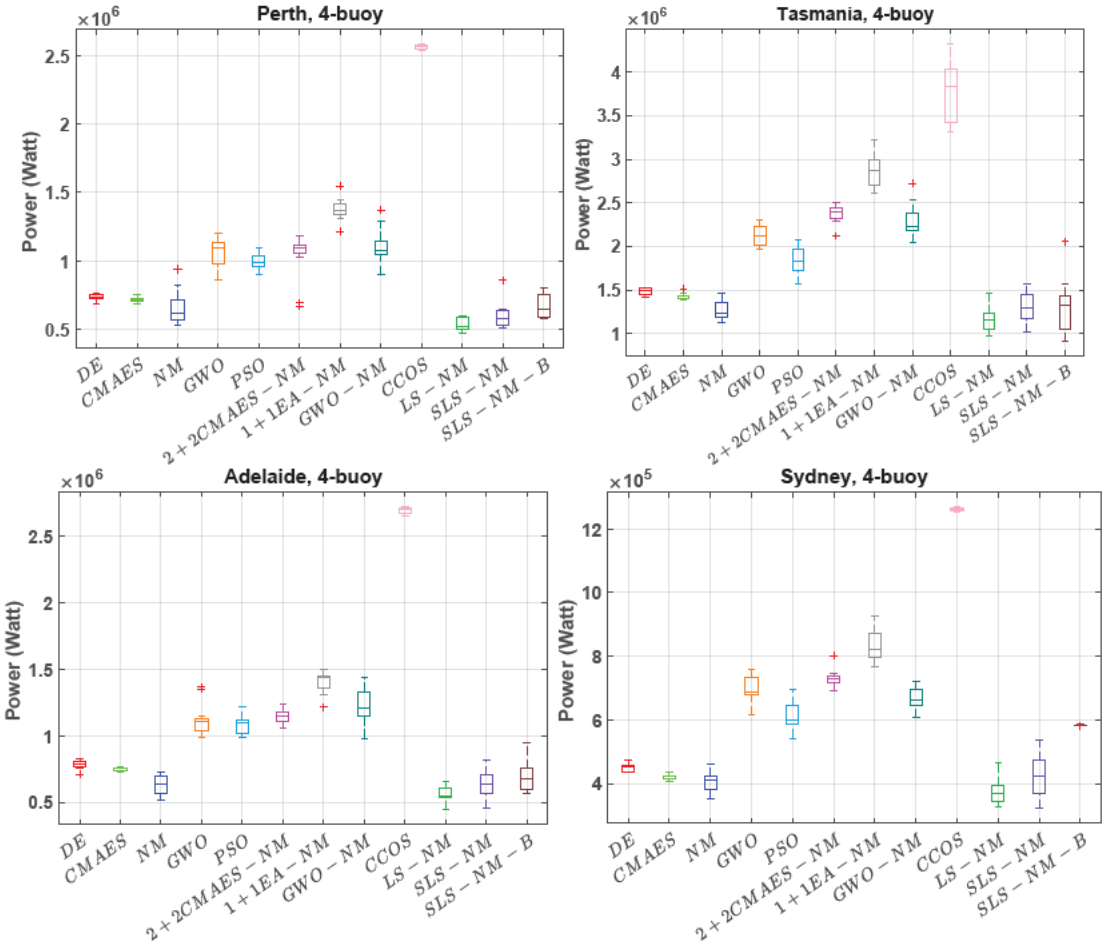

FIGURE 3.23: The comparison of the proposed algorithms' performance for 4-buoy layouts in four real wave models. The optimisation results present the best solution per experiment. (10 independent runs per each method)

The current study, whilst carefully designed and constructed, has several potential limitations. First, while the wave models evaluated are very high fidelity in comparison to related design studies, they still deal with a discrete (but large) range of wave frequencies. It is likely that an even more fine-grained model of the wave environment would improve the accuracy of the model further. The model also assumes a uniform-depth sea floor and tether lengths. While this assumption is a reasonable approximation for some sites, the generality of the model could be improved by allowing for non-uniform depth. Our model also assumes a simplified (spherical) geometry for each buoy. This eliminates the need to model rotational effects. We plan to improve our models to incorporate these effects and allow buoy geometries to be explored. Finally, the three-day running time for the search imposes a realistic but strict limit on how much layouts can be refined. It is expected that larger runtime budgets and faster model evaluations would open up scope for new algorithmic approaches.

In the future, we would also like to combine HCCA with a deep neuro-surrogate model which is trained using a minimum number of samples for speeding up the evaluation

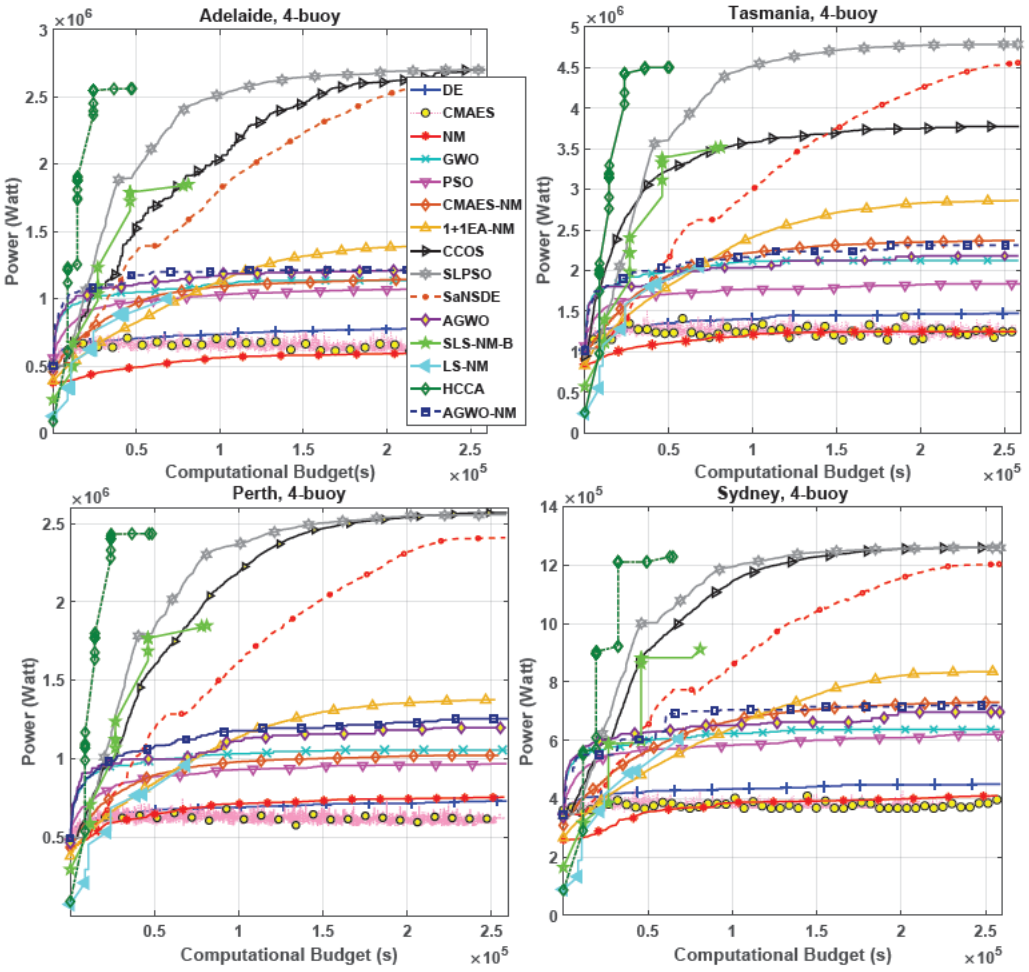

FIGURE 3.24: Comparison of algorithms' effectiveness and convergence rate for 4-buoy layouts in four real wave scenarios.

time of the large wave farm. The neuro-surrogate will aim to quickly estimate the total power output based on the wave scenario's characteristics.

Acknowledgements

We would like to offer our special thanks to Dr. Sergiienko, Dr. Sun, Dr. Hansen, Dr. Kalami and Dr. Mirjalili for sharing and publishing their valuable source codes. Additionally, this research is supported by the supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

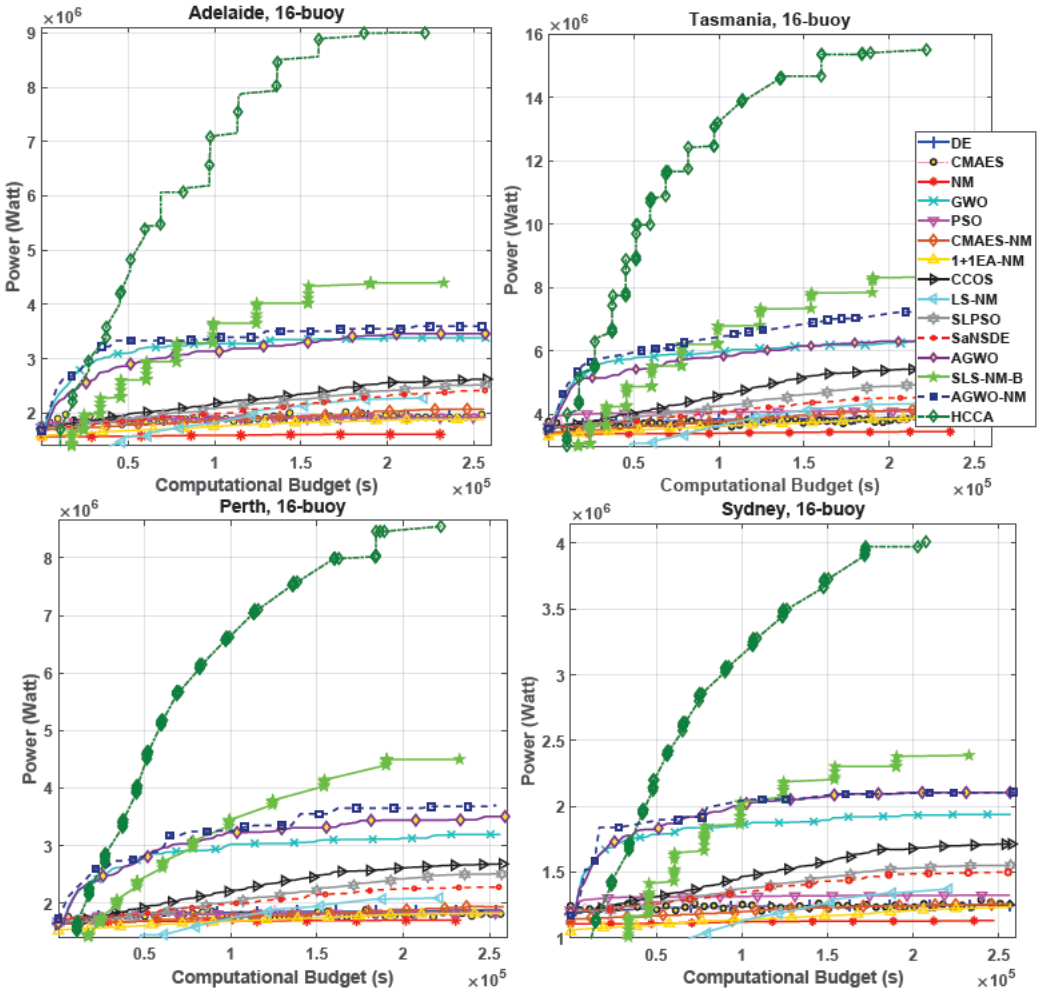

FIGURE 3.25: Comparison of algorithms' performance and convergence rate for 16-buoy layouts in four real wave scenarios.

Chapter 4

Geometric Optimisation of Wave Energy Converters (WECs)

4.1 Design optimisation of a multi-mode wave energy converter

4.1.1 Synopsis

The hydrodynamic characteristics of the Wave Energy Converters and their potential absorbed power are strongly associated with the geometric parameters like shape and dimensions. With this in mind, we use this chapter to propose a hybrid evolutionary algorithm. This algorithm combines self-adaptive differential evolution and the Nelder-Mead (NM) simplex method in order to maximise the total power output of a cylinder-shaped converter and minimise the levelised cost of energy. We assess the performance of the hybrid EA and compare it to that of six more popular optimisation methods. Results of our experiments indicate that this approach converges faster than other algorithms to a near-optimum design.

Refrence

[7] Sergiienko, N. Y., Neshat, M., da Silva, L. S., Alexander, B., & Wagner, M. (2020). Design optimisation of a multi-mode wave energy converter. arXiv preprint arXiv:2001.08966. Accepted for publication by the 39th International Conference on Ocean, Offshore and Arctic Engineering ASME 2020 on [3 March 2020].

LIBRARY NOTE:

The following article (pages 176-191) has been removed due to copyright.

Sergiienko, NY, Neshat, M, da Silva, LSP, Alexander, B, & Wagner, M. "Design Optimisation of a Multi-Mode Wave Energy Converter." Proceedings of the ASME 2020 39th International Conference on Ocean, Offshore and Arctic Engineering. Volume 9: Ocean Renewable Energy. Virtual, Online. August 3–7, 2020.

It is available online to authorised users at:

https://doi.org/10.1115/OMAE2020-19266

Statement of Authorship

Title of Paper	Design optimisation of a multi-mode wave energy converter
Publication Status	☐ Published ☑ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Sergiienko, N. Y., Neshat, M., da Silva, L. S., Alexander, B., & Wagner, M. (2020). Design optimisation of a multi-mode wave energy converter. arXiv preprint arXiv:2001.08966. Accepted for publication by the 39th International Conference on Ocean, Offshore and Arctic Engineering ASME 2020 on [3 March 2020].

Principal Author

Name of Principal Author (Candidate)	Nataliia Y.Sergiienko		
Contribution to the Paper	Initiated the work, developed the ideas and concepts, developed an objective function for optimisation, interpreted results, wrote the manuscript, and acted as a corresponding author.		
Overall percentage (%)	40%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	13/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contr bution.

Name of Co-Author	Mehdi Neshat (30%)		
Contribution to the Paper	Came up with the optimisation ideas, read the existing articles, implemented the optimisation ideas to confirm its efficiency,		
Signature		Date	13/05/2020

Name of Co-Author	Leandro SP da Silva (10%)		
Contribution to the Paper	Contributed to the formulation of the model, helped with some ideas.		
Signature		Date	12/05/2020

Name of Co-Author	Bradley Alexander (5%)		
Contribution to the Paper	Participated in study design, advised on development, revised paper.		
Signature		Date	12/05/20

Name of Co-Author	Markus Wagner (15%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

Part II

Wind Farm Power Forecasting

Chapter 5

An Evolutionary Deep Learning Method for Short-term Wind Speed Prediction: A Case Study of the Lillgrund Offshore Wind Farm

5.1 Synopsis

The article in the chapter demonstrates a hybrid neuro-evolutionary algorithm for short-term wind speed forecasting. This algorithm uses a long short-term memory (LSTM) neural network model plus a popular evolutionary search algorithm, (covariance matrix adaptation evolution strategy (CMA-ES)), to tune the hyper-parameters. The hybrid method is trained on data gathered from an offshore wind turbine in the Baltic Sea. We consider two forecasting horizons (ten-minutes and one-hour ahead), in the experiments. The experimental results show that the proposed method is superior to five other machine learning models, as measured by five performance criteria.

Reference

[9] Neshat, M., Nezhad, M. M., Abbasnejad, E., Tjernberg, L. B., Garcia, D. A., Alexander, B., & Wagner, M. (2020). An Evolutionary Deep Learning Method for Short-term Wind Speed Prediction: A Case Study of the Lillgrund Offshore Wind Farm. arXiv preprint arXiv:2002.09106. Accepted for publication by The 1st Asia Pacific Conference on Sustainable Development of Energy, Water and Environment Systems – SDEWES Conference.

Statement of Authorship

Title of Paper	An Evolutionary Deep Learning Method for Short-term Wind Speed Prediction: A Case Study of the Lillgrund Offshore Wind Farm
Publication Status	☐ Published ☑ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Nezhad, M. M., Abbasnejad, E., Tjernberg, L. B., Garcia, D. A., Alexander, B., & Wagner, M. (2020). An Evolutionary Deep Learning Method for Short-term Wind Speed Prediction: A Case Study of the Lillgrund Offshore Wind Farm. arXiv preprint arXiv:2002.09106. Accepted for publication by The 1st Asia Pacific Conference on Sustainable Development of Energy, Water and Environment Systems – SDEWES Conference.

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	75%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	15/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Meysam Majidi Nezhad (5%)		
Contribution to the Paper	Wrote the literature review section, data analysis and visualization.		
Signature		Date	18/05/2020

Name of Co-Author	Ehsan Abbasnejad (5%)		
Contribution to the Paper	Provided advice on the machine learning methodology and recommendations on the implementations		
Signature		Date	16/05/2020

Name of Co-Author	Lina Bertling Tjernberg (2.5%)		
Contribution to the Paper	Supervised development of the work.		
Signature		Date	18/05/2020

Name of Co-Author	Davide Astiaso Garcia (2.5%)		
Contribution to the Paper	Supervised development of the work.		
Signature		Date	18/05/2020

Name of Co-Author	Bradley Alexander (5%)		
Contribution to the Paper	Supervised development of the work. Advised on data analysis and approaches to outlier removal. Helped revise paper.		
Signature		Date	10/05/2020

Name of Co-Author	Markus Wagner (5%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

5.2 Abstract

Accurate short-term wind speed forecasting is essential for large-scale integration of wind power generation. However, the seasonal and stochastic characteristics of wind speed make forecasting a challenging task. This study adopts a new hybrid evolutionary approach that utilises a popular evolutionary search algorithm, CMA-ES, to tune the hyper-parameters of two long short-term memory (LSTM) neural network models for wind prediction.

The proposed hybrid approach is trained on data gathered from an offshore wind turbine installed in a Swedish wind farm located in the Baltic Sea. Two forecasting horizons, including ten-minutes-ahead (absolute short term) and one-hour ahead (short term), are considered in our experiments. Our experimental results indicate that the new approach is superior to five other applied machine learning models, i.e., polynomial neural network (PNN), feed-forward neural network (FNN), nonlinear autoregressive neural network (NAR) and adaptive neuro-fuzzy inference system (ANFIS), as measured by five performance criteria.

5.3 Introduction

Concerning for global warming and environmental pollution trajectories have motivated intensive efforts to replace fossil fuels [201]. One of the most important clean energy sources is wind. Wind energy has key advantages over technological maturity, cost and life-cycle greenhouse gas emissions [202]. However, wind is a variable resource, so accurate wind power forecasts are crucial in reducing the incidence of costly curtailments, and in protecting system integrity and worker safety [203]. Obtaining an accurate local wind speed prediction can be difficult. This is because nature of wind speed is stochastic, intermittent and non-stationary, all of which characteristics can defeat simple models [48].

In this paper, we propose a hybrid evolutionary deep forecasting model, combining a recurrent deep learning model (LSTM network) [204], coupled with the CMA-ES algorithm [205], (CMAES-LSTM) for predicting short-term wind speed with high accuracy.

As there is no straightforward theory governing the design of an LSTM network for a given problem [47], we tune model structure and hyper-parameters using a combination of grid search and CMA-ES. We demonstrate the performance of the proposed hybrid (CMAES-LSTM) model using a real case study based on data collected from the Lillgrund offshore wind farm to predict wind speeds ten-minutes-ahead and one hour ahead.

The proposed method is compared with the FNN model, ANFIS model, PNN model, NAR model and a static LSTM model. Statistical analyses show that the proposed adaptive method exhibits better performance than these current (static) models.

The remainder this article is structured as follows. The following section briefly surveys related work in the field of predictive wind models. Section 6.7 presents current methodologies, theories and our proposed hybrid evolutionary deep learning (CMAES-LSTM) model. Section 6.6.1 exhibits the performance indices which are applied to evaluate the introduced models. After this, Section 5.8 provides a brief description of the offshore wind farm which is the focus of this paper. Section 5.9 describes and analyses our experimental results. Lastly, we provide a summary and outline future work in Section 5.10.

5.4 Related Work

Today, wind turbine generators (WTGs) are installed in onshore, nearshore and offshore areas worldwide [206–209]. Sweden is one of the leading countries harnessing offshore wind power due to its geographical location, its access to shallow seas and its exposure to the strong North winds on the Baltic Sea. 'Wind energy' refers to variable wind resources that are influenced by several factors including the location of the turbines (on-, near-, offshore); turbine height; seasons, meso-scale and diurnal variations; and climate change. All of these can affect the stable operation of the power grid [210]. Numerous studies have shown how these factors can impact on the reliability of wind energy [211].

Short-term predictive models of wind speed based on historical data have been developed using autoregressive moving average models [212], autoregressive integrated moving average models [213]. Related work by Gani et. al. [214] has combined non-linear models with support vector machines. Wind forecast methodologies using ANNs include Elman neural networks [215], polynomial neural networks (PNN) [216], feed-forward neural networks (FNN) [217] and long short-term memory Network (LSTM)[47], hybrid artificial neural network [218].

Recently deep-learning ANNs solutions for wind forecasting have proven popular. Hu et al. [219] used transfer learning to form short term prediction. Wang et al. [220] introduced a new wind speed forecasting approach using a deep belief network (DBN) based on the deterministic and probabilistic variables. Liu used recurrent deep learning models that were based on LSTM network to forecast wind speed in different time-scales [221, 222]. Chen et al. [223] recommended an ensemble of six different LSTM network configurations for wind speed forecasting, with both ten-min and one-hour interval. More broadly, hybrid nonlinear forecasting models have been explored for the prediction of

wind energy generation, solar energy forecasting and energy market forecasting [224–229].

The work in this paper differs from previous work on wind prediction in that it uses global heuristic search methods to optimise both network structure and tune hyper-parameters. Our approach customises known methodologies from neuro-evolution [230] to improve the performance of predictive wind models.

5.5 Methodology

In this section, we introduce the proposed methodologies and related concepts, including LSTM network, the CMA-ES algorithm and the combination of LSTM network and the CMA-ES algorithm.

5.5.1 Long short-term memory network (LSTM)

An LSTM is a type of recurrent neural network (RNN) which has the capacity to model time series data with different long-term and short-term dependencies [204]. The core of the LSTM network is the memory cell, which refers to the hidden layers in which neurons are traditionally located. LSTM is equipped with three gates (input, output and 'forget' gates), and is therefore able to add or remove information to the cell state. In order to calculate the estimated outputs and update the state of the cell, the following equations can be used:

$$i_t = \sigma(W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{t-1} + b_i) \tag{5.1}$$

$$i_t = \sigma(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f) \tag{5.2}$$

$$c_t = f \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1} + b_c) \tag{5.3}$$

$$o_t = \sigma(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_t + b_o) \tag{5.4}$$

$$m_t = o_t \odot h(c_t) \tag{5.5}$$

$$y_t = W_{ym}m_t + b_y \tag{5.6}$$

where x_t is the input and y_t is the output; i_t, o_t and f_t indicate the input gate, output gate and forget gate respectively. The activation vectors of each cell are shown by c_t, while m_t denotes the activation vectors for any memory block. σ, g and h express the activation function of the gate, input and output (the logistic sigmoid and $tanh$ function are assigned). Lastly, $\odot$ (Hadamard product) indicates the element-wise multiplication between two vectors. Furthermore, b_i, b_f,b_c, b_o are the corresponding bias

vectors. W_{ox},W_{om}, W_{oc}, W_{ix}, W_{im}, W_{ic}, W_{fx}, W_{fm}, W_{fc}, W_{cx}, W_{cm}, and W_{ym} are the corresponding weight coefficients.

5.5.2 Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

The CMA-ES [205] search process for an n-dimensional problem works by adapting an $n \times n$ covariance-matrix C. This matrix defines the shape and orientation of a Gaussian distribution in the search space and a vector x that describes the location of the centre of the distribution. The search is conducted by sampling this distribution for a population of μ individual solutions. These solutions are then evaluated and the relative performance of these solutions is used to update both C and x. This process of sampling and adaptation continues until the search converges or a fixed number of iterations has expired.

CMA-ES relies on three principal operations, which are selection, mutation, and recombination. Recombination and mutation are employed for exploration of the search space and to create genetic variations, whereas the operator of selection is used for exploiting and converging on an optimal solution. The mutation operator plays a significant role in CMA-ES, which utilises a multivariate Gaussian distribution. For a thorough explanation of the different selection operators, we refer the interested reader to [88].

CMA-ES can explore and exploit search spaces due to its self-adaptive mechanism for setting the vector of mutation step sizes (σ) instead of adopting only one global mutation step size. Self-adaptation can also improve convergence speed [205]. The covariance matrix is computed based on the differences in the mean values of two progressive generations. In which case, it expects that the current population includes sufficient information to estimate the correlations. After calculating the covariance matrix, the rotation matrix will derive from the covariance matrix with regard to expanding the distribution of the multivariate Gaussian in the estimated direction of the global optimum. This can be accomplished by conducting an eigen-decomposition of the covariance matrix to receive an orthogonal basis for the matrix [231].

5.6 Adaptive Tuning Process

One of the primary challenges in designing an ANN is setting appropriate values for the hyper-parameters such as the number of the hidden layers, number of neurons in each layer, batch size, learning rate and optimiser type [47]. Tuning the hyper-parameters plays a significant role in improving the performance of the DNN with respect to problem domain. In the domain of wind forecasting, Chen [223] has noted that the forecasting

accuracy of LSTM networks is influenced by structural parameters. There are three main techniques for tuning hyper-parameters. These are 1) manual trial and error, which is costly and cannot be practised adequately, 2) systematic grid search, and 3) meta-heuristic search. In this paper, we compare the performance of both grid search and the meta-heuristic approach (CMAES-LSTM) in tuning LSTM networks for wind forecasting.

In the grid search method, we assign a fixed value for the optimiser type ('adam') [234], the number of LSTM hidden layers and also the number of neurons. These values are listed in Table 6.1. The grid search process determines the batch size and learning rate can fall within ranges of ($10^{-5} \leq LR \leq 10^{-1}$, and $8 \leq BS \leq 1024$). For the search using CMA-ES we apply all of the listed hyper-parameters of the LSTM networks listed in the corresponding section of Table 6.1. In order to avoid a situation in which the search simply on complex network designs that take too long to train, we add a penalty term for model training time to the fitness function f. We frame the optimisation process as follows:

$$\begin{aligned} &Argmin \rightarrow f = fitness(N_{h_1}, N_{h_2}, ..., N_{h_D}, N_{n_1 h_1}, N_{n_2 h_2}, ...N_{n_D h_D}, L_R, B_S, Op), \\ &Subject - to: \\ &\qquad LN_h \leq N_h \leq UN_h, \\ &\qquad LN_n \leq N_n \leq UN_n, \\ &\qquad 10^{-5} \leq L_R \leq 10^{-1}, \\ &\qquad\quad 8 \leq B_S \leq 1024. \end{aligned} \tag{5.7}$$

where $N_{h_i}, \{i = 1, \ldots, D\}$ is the number of hidden layers for the $i{-}^{th}$ LSTM network and $N_{n_i,h_j}, \{j = 1, \ldots, D_l\}$ is the number of neurons in the i^{th} hidden layer of this network. The lower and upper bounds of N_h are shown by LN_h and UN_h , while LN_n and UN_n are the lower and upper bounds of neuron number. The final fitness function is as follows:

$$\begin{aligned} f &= f_1 + \omega f_2 \\ f_1 &= RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - f_o(i))^2} \end{aligned} \tag{5.8}$$

$$f_2 = \begin{cases} Tr_{runtime} - \rho, & if(Tr_{runtime} > \rho) \\ 0, & otherwise \end{cases} \tag{5.9}$$

where RMSE is the root mean square error of the test samples; ρ is the threshold of training runtime by 600(s). ω is the weight coefficient which is used to penalise long training times. In this work we set ω to 10^{-3}, because the range of RMSE is between 0.5

TABLE 5.1: Summary of the predictive models tested in this paper.

Models	**Descriptions**
LSTM [47] + grid search	Long Short-term memory Network: • **LSTM hyper-parameters** – *miniBatchSize*=512 – *LearningRate*= 10^{-3} – *numHiddenUnits1* = 125; – *numHiddenUnits2* = 100; – *Epochs* = 100 – *Optimiser*= 'adam'
ANFIS [232]	Adaptive neuro-fuzzy inference system: • OptMethod= Backpropagation • **Training settings** – *Epochs*=100; – *ErrorGoal*=0; – *InitialStepSize*=0.01; – *StepSizeDecrease*=0.9; – *StepSizeIncrease*=1.1; • **FIS features** – *mf* number=5; – *mf* type='gaussmf';
PNN [216]	Polynomial neural network: • **PNN parameters** – *MaxNeurons*=20 – *MaxLayers*= 5 – *SelectionPressure*= 0.2; – *TrainRatio*= 0.8;
FNN [217]	Feed-forward neural network • **FNN settings** – *hiddenSizes*= 100 – *hiddenLayers*= 2 – *trainFcn*= 'trainlm';
NAR [233]	Nonlinear autoregressive neural network (is similar to FNN settings)
CMAES-LSTM	• **CMAES-LSTM hyper-parameters (Best configuration)** – *miniBatchSize*=655 – *LearningRate*=10^{-3} – *numHiddenUnits1*=177 ; – *numHiddenUnits2*=151 ; – *Epochs* = 100 – *Optimiser*= 'adam' – *PopulationSize*=12 – *MaxEvaluation* =1000

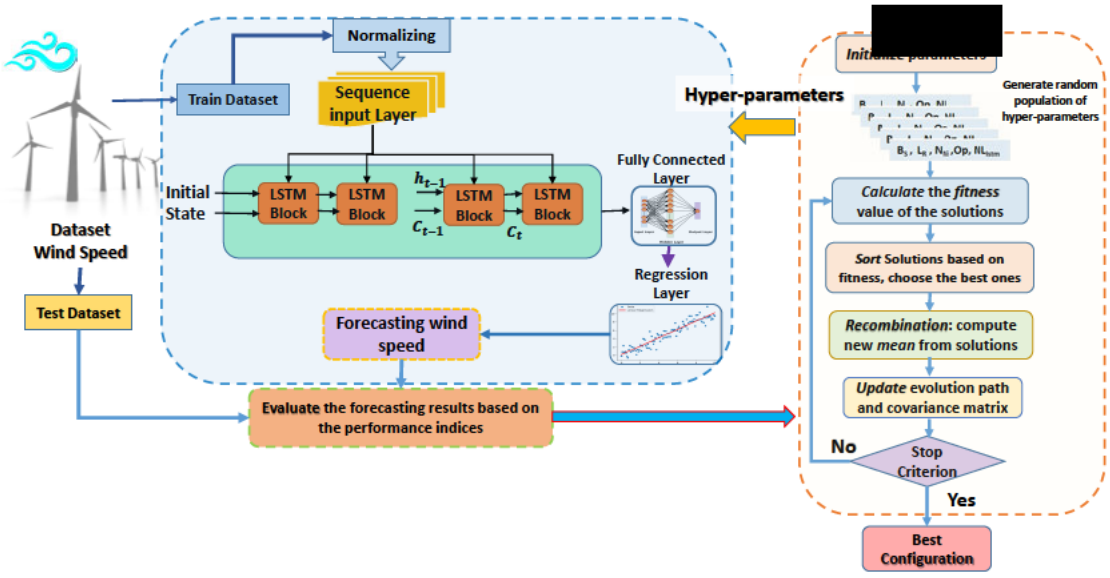

FIGURE 5.1: The forecasting framework of the proposed hybrid CMAES-LSTM model.

and 1.5. The value N is the number of test data. Using the results of recent investigations of the lower and upper bounds the N_h and N_n, which represent the desirable learning performance of LSTM networks [4, 235, 236], we set the upper and lower bounds of N_h and N_n to $\{1, 2\}$ and $\{30, 230\}$ respectively.

The optimisation process is initialised by providing CMA-ES with the search space bounds above and starting parameters of $\sigma = 0.25$ and $\lambda = 12$ (population size). Meanwhile the hyper-parameters of other forecasting models are initialised by the references.

Figure 5.1 shows the overall optimisation process of our hybrid model.

5.7 Performance indices of forecasting models

We use five broadly considered indicators to assess the forecasting performance: the mean square error (MSE), the root mean square error (RMSE), mean absolute error (MAE), the mean absolute percentage error (MAPE) and the Pearson correlation coefficient (R) [237]. The equations of MAE, RMSE, MAPE and R are represented as follows :

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|f_p(i) - f_o(i)| \tag{5.10}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - f_o(i))^2} \tag{5.11}$$

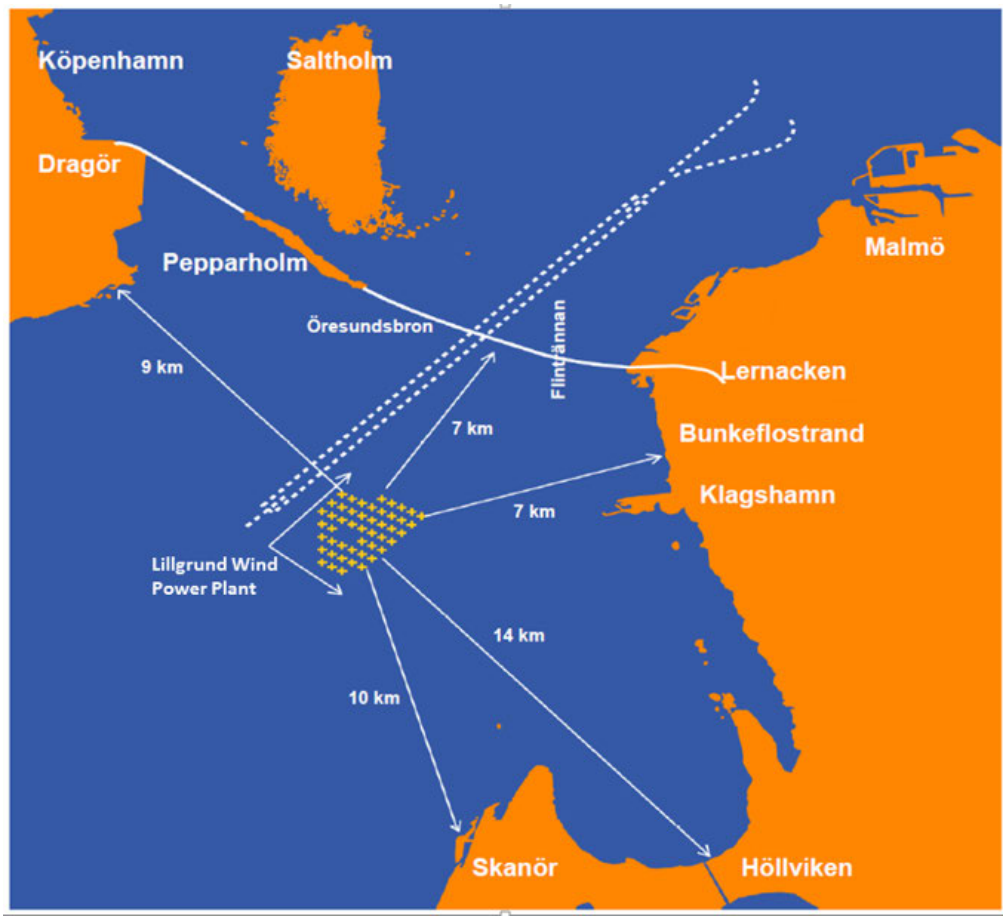

FIGURE 5.2: Location of the Lillgrund offshore wind power plant [14].

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{(f_p(i) - f_o(i))}{f_o(i)} \times 100\% \tag{5.12}$$

$$R = \frac{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - \overline{f}_p)(f_o(i) - \overline{f}_o)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - \overline{f}_p)^2} \times \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_o(i) - \overline{f}_o)^2}} \tag{5.13}$$

where $f_p(i)$ and $f_o(i)$ signify the predicted and observed wind speed values at the i^{th} data point. The total number of observed data points in N. In addition, $\overline{f}_p$ and $\overline{f}_o$ are the average of the projected and observed consequences, respectively. In order to improve the performance of the predicted model, MSE, RMSE, MAE and MAPE should be minimised, while R needs to be maximised.

5.8 Case Study

In this paper, we use the original wind speed data gathered from a large offshore wind farm called Lillgrund [14] , which is situated in a shallow area of Oresund, located 7 km off the coast of Sweden and 7 km south from the Oresund Bridge connecting Sweden

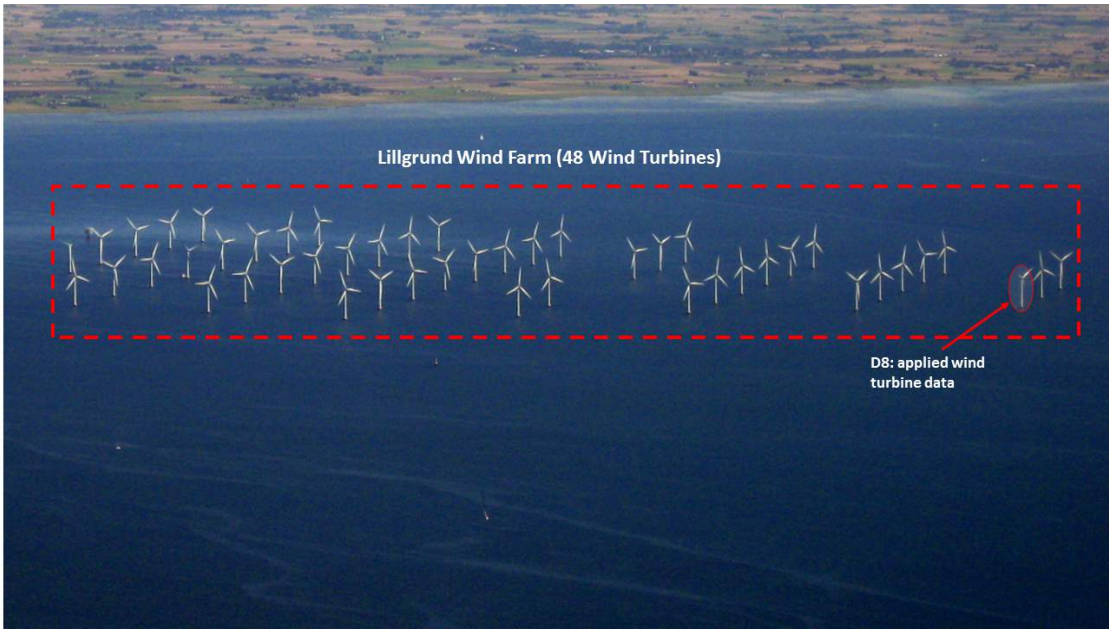

FIGURE 5.3: Lillgrund offshore wind farm in Baltic Sea and the wind turbine position is showed with red cycle which is applied for collecting the real wind speed data [14].

and Denmark (see Figure 5.2). The mean wind speed is around 8.5 m/s at hub height. This wind, together with the low water depth of 4 to 8 (m), makes the installation of wind turbines economically feasible. The Lillgrund offshore wind farm consists of 48 wind turbines, each rated at 2.3 (MW), resulting in a total wind power plant potential of 110 (MW) [238]. A SCADA collects wind power plant information at a 10-minute interval [239]. The wind power system also includes an offshore substation, an onshore substation and a 130 (kV) sea and land cable which connects offshore substation to the shore.

The wind speed data collected from the Lillgrund wind farm (D3 wind turbine position can be seen in Figure 5.3) consists of the period from July 2018 to July 2019 at a ten-minute resolution. Figures 5.4 and 5.5 show the distribution and frequency of the recorded wind speed in Lillgrund Wind farm during these 12 months. The distribution and frequency of the wind speed is strongly anisotropic [240].

Figure 5.6 shows that the predominant wind direction is south-west, and a secondary prevailing direction is south-east. However, there are also occasional North-west winds and sporadic north-east storms. We use two horizons to predict wind speeds: ten minutes and one hour. The wind speed data are randomly divided into three sets using blocks of indices. 80% of the data is used as the training set, and the other 20% is allocated to the test (10%) and validation (10%) sets. We also apply k-fold cross-validation in order to train the LSTM network to predict the time series data.

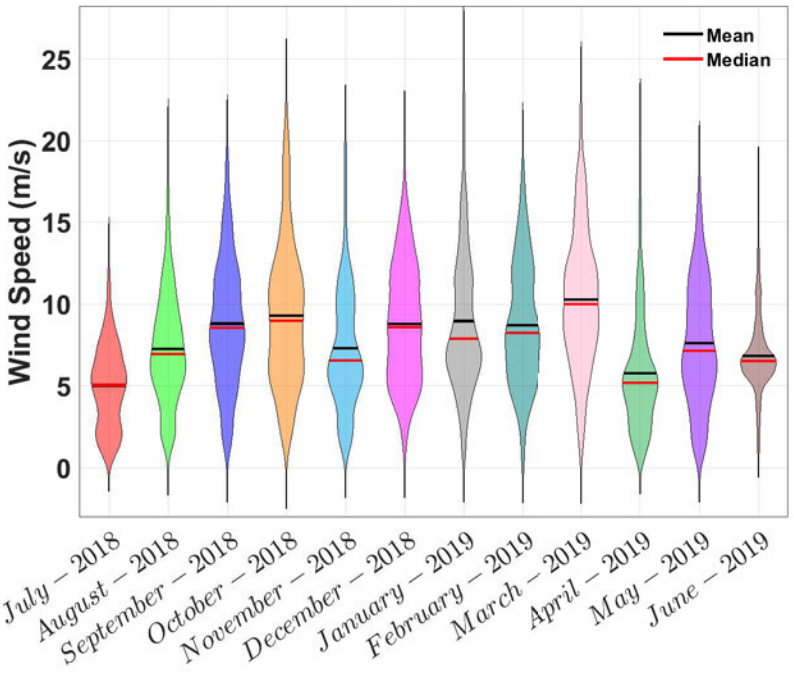

FIGURE 5.4: The distribution and frequency of the wind speed data in Lillgrund Wind coastal site per 12 months.

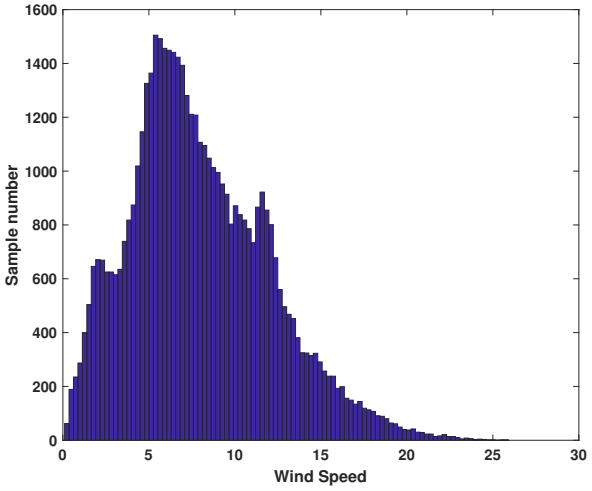

FIGURE 5.5: Total distribution and frequency of the wind speed in Lillgrund Wind coastal site .

5.9 Experiments and analysis

To assess the performance of the proposed CMAES-LSTM hybrid model, we compare its performance with that of four well-known conventional forecasting techniques: FNN, ANFIS, PNN, NAR and one DNN forecasting model (the grid-search tuned LSTM network).

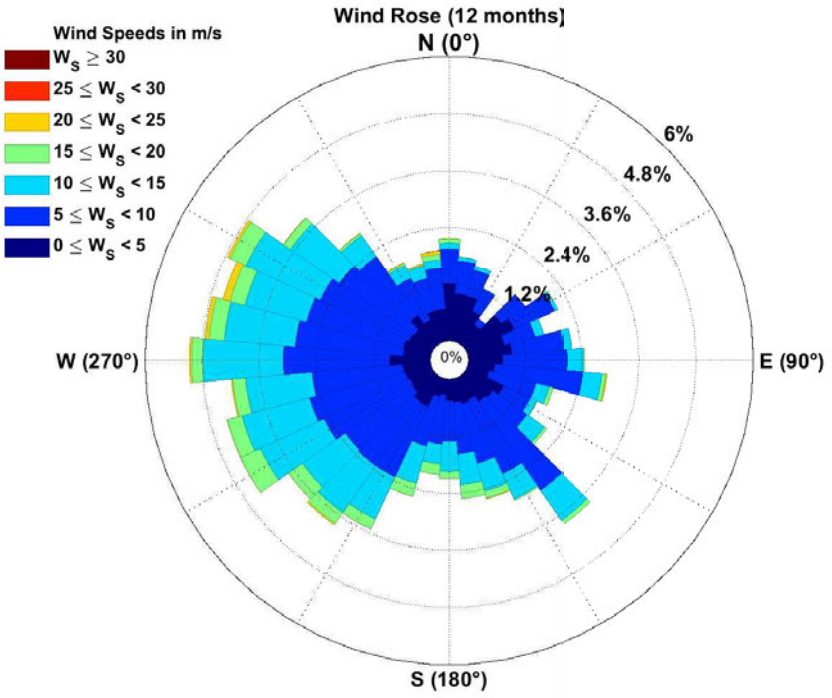

FIGURE 5.6: Wind rose: the speed and directional distribution of wind for the Lillgrund Wind coastal site. The dataset for generating this graph was obtained from [14]

In the first step of this study, a grid search algorithm is used to explore the search space of the hyper-parameters' impact on LSTM Network performance. Conventionally, hyper-parameters tuning is performed by hand and requires skilled practitioners [241]. Here the grid search is limited to tuning learning rate and batch size. Other parameters are fixed, chso as to allow the search be completed in reasonable time.

Table 6.1 shows the final hyper-parameters of the models. We repeat each experiment ten times in order to allow for a reasonable sampling of each method's performance.

Figure 5.7 demonstrates the forecasting results of tuning both the batch size and learning rate in LSTM model performance for the two time-interval prediction datasets. According to the observations, the minimum learning error occurs where the learning rate is between 10^{-4} and 10^{-2}. The optimal size of batches is highly dependent on the selected learning rate values.

Figure 5.8 shows the correlation between the original wind speed data with predicted data for the ten-minute and one-hour forecast period.

The average errors of the testing model obtained by the best configuration of the grid-search-tuned LSTM model are shown in Figure 5.8.

One of the best forecasting models is the adaptive neuro-fuzzy inference system (ANFIS) [232]. In order to model the wind speed using proper membership functions, five Gaussian membership functions are defined to cover whole range of the wind speed dataset (Figure 5.9). The performance of ANFIS is represented in Figure 5.10. We can see that the ANFIS estimation results are competitive. Figure 5.11 shows the results of the four performance indices applied in this work for short-term wind speed forecasting (ten-minute ahead) received by five other models and the proposed hybrid model. Concerning this experiment, the hybrid evolutionary model can outperform its five competitors for short-term wind speed forecasting with the minimum RMSE value of 0.695(m/s), MAE as 0.495(m/s), and MAPE as 8.2% as well as a top R rate of 98.7.

Tables 5.2 and 5.3 summarise the statistical forecasting results for ten-minute and one-hour intervals. For both time intervals, our CMAES-LSTM can accomplish better forecasting outcomes than other applied models in this experiment.

In addition, in Figure 5.12 can be seen that CMAES-LSTM is the most competitive model based on the Friedman statistic test with p-values lower than 0.0001, which signifies that the proposed forecasting method considerably performs better than other models. To evaluate the impact of the penalty factor on the Hybrid forecasting model performance, Figure 5.13 shows the comparison convergence of the hyper-parameters tuning process within (WR) and without (R) applying the penalty factor of the training runtime. Interestingly, both cases are converged to the same learning rate at 10^{-3}. however, the optimisation results are different for the other parameters. It is noted that the whole allocated budget for both cases is the same, but the performance of CMAES-LSTM model with a training runtime penalty is superior to than another strategy.

5.10 Conclusions

Wind speed forecasting plays an essential role in the wind energy industry. In this paper, we introduce a hybrid evolutionary deep learning approach (CMAES-LSTM) to acquire highly accurate, more stationary wind speed forecasting results. For the purpose of tuning the LSTM network hyper-parameters, we propose two different techniques; a grid search and a well-known evolutionary method (CMA-ES). We evaluated the effectiveness of our approach using data from the Lillgrund offshore wind farm and time horizons of ten-minute and one-hour respectively.

Our experiments showed that our approach outperformed others using five performance indices, and that the performance difference is statistically significant.

TABLE 5.2: Performance indices of forecasting outcomes achieved by different models on case ten-minute ahead.

		MSE(m/s)		RMSE(m/s)		MAE(m/s)		MAPE(%)		R	
Model		**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
ANFIS	Mean	6.60E-01	6.64E-01	8.13E-01	8.15E-01	5.69E-01	5.69E-01	9.19E+00	9.17E+00	9.81E-01	9.81E-01
	Min	6.53E-01	6.43E-01	8.08E-01	8.02E-01	5.66E-01	5.61E-01	9.12E+00	8.99E+00	9.80E-01	9.80E-01
	Max	6.71E-01	6.96E-01	8.19E-01	8.34E-01	5.75E-01	5.82E-01	9.27E+00	9.42E+00	9.81E-01	9.81E-01
	Std	4.58E-03	1.55E-02	2.82E-03	9.47E-03	2.49E-03	7.07E-03	4.83E-02	1.39E-01	1.99E-04	3.23E-04
PNN	Mean	1.06E+00	1.06E+00	1.03E+00	1.03E+00	7.25E-01	7.24E-01	1.25E+01	1.25E+01	9.67E-01	9.67E-01
	Min	1.05E+00	1.02E+00	1.03E+00	1.01E+00	7.22E-01	7.19E-01	1.24E+01	1.23E+01	9.66E-01	9.66E-01
	Max	1.08E+00	1.08E+00	1.04E+00	1.04E+00	7.28E-01	7.32E-01	1.26E+01	1.29E+01	9.69E-01	9.70E-01
	Std	7.64E-03	1.78E-02	3.70E-03	8.70E-03	2.04E-03	4.69E-03	7.40E-02	2.22E-01	2.40E-03	3.10E-03
FFNN	Mean	9.73E-01	9.56E-01	9.78E-01	9.68E-01	7.24E-01	7.19E-01	1.22E+01	1.22E+01	9.76E-01	9.77E-01
	Min	6.26E-01	6.28E-01	7.91E-01	7.92E-01	5.61E-01	5.61E-01	9.11E+00	9.22E+00	9.68E-01	9.68E-01
	Max	1.25E+00	1.24E+00	1.12E+00	1.11E+00	8.56E-01	8.53E-01	1.45E+01	1.44E+01	9.81E-01	9.82E-01
	Std	2.77E-01	2.84E-01	1.38E-01	1.42E-01	1.32E-01	1.34E-01	2.30E+00	2.24E+00	5.04E-03	4.84E-03
NAR	Mean	8.69E-01	8.69E-01	9.32E-01	9.32E-01	6.27E-01	6.28E-01	9.57E+00	9.51E+00	9.73E-01	9.73E-01
	Min	8.56E-01	8.47E-01	9.25E-01	9.20E-01	6.24E-01	6.21E-01	9.54E+00	9.30E+00	9.73E-01	9.72E-01
	Max	8.78E-01	8.87E-01	9.37E-01	9.42E-01	6.29E-01	6.32E-01	9.61E+00	9.60E+00	9.74E-01	9.74E-01
	Std	8.83E-03	1.87E-02	4.73E-03	9.98E-03	2.65E-03	4.44E-03	3.66E-02	8.70E-02	3.39E-04	5.43E-04
LSTM	Mean	5.64E-01	5.60E-01	7.51E-01	7.48E-01	5.23E-01	5.24E-01	8.65E+00	8.66E+00	9.83E-01	9.83E-01
	Min	5.55E-01	5.31E-01	7.45E-01	7.29E-01	5.19E-01	5.13E-01	8.50E+00	8.46E+00	9.83E-01	9.82E-01
	Max	5.70E-01	5.76E-01	7.55E-01	7.59E-01	5.26E-01	5.29E-01	8.74E+00	8.77E+00	9.83E-01	9.83E-01
	Std	5.35E-03	1.61E-02	3.56E-03	1.08E-02	2.64E-03	5.60E-03	8.93E-02	1.10E-01	1.42E-04	5.48E-04
CMAES-LSTM	Mean	5.59E-01	5.00E-01	7.61E-01	7.27E-01	5.20E-01	5.07E-01	8.70E+00	8.57E+00	9.83E-01	9.85E-01
	Min	5.46E-01	4.83E-01	7.39E-01	7.19E-01	5.12E-01	4.95E-01	8.58E+00	8.20E+00	9.83E-01	9.84E-01
	Max	5.65E-01	5.22E-01	7.52E-01	7.30E-01	5.25E-01	5.16E-01	8.76E+00	8.71E+00	9.84E-01	9.87E-01
	Std	4.45E-03	1.41E-02	3.56E-03	1.28E-02	3.64E-03	7.60E-03	5.93E-02	1.60E-01	4.42E-04	2.48E-04

TABLE 5.3: Performance indices of forecasting outcomes achieved by different models on the case of one-hour ahead.

		MSE(m/s)		RMSE(m/s)		MAE(m/s)		MAPE(%)		R	
Model		**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
ANFIS	Mean	2.60E+00	2.59E+00	1.61E+00	1.61E+00	1.16E+00	1.16E+00	2.05E+01	2.05E+01	9.19E-01	9.19E-01
	Min	2.58E+00	2.49E+00	1.61E+00	1.58E+00	1.16E+00	1.15E+00	2.04E+01	2.02E+01	9.18E-01	9.16E-01
	Max	2.62E+00	2.66E+00	1.62E+00	1.63E+00	1.17E+00	1.17E+00	2.06E+01	2.08E+01	9.20E-01	9.21E-01
	Std	1.55E-02	6.62E-02	4.80E-03	2.05E-02	2.54E-03	1.09E-02	7.18E-02	2.59E-01	4.99E-04	1.94E-03
PNN	Mean	3.93E+00	3.91E+00	1.98E+00	1.98E+00	1.48E+00	1.48E+00	3.01E+01	3.03E+01	8.73E-01	8.73E-01
	Min	3.90E+00	3.85E+00	1.97E+00	1.96E+00	1.48E+00	1.47E+00	2.98E+01	2.97E+01	8.70E-01	8.72E-01
	Max	3.95E+00	3.96E+00	1.99E+00	1.99E+00	1.49E+00	1.49E+00	3.03E+01	3.06E+01	8.75E-01	8.74E-01
	Std	2.02E-02	4.72E-02	5.10E-03	1.19E-02	3.63E-03	8.71E-03	1.52E-01	3.80E-01	1.90E-03	9.00E-04
FFNN	Mean	3.39E+00	3.41E+00	1.82E+00	1.82E+00	1.36E+00	1.36E+00	2.61E+01	2.62E+01	9.15E-01	8.95E-01
	Min	2.65E+00	2.59E+00	1.63E+00	1.61E+00	1.17E+00	1.14E+00	2.04E+01	1.97E+01	9.05E-01	8.95E-01
	Max	4.65E+00	4.66E+00	2.12E+00	2.12E+00	1.61E+00	1.62E+00	3.20E+01	3.24E+01	9.20E-01	8.99E-01
	Std	1.26E+00	1.26E+00	2.97E-01	2.96E-01	2.56E-01	2.59E-01	5.99E+00	6.22E+00	5.03E-03	4.70E-03
NAR	Mean	3.55E+00	3.57E+00	1.88E+00	1.89E+00	1.44E+00	1.44E+00	3.49E+01	3.59E+01	8.95E-01	8.95E-01
	Min	3.49E+00	3.46E+00	1.87E+00	1.86E+00	1.43E+00	1.41E+00	3.46E+01	3.46E+01	8.95E-01	8.93E-01
	Max	3.59E+00	3.65E+00	1.90E+00	1.91E+00	1.45E+00	1.46E+00	3.52E+01	3.67E+01	8.96E-01	8.97E-01
	Std	3.83E-02	8.08E-02	1.01E-02	2.14E-02	8.42E-03	1.89E-02	2.20E-01	8.66E-01	5.59E-04	1.49E-03
LSTM	Mean	2.50E+00	2.49E+00	1.58E+00	1.58E+00	1.15E+00	1.15E+00	2.27E+01	2.23E+01	9.21E-01	9.22E-01
	Min	2.47E+00	2.43E+00	1.57E+00	1.56E+00	1.15E+00	1.14E+00	2.23E+01	2.15E+01	9.20E-01	9.18E-01
	Max	2.52E+00	2.54E+00	1.59E+00	1.59E+00	1.15E+00	1.16E+00	2.29E+01	2.28E+01	9.22E-01	9.24E-01
	Std	1.80E-02	5.18E-02	5.68E-03	1.64E-02	4.23E-03	1.05E-02	2.50E-01	5.43E-01	5.70E-04	2.29E-03
CMAES-LSTM	Mean	2.46E+00	2.46E+00	1.57E+00	1.57E+00	1.14E+00	1.14E+00	2.15E+01	2.18E+01	9.30E-01	9.28E-01
	Min	2.44E+00	2.40E+00	1.56E+00	1.55E+00	1.14E+00	1.13E+00	2.11E+01	2.13E+01	9.30E-01	9.25E-01
	Max	2.48E+00	2.51E+00	1.57E+00	1.58E+00	1.14E+00	1.15E+00	2.18E+01	2.22E+01	9.31E-01	9.31E-01
	Std	1.59E-02	5.49E-02	5.02E-03	1.74E-02	3.92E-03	9.00E-03	3.65E-01	4.19E-01	6.42E-04	2.63E-03

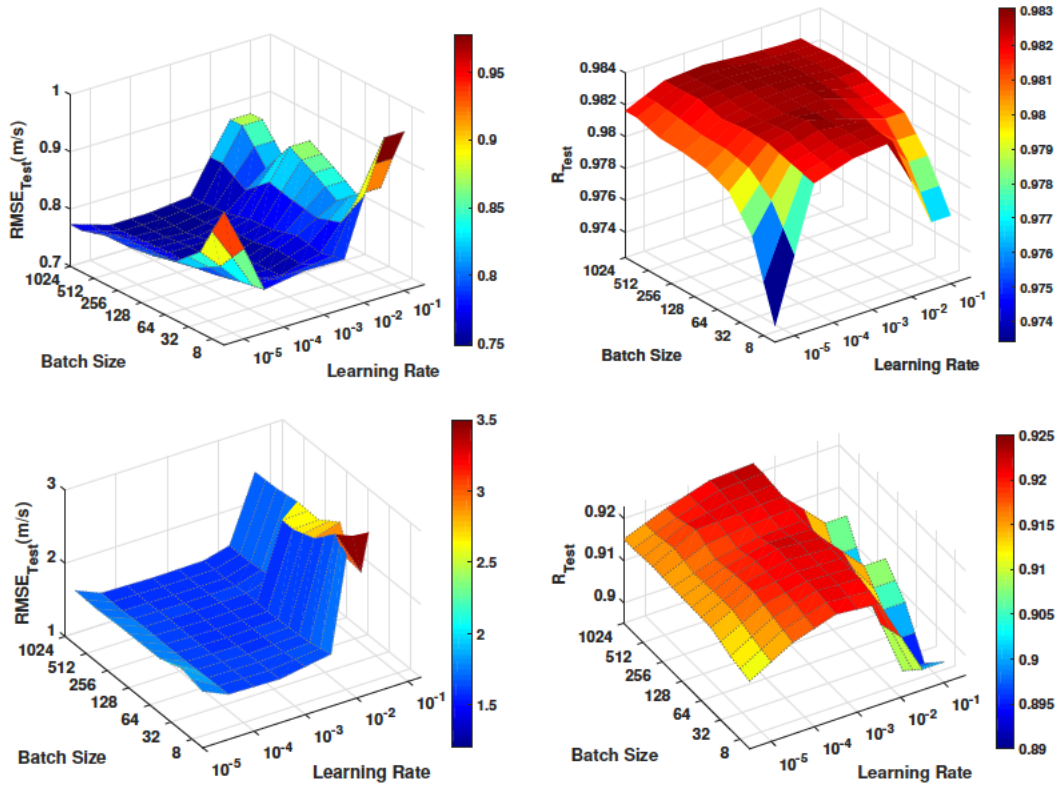

FIGURE 5.7: Hyper-parameters tuning of the applied LSTM network for forecasting the short-term wind speed .(a) the average of RMSE test-set (ten-minute ahead) (b) the average of R-value test-set (ten-minute ahead), (c) the average of RMSE test-set (one-hour ahead) (d) the average of R-value test-set (one-hour ahead)

In the future, we are going to develop the proposed hybrid model by applying a decomposition approach. This will serve to divide the time series wind speed data into sub-groups with more interrelated features. Each sub-group will be absorbed by one independent hybrid method, which will allow independent model to learn the nonlinear model of wind speed. Another future plan can be further investigations to compare the efficiency of different hybrid evolutionary algorithms and deep learning models based on the nonlinear combined mechanism.

Acknowledgements

The authors would like to thank the Vattenfall project the access to the SCADA data of the Lillgrund wind farm which is used in this study. Furthermore, this study is supported with supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

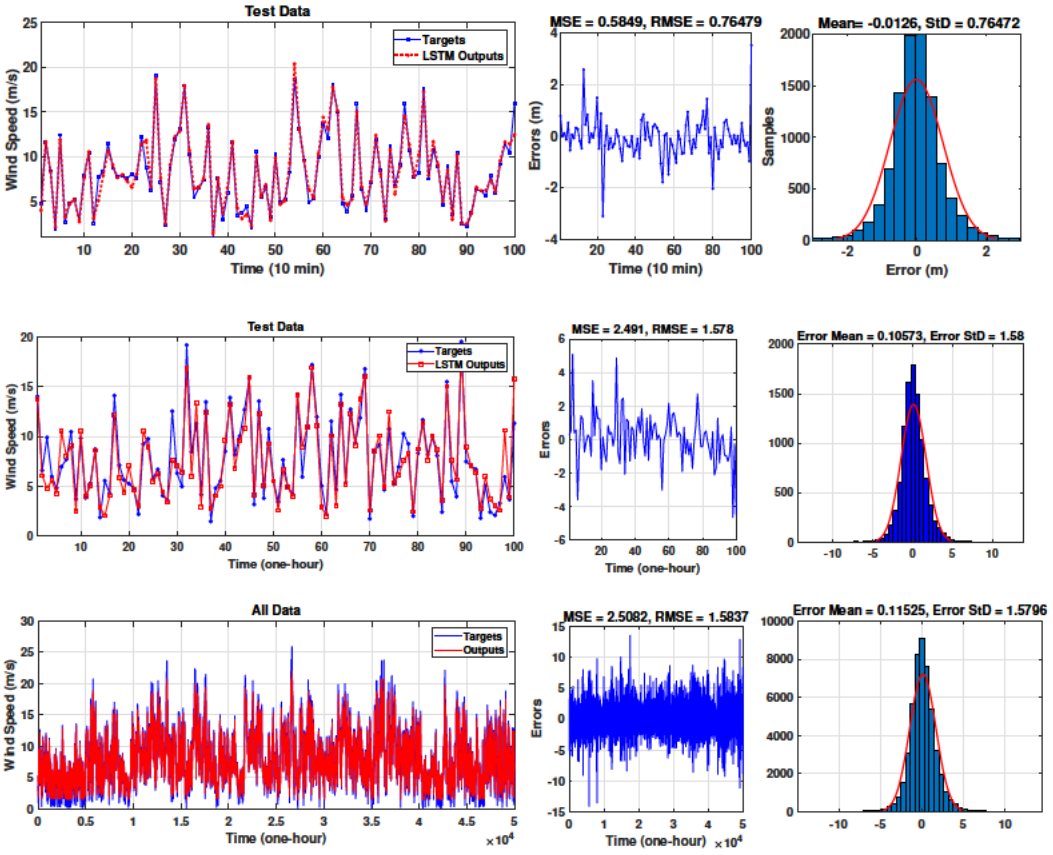

FIGURE 5.8: The wind speed forecasting results achieved by LSTM network with the best tuned hyper-parameters on (a) ten-minute ahead, (b) one-hour ahead (test data-set) and (c) one-hour ahead (all data-set)

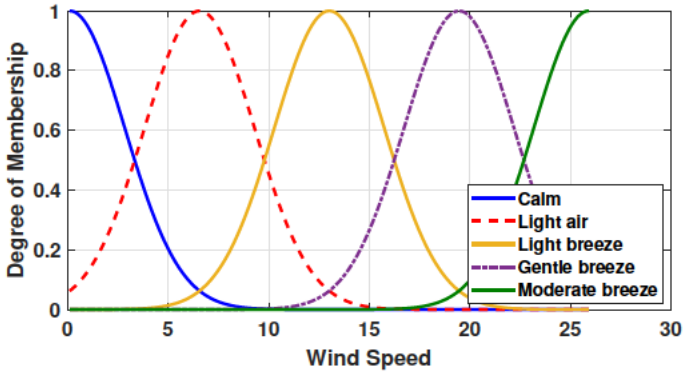

FIGURE 5.9: fuzzy memberships applied for modelling the wind speed in ANFIS.

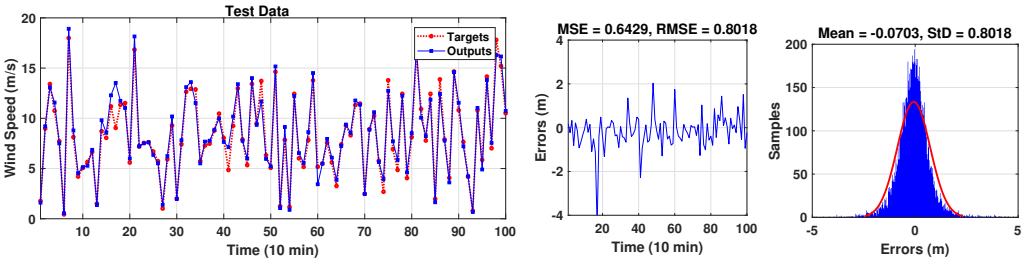

FIGURE 5.10: The wind speed forecasting results achieved by ANFIS network with the best tuned hyper-parameters on ten-minute ahead

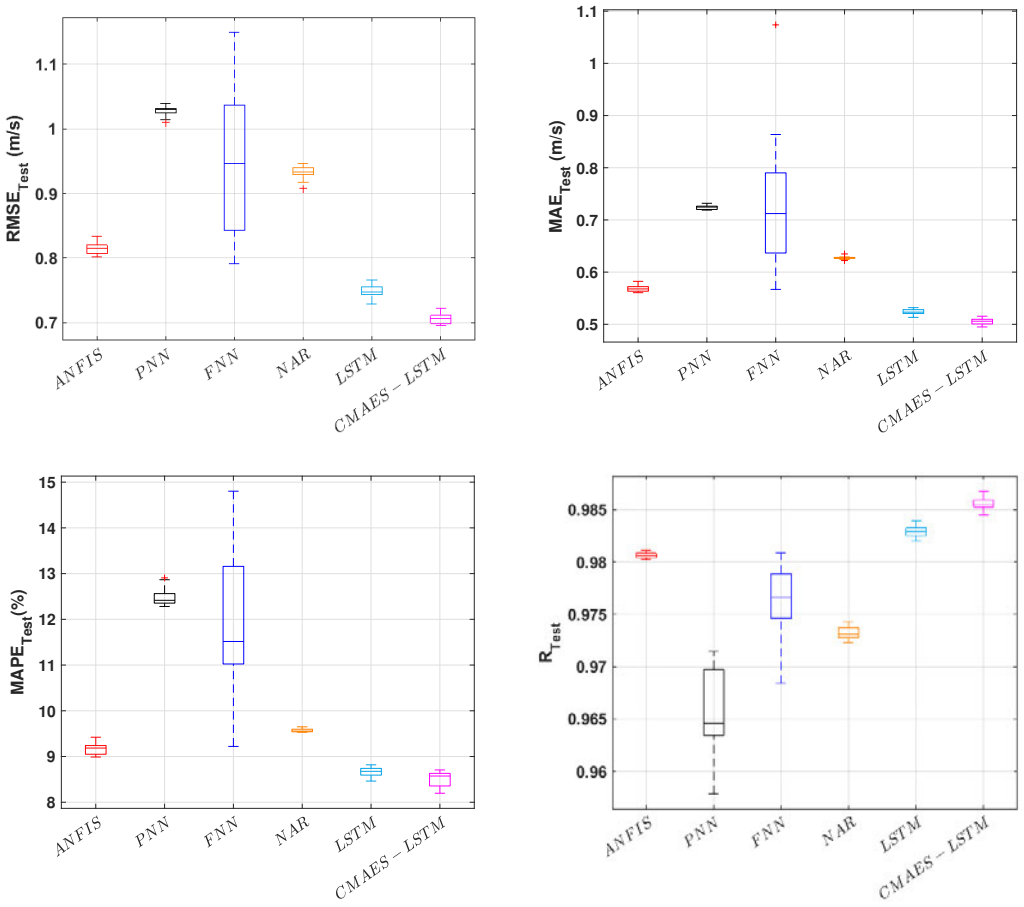

FIGURE 5.11: The performance of different forecasting models results of ten-minute ahead (a) RMSE (b) MAE (c) MAPE and (d) R-value .

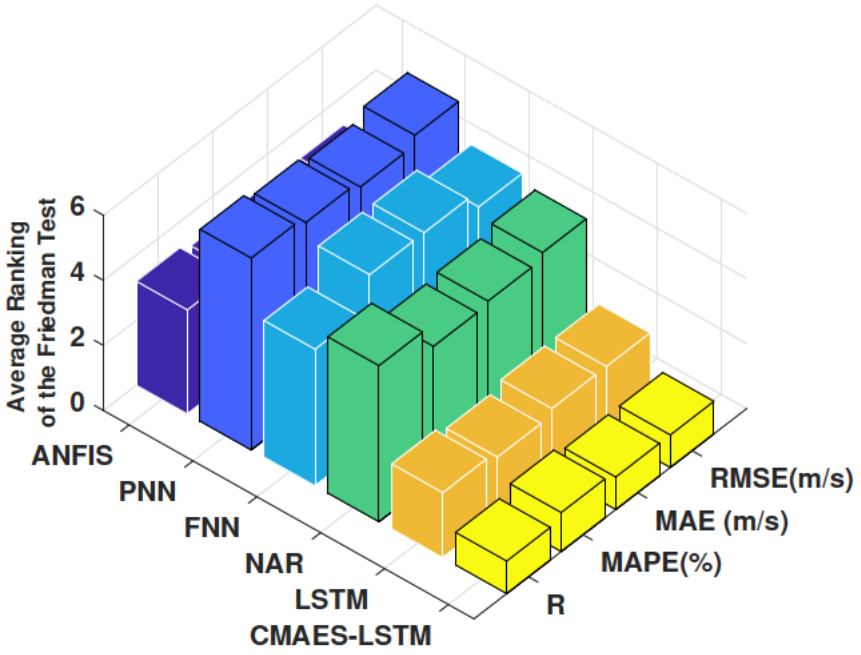

FIGURE 5.12: Average ranking of the Friedman test for performance indices statistical tests achieved by various applied forecasting models (ten-minute ahead).

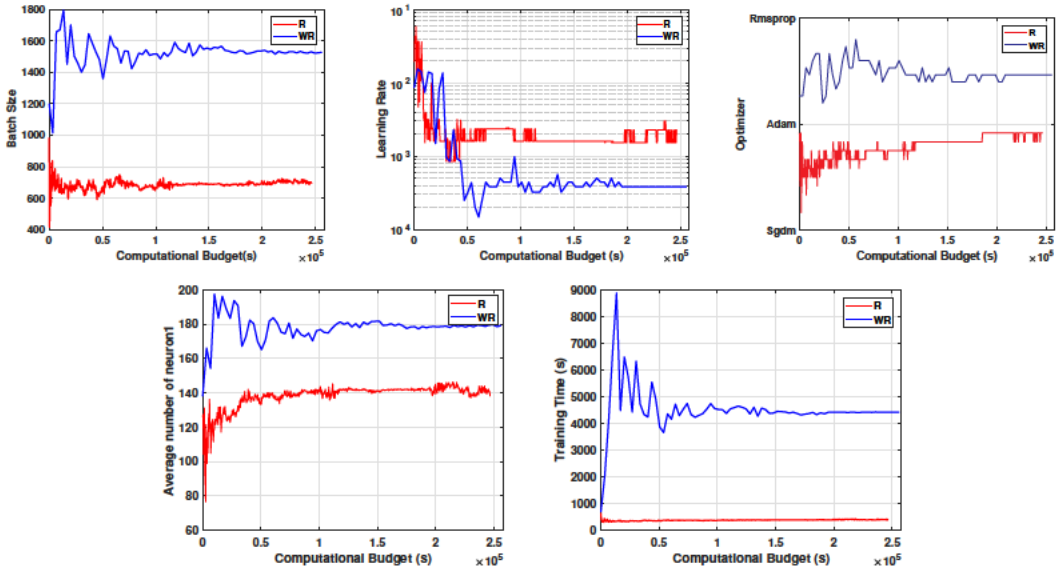

FIGURE 5.13: Comparison of CMAES performance for the optimisation the Hyper-parameters of the applied LSTM network for forecasting the short-term wind speed (10-minute ahead). To evaluate the performance of the LSTM model, two approaches are employed In the first approach, a penalty is applied for the time-consuming training runtime experiment (R). In the second approach no such penalty is applied (WR). (a) Average Batch Size (b) average learning rate, (c) optimiser (Sgdm, Adam and Rmsprop), (d) average number of neurons in the first LSTM layer, (e) average training runtime

Chapter 6

Hybrid Neuro-Evolutionary Method for Predicting Wind Turbine Power Output

6.1 Synopsis

Designing a reliable and accurate wind turbine power forecasting model is challenging because of complex local wind environments, non-linear responses of wind turbines and chaotic behaviours of wind speed. This paper proposes the potential of combined recurrent neural networks and Evolutionary Algorithms (EA) as a hyper-parameter optimiser for accurate forecasting of the power output in wind-turbine farms. The historical data in the supervisory control and data acquisition (SCADA) systems is applied as input to predict the power output from an onshore wind farm in Sweden. Moreover, we propose a hybrid outlier detection method including k-means clustering and an autoencoder. The comparison results indicate that removing the outliers form the original data using the proposed method improves the modeling results. We then complete comprehensive experiments and compared our proposed approach with five hybrid models. According to experimental results, the proposed model outperforms its counterparts in terms of four performance criteria.

Reference

[10] Neshat, M., Nezhad, M.M., Abbasnejad, E., Groppi, D., Heydari, A., Tjernberg, L.B., Garcia, D.A., Alexander, B. and Wagner, M., 2020. Hybrid Neuro-Evolutionary Method for Predicting Wind Turbine Power Output. Accepted for publication by 4th SEE Conference on Sustainable Development of Energy, Water and Environment Systems.

Statement of Authorship

Title of Paper	Hybrid Neuro-Evolutionary Method for Predicting Wind Turbine Power Output
Publication Status	☐ Published ☑ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Nezhad, M.M., Abbasnejad, E., Groppi, D., Heydari, A., Tjernberg, L.B., Garcia, D.A., Alexander, B. and Wagner, M., 2020. Hybrid Neuro-Evolutionary Method for Predicting Wind Turbine Power Output. Accepted for publication by 4th SEE Conference on Sustainable Development of Energy, Water and Environment Systems – SDEWES Conference.

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%)	70%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	05/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contr bution.

Name of Co-Author	Meysam Majidi Nezhad (5%)		
Contribution to the Paper	Wrote the literature review section, data analysis and visualization.		
Signature		Date	18/05/2020

Name of Co-Author	Ehsan Abbasnejad (5%)		
Contribution to the Paper	Provided advice on the machine learning methodology and recommendations on the implementations		
Signature		Date	16/05/2020

Name of Co-Author	Daniele Groppi (2.5%)		
Contribution to the Paper	Data analysis and visualization.		
Signature		Date	18/05/2020

Name of Co-Author	Azim Heydari (2.5%)		
Contribution to the Paper	Worked on the conceptualization of the research concept.		
Signature		Date	18/05/2020

Name of Co-Author	Lina Bertling Tjernberg (2.5%)		
Contribution to the Paper	Supervised development of the work.		
Signature		Date	18/05/2020

Name of Co-Author	Davide Astiaso Garcia (2.5%)		
Contribution to the Paper	Supervised development of the work.		
Signature		Date	18/05/2020

Name of Co-Author	Bradley Alexander (5%)		
Contribution to the Paper	Advised on data analysis and approaches to outlier removal. Helped revise paper.		
Signature		Date	10/05/2020

Name of Co-Author	Markus Wagner (5%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	12/05/2020

Please cut and paste additional co-author panels here as required.

6.2 Abstract

Reliable wind turbine power prediction is imperative to the planning, scheduling and control of wind energy farms for the purpose of stable power production. In recent years, Machine Learning (ML) methods have been successfully applied in a wide range of domains, including renewable energy. However, due to the challenging nature of power prediction in wind farms, the accuracy of current models falls far short of that required by industry. In this paper, we deploy a composite ML approach–namely a hybrid neuro-evolutionary algorithm–for accurate forecasting of the power output in wind-turbine farms. We use historical data from supervisory control and data acquisition (SCADA) systems as input to estimate the power output from an onshore wind farm in Sweden. At the beginning stage, the k-means clustering method and an autoencoder are employed to detect and filter noise in the SCADA measurements. Next, based on prior knowledge that the underlying wind patterns are highly non-linear and diverse, we combine a self-adaptive differential evolution (SaDE) algorithm as a hyper-parameter optimizer, and a recurrent neural network (RNN) called Long Short-term memory (LSTM). This allows us to model the power curve of a wind turbine in a farm. Two short time forecasting horizons, including ten minutes ahead and one hour ahead, are considered in our experiments. We show that our approach outperforms its counterparts.

6.3 Introduction

Renewable wind energy an established but fast-growing technology for the sustainable production of energy at scale. With falling costs and large-scale production of generators, the deployment of wind energy is accelerating. For example, gross installations of onshore and offshore wind farm in the EU were 0.3 GW in 2008, and increased to 3.2 GW in 2017 [242]. With such large increases in the deployment of wind energy forecasting of the power output of installed wind turbines is becoming vitally important. However, as local wind environments in wind farms are complex, and as the responses of wind turbines is non-linear and dependent on the condition of the turbine, wind power forecasting is a challenging problem [243]. Wind power forecasting is fundamental to the effective integration of wind farms into the power grid. For a single turbine, the following equation describes power output: P.

$$P = \frac{1}{2}\rho\pi R^2 C_p u^3 \tag{6.1}$$

The terms ρ, R and u refer to the density of air, the rotor radius and the wind speed respectively. Meanwhile, C_p is the power coefficient the proportion of available power the

turbine is able to extract. The theoretical estimation of wind turbine power is depicted by Equation 6.1. This equation describes a smooth s-shaped power curve that resembles a logistic function with wind on the x-axis.

However, because of the variable nature of wind and complex dynamics within and between turbines, the real power output of individual wind turbines is not precisely described by this curve [244, 245]. A more realistic alternative model for each wind turbine in a farm can be derived by fitting observations to field data [243].

On top of the task of modelling wind turbine power in response to *current* wind conditions, managers of wind farms also need to forecast *future* power output based on current conditions. Recent work has used complex data-driven models such as artificial neural networks (ANNs) to forecast turbine output with some degree of accuracy [50–52]. In this paper we propose an integrated approach that couples self-adaptive differential evolution with ANNs for accurate short term wind power forecasting. The input features used in our modeling are current wind speed, current wind direction and (in some models) current power output. The main contributions of this paper are as follows:

1. a new hybrid Neuro-evolutionary method (SaDE-LSTM) for short-term wind turbine power output forecasting that combines self-adaptive differential evolution (SaDE) [246] to act on a recurrent deep neural network [139] with two forecasting horizons of ten-minute and one-hour;

2. An advanced data filtering technique is implemented on the training observations (from SCADA data) using K-means clustering [247] and autoencoder neural-networks [248] to detect outliers;

3. A comparison of the performance of the models (using raw and clean SCADA datasets) is completed, in terms of the models' ability to assess the impact of the outlier detection method.

4. A comparison of the performance of four forecasting models trained to act on different subsets of SCADA inputs is completed. These sub-sets are wind speed for model one; wind speed and wind direction for model two; wind-speed and current power output for model three; and, finally, wind-speed, wind-direction and current power output for model four.

   Figure 6.1 illustrates the models and their inputs;

5. Finally, as there is no a straightforward theory with regard to the design and tune the hyper-parameters of an LSTM network [47], we tune the model structure and hyper-parameters using grid search; the Gray Wolf Optimizer [11] (GWO) method;

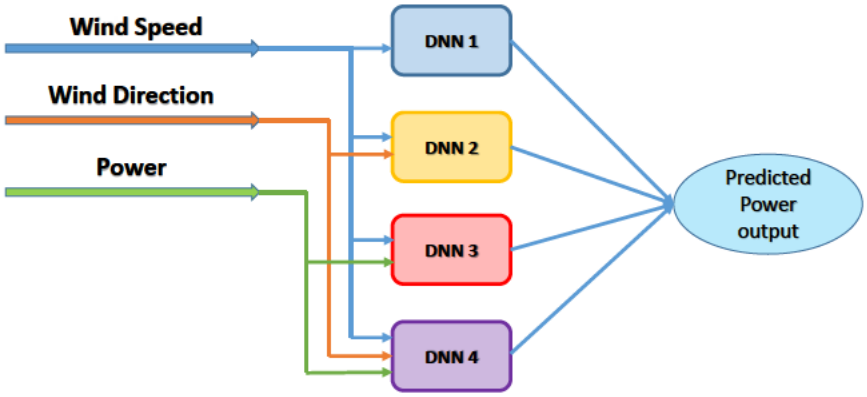

FIGURE 6.1: The proposed four different independent forecasting models.The applied power as an input is the current generated power by the wind turbine.

the Differential Evolution [94] (DE) algorithm; and covariance matrix adaptation evolution strategy [205] (CMA-ES).

The remainder of this article is organized as follows: Section 6.4 reviews current approaches to building wind power forecasting models. Section 6.5 describes the features of the SCADA datasets employed in this research, collected from 42-months' high-frequency monitoring of onshore wind turbines. Section 6.6 describes the new outlier detection method used in this work. Section 6.7 describes the experimental methodology. Next, the power prediction results trained by datasets of both raw and clean SCADA datasets are demonstrated. Finally, Section 6.9 summarises the contributions of this work.

6.4 Related Work

Previous work in this field combined physical models with numerical weather prediction (NWP) to forecast wind turbine output [249]. While such methods are fast, they lack accuracy.

More accurate predictions can be obtained using statistical methods to model the relationship between the inputs of the system and the corresponding outputs. Commonly applied statistical methods include time-series methods [250, 251], machine learning methods [252, 253], the persistence method [254] and Kalman filtering [255].

One set of well-known time series forecasting methods for wind power prediction is known as autoregressive modelling. Such modelling includes autoregressive moving average (ARMA), autoregressive integrated moving average (ARIMA), and a fractional version of ARIMA (f-ARIMA) [53]. Both ARMA and ARIMA models can capture short-range

correlations between inputs and outputs, and the f-ARIMA method is well adapted for representing the time series data with long memory characteristics [251].

Another effective technique for forecasting time series data is recurrent neural networks (RNNs). Olaofe et al. [54] applied RNNs to predict the wind turbine power output one-day ahead. However, the applied 'tanh' activation function used in this work leads to disappearing and exploding gradients, which lead to difficultly in training an accurate model [256]. Long short-term memory networks (LSTMs) were introduced in [139], partly to help avoid these issues. LSTMs can learn the correlations that exist in time series data with some accuracy. In [55], LSTMs were employed for short-term predictions of wind power. That study showed that LSTMs could outperform traditional ANNs and support vector machines (SVMs) in terms of prediction accuracy. A combination of principal component analysis (PCA) and a LSTM forecasting model was proposed in [56], and compared with back-propagation (BP) neural network and an SVM model. The study found that the PCA-LSTM framework results produced higher forecasting accuracy than other methods. Recently, Erick et al. [57] defined a new architecture for wind power forecasting composed of LSTM blocks replaced the hidden units in the Echo State Network (ESN). The authors also used quantile regression to produce a robust estimation of the proposed forecast target. Finally, Yu et al. [58] used an LSTM with an enhanced forget-gate network model (LSTM-EFG) combined with a Spectral Clustering method to forecasting wind power. This technique resulted in considerably increased accuracy.

However, none of above studies used an automated method to tune the hyper-parameters of their ANN models. Such automatic tuning helps with porting models to a new setting and makes it possible to more rigorously compare modelling approaches.

One recent work that has used hyper-parameter tuning is that of Qin et al. [59] who used the Cuckoo Search Optimization (CSO) method to improve performance of a Back Propagation Neural Network (BPNN) by adjusting the connection weights. They reported that the accuracy of the proposed hybrid model was higher than that of other methods for predicting the wind speed time series. Shi et al. [60] used the dragonfly algorithm (DA) to tune RNN hyper-parameters for wind power forecasting.

In another recent work, Peng et al. [61] used Differential Evolution (DE) to optimise LSTM parameters, and the reported results indicated that the hybrid DE-LSTM model is able to outperform traditional forecasting models in terms of prediction accuracy. More recently, Neshat et al. [9] forecasted time series through online hyper-parameter tuning of an LSTM model using CMA-ES. This work improved on earlier work by systematically comparing the impact of tuning strategies, model input sets, and data pre-processing on

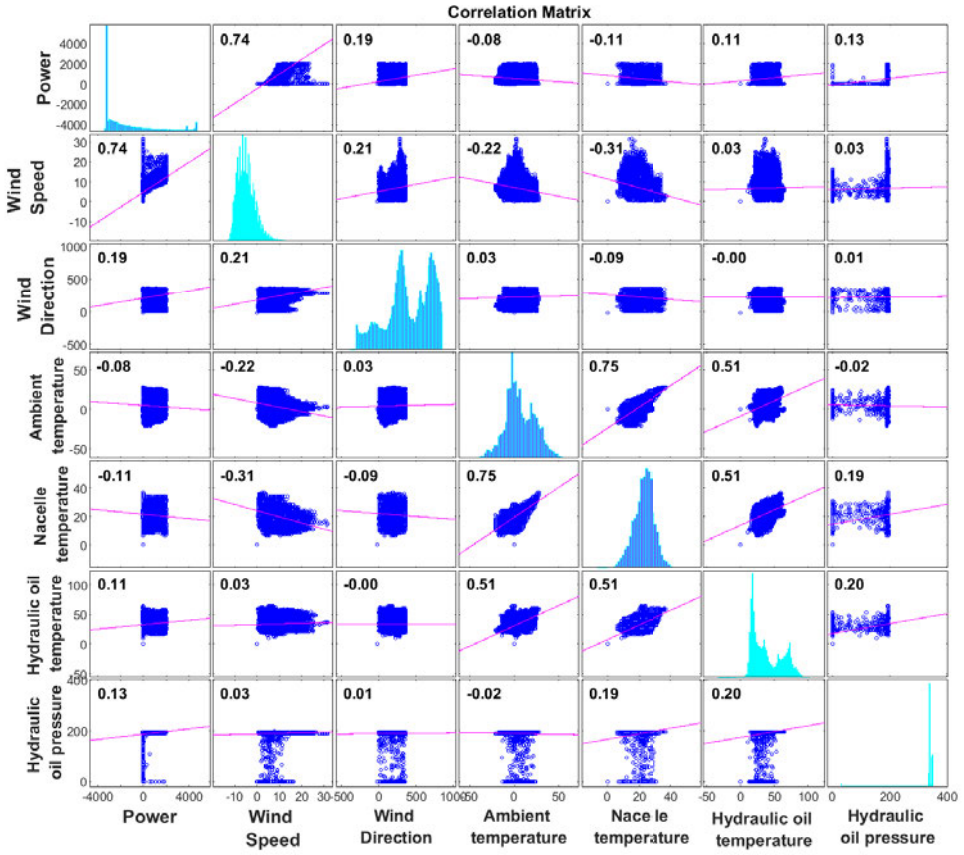

FIGURE 6.2: The Pearson's linear correlation coefficients between all pairs of the wind turbine data (SCADA). The correlation plot shows that wind speed, wind direction and Power are highly correlated.

prediction performance. These comparisons define some of the search landscape in the parameter space of these algorithms.

6.5 SCADA data description and analysis

The data analysed in this research comes from six turbines of one onshore wind farm in north-western Europe (Sweden) [257]. For each turbine, 42 months of data are available from January of 2013 to June of 2016, including 10-minute interval operation data and a log file. Data on faults and maintenance were also stored. In this paper, we select and investigate the SCADA data from the sixth turbine in the wind farm. In order to evaluate and analyse the correlation between power output and other SCADA features, seven features are chosen including wind speed, wind direction, ambient temperature, Nacelle temperature, Hydraulic oil temperature and Hydraulic oil pressure. These are the most recommended SCADA features for power prediction from [50]. Pearson's linear

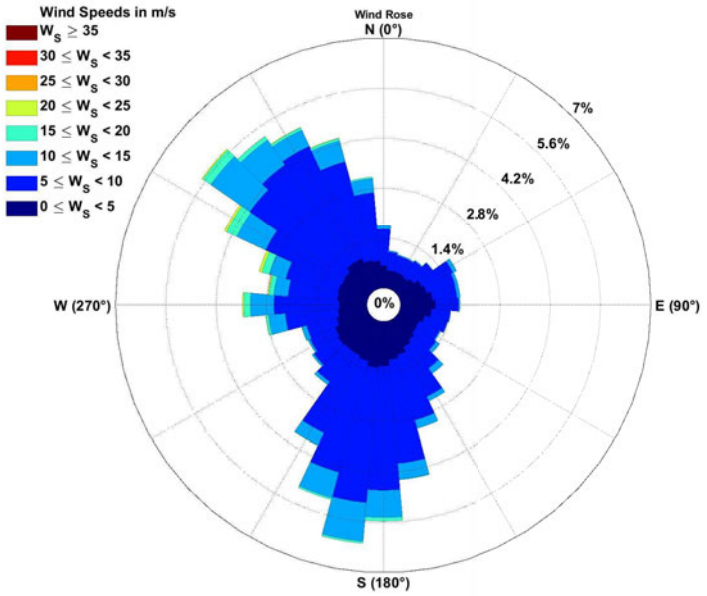

FIGURE 6.3: a large view of how the wind speed and wind direction are distributed at the wind farm (Sweden) from 2013 to 2016 (June).

correlation coefficients between all pairs of the wind turbine data features can be seen in Figure 6.2. The highest correlations are those between Power output and wind speed, as well as between power output and wind direction. Therefore, we select the wind speed and wind direction as ANN inputs designate generated power as the network output. The diagonal in Figure 6.2 shows the distributions of each variable, including power. These distributions show some outliers, which might pose challenging for modelling. It is also of note that there are some negative values for produced power; these values are caused by stationary turbines spinning up.

Figure 6.3 depicts the wind rose for the wind farm. It also shows that the dominant wind direction is North-west, and a secondary prevailing direction is South-east. However, there are also occasional West winds.

6.6 Information preprocessing

In data science, outliers are values that differ from regular observations in a dataset. Figure 6.4 shows the correlation between wind turbine power output and wind speed; moreover, it represents the correlation between wind turbine power output and wind direction during the 42 month data collection period. The outliers can be seen clearly in the scatter plots; they are distributed on the right side of the plot. In this study, we apply a combination of a K-means method which is one of the well-known Clustering

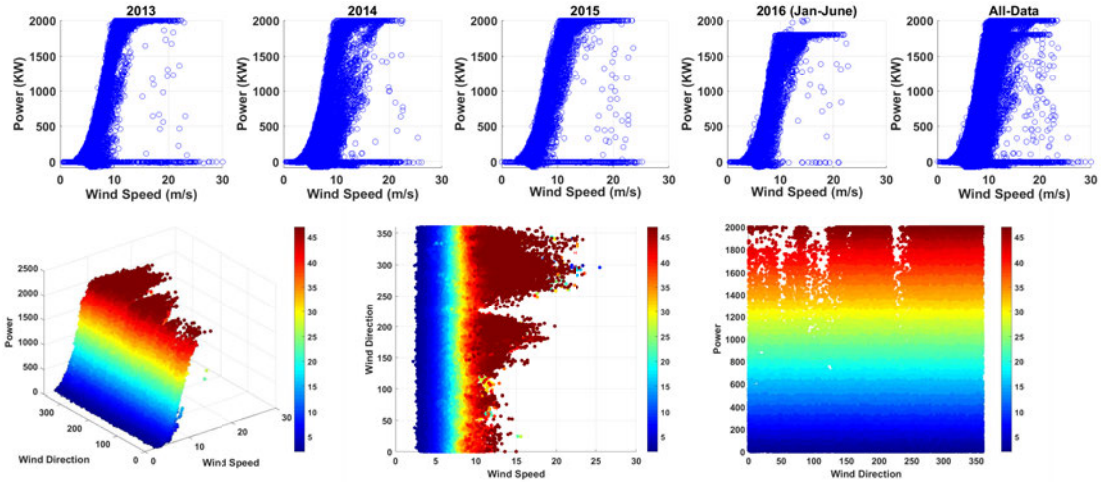

FIGURE 6.4: The correlation between wind turbine power output and wind speed over the 42 months of data collection.(a) Outliers can be seen clearly in the data. (b) 3D figure of power curves, wind speed and wind direction. (c) the correlation between wind speed and direction. (d) power curves and wind direction.

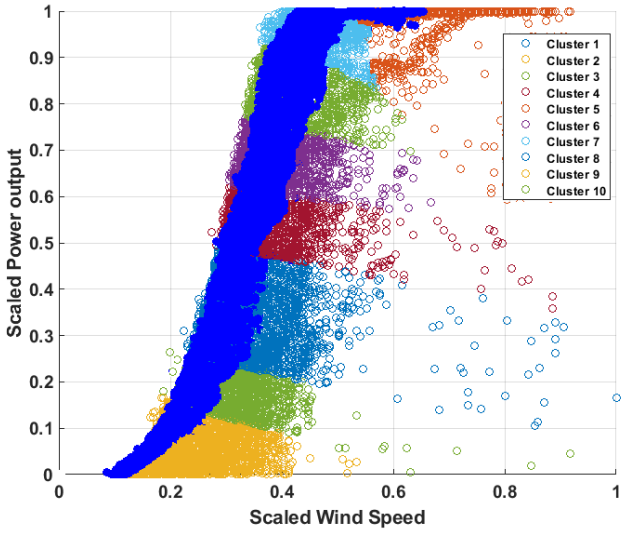

FIGURE 6.5: Clustering the data into 10 groups by K-means and then detecting and removing the outliers by an autodecoder NN. The purified data after removing outliers show by the dark blue region.

Based outlier Detection (CBOD) [258] methods and an autoencoder neural network to detect and remove the outliers from the SCADA dataset. As previous studies listed in Section 6.4, wind speed is the primary factor that determines wind power among the SCADA features. In the data, wind speed is widely distributed; we use the K-Means clustering algorithm to classify the wind power data into K subclasses. Before the clustering, due to the significant differences in numerical values of each type of data which has a great impact on the training of the autoencoder's latent model, the data is normalized between zero and one [259]. The normalization used in this paper is described

in Equation 6.2.

$$\hat{Z} = \frac{Z - Z_{min}}{Z_{max} - Z_{min}} \tag{6.2}$$

For this work, the number of the clusters is set to 10 [259] for wind speed and power output of SCADA data. These clusters indicate the different operation states of the main subsystems, such as the drive train and the control system [260]. Figure 6.5 shows ten clusters of data from Turbine 6. It can be observed that the distribution in each cluster is a horizontal band. Within these bands, outliers are more easily discerned as being relatively far from the main body of the cluster. In order to remove the outliers in each cluster, an autoencoder neural network is used that shows better performance compared with other traditional outliers detection methods [261, 262]. An autoencoder is a particular type of unsupervised feedforward neural network. This network is trained to reconstruct output in such a way that it becomes similar to each input. In this work the autoencoder consists of an input layer and one hidden layer which are fully connected [263]. For training an autoencoder, the input data are mapped to the hidden layer where the encoding of input data takes place, which typically comprises fewer nodes than the input layer and consequently compresses the data. Next, from the hidden layer, the reconstructed data flows through the output layer, which is re-transformed in a process called 'decoding', and the squared restoration error between the network's output and its input is calculated. For detecting the outliers, It is noticed that outliers have higher reconstruction error than the norm of the dataset. Therefore we remove the observations which have higher RMSE than the average of all data RMSE. Figure 6.5 presents the outliers detection and removal process, with the dark blue sections showing the clean data.

6.6.1 Performance criteria of forecasting models

To evaluate and compare the performance of the applied forecasting models, four broad performance indices are used: the mean square error (MSE), the root mean square error (RMSE), mean absolute error (MAE), and the Pearson correlation coefficient (R) [237]. The equations for MAE, RMSE and R are described as follows :

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |f_p(i) - f_o(i)| \tag{6.3}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (f_p(i) - f_o(i))^2} \tag{6.4}$$

$$R = \frac{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - \overline{f}_p)(f_o(i) - \overline{f}_o)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_p(i) - \overline{f}_p)^2} \times \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f_o(i) - \overline{f}_o)^2}} \tag{6.5}$$

where $f_p(i)$ and $f_o(i)$ denote the predicted and observed SCADA values at the i^{th} data point. The total number of observed data points is N. The variables $\overline{f}_p$ and $\overline{f}_o$ are the means of the predicted and perceived power measures, respectively. In order to develop the effectiveness of the predicted model, MSE, RMSE and MAE should be minimised, while R should be maximised.

6.7 Methodology

In this section, we introduce the proposed methodologies and related concepts for short-term wind turbine power output forecasting, including LSTM network details, self-adaptive differential evolution (SaDE) and the hybrid LSTM network and the SaDE algorithm.

6.7.1 Long short-term memory deep neural network (LSTM)

The LSTM network [139] is a special kind of recurrent neural network (RNN) with three thresholds: the input gate, the output gate and the forgetting gate. The unit structure of the LSTM network can be seen in Figure 6.6. The forgetting gate defines the permissible rise or drop of the data flow [15] by setting the threshold, which indicates reservation and forgetting. Considering that an RNN hidden layer has only one state, there are severe difficulties with gradient fading and gradient explosion. Augmenting the RNN, the LSTM adds the structure of the cell state, which can recognise the long-term preservation of the state and emphasises the active memory function of the LSTM network. In the case of massive wind power time series data, the network can significantly enhance the accuracy of wind power prediction. In the forward propagation method of the LSTM network, the output value of the forgetting gate f_t can prepare the information trade-off of the unit state and the functional relationship encoded by Equation 5.1.

$$f_t = \sigma(w_f h_{t-1} + u_f x_t + b_f) \tag{6.6}$$

Both i_t and $\tilde{c}_t$ are generated by the input gate, which are related to the previous moment. The expressions are as shown as Equation 6.7 and 6.8.

$$i_t = \sigma(w_i h_{t-1} + u_i x_t + b_i) \tag{6.7}$$

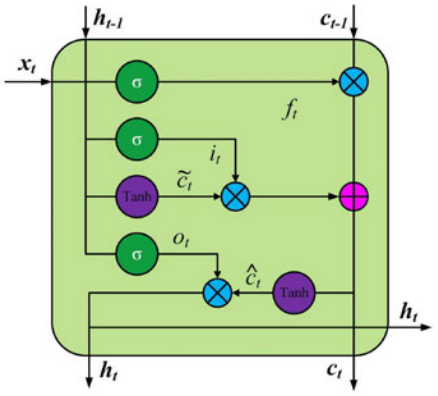

FIGURE 6.6: The internal structure of LSTM network from [15].

$$\tilde{c}_t = tanh(w_c h_{t-1} + u_c x_t + b_c) \tag{6.8}$$

Cell state c_t is the transmission centre of the cell state before and later LSTM, which has the following functional relationship:

$$c_t = c_{t-1} \odot f_t + i_t \odot \tilde{c}_t \tag{6.9}$$

The output h_t of the output gate derives from two components. The first part is the output of the previous moment that is the input of the current moment, and the second part is information of the current cell state and the particular expression model is delivered as Equation 6.10 and 6.11.

$$o_t = \sigma(w_o h_{t-1} + u_o x_t + b_o) \tag{6.10}$$

$$h_t = o_t \odot tanh(c_t) \tag{6.11}$$

where both u and w are the weight values; b and σ are the bias values and activation function respectively, and $\odot$ is the Hadamard product. For the LSTM network training settings, the Adam algorithm [234] is employed to optimise the loss function, and Dropout [264] is used to prevent model overfitting.

6.7.2 Self-adaptive Differential Evolution (SaDE)

SaDE [246] is proposed by Qin et al. to concurrently perform two popular mutation strategies "DE/rand/1" and "DE/current-to-best/1". SaDE adjusts the probability that offspring solutions will be generated using each strategy, depending on the success rates (improved solutions) in the past N_f generations of the algorithm. The aim of this adaptation scheme is to evolve the best mutation strategy as search progresses. This methodology is similar to the ideas proposed in [265], where striving heuristics (including diverse

DE variants, simplex methods and evolution strategies) are adopted simultaneously and probabilities for offspring generation are adjusted dynamically.

In SaDE, the vectors of the mutation factors are generated independently at each iteration based on a normal distribution ($\mu = 0.5$, $\sigma = 0.3$), and trimmed to the interval (0, 2]. This scheme can retain both local (with small F_i values) and global search capability, so as to create potentially suitable mutation vectors during the evolution process. In addition, the crossover probabilities are randomly generated based on an independent normal distribution, with $\mu = C_{Rm}$ and $\sigma = 0.1$. This is in contrast to the Fi and C_{Ri} values, which remain fixed for the last five generations before the next regeneration. The C_{Rm} is initially set to 0.5. in order to tune C_R to suitable values, the authors renew C_{Rm} every 25 generations using the best C_R values from the last C_{Rm} update.

To speed up the SaDE convergence rate, a further local search procedure (quasi-Newton method) is used on some competent solutions after N_s generations. The benefits of Self-adaptive parameter control make the SaDE as one of the most successful evolutionary algorithms, especially in the context of real engineering optimisation problems that have multi-modal search spaces with many local optima [266].

6.7.3 Hybrid Neuro-Evolutionary Deep Learning method

Multiple parameters can influence the precision and performance of LSTM networks. The selected hyper-parameters include maximum training number of LSTM (Epoch), hidden layer size, batch size, initial learning rate and the optimizer type. If the maximum training number is too small, then it will be difficult for the training data to converge; if we set the number to a large value, then the training process might overfit. The hidden layer size can influence the impact of the fitting [61]. Batch size is also an important hyper-parameter. If batch size is set too low, then the training data will struggle to converge, resulting in under-fitting. If the batch size is too large, then the necessary memory will rise significantly. There are also complex interactions between hyper-parameters. Therefore, a reliable optimisation technique should be utilised to tune the optimal combination of hyper-parameters, and to balance forecasting performance and computational efficiency.

There are three main methods for tuning hyper-parameters, including 1) manual trial and error, 2) systematic grid search, and 3) meta-heuristic approaches. In this paper, we apply the grid search and meta-heuristic approach, which is a self-adaptive version of DE (SaDE) that can be used to adjust the optimal configuration of settings for the LSTM. The performance of this hybrid technique (SaDE-LSTM) is compared with that of grid

TABLE 6.1: Summary of the best-found configuration for the predictive models tested in this paper (ten-minute ahead).

Models	Descriptions
ANFIS [232]	Adaptive neuro-fuzzy inference system: • OptMethod= Backpropagation • **Training settings** – *ErrorGoal*=0; – *InitialStepSize*=0.01; – *StepSizeDecrease*=0.9; – *StepSizeIncrease*=1.1; • **FIS features** – *mf* number=5; – *mf* type='gaussmf';
LSTM [47] + grid search	Long Short-term memory Network: • **LSTM hyper-parameters** – *miniBatchSize*=512 – *LearningRate*= 10^{-3} – *numHiddenUnits1* = 100; – *Optimiser*= 'adam'
CMAES-LSTM [9]	• **CMAES-LSTM hyper-parameters (Best configuration)** – *miniBatchSize*=1114 – *LearningRate*= 10^{-4} – *numHiddenUnits1*=201 ; – *numHiddenUnits2*=30 ; – *Optimiser*= 'adam'
DE-LSTM [61]	• **DE-LSTM hyper-parameters (Best configuration)** – *miniBatchSize*=1155 – *LearningRate*=2.2×10^{-3} – *numHiddenUnits1*=141 ; – *numHiddenUnits2*=42 ; – *Optimiser*= 'adam'
GWO-LSTM [4, 267]	• **GWO-LSTM hyper-parameters (Best configuration)** – *miniBatchSize*=1598 – *LearningRate*=0.3×10^{-3} – *numHiddenUnits1*= 150; – *numHiddenUnits2*=235 ; – *Optimiser*= 'rmsprop'
SaDE-LSTM	• **SaDE-LSTM hyper-parameters (Best configuration)** – *miniBatchSize*=727 – *LearningRate*=5.89×10^{-3} – *numHiddenUnits1*= 184; – *numHiddenUnits2*=117 ; – *Optimiser*= 'adam'

search; three hybrid neuro-evolutionary methods: DE-LSTM [61], CMAES-LSTM [9], GWO-LSTM [4, 267]; and ANFIS [232].

In the grid search method, we evaluate and tune only two hyper-parameters of the LSTM: the batch size and the learning rate. Other settings assign a fixed value for the optimizer type, the number of LSTM hidden layers, the hidden layer size, maximum number of epochs by ('adam' [234]) one, 100 and 100 respectively. These values are chosen in order to provide a baseline for the LSTM model evaluation. The ranges of batch size and learning rate are, respectively, selected from $128 \leq BS \leq 2048$ and $10^{-5} \leq LR \leq 10^{-1}$.

The optimization procedures are as follows:

- **Step 1**. Data preprocessing. Detecting and removing the outliers, then dividing the dataset into three subsets: training, validation, and test sets.
- **Step 2**. Initialization. The following parameters are set: maximum iteration number of SaDE, population size (NP), minimum and maximum crossover rate (C_R), mutation rate (F), and the upper and lower bounds of decision variables, the iteration numbers for updating the control parameters (N_f and N_s) are set.
- **Step 3**. Generating offspring: The offspring solution is generated by the mutation, crossover, and selection operations, and is iterated until the offspring population is achieved.
- **Step 4**. Evaluating the offspring: The fitness values of the offspring population are computed by applying the proposed hyper-parameters in the LSTM. The fitness is the root of the mean square error (RMSE) of the validation set; other performance indices are also computed and recorded. The RMSE should be minimized, and the corresponding individual is the current best solution that achieves this.
- **Step 5**. Updating the SaDE control parameters through the historical optimisation process.
- **Step 6**. Stopping criteria: if the maximum iteration is achieved, then SaDE is terminated and the optimum configuration is taken; otherwise, the procedure returns to Step 3.

The fitness function of the optimisation process is defined as follows:

$$
\begin{aligned}
&Argmin \rightarrow f = fitness(N_{h_1}, N_{h_2}, ..., N_{h_D}, N_{n_1h_1}, N_{n_2h_2}, ...N_{n_Dh_D}, L_R, B_S, Op),\\
&Subject - to:\\
&\qquad LN_h \leq N_h \leq UN_h,\\
&\qquad LN_n \leq N_n \leq UN_n,\\
&\qquad 10^{-5} \leq L_R \leq 10^{-1},\\
&\qquad 128 \leq B_S \leq 2048\\
&\qquad 1 \leq Op \leq 3.
\end{aligned} \tag{6.12}
$$

where $N_{h_i}, \{i = 1, \ldots, D\}$ is the number of hidden layers for the $i-^{th}$ LSTM network and $N_{n_i,h_j}, \{j = 1, \ldots, D_l\}$ is the number of neurons in the i^{th} hidden layer of this network. The lower and upper bounds of N_h are presented by LN_h and UN_h , while LN_n and UN_n are the lower and upper bounds of neuron number. The Op is the selected optimizer for optimising the LSTM weights ('sqdm' [268], 'adam' [234], 'rmsprop' [269]).

6.8 Experimentation design

In the first step of the forecasting power output of the wind turbine, we proposed four DNN models with different inputs and the same output. The main aim of proposing these forecasting models is to analyse the impact of three SCADA features, wind speed, wind direction and the currently generated power on the predicting accuracy of the power output both ten-minute and one-hour ahead. In the second step, we compare the performance of the proposed models before and after removing the outliers from the SCADA dataset to illustrate the effectiveness of the outlier detection technique (K-means + Autoencoder). Finally, the proposed hybrid model (SaDE-LSTM) is compared with some of the state-of-the-art forecasting frameworks.

To evaluate the performance of four models using raw SCADA data which are randomly categorised into three training (80%), testing (10%) and validating (10%) sets, the LSTM deep network is used. This network is composed of one sequence input layer, one LSTM layer, fully-connected layer and a regression layer. A grid search method is used to tune the hyper-parameters, batch size and learning rate. Figure 6.7 presents the performance of LSTM framework with a tuned batch size and learning rate parameters for three forecasting models in the interval of ten-minute. The best performance of model 1 (one input) is obtained where the values of batch size are greater than 512 and learning rate placed between the range of 10^{-2} and 10^{-4}. The forecasting behaviours of both models 2 and 3 are similar, and the best accuracy occurs for batch sizes greater than 256 and a learning rate between 10^{-3} and 10^{-5}.

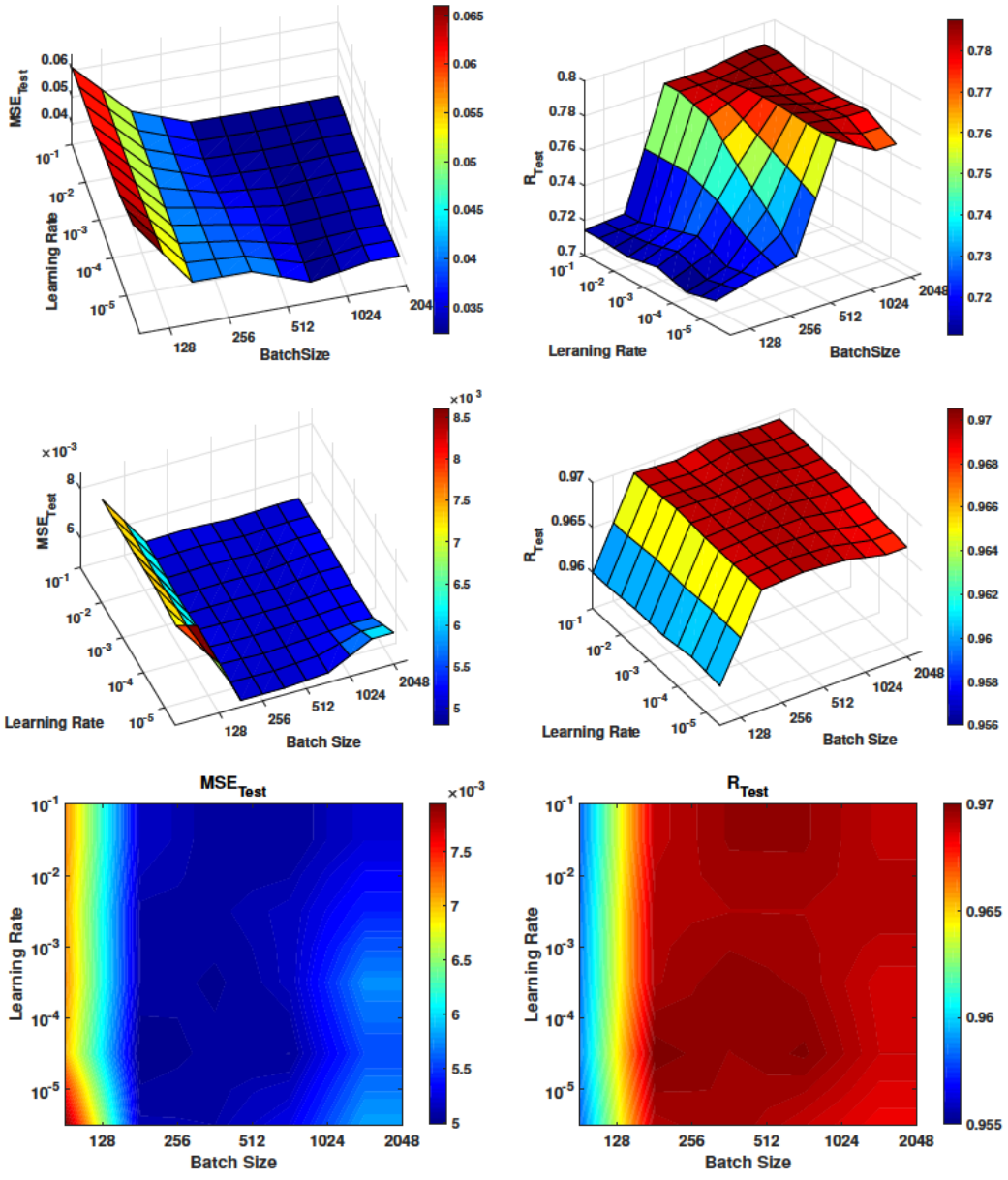

FIGURE 6.7: Hyper-parameter tuning of the applied LSTM network for forecasting the short-term power output of the wind turbine without removing the outliers(Layer number=1, neuron number=100, Optimiser='Adam') .(a) the average of MSE test-set (ten-minute ahead) with one input (wind speed) (b) the average of R-value test-set (ten-minutes ahead) with one input (wind speed).(c) and (d) the average of MSE and R-value test-set with two inputs (wind speed, Power) respectively. (e) and (f)the average of MSE and R-value test-set with two inputs (wind speed, wind direction and Power) respectively

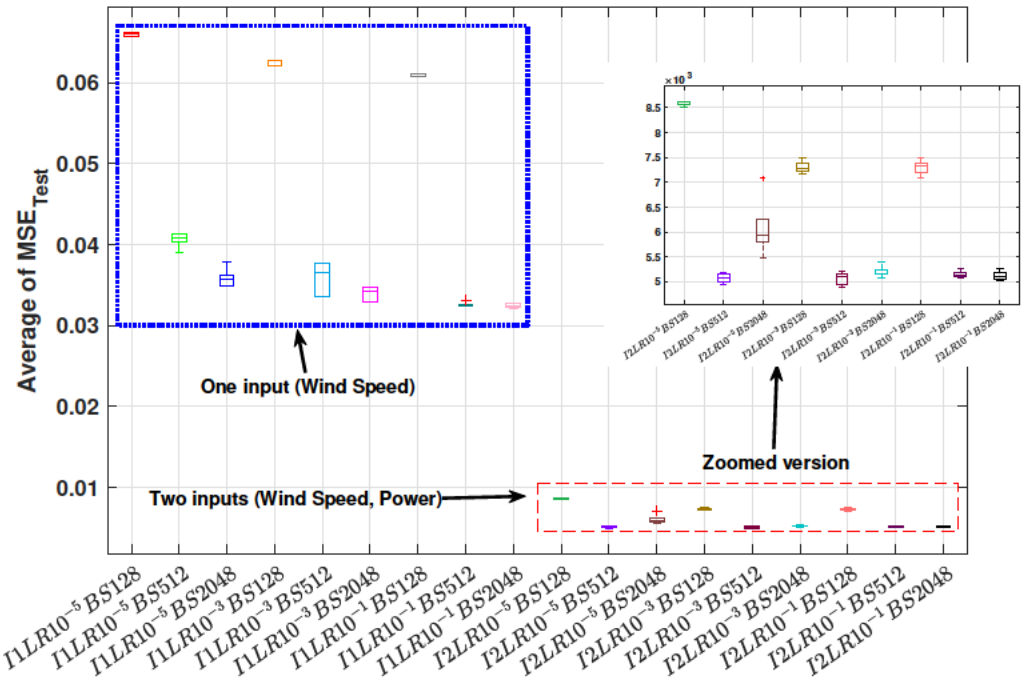

FIGURE 6.8: Comparison of the LSTM performance with one (I_1) and two inputs (I_2) without removing outliers

Figure 6.8 shows an average performance (MSE and R) comparison between two forecasting models (1 and 2 and) shows that using the currently generated power as an input plays a significant role in producing an accurate prediction.

After removing the outliers, we compare the performance of the LSTM framework as a predictor of the power produced by the 6^{th} wind turbine in two models (model 1 with one input and model 3 with two inputs including wind speed and direction) with different ranges of the batch size and learning rate. Figure 6.9 illustrates the 3D forecasting landscape of the correlation between the batch size, learning rates and forecasting accuracy for model 1 (within ten-minute) and model 3 (one-hour interval) . The applied method for tuning the hyper-parameters is the grid search in this experiment. In model 1, the best prediction results happen where the batch size is more than 1024 and the learning rate value is between 10^{-3} and 10^{-4}.

In order to highlight the benefit of using the outlier detection & removal technique (K-means + Autoencoder), a comprehensive comparison for four prediction models is applied (the time interval considered within these models is ten minutes), and the results of this experiment can be seen in Figure 6.10. Meanwhile, it is noticed that removing the noise from the SCADA data results in a significant enhancement in the accuracy of the prediction except two inappropriate configurations (Batch size=128 and Learning rate=10^{-3}, 10^{-1}).

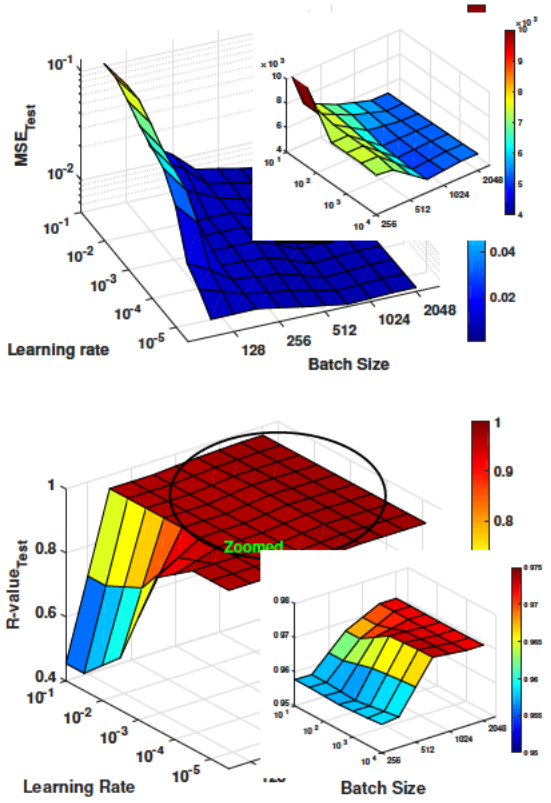

FIGURE 6.9: Hyper-parameter tuning of the applied LSTM network for forecasting the short-term power output of the wind turbine after removing outliers(Layer number=1, neuron number=100, Optimiser='Adam') .(a) the average of MSE test-set (ten-minute ahead) with one input (wind speed) (b) the average of R-value test-set (one-hour ahead) with two inputs (wind speed and direction).

The same experiment was completed in order to compare the performance of four models in forecasting the wind turbine power output in the one-hour ahead. The comparative outcomes are presented in Figure 6.11. In the first and second models, the highest accuracy was observed when batch sizes were large; however, the best-found configurations of the hyper-parameters in model 3 and 4 occurred where the batch size and learning rate of 256 and 10^{-4} respectively. e perform a statistical analysis of the forecasting results above using a Friedman test (non-parametric and multiple comparisons). The statistical analysis results by implementing the Friedman test are shown in Figure 6.12. We rank the four forecasting LSTM models corresponding to their mean value. As shown in Figure 6.12, Model 3 and 4 obtained the first and second ranking (i.e., the lowest value goes the first rank). This was in a comparison featuring all models over the 25 configurations of the hyper-parameters, under the ten-minute ahead forecasting conditions.

The overall comparison of four LSTM models at intervals of ten-minute and one-hour ahead can be seen in Figure 6.13. Each box represents the average RMSE forecasting of all 25 configurations per model. These statistical results show that the 4th model with

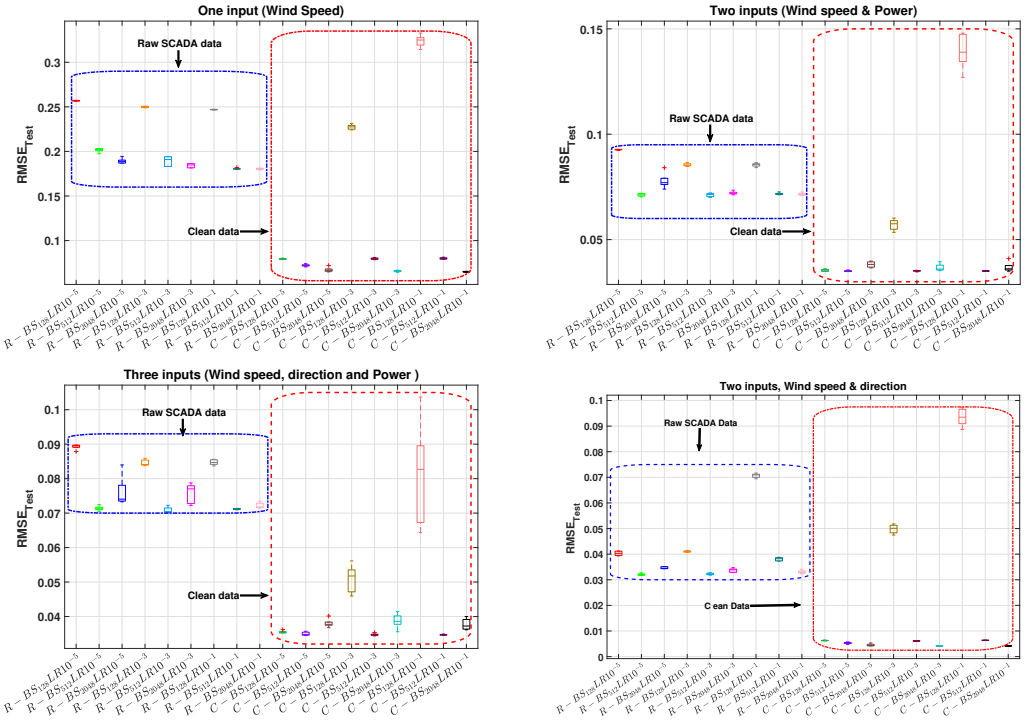

FIGURE 6.10: A comparison of the LSTM network Hyper-parameter tuning performance training on the raw SCADA data (R) and training after removing the outliers (C) for ten-minute ahead forecasting (Layer number=1, neuron number=100, Optimiser='Adam') .(a) the RMSE test-set with one input (wind speed) (b) the RMSE test-set with two inputs (wind speed and current power).

three inputs, including wind speed, wind direction and the current power output of the wind turbine found a configuration with the minimum validation error. However, model 3 outperforms other models on average. According to the statistical results, model 3 with two inputs (wind speed and the current produced power) perform better than other models. Therefore, we then applied this model to developing hybrid neuro-evolutionary methods. To evaluate the performance of the proposed hybrid model, we compare five different forecasting methods, including the best LSTM model which is tuned by the grid search, an adaptive neuro-fuzzy inference system (ANFIS) (its hyper-parameters are assigned based on the study in [232]), and three new hybrid neuro-evolutionary methods (CMAES-LSTM [9], DE-LSTM [61] and GWO-LSTM [4, 267]). Table 6.2 and Table 6.3 summarise the outcomes of the performance indices produced to determine the optimal structure and hyper-parameters of the applied forecasters. It is evident that the SaDE-LSTM hybrid model outperforms other hybrid models and provides more accurate forecasting results. Table 6.1 reports the best-found configurations of the proposed forecasting model and other compared models.

In addition, The actual and forecasting values generated by the hybrid models and the LSTM networks for the one-hour ahead forecasting are shown in Figure 6.14. In the

TABLE 6.2: Performance indices of forecasting wind turbine power output achieved by different models for ten-minutes ahead.

		MSE		RMSE		MAE		R	
Model		**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
ANFIS	Mean	6.981E-03	6.949E-03	8.324E-02	8.343E-02	5.287E-02	5.282E-02	9.615E-01	9.618E-01
	Min	5.657E-03	5.725E-03	7.566E-02	7.521E-02	4.795E-02	4.809E-02	9.593E-01	9.584E-01
	Max	7.648E-03	7.673E-03	8.760E-02	8.745E-02	5.546E-02	5.544E-02	9.665E-01	9.664E-01
	Std	7.860E-04	7.614E-04	4.662E-03	4.834E-03	2.999E-03	2.979E-03	2.825E-03	2.858E-03
LSTM-grid	Mean	1.427E-03	1.407E-03	3.778E-02	3.751E-02	2.834E-02	2.819E-02	9.826E-01	9.829E-01
	Min	1.411E-03	1.382E-03	3.756E-02	3.718E-02	2.822E-02	2.804E-02	9.826E-01	9.826E-01
	Max	1.452E-03	1.450E-03	3.810E-02	3.808E-02	2.867E-02	2.836E-02	9.827E-01	9.830E-01
	Std	1.520E-04	2.552E-04	1.233E-02	1.598E-02	1.908E-03	1.216E-04	4.968E-05	1.606E-04
CMAES-LSTM	Mean	1.236E-03	1.193E-03	3.515E-02	3.454E-02	2.643E-02	2.615E-02	9.926E-01	9.928E-01
	Min	1.227E-03	1.171E-03	3.503E-02	3.422E-02	2.631E-02	2.594E-02	9.926E-01	9.927E-01
	Max	1.256E-03	1.206E-03	3.544E-02	3.473E-02	2.679E-02	2.638E-02	9.927E-01	9.930E-01
	Std	7.280E-06	1.161E-05	1.033E-04	1.682E-04	1.206E-04	1.393E-04	2.597E-05	9.001E-05
DE-LSTM	Mean	1.241E-03	1.163E-03	3.522E-02	3.411E-02	2.648E-02	2.583E-02	9.926E-01	9.930E-01
	Min	1.235E-03	1.159E-03	3.515E-02	3.404E-02	2.639E-02	2.569E-02	9.925E-01	9.929E-01
	Max	1.248E-03	1.180E-03	3.533E-02	3.436E-02	2.667E-02	2.609E-02	9.926E-01	9.932E-01
	Std	4.002E-06	5.727E-06	5.680E-05	8.376E-05	7.153E-05	1.087E-04	2.693E-05	6.738E-05
GWO-LSTM	Mean	1.586E-03	1.495E-03	3.983E-02	3.866E-02	2.944E-02	2.830E-02	9.912E-01	9.911E-01
	Min	1.404E-03	1.403E-03	3.747E-02	3.745E-02	2.932E-02	2.815E-02	9.909E-01	9.904E-01
	Max	1.631E-03	1.549E-03	4.038E-02	3.936E-02	2.977E-02	2.847E-02	9.913E-01	9.914E-01
	Std	2.198E-03	1.154E-03	4.688E-02	3.397E-02	1.908E-03	1.216E-04	1.309E-04	3.870E-04
SaDE-LSTM	Mean	1.167E-03	1.133E-03	3.414E-02	3.365E-02	2.542E-02	2.504E-02	9.931E-01	9.935E-01
	Min	1.006E-03	1.016E-03	3.171E-02	3.187E-02	2.320E-02	2.299E-02	9.928E-01	9.930E-01
	Max	1.244E-03	1.210E-03	3.528E-02	3.478E-02	2.650E-02	2.649E-02	9.936E-01	9.941E-01
	Std	8.014E-05	6.429E-05	1.192E-03	9.620E-04	8.965E-04	8.936E-04	2.257E-04	4.043E-04

TABLE 6.3: Performance indices of forecasting wind turbine power output achieved by different models for one-hour ahead.

		MSE		RMSE		MAE		R	
Model		**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
ANFIS	Mean	6.875E-03	6.789E-03	8.284E-02	8.229E-02	5.319E-02	5.289E-02	9.608E-01	9.610E-01
	Min	6.208E-03	6.013E-03	7.879E-02	7.754E-02	5.071E-02	4.986E-02	9.585E-01	9.575E-01
	Max	7.617E-03	7.905E-03	8.728E-02	8.891E-02	5.620E-02	5.711E-02	9.634E-01	9.636E-01
	Std	6.226E-04	7.344E-04	3.758E-03	4.421E-03	2.348E-03	2.851E-03	2.318E-03	2.314E-03
LSTM-grid	Mean	1.906E-03	1.909E-03	4.366E-02	2.089E-01	3.046E-02	3.053E-02	9.903E-01	9.905E-01
	Min	1.880E-03	1.827E-03	4.336E-02	2.082E-01	3.036E-02	3.024E-02	9.902E-01	9.899E-01
	Max	1.923E-03	2.004E-03	4.386E-02	2.094E-01	3.055E-02	3.083E-02	9.905E-01	9.909E-01
	Std	1.968E-05	8.082E-05	2.458E-04	1.004E-03	6.857E-05	2.415E-04	1.238E-04	4.684E-04
CMAES-LSTM	Mean	1.634E-03	1.547E-03	4.042E-02	3.933E-02	3.061E-02	3.033E-02	9.902E-01	9.907E-01
	Min	1.608E-03	1.520E-03	4.010E-02	3.898E-02	3.041E-02	3.000E-02	9.901E-01	9.905E-01
	Max	1.652E-03	1.589E-03	4.065E-02	3.987E-02	3.084E-02	3.066E-02	9.903E-01	9.910E-01
	Std	1.208E-05	2.141E-05	1.495E-04	2.717E-04	1.180E-04	1.854E-04	6.810E-05	1.496E-04
DE-LSTM	Mean	1.645E-03	1.483E-03	4.056E-02	3.851E-02	3.064E-02	3.005E-02	9.901E-01	9.911E-01
	Min	1.636E-03	1.467E-03	4.045E-02	3.830E-02	3.050E-02	2.963E-02	9.901E-01	9.909E-01
	Max	1.655E-03	1.519E-03	4.068E-02	3.897E-02	3.085E-02	3.026E-02	9.902E-01	9.913E-01
	Std	6.622E-06	1.504E-05	8.163E-05	1.948E-04	1.043E-04	1.679E-04	3.803E-05	1.328E-04
GWO-LSTM	Mean	1.981E-03	1.999E-03	4.451E-02	4.471E-02	3.050E-02	3.053E-02	9.887E-01	9.887E-01
	Min	1.870E-03	1.887E-03	4.324E-02	4.344E-02	3.037E-02	3.034E-02	9.882E-01	9.882E-01
	Max	1.202E-03	1.904E-03	3.467E-02	4.364E-02	3.055E-02	3.097E-02	9.890E-01	9.894E-01
	Std	2.675E-05	2.082E-05	5.172E-03	4.563E-03	6.857E-05	4.415E-04	2.948E-04	4.635E-04
SaDE-LSTM	Mean	1.431E-03	1.413E-03	3.779E-02	3.755E-02	2.884E-02	2.833E-02	9.919E-01	9.921E-01
	Min	1.242E-03	1.237E-03	3.525E-02	3.517E-02	2.761E-02	2.757E-02	9.911E-01	9.912E-01
	Max	1.636E-03	1.632E-03	4.045E-02	4.040E-02	2.968E-02	2.896E-02	9.932E-01	9.931E-01
	Std	1.457E-04	1.394E-04	1.923E-03	1.850E-03	6.034E-04	4.214E-04	6.733E-04	6.051E-04

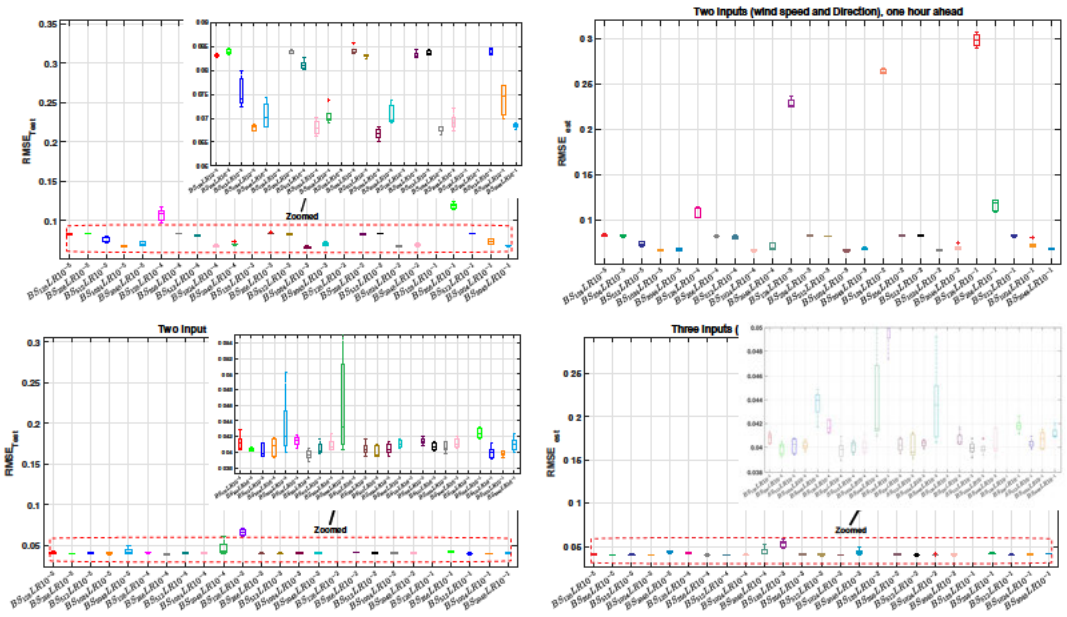

FIGURE 6.11: A comparison of four forecasting LSTM network models performance with various Hyper-parameters for forecasting the power output in one-hour ahead (Layer number=1, neuron number=100, Optimiser='Adam') .(a) the RMSE test-set with one input (wind speed) (b) the RMSE test-set with two inputs (wind speed and direction), (c) the RMSE test-set with two inputs (wind speed and current power), (d) the RMSE test-set with three inputs (wind speed, direction and current power).

zoomed versions of Figure 6.14(b, c and d), it can be seen that the SaDE-LSTM estimates the power output with considerable accuracy compared with other models.

6.9 Conclusions

Due to multiple systems and meterological factors, wind power time series data exhibit chaotic behaviours which are hard to predict. In this paper, a combination of autoencoder and clustering was adopted to reduce the stochastic noise inherent in raw data series. Subsequently a neuro-evolutionary approach (SaDE-LSTM) consisting of the self-adaptive differential evolution (SaDE) and LSTM network was used for modeling the wind behavior. We then conducted extensive experiments and compared our proposed approach with five alternative hybrid models. As our experiments suggests, the proposed SaDE-LSTM model outperforms its counterparts in terms of four performance criteria, in both ten-minute and one-hour intervals.

In the future, a greater variety of power curve datasets derived from different kinds of wind turbines from different regions will be evaluated to further enhance our model. Ultimately, another investigation of this study is to employ various outlier detection methods and optimization approaches to improve the forecasting results.

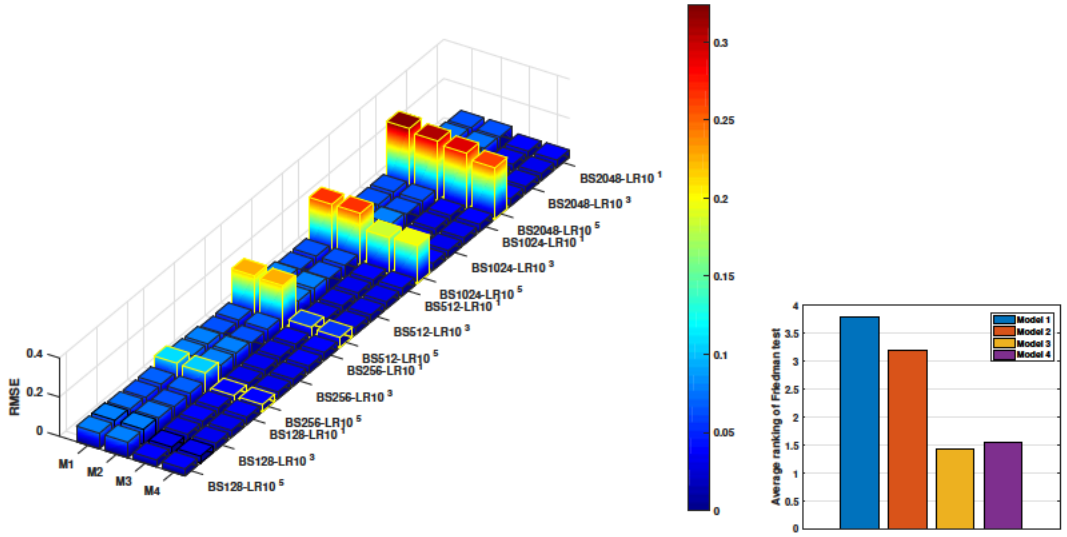

FIGURE 6.12: A comparison of four proposed forecasting models with 25 different configurations of hyperparameters. (a) A comparison of various LSTM settings based on RMSE. (b) The average ranking of the Friedman test for four applied models.

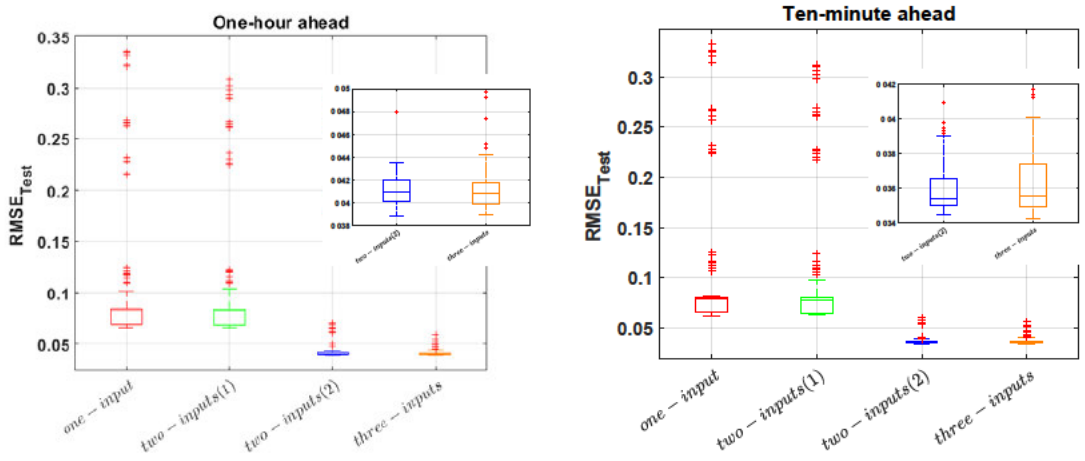

FIGURE 6.13: The total performance comparison of four LSTM forecasting models with ten-minute and one-hour ahead prediction. two-input(1) is the wind speed and direction, two-input(2) mentions the wind speed and current power of wind turbine.

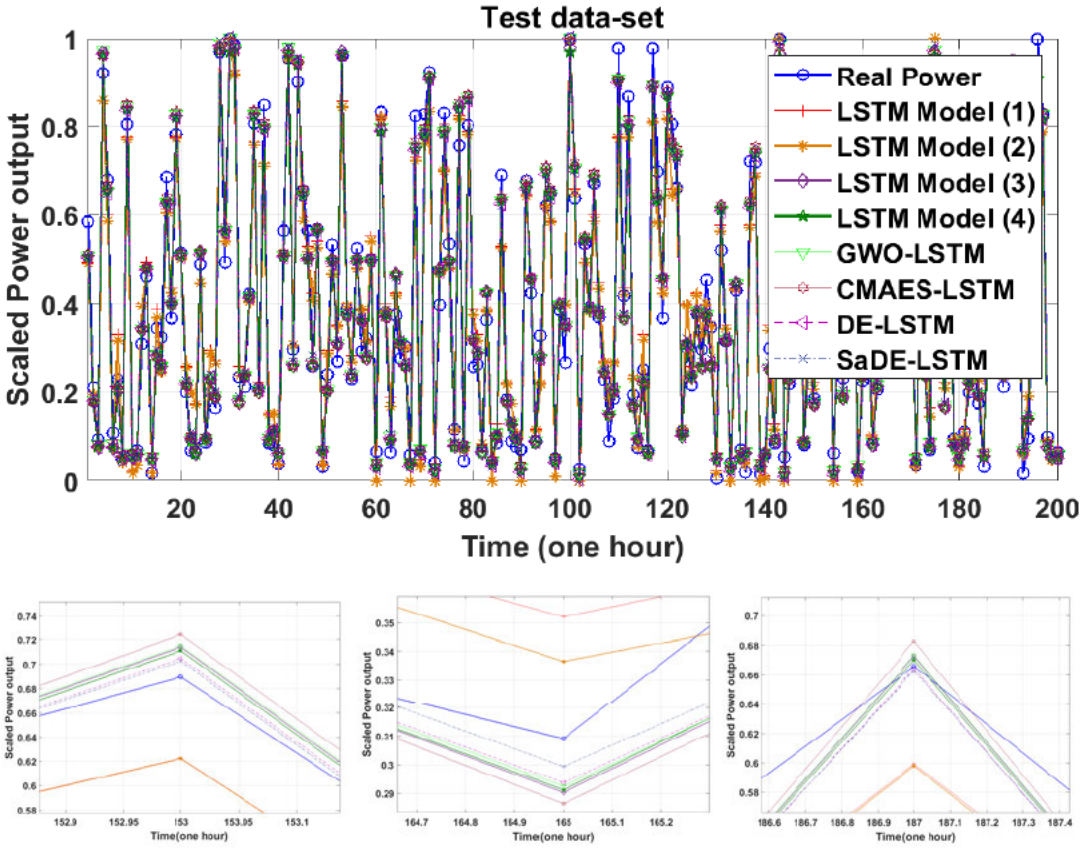

FIGURE 6.14: The best estimated power output values from the proposed hybrid models and the corresponding measured values in SCADA system. The initial values of the weights are kept the same.

Part III

Water Distribution Networks (WDNs) Design Optimisation

Chapter 7

Covariance Matrix Adaptation Greedy Search Applied to Water Distribution System optimisation

7.1 Synopsis

The article in this chapter proposes a new hybrid meta-heuristic framework for minimising the design cost of the real Water distribution systems (WDSs). The WDSs optimisation problem is challenging because of its non-linear nature, the large number of pipes, its use of discrete pipe sizes, its reliance on a multi-modal search space and dynamic constraints. In order to tackle these challenges, we design a hybrid optimisation method consisting of three parts. This method includes a self-adaptive EA called CMA-ES, a new upward greedy search algorithm for amending violations of the nodal pressure head constraints and a downward greedy search designed to decrease pipe sizes that might be larger than needed. The performance of the proposed hybrid framework is evaluated on five WDS cases studies of varying size and compared with other heuristic methods. The results show that our proposed method performs best overall in terms of computational efficiency and has a competitive ability to find near-optimal solutions.

Reference

[8] Neshat, M., Alexander, B., & Simpson, A. (2019). Covariance Matrix Adaptation Greedy Search Applied to Water Distribution System optimisation. arXiv preprint arXiv:1909.04846. Submitted for publication to Engineering Applications of Artificial Intelligence – journal. on [5 April 2020].

Statement of Authorship

Title of Paper	Covariance Matrix Adaptation Greedy Search Applied to Water Distribution System Optimization
Publication Status	☐ Published ☐ Accepted for Publication ☑ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Neshat, M., Alexander, B., & Simpson, A. (2019). Covariance Matrix Adaptation Greedy Search Applied to Water Distribution System Optimization. arXiv preprint arXiv:1909.04846. Submitted for publication to Engineering Applications of Artificial Intelligence – journal. On [5 April 2020].

Principal Author

Name of Principal Author (Candidate)	Mehdi Neshat		
Contribution to the Paper	Came up with the idea, read the existing articles, implemented the ideas to confirm its efficiency, wrote the first draft and applied comments from Co-authors.		
Overall percentage (%).	80%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	05/05/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Bradley Alexander (10%)		
Contribution to the Paper	Helped with methodology. Feedback on the experimental design and advised on refinements of the search processes and helped with revisions of paper.		
Signature		Date	12/05/20

Name of Co-Author	Angus R. Simpson (10%)		
Contribution to the Paper	Supervised development of the work, read the paper, provided comments and editing the paper.		
Signature		Date	22/05/2020

7.2 Abstract

Water distribution system design is a challenging optimisation problem with a high number of search dimensions and constraints. As a result, Evolutionary Algorithms (EAs) have been widely applied to optimise WDS to minimise the cost subject whilst meeting pressure constraints. This paper proposes a new hybrid evolutionary framework that consists of three distinct phases. The first phase applied the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES), a robust adaptive meta-heuristic for continuous optimisation. This is followed by an upward-greedy search phase to remove pressure violations. Finally, a downward greedy search phase is used to reduce oversized pipes.

To assess the effectiveness of the hybrid method, it has been applied to five well-known WDSs case studies. The results reveal that the new framework outperforms CMA-ES by itself and other previously applied heuristics on most benchmarks in terms of both optimisation speed and network cost.

7.3 Introduction

Water distribution systems (WDSs) are expensive to construct [63] and difficult to modify once in place. Careful design of WDS systems can lead to significant cost savings. As a consequence, optimisation of WDS design is a long-standing topic for research. WDS design can be framed as an optimisation problem of searching for an assignment of pipe diameters in the WDS that minimises the cost of construction subject to given constraints in the pressure head at each node in the system. This optimisation problem is challenging because relationships between individual pipe sizes and pressure at nodes are non-linear. In addition, the search landscape, for non-trivial systems is multi-modal with many local minima in the cost-function. Moreover, problem constraints are quite complex, with detailed requirements for pressures head and, commonly, restricts pipe sizes to discrete values.

During the last two decades, a wide variety of Evolutionary Algorithms (EAs) have been successfully applied to the problem of optimising the design of WDSs. EAs offer flexibility in their design parameters and work robustly in multi-modal, nonlinear and non-convex fitness spaces compared with traditional optimisation methods such as linear programming [270] and nonlinear programming [271]. In early work with EAs good results have been achieved by standard genetic algorithms (GA) [68, 272, 273]. Later work [274] modified a standard GAs with an additional heuristic-based, local representative cellular automata method to present a proper initial population for GA runs.

Another relative study developed a generic GA with an added heuristic selection phase (the Prescreened Heuristic Sampling Method (PHSM)) for initializing the population of the GA [275]; Other EAs applied include Ant Colony optimisation (ACO)[77, 276] and modified versions of ACO including the Max-Min Ant Systems [79, 80] and adaptive-convergence-trajectory ACO [277]. Other heuristic approaches have included, Simulated Annealing (SA) [70], Scatter Search [278], the Shuffled Frog Leaping algorithm [74], standard Particle Swarm optimisation (PSO) and its adaptive versions [76, 279, 280]; and also heuristics embedding Differential Evolution (DE) [83, 85, 281]. Still other approaches have combined heuristics including: DE and linear-programming [282]; DE and non-linear-programming [16]; and PSO and DE [283].

The best-performing frameworks described above, are quite varied in approach. Zheng et. al.[281] obtained good performance on the moderately sized Hanoi benchmark using standard Differential-Evolution (SDE). Sedkai et al. [283] obtained slightly better results on the Hanoi and NYTP benchmarks by interleaving Particle Swarm optimisation and DE (PSO-DE). In this work the addition of PSO appears to enhance global search capabilities.

Zheng et al. [16, 282] obtained strong results by partitioning the WDS network topology into trees, which do not contain loops, and the core, which does contain loops. Trees are relatively easy to optimise through convex search operations such as binary linear programming (BLP)[282] and non-linear-programming (NLP)[16]. In both approaches the optimisation of the looped network core is solved using DE. The resulting algorithms BLP-DE and NLP-DE benefit from the smaller problem size provided by problem decomposition and exhibit best-so-far performance on the Hanoi and Balerma (BN) [284] benchmarks.

Good performance on existing benchmarks was also obtained through the use of adaptive search algorithms. Adaptive algorithms contain logic to adjust search hyper-parameters in response to fitness distributions obtained during search. Tolson et al. [285] developed the Hybrid Discrete Dynamically Dimensioned search algorithm (HD-DDS) which specialised the adaptive DDS algorithm [286] to the problem WDS design. Later Zheng et. al. [85] developed a self-adaptive differential evolution (SADE) algorithm which adjusted DE hyper-parameters in response to search progress. Both of these frameworks obtained results comparable to the best frameworks on their tested benchmarks.

While the published frameworks to date have used a wide variety of popular meta-heuristic search frameworks there is currently no work that applies one of the most effective stochastic optimisers: CMA-ES [205] to the problem designing a WDS network.

TABLE 7.1: Software availability

Name of the Software:	WDSOP
Version:	1.00
Available from:	https://github.com/a1708192/WDSOP
Language:	Matlab
Supported System:	Windows, MacOS, Linux, Unix
Year first available:	2019

CMA-ES is a self-adaptive, global search algorithm designed for searching spaces consisting of many continuous variables. CMA-ES is one of the fastest and versatile meta-heuristic search algorithms. Variants of CMA-ES often lead black-box optimisation competitions and also it has been shown to perform well in noisy, non-convex and non-separable search spaces [287].

Within the field of water management CMA-ES has been applied a diverse range of settings. Burger and Bayer et. al. [288, 289] applied CMA-ES to the problem of optimising ground water remediation and extraction. Belaqziz et al. [290] used CMA-ES to improve pumping schedules in an irrigation project. This approach was improved by Ikudayisi et al. [291] to take account of plant water-stress levels in the objective function. Similarly, Grundmann et al. [292] used CMA-ES to optimise a groundwater irrigation system for both profit and sustainability. More recently, Romero et al. [293] applied CMA-ES to optimising the location of leak-detecting sensors in a WDS network. Maier et al. [294] used CMA-ES to calibrate model parameters for vertical water flows in wetlands used for water cleaning.

However, to date, there has been no published work applying CMA-ES to the problem of optimising pipe diameters. This paper addresses this gap by deploying CMA-ES in pure and hybrid form to a range of small and large scale WDS benchmark networks and performing a systematic comparison with previous results. The hybrid search methods combine CMA-ES with novel greedy search heuristics for refining solutions found by CMA-ES. We show that CMA-ES, when combined with greedy search has performance comparable to the current best search heuristics in terms of execution time and network costs. This work also, for the first time explores, as a baseline, the performance of simple randomised local search (RLS), and a 1+1 Evolutionary Algorithm (1+1 EA).

The software code used to produce the results in this paper, called WDSOP, has been made available. Details of the distribution used to produce the results in this paper are shown in Table 7.1. WDSOP is written in the MATLAB programming language and has an, easily extensible, modular design. WDSOP is set up to run on the five benchmark networks considered in this paper.

The primary inputs for running the framework are the water distribution network name and the selected optimisation method. Other control parameters are selected based on the optimisation process.

The organization of the rest of this paper is as follows. Section 2 outlines our methodologies. Section 3 presents the details of the case studies used in this research. The experimental results are presented and analysed in section 4. Finally, section 5 presents our conclusions and directions for future work.

7.4 Methodology

This section outlines the search algorithms used in this work.

7.4.1 Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

CMA-ES is a high-performance adaptive heuristic search meta-heuristic designed to search continuous, multi-dimensional search spaces. CMA-ES is a highly-refined variant of Evolutionary-Strategies (ES) [88]. Evolutionary Strategies are evolutionary search meta-heuristics that rely solely on mutation operators to change individuals in their populations. CMA-ES searches for solutions in a continuous n dimensional space by cyclically updating a population of m individual n-dimensional solution vectors. In each cycle (generation): g these vectors are randomly sampled from an n-dimensional Gaussian distribution $\mathcal{N}(\mathbf{m}^{(g)}, \mathbf{C}^{(g)})$ where $\mathbf{m}^{(g)}$ is the current mean of the distribution and $\mathbf{C}^{(g)} \in \mathbb{R}^{n \times n}$ is the covariance matrix describing the current shape and orientation of the sampling distribution. The distribution defined by $\mathbf{m}^{(g)}$ and $\mathbf{C}^{(g)}$ biases the sampling process towards a particular region of the search space. After a population of m sampled solutions is made they are evaluated according to a fitness function $f \in \mathbb{R}^n \to \mathbb{R}$ and the distribution of solutions' fitness within the search space is used to update $\mathbf{m}$ and $\mathbf{C}$ for generation $g + 1$'s round of sampling [231].

This action of adaptively biasing the location, shape and orientation of sampling process allows search to proceed to where it is currently most productive. CMA-ES also has built-in mechanisms that enlarge the search space if the search stagnates quickly, thus providing some capability to search multi-modal spaces. The initial value of $\mathbf{m}^{(1)}$ is set by the user and $\mathbf{C}^{(1)} = \sigma^2\mathbf{I}$ where σ is a starting standard deviation for the sampling distribution, also provided by the user. Note that a small value for σ, relative to the solution space, will lead to fast convergence while a large value for σ will favour a more thorough global search – at the cost of speed of convergence.

In this work we favour exploration at first so σ is set to be half the range of possible pipe sizes. The starting vector $\mathbf{m}^{(1)}$ is a vector consisting of randomly generated pipe diameters within the range of valid pipe diameters for the problem. CMA-ES can be configured to abide by given constraints on $\mathbf{m}$ in this work we constrain $L \geq \mathbf{m}_i \geq U$ where L is the minimum (lower-bound) allowable pipe diameter for the given design problem and U is the maximum (upper-bound) pipe diameter allowable for the design problem. As previously mentioned, CMA-ES is a continuous optimisation framework and we use it directly for our experiments for continuous pipe sizing. However, most network design problems select pipe diameters from a set of discrete of sizes. For the design of the core elements of large diameter networks this set usually contains pipe diameter at $300mm$ intervals starting from zero. In this work, for discrete design problems we round pipe-diameters up to the nearest available discrete diameter.

7.4.2 Greedy Search

The pipe diameters produced by rounding the results of CMA-ES search are not guaranteed to deliver the required minimum allowable pressure heads for each problem. Moreover, in some cases, there is scope in the CMA-ES solutions for reduction in the sizes of some pipes. To address these issues we propose a greedy search solution that follows the application of the CMA-ES search described in section 7.4.1 above. The first stage of the greedy search solution is an upward greedy search (GS_U) algorithm to round *up* pipe diameters to produce a feasible solution. This is followed by an analogous downward greedy search stage (GS_D) that, where feasible, rounds *down* pipe diameters. The combined hybrid search, is called CMA_{ES}-GS_U-GS_D.

The upward greedy search works by speculatively incrementing the diameter of each pipe in the network in turn and checking its impact on both pressure violation and cost. The pipe whose upward increase of diameter size reduces the pressure violation per-unit cost is selected to be enlarged by one discrete size. The GS_U enhances the infeasibility amount of the CMA-ES achieved solutions and pushes up the infeasible layouts toward the feasible area by increasing the discrete size of pipe diameters based on a greedy selection of those solutions with the largest reduction in the sum of pressure violations for the least cost. The maximisation problem can be stated mathematically as:

$$\begin{aligned} Argmax \rightarrow f(\Theta) = (\frac{\sum_{i=1}^{M} \triangle PV_i}{\sum_{j=1}^{N} \triangle PipeCost_j}) \\ \forall i = \{1, ..., M\}/j = \{1, ..., N\} \\ Subject - to : \triangle PV_i \leq 0, \triangle PipeCost_j \geq 0 \end{aligned} \tag{7.1}$$

Algorithm 7.1 Upward Greedy Search (Fixing up the nodal pressure head violations)

procedure THE UPWARD GREEDY SEARCH
Initialization
  Initialize τ ▷ initialize Pressure head constraint
  $Layout_{iter}$=Pipe-Network ▷ Read Network data
  $Nodal_{Pressure}$=$Eval(Layout_{iter})$ ▷ Evaluate Network by EPANET
  $N = Size(Layout_{iter})$ *and* $M = Size(Nodal_{Pressure})$
  $Sum_{PV} = \sum_{i=1}^{M}(|\tau - Nodal^{i}_{Pressure}|) \; \forall \; Nodal^{i}_{Pressure} < \tau$ ▷ Calculate sum of the nodal pressure violation
  while $Sum_{PV} > 0$ **do**
    $NetPipe = Layout_{iter}$
    while $i \leq N$ **do**
      Increase the i^{th}Pipe diameter of the $Netpipe_{iter}$ based on the possible diameters
      $Nodal_{Pressure}$=$Eval(Netpipe_{iter})$
      Update Sum_{PV}
      $Improvement^{i}_{rate} = \frac{\sum_{i=1}^{M} \triangle PV_i}{\sum_{i=1}^{N} \triangle PipeCost_i}$
    end while
    $Layout_{iter}$=Max($Improvement_{rate}$) ▷ Choose the best design based on Improvement rate
  end while
end procedure

where a layout Θ can be defined as a set of sequential pipe diameters, so that $\Theta = \{D_1, D_2, ..., D_N\}$ where N is the number of pipes and M is the number of network nodes. $f(\Theta)$ describes the feasibility of the network function, which should be maximised by the GS_U. This greedy heuristic search method is able to guarantee to produce a feasible design based on the constraints, and also it will yield the locally optimal design in a reasonable runtime. The procedure of the GS_U is shown in the Algorithm 7.1.

Despite all positive points of GS_U, sometimes its proposed solutions require improvements because of the greedy selection behaviours without looking at the future or past situations. Therefore, the third phase is proposed to reduce the extra cost of the some of the obtained solutions. This part is made up of another Greedy Search idea.

The idea of the hybrid framework third part is a Downward Greedy Search (GS_D)(Algorithm 7.2). The main purpose of the GS_D is smoothing the pipe cost with respect to the constraints by decreasing the diameter of the pipes one by one. In other words, GS_D is looking for improvements that give us the least reduction in pressure violations for the most significant reduction in the pipe cost. The purpose is maximising

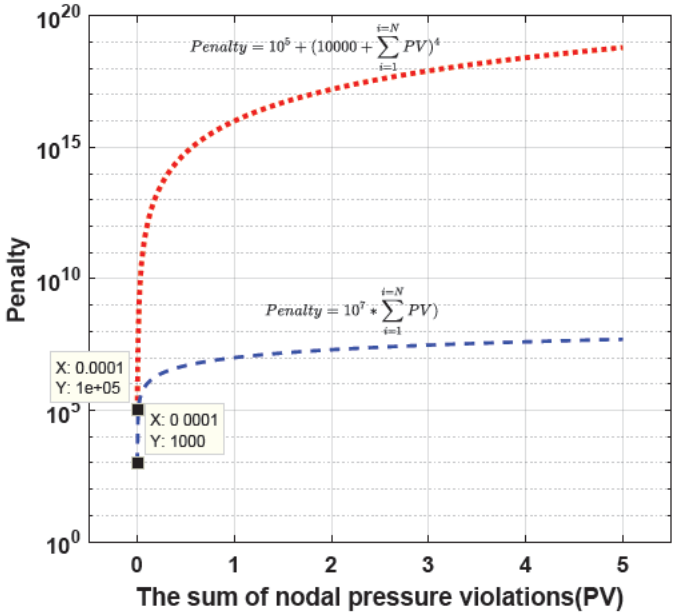

FIGURE 7.1: Two applied penalty functions for handling the pressure violations. The red line is a severe penalty function.

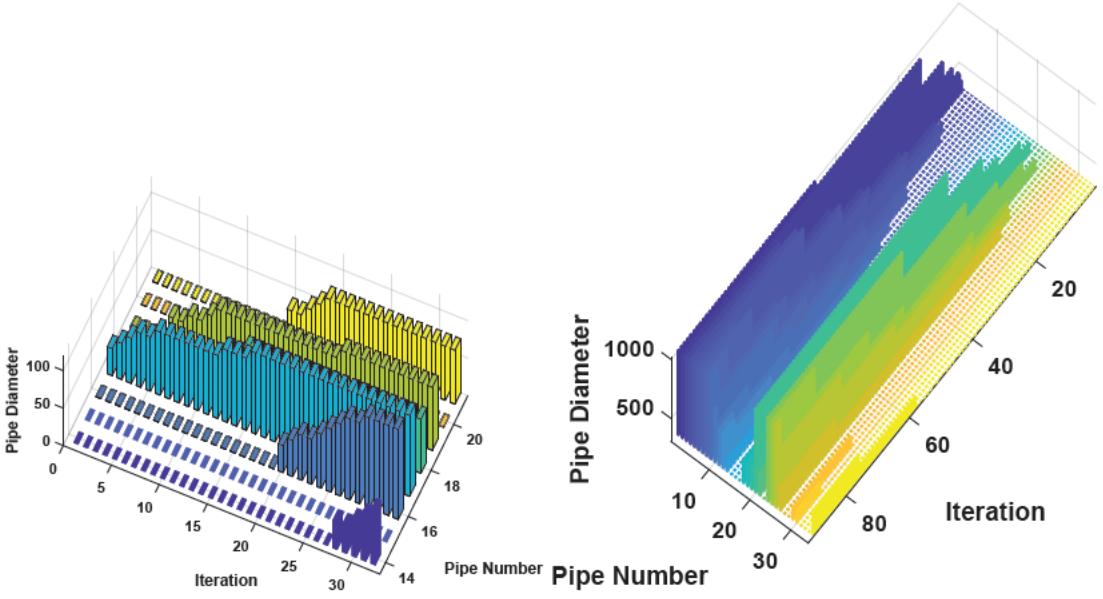

FIGURE 7.2: (a) A 3D bar chart of the greedy search performance when all pipes diameters (NYTP) are equal to Zero.(b) for optimising the Hanoi network when all pipes diameters are initialized to 304.8(mm), maximum number of evaluation is 3094 and the proposed layout cost=$6,312,405.

according to the Equation 7.2.

$$Argmax \rightarrow f(\Theta) = \left(\frac{\sum_{j=1}^{N} \triangle PipeCost_j}{\sum_{i=1}^{M} \triangle PV_i}\right) \qquad (7.2)$$
$$Subject - to : \triangle PV_i \geq 0, \triangle PipeCost_i \geq 0$$

Practically, GS_D is so fast and effective to attain the near best lowest cost solution.

Algorithm 7.2 Downward Greedy Search (minimising the network cost)

procedure THE DOWNWARD GREEDY SEARCH (GS_D)
Initialization
    Initialize τ ▷ initialize Pressure head constraint
    $NetPipe_{iter}$=Pipe-Network ▷ Read Network data
    $Nodal_{Pressure}$=$Eval(NetPipe_{iter})$ ▷ Evaluate Network by EPANET
    $N = Size(NetPipe_{iter})$ *and* $M = Size(Nodal_{Pressure})$
    $Sum_{PV} = \sum_{i=1}^{M}(|\tau - Nodal^i_{Pressure}|) \; \forall \; Nodal^i_{Pressure} < \tau$ ▷ Calculate sum of the nodal pressure violation
    while $Sum_{PV} \geq 0$ **do**
        while $i \leq N$ **do**
            Decrease the i^{th}Pipe diameter of the $Netpipe_{iter}$ based on the possible diameters
            $Nodal_{Pressure}$=$Eval(Netpipe_{iter})$
            Update Sum_{PV}
            if $Sum_{PV} = 0$ **then**
                Add the $NetPipe_{iter}$ to feasible solution set
                $Improvement^i_{rate} = \frac{\sum_{i=1}^{N} \triangle PipeCost_i}{\sum_{i=1}^{M} \triangle PV_i}$
            end if
        end while
        $NetPipe_{iter}$=Max($Improvement_{rate}$) ▷ Select the best feasible solution
    end while
    return $NetPipe_{iter}$
end procedure

7.4.3 Randomized Local Search(RLS) and 1+1EA

Randomized Local Search (RLS) is the simplest single-based solution EAs. According to the practical results, sometimes applying the simple EAs can be more efficient than complicated approaches and also RLS can be a proper choice when the fitness function is a combinatorial optimisation problem [295]. RLS begins with a candidate solution (x) and provides in each iteration a new solution (y) by flipping one chosen variable of x randomly. In the standard version of RLS, the mutation is done by a uniform distribution which leads to a non-curved and noisy local search, but we prefer to use a normally distributed mutation. The advantage of RLS is that in each iteration, just one pipe size of the network is changed. This attribute leads to approaching a near-optimal solution step by step; however, it can be so costly for a large-scale network. In the following, the pseudo-code of RLS can be seen by the Algorithm 7.4 where UB and LB are the upper and lower bound of the variable, and also n is the number of variables.

Undoubtedly, after RLS, the most simple evolutionary algorithm is (1+1)EA because there is just one solution in each iteration and a standard bit mutation applies for providing a new solution with mutation probability $\frac{1}{N}$ that N is the number of variables. Its benefits include simplicity and performance which makes (1+1)EA one of the most

Algorithm 7.3 CMA_{ES}-GS_U-GS_D

procedure THE CMA_{ES}-GS_U-GS_D
Initialization
    Initialize the CMA-ES Parameters
    λ ▷ Offspring population size
    μ ▷ Parent population size ($floor(\lambda/2)$)
    σ_{start} ▷ Initial standard deviation($0.5 \times (UB - LB)$)
    c_c ▷ Covariance learning rate
    while $FunctionTolerance \leq \xi$ **do**
        Update the Covariance Matrix $C^{(g+1)}$
        Update the mutation step size σ^g
        Generate sample population for next generation (g+1)
        $x_k^{(g+1)} \sim \mathcal{N}(m^{(g)}, (\sigma^{(g)})^2 C^{(g)}) \forall k = 1, ..., \lambda$
        if ($x_k^{(g+1)}$ is not feasible) **then**
            Impose the penalty
        end if
        Update the mean for next generation (g+1)
        Update best ever solution ($Best_{Solution}$)
        $Possible_{Solution}$=round($Best_{Solution}$) ▷ Convert continuous design to possible pipe size
        Sum_{PV}=Sum-Violation($Possible_{Solution}$) ▷ Compute sum of the nodal pressure violation
        if ($f(Possible_{Solution}) < \phi$ & $Sum_{PV} > 0$)) **then**
            Apply the Upward Greedy Search (GS_U) ▷ Fix the violation of nodal pressure heads
        end if
    end while
    Apply the Downward Greedy Search(GS_D)
end procedure

Algorithm 7.4 Randomized Local Search

procedure THE RLS
Initialization
    LB=Min(Diameters);UB= Max(Diameters)
    $X_{iter} \in \{LB, UB\}^N$ uniformly at random ▷ Generate first feasible design
    while Stopping Criteria **do**
Mutation
        Create $Y_{iter} = X_{iter}$ independently for each $i \in \{1, 2, ..., N\}$
        $Y_{iter} = N(\mu, \sigma = 0.5 * (UB - LB))$ ▷ Mutate one random variable of $Y_{iter}$by normally distributed random
Selection
        if ($f(Y_{iter}) \leq f(X_{iter})$) **then**
            $X_{iter+1} = Y_{iter}$
        else
            $X_{iter+1} = X_{iter}$
        end if
    end while
end procedure

attractive EAs which can often be generalized and extended to more complex EAs. As the performance of (1+1)EA can be better than complex EAs like CMA-ES in some cases [1]. in this investigation, a fine-tuned mutation step size of (1+1)EA version is implemented for optimising WDSs and for analyzing its pros and cons. Besides, since there is the probability that in one mutation, none of the variables is mutated, a substitute mutation strategy is considered, which is flipping the size of one pipe randomly at least. The pseudo-code of (1+1)EA is presented in Algorithm 7.5.

Algorithm 7.5 (1+1)EA

procedure THE (1+1)EA
Initialization
    LB=Min(Diameters);UB= Max(Diameters)
    $X_{iter} \in \{LB, UB\}^N$ uniformly at random ▷ Generate first feasible design
    while Stopping Criteria **do**
Mutation
        Create $Y_{iter} = X_{iter}$ independently for each $i \in \{1, 2, ..., N\}$
        Mutate each variable of $Y_{iter}(i)$ with probability $\frac{1}{N}$ (Normal random distribution: $\sigma = C$)
        if Mutation Number= 0 **then**
            Mutate one variable of Y_{iter} randomly
        end if
Selection
        if $(f(Y_{iter}) \leq f(X_{iter}))$ **then**
            $X_{iter+1} = Y_{iter}$
        else
            $X_{iter+1} = X_{iter}$
        end if
    end while
end procedure

7.5 WDS design formulation and constraints

In this research, the main optimisation objective is minimising the cost of the pipes of the WDS network whilst satisfying given pressure constraints. The WDSs optimisation problem is a combinatorial optimisation problem which can be defined as searching for the best network design parameters, whilst minimising costs and being subject to constraints such as minimum nodal pressure.

The optimisation problem is defined mathematically as:

$$Argmin \rightarrow C_{pipe}(\Theta) = \sum_{i=1}^{N} (c_i \times D_i) \times L_i \tag{7.3}$$

$Subject - to:$
$(1): H_j \geq H_j^{min}, \quad \forall \quad j = 1, ..., M$
$(2): D_k \in \{D\}, \quad \forall \quad k \in N$
where C_{pipe} is the primary fitness function including the layout cost and also L_i and D_i are the i^{th}pipe length and diameter respectively; c_i is the cost per unit length of pipe in the network. The above cost function is constrained to the following.

Firstly, it is the minimum nodal pressure head constraint that should be imposed for all nodes of the network. The H_j indicates the pressure head level for the j^{th} node, and the minimum needed j^{th} nodal pressure head based on the demand pattern is shown as H_j^{min}. If this constraint is not satisfied the sum of nodal head pressure violation will be computed (Equation 7.4).

$$Sum_{PV} = \begin{cases} 0 & if(H_j \geq H_j^{min}) \\ \sum_{j=1}^{M}(H_j^{min} - H_j) & otherwise \end{cases} \tag{7.4}$$

The second constraint is the possibility of the discrete pipe sizes, which are usually defined commercially. Thus, if the diameter of pipes is not included from the discrete sizes set, the fitness function must be penalized by the Equation 7.5 where D_k is the diameter of pipe i that is chosen from a deterministic set (D).

$$Sum_{DV} = \sum_{i=1}^{N} \begin{cases} 0 & D_i = D_a || D_b \\ 1 & D_i = (D_a + D_b)/2 \\ \frac{D_i - D_a}{((D_a+D_b)/2)-D_a} & D_i < (D_a + D_b)/2 \\ \frac{D_b - D_i}{D_b-((D_a+D_b)/2)} & D_i > (D_a + D_b)/2 \end{cases} \tag{7.5}$$

where both D_a and D_b are sequential possible (standard) pipe diameters ($D_a < D_b$) and D_i is the size of pipe i which is determined from a set of possible diameters or considered as a continuous values. When the sum of the pressure head violation is not equal to zero, the constrained model is transformed into an unconstrained one by inserting the sum of constraint violations value into the fitness function as a penalty.

Optimal design and rehabilitation of a water distribution network is a constrained nonlinear optimization problem. A penalty function is usually applied to modify a constrained into a nonconstrained optimization problem within the framework. However, the penalty function requires tuning the penalty factor. Mainly due to its simplicity and ease of implementation, it is a common technique for dealing with the constraints. There are many different constraint handling approaches like Repair approaches, Separatist approaches (These approaches do not combine objective function and constraints, but handle them

separately) and Hybrid approaches. While all computations of the hydraulic simulation are done via EPANET 2.0. Consequently, the total cost is acknowledged as the sum of the pipe cost, a penalty cost of pressure head violation and the violation of discrete pipe diameters represented as:

$$\begin{aligned} C_{PV}(\Theta) &= (P_f \times Sum_{PV}) \\ C_{DV}(\Theta) &= (P_D \times Sum_{DV}) \\ minimise \rightarrow C_t(\Theta) &= C_{pipe}(\Theta) + C_{PV}(\Theta) + C_{DV}(\Theta) \end{aligned} \tag{7.6}$$

where C_t is the total cost of the penalized or non-penalized fitness function cost, and also both P_f and P_D are the penalty factors. For instance, Algorithm 7.6 shows how the penalty of the continuous pipe diameters is computed for the NYTP and NYTP2 case studies.

Algorithm 7.6 Handling constraint violations for continuous pipe diameters of NYTP

procedure THE PENALTY FUNCTION
  if $rem(D_i, 12) \neq 0$ **then**
    if $D_i > 36$ **then**

$$Sum_{DV} = \sum_{i=1}^{N} \begin{cases} 0 & D_i = D_a || D_b \\ 1 & D_i = (D_a + D_b)/2 \\ \frac{D_i - D_a}{((D_a+D_b)/2) - D_a} & D_i < (D_a + D_b)/2 \\ \frac{D_b - D_i}{D_b - ((D_a+D_b)/2)} & D_i > (D_a + D_b)/2 \end{cases}$$

    else

$$Sum_{DV} = \sum_{i=1}^{N} \begin{cases} 0 & D_i = 0 | D_i = 36 \\ \frac{D_i}{18} * 3 & D_i < 18 \\ 3 & D_i = 18 \\ \frac{36 - D_i}{18} * 3 & D_i > 18 \end{cases}$$

    end if
  end if
end procedure

7.6 Case Studies Results and Discussions

For evaluating the proposed hybrid framework effectiveness, five well-known WDS case studies have been considered including the New York Tunnels Problem (NYTP) [67], the Doubled New York Tunnels Problem (NYTP2) [78], 50NYTP [296], the Hanoi Problem (HP) [271] moreover, one large-scale network called the Balerma Network (BN) [284]. The details of the case studies can be seen in Table 7.2.

TABLE 7.2: The characteristics of Case Studies Summery

WDS	No. of Decision Variables	No. of options	No. of Nodes	Search Space Size
NYTP	21	16	20	1.934×10^{25}
HP	34	6	32	2.865×10^{26}
NYTP2	42	16	20	3.741×10^{50}
BN	454	10	447	10^{454}
NYTP50	1050	16	1000	2.12×10^{1264}

7.6.1 Case Study 1: New York Tunnel problem (NYTP)

The NYTP layout is a fundamental benchmark of the water distribution system problem which is inspired by the real New York water network. The number of existing tunnels is 21 with 20 nodes supported by a fixed-head reservoir. The detailed information of NYTP provided by Dandy et al. [67] such as the cost of pipes, the nodal demand pattern and the pressure head constraints. The principal purpose is minimising the total pipe cost of the new installed parallel pipes, to reinforce the existing pipes. Meanwhile, the constraints should be met (minimum nodal pressure head).

In the NYTP, pipes diameter size can be allocated among 15 actual different sizes plus a zero size. Therefore, the search space size is 16^{21}. However, in this research, a continuous search space is considered as well. The benchmark networks have been optimised using different granularities of search space including continuous, discrete (interval=1 inch) and commercially available (interval= 12 inches) pipe sizes. With regard to assessing the ability of the proposed CMA_{ES}-GS_U-GS_D algorithm to achieve an appropriate balance between exploration and exploitation in the decision space a range of population sizes, as specified by the maximum iteration number, are considered for each case study network such as λ =10, 20, 50, 100, 200 and 400. Since the CMA-ES is a self-adaptive method, all control parameters have been adjusted during the optimisation process except σ. The σ value is initialized by the half of the decision variables length. Thus, the CMA-ES is started with the ability to explore the search space.

In the first step of the proposed hybrid framework, the CMA-ES efficiency is evaluated by three kinds of decision variables: continuous, discrete and possible. Where the continuous pipe sizes are used, the only nodal pressure head constraint should be satisfied, so the penalty factor (PF) for NYTP is 10^7. Moreover, a severe penalty factor is imposed too, which can be seen in the Algorithm 7.7. According to the achieved results, the performance of the severe PF is not competitive.

The best feasible configuration of the NYTP cost which is obtained by continuous CMA-ES is $38.00 million based on the nodal pressure head constraint by different population

sizes (The best-known NYTP cost is \$38.64 million [297]). Thus, CMA-ES can overcome all previous optimisation methods when continuous pipe sizes are considered. The proposed new feasible continuous designs of NYTP are shown in Table 7.3 and also a comprehensive review comparison of the previous best NYTP layouts including some infeasible networks after evaluating the nodal pressure head constraints are listed in Table 7.10.

In EAs, the convergence rate is another significant evolutionary parameter to reflect how fast the EA converges to the optimal solutions per generation. In this way, Figure 7.6 represents the average convergence rate of the proposed methods by 30 independent runs. We can see the CMA-ES with small population sizes has converged faster compared with the large population sizes. However, this configuration of search often prematurely converges to a local optimum.

Algorithm 7.7 Penalizing the pressure violations

procedure THE PENALTY FUNCTION FOR HANDLING THE PRESSURE VIOLATIONS(PV)
if $\sum_{i=1}^{i=N} PV_{(i)} > 0$ **then**
$Penalty_{(PV)} = 10^5 + (10^4 * \sum_{i=1}^{i=N} PV_{(i)})^4$
end if
end procedure

When the discrete interval is 1 inch in the discrete pipe size scenario, the performance of CMA-ES is better than the rounded (possible) pipe size. Figure 7.3 shows the comparison of the three strategies and the impact of large population sizes. As the CMA-ES is able to discover very low cost designs when the pipe sizes are discrete or rounded (possible), but they are not feasible in terms of nodal head constraints, the Upward Greedy search is applied to repair the designs.

For saving the computational budgets of the CMA-ES, the termination criterion is configured as a function tolerance value (ξ) at $1/10^5 \times C_t(\Theta)$. The GS_U will be begun if the network cost of the rounded pipes is less than ϕ (well-known or estimated network cost). The efficiency of GS_U is independently tested by the initialization of zero pipe sizes. The GS_U is a super fast search method and can find an NYTP layout at \$42.36 millions in just 714 evaluations (Figure 7.2). Substantially, the GS_U is able to repair the violation of nodal pressure head by increasing some of the pipe sizes. This combination of CMA-ES and GS_U leads to a robust and powerful framework that finding the optimal feasible solutions are guaranteed by the termination conditions of the greedy algorithm. The best results occurred when the CMA_{ES}-GS_U population size is 400 in these runs the best-known NYTP solution is reached 100% of the time. Figure 7.2 shows that the strong penalty factor does not let to keep good infeasible designs with a few violation in the population and it results in decreasing the diversity of the population in the initial

TABLE 7.3: The achieved best feasible continuous (C) and Discrete (D) NYTP design by CMA-ES. Next, the best solutions are rounded to possible pipe diameters (r), but rounded layouts have a negligible nodal pressure violation ($\sim$).

Pipe	**CMA-ES** (C,λ=**10**)	**CMA-ES** (C,λ=**20**)	**CMA-ES** (C,λ=**50**)	**CMA-ES** (C,λ=**100**)	**CMA-ES** (D,λ=**10**)	**CMA-ES** (D,λ=**20**)	**CMA-ES** (D,λ=**50**)	**CMA-ES** (D,λ=**100**)
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	119 06 $\rightarrow 120_r$	118 99 $\rightarrow 120_r$	118 98 $\rightarrow 120_r$	118 99 $\rightarrow 120_r$	0	118 12 $\rightarrow 120_r$	117 19 $\rightarrow 120_r$	122 09 $\rightarrow 120_r$
8	0	0	0	0	0	0	0	0
9	0	0	0	0	5 084 $\rightarrow 0.0_r$	0	0	0
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	0	6 895 $\rightarrow 0.0_r$	0	0	0
13	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0
15	0	0	0	0	120 01 $\rightarrow 120_r$	0	0	0
16	99 95 $\rightarrow 96_r$	99 97 $\rightarrow 96_r$	99 98 $\rightarrow 96_r$	99 97 $\rightarrow 96_r$	82 01 $\rightarrow 84_r$	100 14 $\rightarrow 108.0_r$	100 76 $\rightarrow 96.0_r$	99 06 $\rightarrow 96_r$
17	99 29 $\rightarrow 96_r$	99 28 $\rightarrow 96_r$	99 28 $\rightarrow 96_r$	99 28 $\rightarrow 96_r$	101 31 $\rightarrow 96.0_r$	99 97 $\rightarrow 96.0_r$	98 71 $\rightarrow 96.0_r$	99 31 $\rightarrow 96_r$
18	79 04 $\rightarrow 84_r$	79 08 $\rightarrow 84_r$	79 09 $\rightarrow 84_r$	79 08 $\rightarrow 84_r$	74 32 $\rightarrow 72.0_r$	78 86 $\rightarrow 84.0_r$	80 04 $\rightarrow 84.0_r$	79 01 $\rightarrow 84_r$
19	75 04 $\rightarrow 72_r$	75 07 $\rightarrow 72_r$	75 05$\rightarrow 72_r$	75 06 $\rightarrow 72_r$	68 14 $\rightarrow 72.0_r$	81 63 $\rightarrow 84.0_r$	75 96 $\rightarrow 72.0_r$	76 23 $\rightarrow 72_r$
20	0	0	0	0	0	0	0	0
21	70 61 $\rightarrow 72_r$	70 60 $\rightarrow 72_r$	70 61 $\rightarrow 72_r$	70 61 $\rightarrow 72_r$	73 65 $\rightarrow 72.0_r$	69 03 $\rightarrow 72.0_r$	70 74 $\rightarrow 72.0_r$	70 13 $\rightarrow 72_r$
Total Cost	**38.00** $\rightarrow 37.63_r^{\sim}$	**38.00** $\rightarrow 37.63_r^{\sim}$	**38.00** $\rightarrow 37.63_r^{\sim}$	**38.00** $\rightarrow 37.63_r^{\sim}$	**38.57** $\rightarrow 37.69_r^{\sim}$	**38.26** $\rightarrow 39.61_r^{\sim}$	**38.09** $\rightarrow 37.63_r^{\sim}$	**38.05**
16^{th}	1 42e-07 $\rightarrow$ -0 042	5 35e-07 $\rightarrow$ -0 042	2 49e-07 $\rightarrow$ -0 042	1 15e-07 $\rightarrow$ -0 042	0 08 $\rightarrow$ -0 255	0 16 $\rightarrow$ 0 68	0 073 $\rightarrow$ -0 042	0 025
17^{th}	1 13e-09 $\rightarrow$ -0 046	5 55e-10 $\rightarrow$ -0 046	1 04e-08 $\rightarrow$ -0 046	2 63e-08 $\rightarrow$ -0 046	0 003 $\rightarrow$ 0 047	8 20e-04 $\rightarrow$ 0 085	0 002 $\rightarrow$ -0 046	0 002
19^{th} Excess Nodal Pressure	1 47e-06 $\rightarrow$ -0 044	3 03e-07 $\rightarrow$ -0 044	3 04e-07 $\rightarrow$ -0 044	6 02e-07 $\rightarrow$ -0 044	0 036 $\rightarrow$ -1 38	0 09 $\rightarrow$ -0 0455	0 015$\rightarrow$ -0 044	0 008

TABLE 7.4: A comparison results of CMA-ES and CM$A_{ES} - GS_U$ (Hybrid method of CMA-ES and Upward Greedy search) for NYTP, All proposed networks by CM$A_{ES} - GS_U$ are feasible in terms of both pipe diameter and nodal pressure.

Pipe	**CMA-ES**	**CM$A_{ES} - GS_U$**	**CMA-ES**	**CM$A_{ES} - GS_U$**	**CMA-ES**	**CM$A_{ES} - GS_U$**	**CMA-ES**	**CM$A_{ES} - GS_U$**
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0→	**36**
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	120 →	**144**	144	144	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0→	**60**
15	0	0	0	0	132→	**144**	0	0
16	96	96	84→	**96**	72	72	108 →	**132**
17	96	96	96	96	96	96	96	96
18	84	84	84	84	72→	**84**	84	96
19	72	72	72	72	60→	**72**	60	60
20	0	0	0	0	0	0	0	0
21	72	72	72	72	72	72	84	84
Total Pipe Cost ($M)	**37.63**	**38.64**	**37.37**	**38.64**	**36.63**	**39.22**	**35.52**	**43.81**
16_{th},	0.048	0.033	0.033	0.033	-0.894	0.513	0.033	0.425
17_{th}	−0.0463	0.0296	−0.1664	0.0296	−0.1266	0.0094	−0.0257	0.0022
19_{th}	−0.0432	0.0234	0.0234	0.0234	−1.2365	0.6129	−0.3635	0.8309
Excess Nodal Pressure								

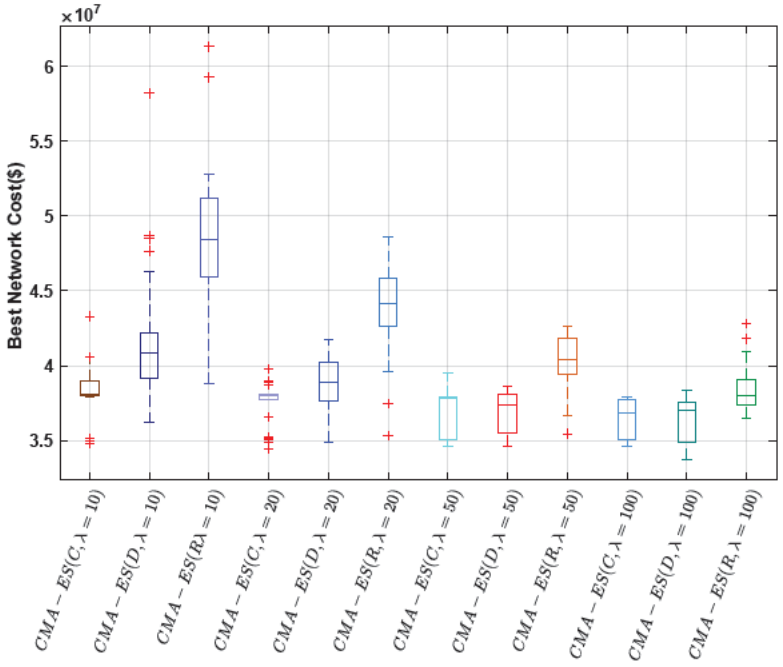

FIGURE 7.3: A comparison of the three different strategies for optimising the New York Tunnels problem for different CMA-ES population sizes ($\lambda = 10, 20, 50, 100$). C= Continuous, D= Discrete and R= Rounded (Possible), the best solution per experiment, 30 independent runs. According to the results, large population sizes can be a better choice like 50 or 100, and also continuous pipe diameter strategy performs better than both discrete and rounded pipe size methods. However, the sum of the nodal head violation is not zero in some of the lower cost designs.

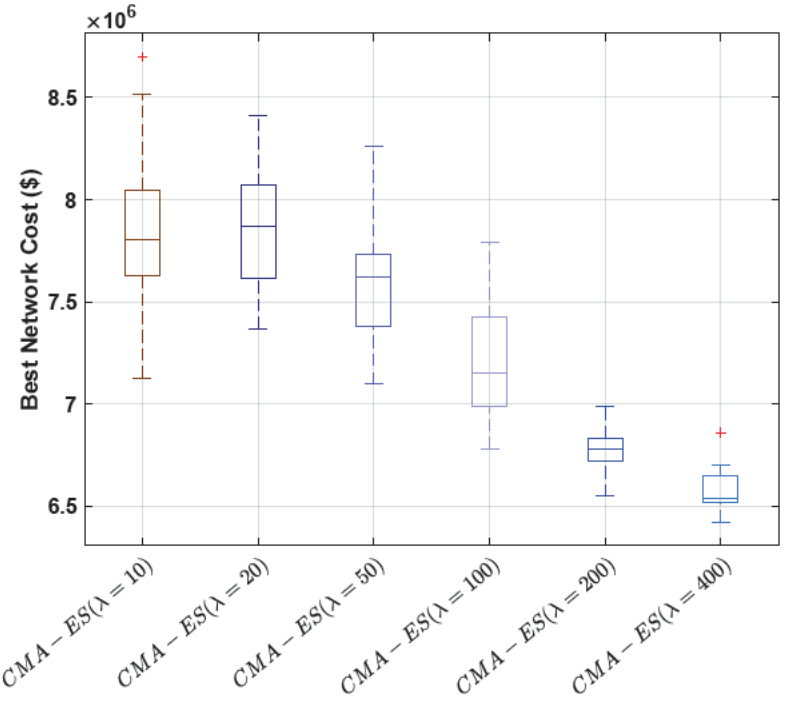

FIGURE 7.4: The impact of different population sizes of CMA-ES with the discrete penalty cost in the Hanoi network.

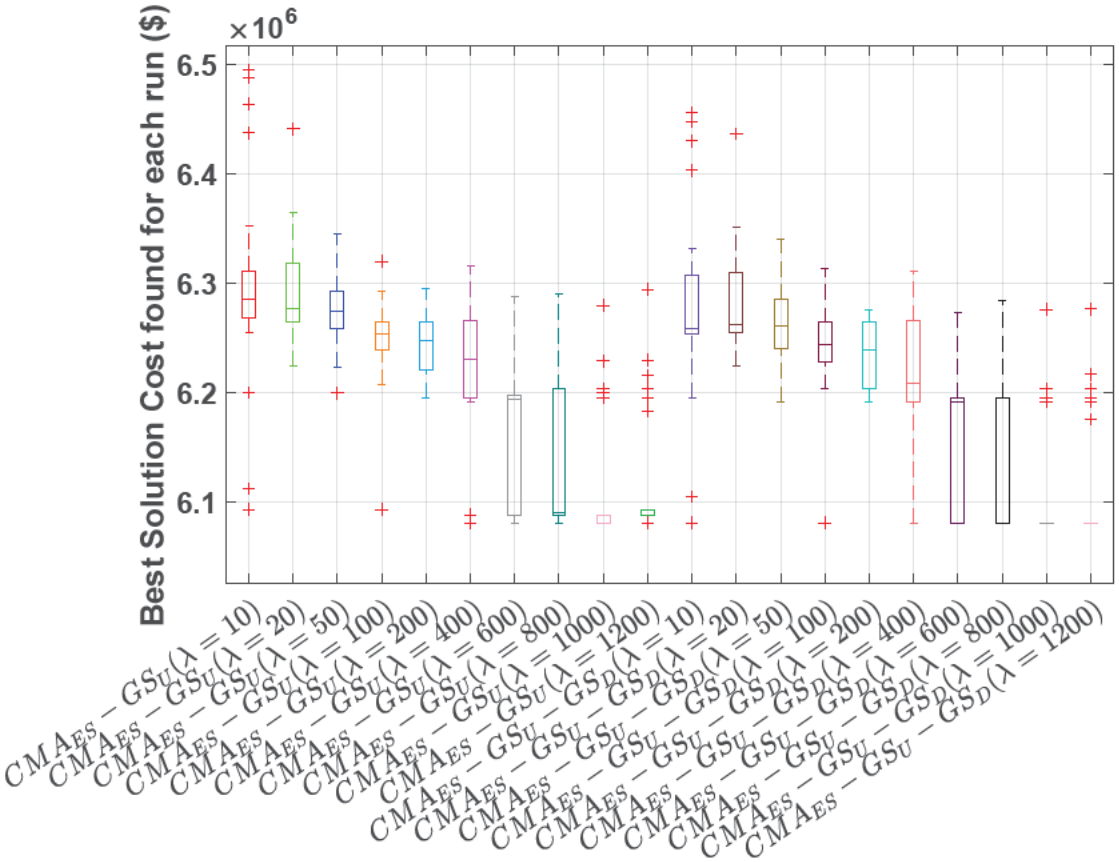

FIGURE 7.5: The impact of different population sizes on the $\text{CMA}_{ES} - GS_U$ and $\text{CMA}_{ES} - GS_U - GS_D$ in the Hanoi network.

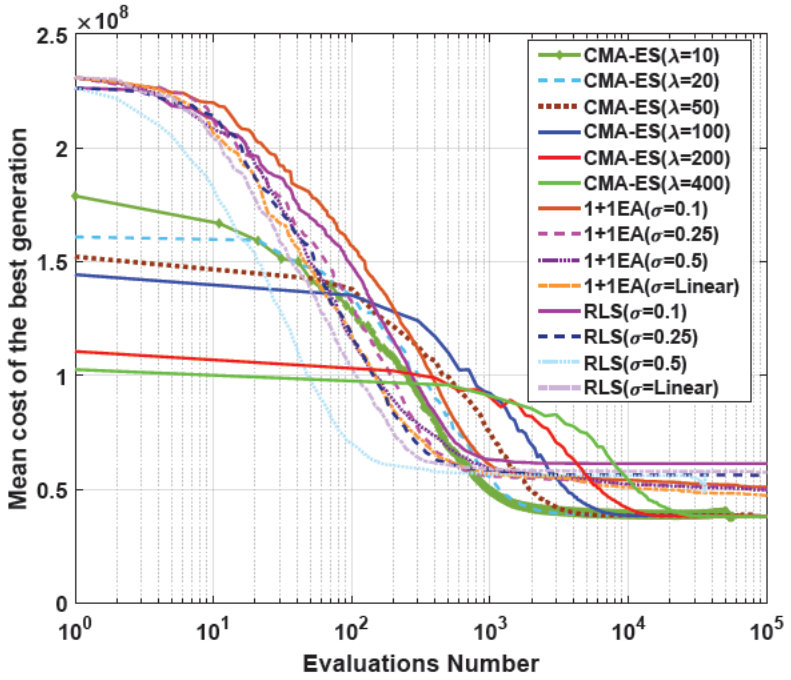

FIGURE 7.6: The comparison of the average convergence rate of CMA-ES ,RLS and 1+1EA for the continuous NYTP design where $\lambda = 10, 20, 50, 100, 200, 400$ and $\sigma = 0.1, 0.25, 0.5$, Linearly decreased.

iterations and converging to a local optimum. Table 7.4 illustrates how CMA_{ES}-GS_U explores and finds the feasible NYTP designs with possible pipe diameter sizes compared with the CMA-ES.

The configuration using CMA_{ES}-GS_U with $\lambda = 20$ exhibited the fastest average speed of convergence to the best-known solutions at 5500 evaluations. However, this configuration only reached the best-known solution 43% of the time. The first configuration which reached the optimum layout 100% was CMA_{ES}-GS_U with $\lambda = 400$ and for this the average number of evaluations was 22000.

Furthermore, where the population sizes are 10, 20, 50 and 100, the maximum admissible number of evaluations for NYTP, NYTP2 and HP case studies is 10^5, and for larger population sizes, that number increases to 2×10^5. In Table 7.5 it can be seen that the large population CMA variants require substantially more evaluations than SADE to find the best WEN design because the performance of standard, adaptive and self-adaptive versions of DE is better than the standard of CMA-ES in this case study. However, our proposed hybrid method outperforms other applied methods in Table 7.5. Meanwhile, there is a direct relationship between the convergence rate and the population size in population-based optimisation methods.

Both fine-tuned simple EAs have been assessed to minimise the cost of NYTP by four various mutation step sizes including $\sigma = 0.1, 0.25, 0.5 \times length(decision-variables)$ and a linear mutation step size which is decreased linearly. The best cost of the RLS found solutions is \$39.43 million with a linear σ and for the (1+1)EA is \$38.88 million when σ is equal to 0.5. Both methods are not able to find the best-known solution of NYTP, and it shows the significant complexity of the problem (Figure 7.6). The summary of the detailed outcomes of NYTP case study can be presented in Table 7.5. For carrying out a comprehensive analysis of the proposed hybrid framework and both RLS and (1+1)EA effectiveness, the box plots of the NYTP results are shown with continuous decision variables (Figure 7.8) as well as with the discrete (interval=1(inch)) pipe sizes (Figure 7.7) .

7.6.2 Case Study 2: Doubled New York Tunnel problem (NYTP2)

The NYTP2 consists of two independent NYTP problems hydraulically connected to one reservoir. The number of decision variables is 42, and the design options number are identical to the single NYTP. The best-known design cost is \$77.276 million. The best-found result for NYTP2 was reported for a Self-Adaptive DE (SADE [85]). The SADE success rate for finding the best-known solution is 90% of the time, and the average number of evaluations is 33,810.

TABLE 7.5: Summary of the proposed methods and other EAs assessed as applied to the NYTP,*CMA-ES results are feasible in terms of pipe sizes and nodal pressure head.

Algorithm	Number of runs	Best solution ($ M)	Success rate (%) (Global Optimum)	Average Cost ($ M)	Average evaluations to discover the first best solution	Maximum number of evaluations
SDE [281]	100	38.64	97%	38.65	1.29×10^4	2.0×10^5
DDE [281]	100	38.64	93%	38.66	1.32×10^4	2.0×10^5
SADE [85]	50	38.64	92%	38.64	0.66×10^4	*NA*
GHEST [298]	60	38.64	92%	38.64	1.15×10^4	NA
HD-DDS [285]	50	38.64	86%	38.64	4.70×10^4	0.5×10^5
DE [83]	300	38.64	71%	NA	0.55×10^4	1.0×10^5
Scatter Search [278]	100	38.64	65%	NA	5.76×10^4	NA
MMAS [79]	20	38.64	60%	38.64	3.07×10^4	0.5×10^5
CGA [281]	100	38.64	50%	39.04	4.43×10^4	2.0×10^5
SGA [281]	100	38.64	45%	39.25	5.48×10^4	2.0×10^5
PSO [76]	2000	38.64	30%	NA	NA	1.0×10^5
CMA-ES*						
$\lambda = 10$		38.64	6.7%	40.56	0.63×10^4	1.0×10^5
$\lambda = 20$		38.64	3.3%	41.97	0.54×10^4	1.0×10^5
$\lambda = 50$	30	38.64	3.3%	45.79	0.72×10^4	1.0×10^5
$\lambda = 100$		38.64	6.7%	43.80	1.01×10^4	1.0×10^5
$\lambda = 200$		38.64	13.3%	42.92	1.8×10^4	2.0×10^5
$\lambda = 400$		38.64	27.0%	40.53	2.1×10^4	2.0×10^5
CMA_{ES}-GS_U						
$\lambda = 10$		38.64	33.3%	39.44	0.64×10^4	1.0×10^5
$\lambda = 20$		38.64	43.3%	39.15	0.55×10^4	1.0×10^5
$\lambda = 50$	30	38.64	40.0%	39.80	0.73×10^4	1.0×10^5
$\lambda = 100$		38.64	53.3%	39.27	1.1×10^4	1.0×10^5
$\lambda = 200$		38.64	83.3%	38.85	1.9×10^4	2.0×10^5
$\lambda = 400$		38.64	**100%**	38.64	2.2×10^4	2.0×10^5
CMA_{ES}-GS_U-GS_D						
$\lambda = 10$		38.64	36.6%	39.30	0.65×10^4	1.0×10^5
$\lambda = 20$		38.64	43.3%	39.10	0.61×10^4	1.0×10^5
$\lambda = 50$	30	38.64	43.3%	39.49	0.74×10^4	1.0×10^5
$\lambda = 100$		38.64	56.7%	39.19	1.2×10^4	1.0×10^5
$\lambda = 200$		38.64	86.7%	38.80	2.0×10^4	2.0×10^5
$\lambda = 400$		38.64	**100%**	38.64	2.3×10^4	2.0×10^5
RLS						
$\sigma = 0.1$		53.42	0.0%	61.32	1.44×10^4	1.0×10^5
$\sigma = 0.25$	30	39.93	0.0%	56.26	7.58×10^4	1.0×10^5
$\sigma = 0.5$		39.52	0.0%	55.99	1.50×10^4	1.0×10^5
$\sigma = Linear$		39.43	0.0%	57.49	8.50×10^4	1.0×10^5
1+1EA						
$\sigma = 0.1$		44.04	0.0%	50.93	9.10×10^4	1.0×10^5
$\sigma = 0.25$	30	39.74	0.0%	49.76	6.89×10^4	1.0×10^5
$\sigma = 0.5$		38.88	0.0%	49.76	5.86×10^4	1.0×10^5
$\sigma = Linear$		39.47	0.0%	47.31	5.96×10^4	1.0×10^5

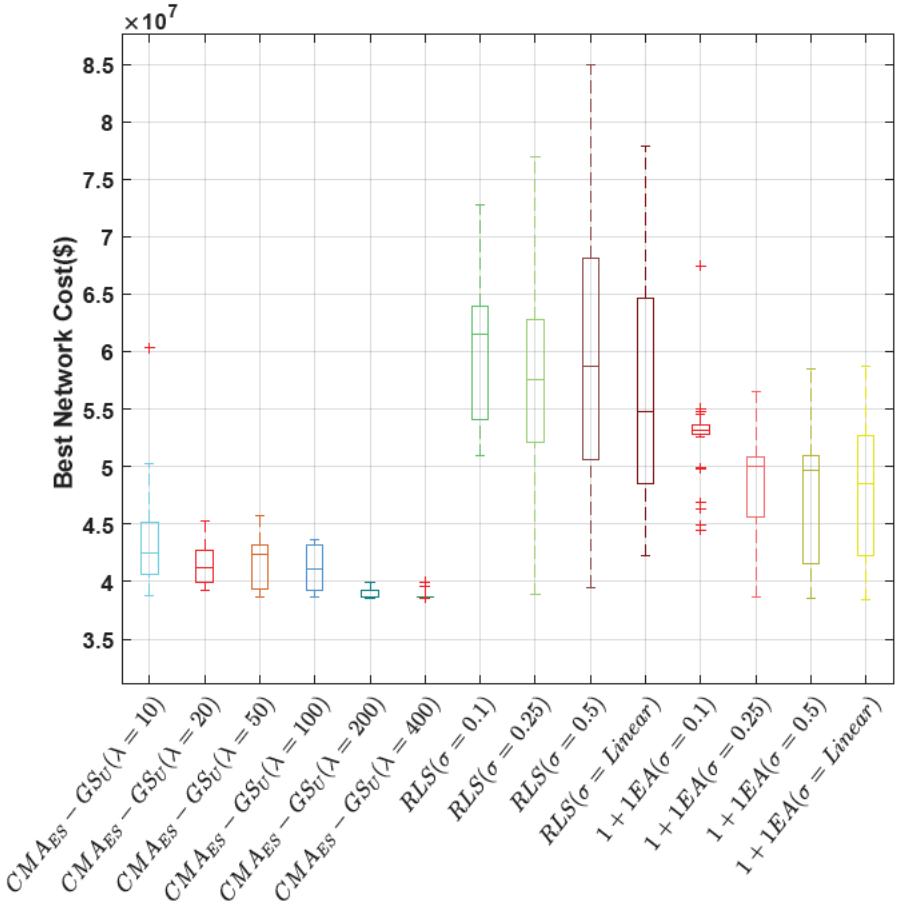

FIGURE 7.7: optimising the Discrete (interval=1(inch)) New York Tunnel Network design for different CMA-ES population sizes (λ=10, 20, 50, 100, 200, 400), CMA_{ES} − GS_U, RLS and 1+1EA(σ=0.1, 0.25, 0.5 ×(UB-LB) and Linearly decreased), the best solution per experiment, 30 independent runs, maximum evaluation number=10^5(λ=10, 20, 50, 100) and 2×10^5($\lambda = 200$) as well as $\lambda = 400$. Remark: all discovered solutions by CMA_{ES}-GS_U are feasible in terms of both pipe diameters and pressure head constraints, Best NYTP cost= \$38.64 (Million)

In this paper, the performance of CMA_{ES}-GS_U for NYTP2 is also observed to be considerable, as CMA_{ES}-GS_U is able to find the optimal solution in 100% of the 30 independent runs with a uniform random scenario for initializing the decision variables (Table 7.6). Moreover, CMA_{ES}-GS_U performs significantly better than the CMA-ES in terms of both the quality and efficiency of the obtained solutions. Indeed, the CMA-ES performance can be satisfactory if the right population size is chosen. For both case studies (NYTP and NYTP2), large population sizes ($\lambda = 10, 20 \times Dimension$) have better performance because it has a more robust exploration strength compared with the exploitation ability.

In spite of the simplicity of RLS and (1+1)EA, two methods can find the relative near-optimal designs at \$77.69 ($\sigma = 0.5$) and \$79.38 million ($\sigma = Linear$) respectively. These corresponding closeness in solutions show that the RLS and (1+1)EA are exploring in the neighbourhood space of the known-optimum solution.

TABLE 7.6: Summary of the proposed methods and other EAs evaluated to the DNYTP (NYTP2)

Algorithm	**Number of runs**	**Best solution ($ M)**	**Success rate (%) (Best Solution) Found**	**Average Cost ($ M)**	**Average evaluations to discover the first best solution**	**Maximum number of evaluations**
SADE [85]	50	77.28	90%	77.28	0.34×10^5	NA
HD-DDS [285]	20	77.28	85%	77.28	3.10×10^5	3.1×10^5
DE [85]	50	77.28	86%	77.28	0.70×10^5	1.0×10^5
MMAS [79]	20	77.28	5%	78.20	2.38×10^5	3.0×10^5
CMA-ES						
$\lambda = 10$		78.04	0.0%	84.17	0.18×10^5	1.0×10^5
$\lambda = 20$		81.15	0.0%	85.11	0.19×10^5	1.0×10^5
$\lambda = 50$	30	81.73	0.0%	89.23	0.23×10^5	1.0×10^5
$\lambda = 100$		77.28	3.33%	93.05	0.32×10^5	1.0×10^5
$\lambda = 200$		77.28	3.33%	96.45	0.48×10^5	2.0×10^5
$\lambda = 400$		77.28	3.33%	93.44	0.71×10^5	2.0×10^5
$\lambda = 600$		77.28	3.33%	86.11	0.92×10^5	2.0×10^5
$\lambda = 800$		79.74	0.0%	85.87	0.98×10^5	2.0×10^5
$\lambda = 1000$		77.28	6.66%	80.20	1.01×10^5	2.0×10^5
CMA_{ES}**-G**S_U						
$\lambda = 10$		77.45	0.0%	79.76	0.19×10^5	1.0×10^5
$\lambda = 20$		77.28	13.33%	80.05	0.20×10^5	1.0×10^5
$\lambda = 50$	30	77.28	13.33%	79.53	0.24×10^5	1.0×10^5
$\lambda = 100$		77.28	13.33%	79.50	0.33×10^5	1.0×10^5
$\lambda = 200$		77.28	23.33%	79.48	0.49×10^5	2.0×10^5
$\lambda = 400$		77.28	33.33%	78.93	0.72×10^5	2.0×10^5
$\lambda = 600$		77.28	70.00%	77.82	0.93×10^5	2.0×10^5
$\lambda = 800$		77.28	90.00%	77.45	0.99×10^5	2.0×10^5
$\lambda = 1000$		77.28	**100%**	77.28	1.03×10^5	2.0×10^5
CMA_{ES}**-G**S_U**-G**S_D						
$\lambda = 10$		77.45	0.0%	79.56	0.20×10^5	1.0×10^5
$\lambda = 20$		77.28	16.67%	79.77	0.21×10^5	1.0×10^5
$\lambda = 50$	30	77.28	13.33%	79.18	0.25×10^5	1.0×10^5
$\lambda = 100$		77.28	13.33%	79.28	0.34×10^5	1.0×10^5
$\lambda = 200$		77.28	23.33%	79.20	0.50×10^5	2.0×10^5
$\lambda = 400$		77.28	36.66%	78.56	0.73×10^5	2.0×10^5
$\lambda = 600$		77.28	73.33%	77.71	0.94×10^5	2.0×10^5
$\lambda = 800$		77.28	90.00%	77.44	1.01×10^5	2.0×10^5
$\lambda = 1000$		77.28	**100%**	77.28	1.04×10^5	2.0×10^5
RLS						
$\sigma = 0.1$		106.92	0.0%	122.64	0.26×10^5	1.5×10^5
$\sigma = 0.25$	30	79.87	0.0%	112.53	0.13×10^5	1.5×10^5
$\sigma = 0.5$		77.69	0.0%	111.99	0.29×10^5	1.5×10^5
$\sigma = Linear$		78.87	0.0%	114.98	1.5×10^5	1.5×10^5
1+1EA						
$\sigma = 0.1$		88.09	0.0%	101.88	1.46×10^5	1.5×10^5
$\sigma = 0.25$	30	79.48	0.0%	99.53	1.17×10^5	1.5×10^5
$\sigma = 0.5$		79.49	0.0%	98.40	1.11×10^5	1.5×10^5
$\sigma = Linear$		79.38	0.0%	94.63	1.01×10^5	1.5×10^5

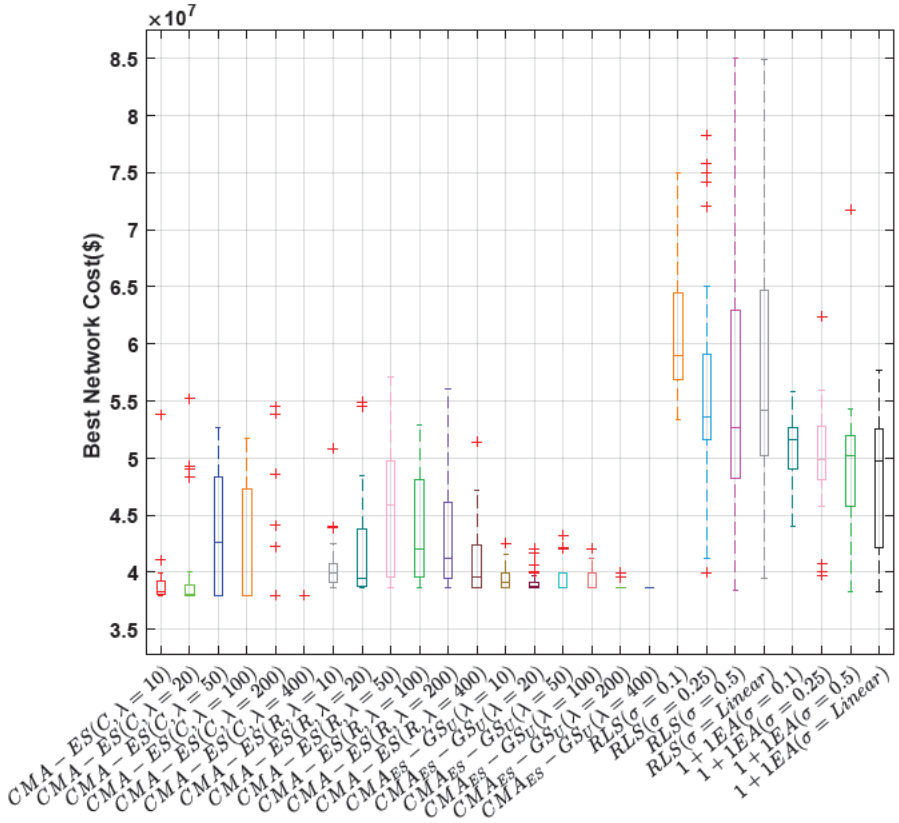

FIGURE 7.8: optimising the NYTP continuous design by different CMA-ES population sizes (λ=10, 20, 50, 100, 200, 400), $CMA_{ES} - GS_U$, RLS and 1+1EA (σ= 0.1, 0.25, 0.5 × (UB-LB) and Linear) , the best solution per experiment, 30 independent runs, Maximum Evaluation number=10^5 (λ=10, 20, 50, 100), 2×10^5 for $\lambda = 200$ as well as $\lambda = 400$. C and R show continuous and rounded pipe diameters designs. All solutions are feasible in terms of nodal pressure head.

7.6.3 Case Study 3: 50 * New York Tunnel problem (NYTP50)

The NYTP50 [296] includes 50 individual NYTP problems that are joined to one reservoir. This problem can be a large-scale optimisation benchmark where the number of decision variables is 50 × 21 with the same design options of the original NYTP. The best-known design cost can be calculated at $1932 million.

Table 7.7 shows the best and average found design costs are $2022(M) and $2030(M) by CMA_{ES}-GS_U-GS_D of the ten independent runs with a uniform random scenario for initializing the decision variables where the population size is 500. These relative near-optimal designs are considerably better than the 50 NYTP designs from [296]. The method presented here is able to explore this high-dimensional search space very effectively with a relatively small number of evaluations.

TABLE 7.7: Summary of the proposed methods and other EAs evaluated to the $50 \times NYTP$ (NYTP50), the best-known =\$1932(M).(* shows the designs are continuous and the average pressure violation per each NYTP is 0.09, 0.15 and 0.2 where $\lambda = 200, 500$ and 1000)respectively.

Algorithm	Number of runs	Best solution (\$ M)	Success rate (%) (Best solution found)	Average Cost (\$ M)	Average evaluations to discover the first best solution	Maximum number of evaluations
GA [296]	100	2238	0.0%	2321	NA	40.0×10^6
CMA_{ES}-GS_U						
$\lambda = 200$	10	2033	0.0%	2037	0.436×10^6	1.0×10^6
$\lambda = 500$	10	2026	0.0%	2037	0.752×10^6	1.0×10^6
$\lambda = 1000$	10	2068	0.0%	2090	0.942×10^6	1.0×10^6
CMA_{ES}-GS_U-GS_D						
$\lambda = 200$	10	2030	0.0%	2033	0.485×10^6	1.0×10^6
$\lambda = 500$	10	2022	0.0%	2030	0.771×10^6	1.0×10^6
$\lambda = 1000$	10	2055	0.0%	2069	0.981×10^6	1.0×10^6

7.6.4 Case Study 4: Hanoi (HP)

The Hanoi Network (HP) is made up of 34 pipes, 32 nodes, and three loops. A gravity-fed system has been designed which is fed from a single fixed tank and is produced to fulfil assigned water demands at the necessary minimum allowable pressures. Six possible sizes of the industrial pipe diameters are assigned for this network. And also the cost of i^{th} pipe with diameter D_i and particular length L_i can be computed by the formula ($C_{pipe_i} = 1.1 \times D_i^{1.5} \times L_i$) where the pipe diameter is in inches, and the pipe length is in meters. The Hazen-Williams coefficient is deterministic at 130 for total pipes. All required data can be obtained from the reference [271]. The best-found feasible solution, to-date, for the optimisation of the Hanoi network cost is \$6.081 million; it is referred to the literature. The Hanoi Problem has been taken into account as three different aspects of optimisation problem such as a continuous [271], split-pipe [271], and discrete pipes [68, 70, 272]. Some of the best achieved HP layouts, which are introduced by the authors are listed in Table 7.11. It can be seen; where the pipe diameters are considered as continuous, the best performance is allocated to the CMA-ES which is able to find the cheapest feasible continuous solution by \$5.959 million. On the other hand, most of the researchers have been focused on the discrete pipe sizes recently, so in Table 7.8, the discrete results are reported only. Since Hanoi search space is super noisy with many local optima in the discrete search space, CMA-ES trapped into one of them and can not escape.

According to the previous optimisation results of case studies (NYTP, NYTP2 and NYTP50), the efficiency of the CMA-ES with the continuous decision variables is better than discrete. A comparison of CMA-ES effectiveness with discrete pipe sizes can be illustrated in Figure 7.4 by diverse population sizes for minimising the pipe cost of HP.

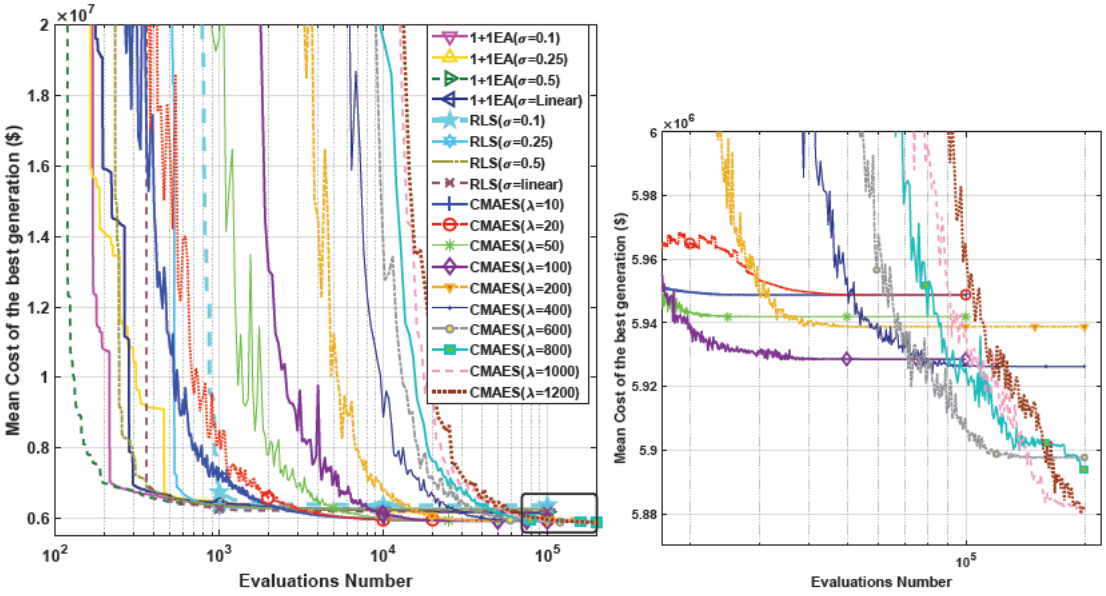

FIGURE 7.9: The comparison of the average convergence rate of CMA-ES ,RLS and 1+1EA for the HP design where $\lambda = 10, 20, 50, 100, 200, 400, 600, 800, 1000, 1200$ and $\sigma = 0.1, 0.25, 0.5$, Linearly decreased. (a) and a zoomed version of the last iterations is shown (b)

The large population size found better solutions similar to previous case studies, but as a matter of fact, the continuous version of CMA-ES is more robust in exploring the search space and finding the optimal solutions which are placed very near the constraint edges. Thus initially, CMA-ES is applied by considering continuous pipe sizes and then obtained solutions are rounded to commercial pipe sizes. As a result, a set of near-optimal feasible layouts is exploited by the CMA-ES (such as \$6.173, \$6.204 million) and the convergence rate is considerable. However, the best-known discrete solution is not found by the CMA-ES. Promisingly, CMA-ES can discover a range of very low cost feasible continuous layouts of the HP network. This is the primary motivation for applying the next step. For improving the optimisation process, two other parts of the proposed hybrid method are employed. Initially, GS_U tries to repair the violation of the nodal pressure head by increasing the size of pipes discretely. Secondly, GS_D is evaluated independently to reduce the extra imposed pipe cost of the GS_U step. Interestingly, the proposed hybrid method is able to find the well-known HP design 83.33% of the time over 30 independent runs with different initializing random number seeds, when the population size is 1000.

Figure 7.9 describes the average convergence rate of the RLS, 1+1EA and CMAES methods for 30 independent runs. The 1+1EA convergence speed is faster than others with $\sigma = 0.5$ and 0.1. Moreover, CMAES with small population sizes can converge quicker than large population sizes. However, the performance of large population sizes for exploiting the search space are more successful.

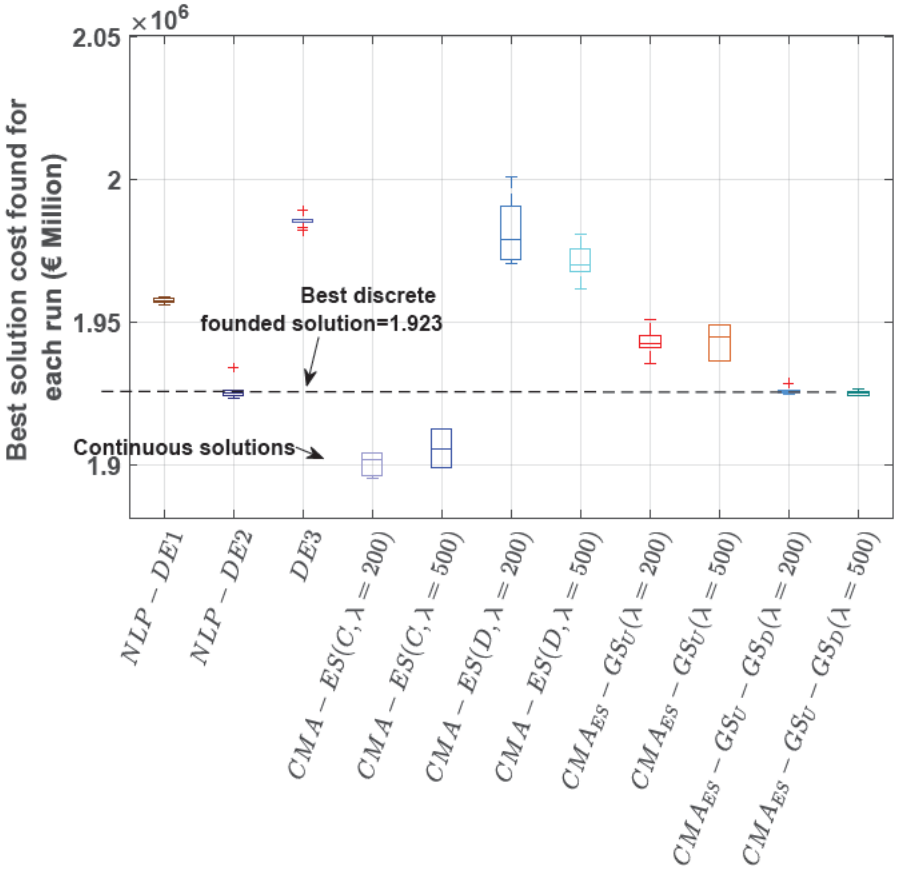

FIGURE 7.10: The efficiency comparison of the proposed Hybrid framework: CMA-ES: Continuous and Discrete ($\lambda = 200, 500$), CMA_{ES}-GS_U and CMA_{ES}-GS_U-GS_D, with previous best methods [16] ($NLP - DE_1$, $NLP - DE_2$ and DE_3) for Balerma Network. The average performance of CMA_{ES}-GS_U-GS_D is better than the previous best method ($NLP - DE_2$).

Meanwhile, using the third step of the hybrid framework (GS_D) leads to increasing the percentage of success rate in finding the optimum solution to 19% compared with the CMA_{ES}-GS_U while spending just 8.2% more in terms of computational budgets.

According to the observed results from the Table 7.8, RLS and (1+1)EA are able to find semi-optimum solutions means very close to the best-known solution. The performance of both RLS and (1+1)EA are acceptable based on the expectations; however, the best previously found solution of HP is not found by them. The (1+1)EA efficiency (\$6.115 million, ($\sigma = 0.25$)) is better than the RLS (\$6.128 million, ($\sigma = 0.1$)) in terms of the quality of the achieved solution. However, the average performance of RLS with regard to the mean evaluation number and solution quality is better than the (1+1)EA. Interestingly, the (1+1)EA, which is a simple EA, performs better than a complex method like MMAS [79].

TABLE 7.8: Summary of the proposed methods and other EAs evaluated to the Hanoi network (all networks satisfy the pressure constraint)

Algorithm	Number of runs	Best solution ($ M)	Success rate (%)	Average Cost ($ M)	Average evaluations to discover the first best solution	Maximum number of evaluations
BLP-DE [282]	100	6.081	98%	6.085	3.31×10^4	0.4×10^5
NLP-DE [282]	100	6.081	97%	6.082	3.46×10^4	0.8×10^5
SDE [281]	100	6.081	92%	NA	7.72×10^4	5.0×10^5
SADE [85]	50	6.081	84%	6.090	6.05×10^4	2.0×10^5
DE [83]	300	6.081	80%	NA	4.87×10^4	1.0×10^5
DDE [281]	100	6.081	80%	NA	6.37×10^4	5.0×10^5
Scatter Search [278]	100	6.081	64%	NA	4.31×10^4	NA
GHEST [298]	60	6.081	38%	6.175	5.01×10^4	NA
GENOME [284]	10	6.081	10%	6.248	NA	1.5×10^5
HD-DDS [285]	50	6.081	8%	6.252	10.00×10^4	1.0×10^5
PSO [76]	2000	6.081	5%	6.310	NA	0.8×10^5
CGA [281]	100	6.109	0.0%	6.274	32.12×10^4	5.0×10^5
SGA [281]	100	6.112	0.0%	6.287	3.85×10^4	5.0×10^5
MMAS [79]	20	6.134	0.0%	6.386	8.50×10^4	1.0×10^5
CMA-ES						
$\lambda = 10$		7.371	0.0%	7.573	0.73×10^4	1.0×10^5
$\lambda = 20$		6.840	0.0%	6.950	0.72×10^4	1.0×10^5
$\lambda = 50$		6.733	0.0%	6.931	0.96×10^4	1.0×10^5
$\lambda = 100$		6.494	0.0%	6.695	1.41×10^4	1.0×10^5
$\lambda = 200$	30	6.281	0.0%	6.295	2.28×10^4	2.0×10^5
$\lambda = 400$		6.220	0.0%	6.319	4.70×10^4	2.0×10^5
$\lambda = 600$		6.227	0.0%	6.287	8.05×10^4	2.0×10^5
$\lambda = 800$		6.290	0.0%	6.315	1.01×10^5	2.0×10^5
$\lambda = 1000$		6.210	0.0%	6.279	1.39×10^5	2.0×10^5
$\lambda = 1200$		6.129	0.0%	6.294	1.6×10^5	2.0×10^5
CMA$_{ES}$-GS$_U$						
$\lambda = 10$		6.094	0.0%	6.293	0.80×10^4	1.0×10^5
$\lambda = 20$		6.224	0.0%	6.296	0.79×10^4	1.0×10^5
$\lambda = 50$		6.199	0.0%	6.276	1.05×10^4	1.0×10^5
$\lambda = 100$		6.093	0.0%	6.250	1.48×10^4	1.0×10^5
$\lambda = 200$	30	6.196	0.0%	6.247	2.47×10^4	2.0×10^5
$\lambda = 400$		6.081	6.67%	6.219	4.81×10^4	2.0×10^5
$\lambda = 600$		6.081	10.00%	6.156	8.16×10^4	2.0×10^5
$\lambda = 800$		6.081	13.33%	6.149	1.05×10^5	2.0×10^5
$\lambda = 1000$		6.081	36.67%	6.112	1.42×10^5	2.0×10^5
$\lambda = 1200$		6.081	33.33%	6.118	1.71×10^5	2.0×10^5
CMA$_{ES}$-GS$_U$-GS$_D$						
$\lambda = 10$		6.081	6.67%	6.272	0.83×10^4	1.0×10^5
$\lambda = 20$		6.224	0.0%	6.282	0.81×10^4	1.0×10^5
$\lambda = 50$		6.192	0.0%	6.265	1.12×10^4	1.0×10^5
$\lambda = 100$		6.081	3.33%	6.241	1.53×10^4	1.0×10^5
$\lambda = 200$	30	6.192	0.0%	6.237	2.50×10^4	2.0×10^5
$\lambda = 400$		6.081	16.67%	6.207	4.91×10^4	2.0×10^5
$\lambda = 600$		6.081	43.33%	6.150	8.28×10^4	2.0×10^5
$\lambda = 800$		6.081	53.33%	6.142	1.07×10^5	2.0×10^5
$\lambda = 1000$		6.081	83.33%	6.106	1.48×10^5	2.0×10^5
$\lambda = 1200$		6.081	80.00%	6.110	1.82×10^5	2.0×10^5
RLS						
$\sigma = 0.1$		6.128	0.0%	8.355	7.45×10^4	1.0×10^5
$\sigma = 0.25$	30	6.170	0.0%	7.123	4.82×10^4	1.0×10^5
$\sigma = 0.5$		6.143	0.0%	7.863	6.71×10^4	1.0×10^5
$\sigma = Linear$		6.136	0.0%	8.286	6.12×10^4	1.0×10^5
1+1EA						
$\sigma = 0.1$		6.134	0.0%	9.39	8.55×10^4	1.0×10^5
$\sigma = 0.25$	30	6.115	0.0%	9.184	8.88×10^4	1.0×10^5
$\sigma = 0.5$		6.202	0.0%	9.167	8.83×10^4	1.0×10^5
$\sigma = Linear$		6.170	0.0%	9.045	9.84×10^4	1.0×10^5

TABLE 7.9: Summary of the proposed methods and other EAs evaluated to the Balerma network (BN).

Algorithm	Number of runs	Best solution (€M)	Average Cost (€M)	Average evaluations to discover the first best solutions	Maximum number of evaluations
NLP-DE2 [16]	10	1.923	1.927	1.428×10^6	2.0×10^6
HD-DDS-1 [285]	1	1.941	NA	30.00×10^6	30.0×10^6
NLP-DE1 [16]	10	1.956	1.957	4.12×10^3	1.0×10^6
HD-DDS-2 [285]	10	1.956	NA	30.00×10^6	10.0×10^6
DE3 [16]	10	1.982	1.986	9.21×10^6	10.0×10^6
SADE [85]	10	1.983	1.995	1.20×10^6	1.3×10^6
CSHS [299]	10	1.988	2.031	3.00×10^6	5.0×10^6
DE [85]	10	1.998	2.031	2.30×10^6	2.4×10^6
GHEST [298]	10	2.002	2.055	0.25×10^6	10.0×10^6
HS [276]	NA	2.018	NA	10.00×10^6	10.0×10^6
CS [299]	10	2.036	2.079	4.50×10^6	5.0×10^6
GAs [275]	10	2.061	NA	NA	2.00×10^6
GENOME [284]	10	2.302	2.334	10.00×10^6	10.0×10^6
CMA-ES(Continuous)	10				
$\lambda = 200$		1.895	1.900	0.84×10^6	2×10^6
$\lambda = 500$		1.899	1.906	0.98×10^6	2×10^6
CMA-ES (Discrete)	10				
$\lambda = 200$		1.974	1.990	0.56×10^6	2×10^6
$\lambda = 500$		1.961	1.971	0.69×10^6	2×10^6
CMA_{ES}**-G**S_U	10				
$\lambda = 200$		1.936	1.943	0.86×10^6	2×10^6
$\lambda = 500$		1.937	1.942	1.10×10^6	2×10^6
CMA_{ES}**-G**S_U**-G**S_D	10				
$\lambda = 200$		1.9245	1.9259	0.89×10^6	2×10^6
$\lambda = 500$		1.9243	1.9249	1.15×10^6	2×10^6

7.6.5 Case Study 5: Balerma (BN)

The fifth case study is the Balerma Network (BN), which is an irrigation WDS established in the province of Almeria (Spain) [284]. Its components are four reservoirs, 454 pipes, eight loops and 443 demand nodes. There are 10 allowable PVC commercial pipes diameter sizes 125 to 600 mm. Therefore, the search space is 10^{454}, which is considerably larger than the previous three case studies in this paper, and it is categorized as a large-scale optimisation problem. The minimum required nodal pressure is 20 m. Pipe costs and other details are given in the reference [284].

The current best layout of the BN, which is found by Zheng et al. is at €1.923 million. This functional design is achieved by a combination of the DE and nonlinear programming (NLP-DE). As demonstrated by the results from the Table 7.9, the average performance of the proposed hybrid framework is clearly better than all previous methods in terms of quality, efficiency and the convergence rate, mainly where CMA-ES is applied by the continuous decision variables. The best solution cost of the continuous CMA-ES

found here is €1.895 million ($\lambda = 500$). The main objective of the study is evaluating the performance of the hybrid framework with discrete pipe sizes (commercial), so the discrete results of CMA_{ES}-GS_U and CMA_{ES}-GS_U-GS_D are reported in the Table 7.9 too.

According to the results of Table 7.9, the CMA-ES (continuous) produces the lowest cost design. Additionally, the CMA-ES variants rank highly for the discrete design problem, with the CMA_{ES}-GS_U-GS_D ($\lambda = 500$) producing a network that cost only 0.01% more than the best solution to date.

Moreover, the saving rate of computational cost is 60%. This feature illustrates the ability of exploitation of both CMA_{ES}-GS_U and CMA_{ES}-GS_U-GS_D methods and indicates that the proposed optimisation framework is able to locate reasonable quality solutions with substantially developed computational effectiveness when faced with the large-scale WDS. It is noted that (Table 7.9), in terms of the success rate, the proposed hybrid framework does not outperform the NLP-DE2.

According to the experimental results in Table 7.9, it can be seen that the proposed hybrid method achievements achieve overall lower cost BN layouts compared to the previous methods with less computational budgets. Meanwhile, in terms of convergence speed, the discrete CMA-ES is highest ranked when the average evaluation number is just 0.56×10^6 and the best solution cost (near-optimum) is with a value of €1.961 million.

The CMA_{ES}-GS_U converged slightly slower than the discrete CMA-ES, but the quality of their outcomes is better and feasible (possible pipe sizes) as seen in Figure 7.10. Finally, the third part of the hybrid framework is evaluated for analyzing its impact on improving the results of the previous step (CMA_{ES}-GS_U). Where the dimension of the problem (BN) is high that leads to high value for branching factor of the tree structure of search space, it is recommended that applying the Downward Greedy Search (GS_D) is efficient because of its computational complexity and memory usage. One of the most important advantages of the GS_D is reducing the cost of BN layout by 0.6% and 0.68% (λ=200, 500) respectively on average while spending slightly more in terms of the computational budgets.

For comparing the robustness of the hybrid framework convergence rate for the large-scale Balerma network with the best previous methods, Figure 7.10 is provided. Except for the performance of discrete the CMA-ES, all three parts of the proposed method performed well. The best cost of the BN design that is found by the third step of the hybrid framework is €1.9243 million.

7.7 Conclusion

The WDS design problem is extremely computationally demanding. This optimisation problem belongs to a set of inherently intractable problems referred to as NP-complete problems with nonlinear constraints. This work proposes a new hybrid optimisation framework for WDSs design problems. The optimisation process of the new framework is divided into three phases:

1. Applying a the robust and self-adaptive EA called CMA-ES with different pipe diameter size granularities.

2. Carrying out a new Upward Greedy Search (GS_U) algorithm for repairing violations of the nodal pressure head constraints that are being optimised; and

3. Carrying out a new Downward Greedy Search (GS_D) to reduce pipe sizes that might be larger than required.

According to the optimisation results, it can be shown that the proposed new combined framework has faster convergence characteristics for the large-scale network. For both the NYTP and NYTP2 case studies, the hybrid approach is able to reliably find the best known discrete feasible designs costing \$38.64 and \$77.28 million. In addition, the the proposed method also finds the best known solution to the HP case study. In all benchmark networks the performance of the hybrid CMA_{ES}-GS_U-GS_D method is better than the standard CMA-ES in terms of the quality of discovered solutions.

For the BN case study, the proposed new framework produces solutions with a better average cost than the previously best published frameworks. The best-found solution from CMA_{ES}-GS_U-GS_D is within 0.01% of the cost of the best-published result – whilst being a quite different network design. Moreover, this framework produces solutions of consistently low cost – indicating that this framework is robust for this problem.

Future improvements to this work can include: focusing the greedy search processes on a sub-set of pipes most likely to improve network performance and cost; analysing the features of the diverse range of good solutions found in the larger network problems to look for features that might be used to inform smart search strategies and; finally, extend this work to more complex WDS designs including pumps and other network components.

TABLE 7.10: The review of the best proposed solutions for NYTP (The best continuous solution is $38,001,951 which is achieved by CMA-ES), LP=Linear Programming, PE=Partial Enumeration, BC=Branched Configuration, DC= Decomposition, NLP=Non-Linear Programming, IGA= Improved Genetic Algorithm, GA-O= optimiser GA, SFL= Shuffled Frog Leaping.(CMAES solution is feasible in terms of nodal pressure with the continuous pipe diameter)

Pipe/ Authors	**Schaake Lai**	**Quindry et al.**	**Gessler**	**Bhave**	**Morgan, Goulter**	**Kessler**	**Fujiwara Khang**	**Dandy et al.**	**Savic, Walters**	**Savic, Walters**	**Lippai et al.**	**Lippai et al.**	**Eusuff et al.**	**Maier et al.**	**Sedki, Quazar**	**Aghdam et al.**	**Our solution**
Year	**1969**	**1981**	**1982**	**1985**	**1985**	**1988**	**1990**	**1996**	**1997**	**1997**	**1999**	**1999**	**2003**	**2003**	**2012**	**2014**	**2019**
Feasibility	**Feasible**	**Feasible**	**Feasible**	**Feasible**	**Feasible**	**Clearly infeasible**	**Clearly infeasible**	**Feasible**	**Infeasible**	**Feasible**	**Very Slightly infeasible**	**Slightly infeasible**	**Very Slightly infeasible**	**Feasible**	**Feasible**	**Slightly infeasible**	**Feasible**
1	52.02	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
2	49.90	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
3	63.41	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
4	55.59	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
5	57.25	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
6	59.19	0.0	0	0.0	0	0.0	73.62	0	0	0	0	0	0	0	0	0	0
7	59.06	0.0	100	0.0	144	0.0	0.0	0	108	0	132	124	132	144	144	0	118.99
8	54.95	0.0	100	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
9	0.0	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
10	0.0	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
11	116.21	119.02	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
12	125.25	134.39	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
13	126.87	132.49	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
14	133.07	132.87	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
15	126.52	131.37	0	136.43	0	156.11	0.0	120	0	144	0	0	0	0	0	96	0
16	19.52	19.26	100	87.37	96	72.00	99.01	84	96	84	96	96	96	96	96	96	99.97
17	91.83	91.71	100	99.23	96	96.60	98.75	96	96	96	96	96	96	96	96	96	99.28
18	72.76	72.76	80	78.17	84	78.00	78.97	84	84	84	84	84	84	84	84	84	79.09
19	72.61	72.64	60	54.40	60	59.78	83.82	72	72	72	72	72	72	72	72	72	75.06
20	0.0	0.0	0	0.0	0	0.0	0.0	0	0	0	0	0	0	0	0	0	0
21	54.82	54.97	80	81.50	84	72.27	66.59	72	72	72	72	72	72	72	72	72	70.60
Node 16 Excess	+1.099	+1.0266	+0.3902	+0.9154	+1.6220	-1.4211	-0.8765	+0.5894	-0.2061	+1.1849	-0.0021	-0.0634	-0.0021	+0.0771	+0.0771	+0.0194	**0.0000**
Node 17 Excess	+1.0704	+0.9061	+0.3552	+0.6290	+0.0447	+0.2879	-0.7332	+0.1099	-0.2174	+0.7020	-0.0116	-0.0734	-0.0116	+0.0684	+0.0684	-0.0036	**0.0000**
Node 19 Excess	+1.2217	+1.1083	+0.9317	+1.0296	+0.0667	+0.2911	-0.8979	+0.7782	-0.1977	+1.4133	-0.0164	-0.0709	-0.0164	+0.0540	+0.0540	+0.1714	**0.0000**
Total Cost($M)	78.09	63.58	41.8	40.18	39.20	38.96	38.30	38.80	37.13	40.42	38.13	37.83	38.13	38.64	38.64	38.52	**38.00**
Methods	LP	LP	PE	BC	LP	DC	NLP	I-GA	GA	GA	GA-O	GA-O	SFL	ACO	PSO	AMPSO	**CMA-ES**

TABLE 7.11: The summery of the best proposed solutions for Hanoi Network (The best discovered solution is \$6.081(M\$))

Pipe/ Authors	Fujiwara		Savic & Walters		Cunha & Sousa		Wu et al.		Zecchin & Simpson		Zecchin & Simpson		Sedki& Ouazar		Our solution	
Year	**1990**		**1997**		**1999**		**2001**		**2005**		**2006**		**2012**		**2019**	
Feasibility	**Clearly infeasible**		**Infeasible**		**Infeasible**		**Feasible**		**Feasible**		**Feasible**		**Feasible**		**Feasible**	
#	Diameter	Head	Diameter	Head	Diameter	Head	Diameter	Head	Diameter	Head	Diameter	Head	Diameter	Head	Diameter	Head
1	40	100	40	100	40	100	40	100	40	100	40	100	40	100	40	100
2	40	97.14	40	97.14	40	97.14	40	97.14	40	97.14	40	97.14	40	97.14	40	97.14
3	38.8	61.67	40	61.67	40	61.67	40	61.67	40	61.67	40	61.67	40	61.67	40	61.67
4	38.7	56.17	40	56.88	40	56.87	40	57.22	40	57.63	40	57.08	40	56.92	40	57.17
5	37.8	49.27	40	50.94	40	50.92	40	51.70	40	52.63	40	51.38	40	51.03	40	51.61
6	36.3	41.1	40	44.68	40	44.64	40	45.93	40	47.45	40	45.40	40	44.81	40	45.77
7	33.8	38.77	40	43.21	40	43.16	40	44.59	40	46.27	40	44.00	40	43.35	37.68	44.42
8	32.8	34.84	40	41.45	40	41.39	40	43.03	40	44.96	40	42.36	40	41.62	36.29	42.30
9	31.5	31.2	40	40.04	40	39.98	40	41.81	30	43.95	40	41.06	40	40.23	34.84	40.30
10	25	*27.94*	30	38.2	30	38.93	30	40.92	30	41.07	30	40.11	30	39.20	29.17	38.52
11	23	*24.15*	24	37.44	24	37.37	24	39.36	24	39.51	24	38.55	24	37.64	26.83	36.73
12	20.2	*19.93*	24	34.01	24	33.94	24	35.94	24	36.08	24	35.12	24	34.22	23.42	34.74
13	19	*10.19*	20	*29.80*	20	*29.74*	16	31.73	12	31.87	20	30.91	20	30.01	16.67	30.00
14	14.5	*23.29*	16	35.13	16	35.00	12	34.22	12	31.31	12	37.21	16	35.52	12.35	32.58
15	12	*20.45*	12	33.14	12	32.95	12	31.99	12	31.51	12	32.89	12	33.72	12.00	30.15
16	19.9	*18.2*	12	30.23	12	*29.87*	12	31.96	24	34.86	12	32.16	12	31.30	12.00	30.00
17	23.1	37.01	16	30.33	16	30.03	20	44.73	20	38.38	20	41.36	16	33.41	17.70	37.24
18	26.6	50.72	20	43.97	20	43.87	24	52.54	40	56.13	24	48.55	24	49.93	22.32	49.60
19	26.8	58.04	20	55.58	20	55.54	24	58.54	24	56.9	20	54.33	20	55.09	22.49	57.62
20	35.2	47.15	40	50.44	40	50.49	40	50.4	40	51.1	40	50.61	40	50.61	40	50.28
21	16.4	*22.57*	20	41.09	20	41.14	20	41.05	20	41.75	20	41.26	20	41.26	17.74	33.50
22	12	*17.41*	12	35.93	12	35.98	12	35.88	12	36.58	12	36.1	12	36.1	12.99	30.00
23	29.5	31.91	40	44.21	40	44.3	40	44.13	40	45.41	40	44.53	40	44.53	37.97	42.08
24	19.3	*20.02*	30	38.9	30	38.57	30	38.66	30	41.33	30	39.39	30	38.93	32.66	38.21
25	16.4	*12.39*	30	35.56	30	34.86	30	35.17	24	34.36	30	36.19	30	35.34	30.26	34.55
26	12	*13.93*	20	31.53	20	30.95	24	33.48	20	31.73	20	32.55	20	31.7	20.64	30.79
27	20	*15.08*	12	30.10	12	*29.66*	12	31.91	12	31.65	12	31.61	12	30.76	14.49	30.00
28	22	*24.36*	12	35.5	12	38.66	12	36.22	12	40.93	12	35.90	12	38.94	12.00	36.10
29	18.9	*15.12*	16	30.75	16	*29.72*	16	32.17	20	31.89	16	31.23	16	30.13	16.17	30.00
30	17.1	*9.88*	16	*29.73*	12	*29.98*	16	31.72	16	30.20	16	30.29	12	30.42	13.54	30.03
31	14.6	*9.83*	12	30.19	12	30.26	12	31.87	16	30.24	12	30.77	12	30.70	12.00	30.28
32	12	*9.88*	12	31.44	16	32.72	16	33.37	12	31.42	12	32.4	16	33.18	15.84	31.86
33	12		16		16		16		12		16		16		17.02	
34	19.5		20		24		24		20		20		24		22.67	
Total Cost(\$M)	**5.351**		**6.072**		**6.056**		**6.183**		**6.367**		**6.134**		**6.081**		**5.959**	
Methods	**NLPG-ILOS**		**GANET**		**SA**		**GA**		**ACO**		**Two-ACO**		**PSO-DE**		**CMA-ES,**$\lambda = 200$	

Part IV

Conclusions and and Future Work

Chapter 8

summary

In this thesis, we have contributed new insights into real-world optimisation problems for various kinds of bio-inspired methods. Several new heuristic algorithms are proposed to solve problems associated with wave energy farm, wind farm and water distribution networks in Part I, Part II and Part III, respectively. We faced various challenges in attempting to solve the aforementioned real-world problems. These challenges related to the large number of decision variables, the multi-factorial nature of real problems, a complex and multi-model search spaces, the computational expense of the evaluation process (in some cases taking three hours per evaluation), several complex constraints and the difficulty of modelling dynamic systems. The performance of a large number of bio-inspired algorithms is evaluated and compared with that of our proposed algorithms. This allows us to make a systematic comparison demonstrating the quality and the convergence speed of these new heuristics.

In Part I, we design and propose several fast and effective single-objective optimisation frameworks. These frameworks are designed to improve the total absorbed power output of wave farms consisting of an array of fully submerged three-tether buoys (the CETO 6 [46] system which is currently under development by Carnegie Clean Energy) with small (4-buoy), large (16-buoy) and very large (50 and 100-buoy) farm sizes in real wave scenarios. In the context of WEC optimisation, we find that if the number of WECs is low (e.g., 4 buoys), the performance of population-based global optimisation methods can be competitive. However, in larger wave farms, we need smarter search techniques and surrogate model optimization methods. Generally, experimental results show that individual-based optimisation methods outperform population-based methods in terms of optimising the position of the large wave farm. Another important observation is that when other factors relating to WEC optimisation are involved (such as power take-off

settings, geometric configurations and levelised cost of energy), individual-based optimisation methods require more advances and hybridisation. This is because such factors make the search space highly complex. Therefore, we investigate and suggest various hybrid ideas of local search (Section 2.1), including a partial evaluation technique (Section 2.1), custom evolutionary algorithms (Section 2.1, Section 2.2 and Chapter 4), neuro and surrogate-evolutionary methods (Section 2.3), alternative (Section 3.1) and cooperative co-evolutionary (Section 3.2) approaches to speed up and develop the optimisation process of large wave energy farms with multi-factor parameters. In future, we aim to investigate the potential benefits, or disadvantages, of moving to a fully bi-objective and multi-objective variant formulation of this problem. As well as considering the total produced power, it is important to bear in mind, the required area, the cable/pipe length required to connect all WECs and the total cost of WECs in the context of optimising wave farms. Moreover, one of the best future plans can be considering the robustness of the best-found configurations, which are achieved based on the simulator, in the real sea site by the renewable energy companies. Another valuable plan that can be considered in the future is focusing on developing the adaptive and self-adaptive optimisation frameworks in the field of renewable energy which are able to adjust the setting of control parameters based on the various situations of the optimisation process. In Part II of this thesis, we investigate the use of Deep neural networks to model wind speed (Chapter 5) and power (Chapter 6) in an offshore and onshore wave farm. However, the seasonal and stochastic characteristics of wind speed make forecasting a challenging task. The initial observation is that recurrent neural networks (RNN) like LSTM outperform other machine learning techniques. Moreover, we find that hyper-parameters play a pivotal role in RNN performance. Hence, we evaluate and compare two tuning techniques, including grid search and evolutionary algorithms. Our experimental results show that evolutionary algorithms can propose more efficient hyper-parameters set than grid search. These proposed hybrid forecasting models outperform others using different performance indices, and the performance difference is statistically significant. Our future plan is to develop the proposed hybrid model by using a decomposition approach, in order to divide the time series wind data into some sub-groups holding more interrelated features than whole wind data as one group. Next, each sub-group is employed by one independent hybrid method to learn the nonlinear model of wind speed. Another later plan can be further investigations to compare the effectiveness of divers hybrid evolutionary algorithms and deep learning model based on the nonlinear combined mechanism.

In Part III, we provided new insights into the Water Distribution Systems (WDS) optimisation problem (Chapter 7). The WDS design problem is notably computationally challenging in regard to an assortment of essentially intractable problems associated with NP-complete problems relating to nonlinear pressure head constraints. In order to deal

with these challenges, we propose a new hybrid optimisation framework for WDSs design problems. This framework consists of a robust and self-adaptive EA called CMA-ES with different pipe diameter size granularities, with a new upward greedy search algorithm for repairing violations of the nodal pressure head constraints, and a new downward greedy search to decrease pipe sizes that might be larger than expected. This will serve to reduce extra costs. The optimisation results show that the proposed new combined framework has faster convergence characteristics than previously proposed methods. It also presents solutions with a better average cost than those achieved using standard CMA-ES and the previous best-available frameworks for the large-scale network. Moreover, in small networks, our framework can reliably find the best-known discrete feasible designs, and produces consistently low-cost solutions – indicating that the framework is robust for the small networks. The proposed future developments include (a) focusing the improved greedy search techniques on a sub-set of pipes which are most likely to enhance network performance and cost; (b) analysing the features of the diverse range of reasonable designs found in the larger network problems to consider features that might be adopted to inform smart search strategies and (c) extending the proposed method to more complex WDS designs including pumps and other network components.

Bibliography

[1] Mehdi Neshat, Bradley Alexander, Markus Wagner, and Yuanzhong Xia. A detailed comparison of meta-heuristic methods for optimising wave energy converter placements. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1318–1325. ACM, 2018. doi: 10.1145/3205455.3205492. URL https://doi.org/10.1145/3205455.3205492.

[2] Mehdi Neshat, Bradley Alexander, Nataliia Y Sergiienko, and Markus Wagner. New insights into position optimisation of wave energy converters using hybrid local search. *Swarm and Evolutionary Computation*, page 100744, 2020. doi: 10.1016/j.swevo.2020.100744. URL https://doi.org/10.1016/j.swevo.2020.100744.

[3] Mehdi Neshat, Bradley Alexander, Nataliia Y Sergiienko, and Markus Wagner. A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1293–1301, 2019. doi: 10.1145/3321707.3321806. URL https://doi.org/10.1145/3321707.3321806.

[4] Mehdi Neshat, Ehsan Abbasnejad, Qinfeng Shi, Bradley Alexander, and Markus Wagner. Adaptive neuro-surrogate-based optimisation method for wave energy converters placement optimisation. In *Proceedings of the International Conference on Neural Information Processing*, pages 353–366. Springer, 2019. doi: 10.1007/978-3-030-36711-4_30. URL https://doi.org/10.1007/978-3-030-36711-4_30.

[5] Mehdi Neshat, Bradley Alexander, and Markus Wagner. A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters. *Information Sciences*, 534:218–244, 2020. doi: 10.1016/j.ins.2020.03.112. URL https://doi.org/10.1016/j.ins.2020.03.112.

[6] Mehdi Neshat, Bradley Alexander, Nataliia Y. Sergiienko, and Markus Wagner. Optimisation of large wave farms using a multi-strategy evolutionary framework. In Carlos Artemio Coello Coello, editor, *GECCO '20: Genetic and Evolutionary Computation Conference, Cancún Mexico, July 8-12, 2020*, pages 1150–1158.

ACM, 2020. doi: 10.1145/3377930.3390235. URL https://doi.org/10.1145/3377930.3390235.

[7] Nataliia Y Sergiienko, Mehdi Neshat, Leandro SP da Silva, Bradley Alexander, and Markus Wagner. Design optimisation of a multi-mode wave energy converter. *arXiv preprint arXiv:2001.08966*, pages 1–19, 2020. URL https://arxiv.org/abs/2001.08966.

[8] Mehdi Neshat, Bradley Alexander, and Angus Simpson. Covariance matrix adaptation greedy search applied to water distribution system optimization. *arXiv preprint arXiv:1909.04846*, 2019. URL http://arxiv.org/abs/1909.04846.

[9] Mehdi Neshat, Meysam Majidi Nezhad, Ehsan Abbasnejad, Lina Bertling Tjernberg, Davide Astiaso Garcia, Bradley Alexander, and Markus Wagner. An evolutionary deep learning method for short-term wind speed prediction: A case study of the lillgrund offshore wind farm. *arXiv preprint arXiv:2002.09106*, 2020. URL https://arxiv.org/abs/2002.09106.

[10] Mehdi Neshat, Meysam Majidi Nezhad, Ehsan Abbasnejad, Daniele Groppi, Azim Heydari, Lina Bertling Tjernberg, Davide Astiaso Garcia, Bradley Alexander, and Markus Wagner. Hybrid neuro-evolutionary method for predicting wind turbine power output. *arXiv preprint arXiv:2004.12794*, 2020. doi: 10.13140/RG.2.2.35400.88323. URL https://arxiv.org/abs/2004.12794.

[11] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis. Grey wolf optimizer. *Advances in Engineering Software*, 69:46–61, 2014. URL https://doi.org/10.1016/j.advengsoft.2013.12.007.

[12] Nitin Mittal, Urvinder Singh, and Balwinder Singh Sohi. Modified grey wolf optimizer for global engineering optimization. *Applied Computational Intelligence and Soft Computing*, 2016:1–16, 2016.

[13] Australian Wave Energy Atlas, 2017. URL http://nationalmap.gov.au/renewables/. Accessed 24 October 2017.

[14] Joakim Jeppsson, Poul Erik Larsen, and Aake Larsson. Technical description lillgrund wind power plant: Lillgrund pilot project, 2008.

[15] Bowen Zhou, Xiangjin Ma, Yanhong Luo, and Dongsheng Yang. Wind power prediction based on lstm networks and nonparametric kernel density estimation. *IEEE Access*, 7:165279–165292, 2019.

[16] Feifei Zheng, Angus R Simpson, and Aaron C Zecchin. A combined nlp-differential evolution algorithm approach for the optimization of looped water distribution systems. *Water Resources Research*, 47(8):1–18, 2011.

[17] Shahrzad Saremi, Seyedali Mirjalili, and Andrew Lewis. Biogeography-based optimisation with chaos. *Neural Computing and Applications*, 25(5):1077–1097, 2014.

[18] Benjamin Drew, Andrew R Plummer, and M Necip Sahinkaya. A review of wave energy converter technology, 2009.

[19] Marianna Giassi. *Optimization of Point Absorber Wave Energy Parks*. PhD thesis, Uppsala University, Department of Engineering Sciences, 2018.

[20] Diana Bull and Margaret E Ochs. Technological cost-reduction pathways for attenuator wave energy converters in the marine hydrokinetic environment. *Sandia National Laboratories*, pages 1–51, 2013.

[21] AD De Andrés, R Guanche, L Meneses, C Vidal, and IJ Losada. Factors that influence array layout on wave energy farms. *Ocean Engineering*, 82:32–41, 2014. URL https://doi.org/10.1016/j.oceaneng.2014.02.027.

[22] Malin Göteman. Advances and challenges in wave energy park optimization-a review. *Frontiers in Energy Research*, 8(26), 2020. URL https://doi.org/10.3389/fenrg.2020.00026.

[23] Mehdi Neshat, Bradley Alexander, Nataliia Sergiienko, and Markus Wagner. A new insight into the position optimization of wave energy converters by a hybrid local search. *arXiv preprint arXiv:1904.09599*, 2019.

[24] Nataliia Sergiienko. *Three-tether wave energy converter: Hydrodynamic modelling, performance assessment and control*. PhD thesis, School of Mechanical Engineering, 04 2018.

[25] Johannes Falnes and Jørgen Hals. Heaving buoys, point absorbers and arrays. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1959):246–277, 2012.

[26] Paula B Garcia-Rosa, Giorgio Bacelli, and John V Ringwood. Control-informed geometric optimization of wave energy converters: The impact of device motion and force constraints. *Energies*, 8(12):13672–13687, 2015.

[27] Johannes Falnes and Adi Kurniawan. *Ocean waves and oscillating systems: linear interactions including wave-energy extraction*, volume 8. Cambridge university press, 2020.

[28] Amélie Têtu. Power take-off systems for wecs. In *Handbook of Ocean Wave Energy*, pages 203–220. Springer, Cham, 2017.

[29] Paresh Halder, Mohamed H Mohamed, and Abdus Samad. Wave energy conversion: design and shape optimization. *Ocean Engineering*, 150:337–351, 2018. URL https://doi.org/10.1016/j.oceaneng.2017.12.072.

[30] Sheng-chao Jiang, Ying Gou, Bin Teng, and De-zhi Ning. Analytical solution of a wave diffraction problem on a submerged cylinder. *Journal of Engineering Mechanics*, 140(1):225–232, 2014. URL https://doi.org/10.1061/(ASCE)EM.1943-7889.0000637.

[31] BFM Child and Vengatesan Venugopal. Optimal configurations of wave energy device arrays. *Ocean Engineering*, 37(16):1402–1417, 2010. URL https://doi.org/10.1016/j.oceaneng.2010.06.010.

[32] Chris Sharp and Bryony DuPont. Wave energy converter array optimization: A review of current work and preliminary results of a genetic algorithm approach introducing cost factors. In *Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, page V02AT03A025. American Society of Mechanical Engineers, 2015.

[33] Marianna Giassi, Malin Göteman, Simon Thomas, Jens Engström, Mikael Eriksson, and Jan Isberg. Multi-parameter optimization of hybrid arrays of point absorber wave energy converters. In *Proceedings of the 12th European Wave and Tidal Energy Conference.*, 2017.

[34] Soheil Esmaeilzadeh and Mohammad-Reza Alam. Shape optimization of wave energy converters for broadband directional incident waves. *Ocean Engineering*, 174:186–200, 2019. ISSN 0029-8018.

[35] Pau Mercadé Ruiz, Vincenzo Nava, Mathew BR Topper, Pablo Ruiz Minguela, Francesco Ferri, and Jens Peter Kofoed. Layout optimisation of wave energy converter arrays. *Energies*, 10(9):1262, 2017. URL https://doi.org/10.3390/en10091262.

[36] Chris Sharp and Bryony DuPont. Wave energy converter array optimization: A genetic algorithm approach and minimum separation distance study. *Ocean Engineering*, 163:148–156, 2018. URL https://doi.org/10.1016/j.oceaneng.2018.05.071.

[37] Marianna Giassi and Malin Göteman. Layout design of wave energy parks by a genetic algorithm. *Ocean Engineering*, 154:252–261, 2018. URL https://doi.org/10.1016/j.oceaneng.2018.01.096.

[38] Ossama Abdelkhalik and Shadi Darani. Optimization of nonlinear wave energy converters. *Ocean Engineering*, 162:187–195, 2018. URL https://doi.org/10.1016/j.oceaneng.2018.05.023.

[39] Junhua Wu, Slava Shekh, Nataliia Y Sergiienko, Benjamin S Cazzolato, Boyin Ding, Frank Neumann, and Markus Wagner. Fast and effective optimisation of arrays of submerged wave energy converters. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference*, pages 1045–1052. ACM, 2016. URL https://doi.org/10.1145/2908812.2908844.

[40] Francesco Ferri. Computationally efficient optimisation algorithms for wecs arrays. In *Proceedings of the 12th European Wave and Tidal Energy Conference European Wave and Tidal Energy Conference*, pages 1–7, 2017.

[41] Hong-Wei Fang, Yu-Zhu Feng, and Guo-Ping Li. Optimization of wave energy converter arrays by an improved differential evolution algorithm. *Energies*, 11(12): 3522, 2018. URL https://doi.org/10.3390/en11123522.

[42] Russell Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995.

[43] Jorge Izquierdo-Pérez, Bruno M Brentan, Joaquín Izquierdo, Niels-Erik Clausen, Antonio Pegalajar-Jurado, and Nis Ebsen. Layout optimization process to minimize the cost of energy of an offshore floating hybrid wind–wave farm. *Processes*, 8(2): 139, 2020. doi: 10.3390/pr8020139.

[44] Duc-Cuong Dang and Per Kristian Lehre. Runtime analysis of non-elitist populations: From classical optimisation to partial information. *Algorithmica*, 75(3): 428–461, 2016. URL https://doi.org/10.1007/s00453-015-0103-x.

[45] Mehdi Neshat, Bradley Alexander, and Markus Wagner. A hybrid cooperative co-evolution algorithm framework for optimising power take off and placements of wave energy converters. *arXiv preprint arXiv:1910.01280*, 2019.

[46] Carnegie Clean Energy. Ceto 6 design update @ONLINE. url = https://s3-ap-southeast-2.amazonaws.com/website-sydney-1/media/2017/11/13134305/1738756.pdf, November 2017.

[47] Ya-Lan Hu and Liang Chen. A nonlinear hybrid wind speed forecasting model using lstm network, hysteretic elm and differential evolution algorithm. *Energy Conversion and Management*, 173:123–142, 2018. URL https://doi.org/10.1016/j.enconman.2018.07.070.

[48] Vikas Khare, Savita Nema, and Prashant Baredar. Solar–wind hybrid renewable energy system: A review. *Renewable and Sustainable Energy Reviews*, 58:23–33, 2016.

[49] Monaaf DA Al-Falahi, SDG Jayasinghe, and HJEC Enshaei. A review on recent size optimization methodologies for standalone solar and wind hybrid renewable energy system. *Energy Conversion and Management*, 143:252–274, 2017. URL https://doi.org/10.1016/j.enconman.2017.04.019.

[50] Zi Lin, Xiaolei Liu, and Maurizio Collu. Wind power prediction based on high-frequency scada data along with isolation forest and deep learning neural networks. *International Journal of Electrical Power & Energy Systems*, 118:105835, 2020.

[51] Fermín Rodríguez, Ane M Florez-Tapia, Luis Fontán, and Ainhoa Galarza. Very short-term wind power density forecasting through artificial neural networks for microgrid control. *Renewable Energy*, 145:1517–1527, 2020.

[52] Jordan Nielson, Kiran Bhaganagar, Rajitha Meka, and Adel Alaeddini. Using atmospheric inputs for artificial neural networks to improve wind turbine power prediction. *Energy*, 190:116273, 2020.

[53] Xiaohui Yuan, Qingxiong Tan, Xiaohui Lei, Yanbin Yuan, and Xiaotao Wu. Wind power prediction using hybrid autoregressive fractionally integrated moving average and least square support vector machine. *Energy*, 129:122–137, 2017.

[54] ZO Olaofe and KA Folly. Wind power estimation using recurrent neural network technique. In *Proceedings of the IEEE Power and Energy Society Conference and Exposition in Africa: Intelligent Grid Integration of Renewable Energy Resources*, pages 1–7. IEEE, 2012.

[55] Q. Zhu, H. Li, Z. Wang, J. Chen, and B. Wang. Short-term wind power forecasting based on lstm. *Dianwang Jishu/Power System Technology*, 41:3797–3802, 12 2017.

[56] Qu Xiaoyun, Kang Xiaoning, Zhang Chao, Jiang Shuai, and Ma Xiuda. Short-term prediction of wind power based on deep long short-term memory. In *2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, pages 1148–1152. IEEE, 2016.

[57] Erick López, Carlos Valle, Héctor Allende, Esteban Gil, and Henrik Madsen. Wind power forecasting based on echo state networks and long short-term memory. *Energies*, 11(3):526, 2018.

[58] Ruiguo Yu, Jie Gao, Mei Yu, Wenhuan Lu, Tianyi Xu, Mankun Zhao, Jie Zhang, Ruixuan Zhang, and Zhuo Zhang. Lstm-efg for wind power forecasting based on

sequential correlation features. *Future Generation Computer Systems*, 93:33–42, 2019.

[59] Qin Shanshan, Feng Liu, Jianzhou Wang, and Yiliao Song. Interval forecasts of a novelty hybrid model for wind speeds. *Energy Reports*, 1:8–16, 11 2015.

[60] Zhichao Shi, Hao Liang, and Venkata Dinavahi. Direct interval forecast of uncertain wind power based on recurrent neural networks. *IEEE Transactions on Sustainable Energy*, 9(3):1177–1187, 2017.

[61] Lu Peng, Shan Liu, Rui Liu, and Lin Wang. Effective long short-term memory with differential evolution algorithm for electricity price prediction. *Energy*, 162: 1301–1314, 2018.

[62] Nikolaos Gorgolis, Ioannis Hatzilygeroudis, Zoltan Istenes, and Lazlo-Grad Gyenne. Hyperparameter optimization of lstm network models through genetic algorithm. In *Proceedings of the 10th International Conference on Information, Intelligence, Systems and Applications.*, pages 1–4. IEEE, 2019.

[63] Elyahu Alperovits and Uri Shamir. Design of optimal water distribution systems. *Water Resources Research*, 13(6):885–900, 1977.

[64] Gerald E Quindry, Jon C Liebman, and E Downey Brill. Optimization of looped water distribution systems. *Journal of the Environmental Engineering Division*, 107(4):665–679, 1981.

[65] Laurie J Murphy, Angus R Simpson, and Graeme C Dandy. *Pipe network optimization using an improved genetic algorithm*. Department of Civil and Environmental Engineering, University of Adelaide . . . , 1993.

[66] Angus R Simpson, Graeme C Dandy, and Laurence J Murphy. Genetic algorithms compared to other techniques for pipe optimization. *Journal of Water Resources Planning and Management*, 120(4):423–443, 1994. URL https://doi.org/10.1061/(ASCE)0733-9496(1994)120:4(423).

[67] Graeme C Dandy, Angus R Simpson, and Laurence J Murphy. An improved genetic algorithm for pipe network optimization. *Water Resources Research*, 32(2):449–458, 1996.

[68] Dragan A Savic and Godfrey A Walters. Genetic algorithms for least-cost design of water distribution networks. *Journal of Water Resources Planning and Management*, 123(2):67–77, 1997.

[69] GV Loganathan, JJ Greene, and TJ Ahn. Design heuristic for globally minimum cost water-distribution systems. *Journal of Water Resources Planning and Management*, 121(2):182–192, 1995. URL https://doi.org/10.1061/(ASCE)0733-9496(1995)121:2(182).

[70] Maria da Conceicao Cunha and Joaquim Sousa. Water distribution network design optimization: simulated annealing approach. *Journal of Water Resources Planning and Management*, 125(4):215–221, 1999.

[71] Istvan Lippai, James P Heaney, and Manuel Laguna. Robust water system design with commercial intelligent search optimizers. *Journal of Computing in Civil Engineering*, 13(3):135–143, 1999.

[72] Zong Woo Geem, Joong Hoon Kim, and GV Loganathan. Harmony search optimization: application to pipe network design. *International Journal of Modelling and Simulation*, 22(2):125–133, 2002. URL https://doi.org/10.1080/02286203.2002.11442233.

[73] Zong Woo Geem. Optimal cost design of water distribution networks using harmony search. *Engineering Optimization*, 38(03):259–277, 2006. URL https://doi.org/10.1080/03052150500467430.

[74] Muzaffar M Eusuff and Kevin E Lansey. Optimization of water distribution network design using the shuffled frog leaping algorithm. *Journal of Water Resources Planning and Management*, 129(3):210–225, 2003.

[75] CR Suribabu and TR Neelakantan. Design of water distribution networks using particle swarm optimization. *Urban Water Journal*, 3(2):111–120, 2006. URL https://doi.org/10.1080/15730620600855928.

[76] Idel Montalvo, Joaquín Izquierdo, Rafael Pérez, and Michael M Tung. Particle swarm optimization applied to the design of water supply systems. *Computers & Mathematics with Applications*, 56(3):769–776, 2008.

[77] Holger R Maier, Angus R Simpson, Aaron C Zecchin, Wai Kuan Foong, Kuang Yeow Phang, Hsin Yeow Seah, and Chan Lim Tan. Ant colony optimization for design of water distribution systems. *Journal of water Resources Planning and Management*, 129(3):200–209, 2003.

[78] Aaron C Zecchin, Angus R Simpson, Holger R Maier, and John B Nixon. Parametric study for an ant algorithm applied to water distribution system optimization. *IEEE Transactions on Evolutionary Computation*, 9(2):175–191, 2005.

[79] Aaron C Zecchin, Holger R Maier, Angus R Simpson, Michael Leonard, and John B Nixon. Ant colony optimization applied to water distribution system design: Comparative study of five algorithms. *Journal of Water Resources Planning and Management*, 133(1):87–92, 2007.

[80] Aaron C Zecchin, Angus R Simpson, Holger R Maier, Michael Leonard, Andrew J Roberts, and Matthew J Berrisford. Application of two ant colony optimisation algorithms to water distribution system optimisation. *Mathematical and Computer Modelling*, 44(5-6):451–468, 2006.

[81] Avi Ostfeld and Ariel Tubaltzev. Ant colony optimization for least-cost design and operation of pumping water distribution systems. *Journal of Water Resources Planning and Management*, 134(2):107–118, 2008. URL https://doi.org/10.1061/(ASCE)0733-9496(2008)134:2(107).

[82] R Baños, C Gil, JI Agulleiro, and J Reca. A memetic algorithm for water distribution network design. In *Soft computing in industrial applications*, pages 279–289. Springer, 2007.

[83] CR Suribabu. Differential evolution algorithm for optimal design of water distribution networks. *Journal of Hydroinformatics*, 12(1):66–82, 2010.

[84] A Vasan and Slobodan P Simonovic. Optimization of water distribution network design using differential evolution. *Journal of Water Resources Planning and Management*, 136(2):279–287, 2010.

[85] Feifei Zheng, Aaron C Zecchin, and Angus R Simpson. Self-adaptive differential evolution algorithm applied to water distribution system optimization. *Journal of Computing in Civil Engineering*, 27(2):148–158, 2012.

[86] Riham M Ezzeldin and Berge Djebedjian. Optimal design of water distribution networks using whale optimization algorithm. *Urban Water Journal*, 17(1):14–22, 2020. URL https://doi.org/10.1080/1573062X.2020.1734635.

[87] Tom W Thorpe et al. *A brief review of wave energy*. Harwell Laboratory, Energy Technology Support Unit, 1999.

[88] Hans-Georg Beyer and Hans-Paul Schwefel. Evolution strategies–a comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002. URL https://doi.org/10.1023/A:1015059928466.

[89] Nikolaus Hansen. The cma evolution strategy: a comparing review. *Towards a new evolutionary computation*, pages 75–102, 2006.

[90] JT Scruggs, SM Lattanzio, AA Taflanidis, and IL Cassidy. Optimal causal control of a wave energy converter in a random sea. *Applied Ocean Research*, 42:1–15, 2013. URL https://doi.org/10.1016/j.apor.2013.03.004.

[91] N Yu Sergiienko. Frequency domain model of the three-tether wecs array, 2016.

[92] GX Wu. Radiation and diffraction by a submerged sphere advancing in water waves of finite depth. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 448, pages 29–54. The Royal Society, 1995.

[93] Markus Wagner, Jareth Day, and Frank Neumann. A fast and effective local search algorithm for optimizing the placement of wind turbines. *Renewable Energy*, 51: 64–70, 2013. URL https://doi.org/10.1016/j.renene.2012.09.008.

[94] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[95] Aguston Eiben, Zbigniew Michalewicz, Marc Schoenauer, and Jim Smith. Parameter control in evolutionary algorithms. *Parameter setting in evolutionary algorithms*, pages 19–46, 2007.

[96] Jeffrey Lagarias, James Reeds, Margaret Wright, and Paul Wright. Convergence properties of the nelder–mead simplex method in low dimensions. *SIAM Journal on Optimization*, 9(1):112–147, 1998.

[97] L. D. Mann, A. R. Burns, , and M. E. Ottaviano. Ceto, a carbon free wave power energy provider of the future. In *Proceedings of the 7th European Wave and Tidal Energy Conference*, 2007.

[98] Nataliia Y Sergiienko, Benjamin S Cazzolato, Boyin Ding, and Maziar Arjomandi. Three-tether axisymmetric wave energy converter: estimation of energy delivery. In *Proceedings of The 3rd Asian Wave and Tidal Energy Conference*, 2016.

[99] RPG Mendes, MRA Calado, and SJPS Mariano. Particle swarm and boxs complex optimization methods to design linear tubular switched reluctance generators for wave energy conversion. *Swarm and Evolutionary Computation*, 28:29–41, 2016. URL https://doi.org/10.1016/j.swevo.2015.12.003.

[100] MJ Box. A new method of constrained optimization and a comparison with other methods. *The Computer Journal*, 8(1):42–52, 1965.

[101] KN Krishnanand and Debasish Ghose. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intelligence*, 3(2):87–124, 2009. URL https://doi.org/10.1007/s11721-008-0021-5.

[102] Dídac Rodríguez Arbonès, Nataliia Y Sergiienko, Boyin Ding, Oswin Krause, Christian Igel, and Markus Wagner. Sparse incomplete lu-decomposition for wave farm designs under realistic conditions. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pages 512–524. Springer, 2018.

[103] Noor H Awad, Mostafa Z Ali, and Ponnuthurai N Suganthan. Ensemble of parameters in a sinusoidal differential evolution with niching-based population reduction. *Swarm and Evolutionary Computation*, 39:141–156, 2018. URL https://doi.org/10.1016/j.swevo.2017.09.009.

[104] Dongping Tian, Xiaofei Zhao, and Zhongzhi Shi. Chaotic particle swarm optimization with sigmoid-based acceleration coefficients for numerical function optimization. *Swarm and Evolutionary Computation*, 51:100573, 2019. URL https://doi.org/10.1016/j.swevo.2019.100573.

[105] Gurcan Yavuz and Doğan Aydın. Improved self-adaptive search equation-based artificial bee colony algorithm with competitive local search strategy. *Swarm and Evolutionary Computation*, 51:100582, 10 2019. URL https://doi.org/10.1016/j.swevo.2019.100582.

[106] Jianyang Lyu, Ossama Abdelkhalik, and Lucia Gauchia. Optimization of dimensions and layout of an array of wave energy converters. *Ocean Engineering*, 192: 106543, 2019. URL https://doi.org/10.1016/j.oceaneng.2019.106543.

[107] Marcos Blanco, Pablo Moreno-Torres, Marcos Lafoz, and Dionisio Ramírez. Design parameters analysis of point absorber wec via an evolutionary-algorithm-based dimensioning tool. *Energies*, 8(10):11203–11233, 2015. URL https://doi.org/10.3390/en81011203.

[108] Dídac Rodríguez Arbonès, Boyin Ding, Nataliia Y Sergiienko, and Markus Wagner. Fast and effective multi-objective optimisation of submerged wave energy converters. In *International Conference on Parallel Problem Solving from Nature*, pages 675–685. Springer, 2016. doi: 10.1007/978-3-319-45823-6_63.

[109] Dídac Rodríguez Arbonès, Nataliia Y. Sergiienko, Boyin Ding, Oswin Krause, Christian Igel, and Markus Wagner. Sparse incomplete lu-decomposition for wave farm designs under realistic conditions. In *Proceedings of the Parallel Problem Solving from Nature.*, pages 512–524, Cham, 2018. Springer. ISBN 978-3-319-99253-2. doi: 10.1007/978-3-319-99253-2_41.

[110] Marcos Blanco, Marcos Lafoz, Dionisio Ramirez, Gustavo Navarro, Jorge Torres, and Luis Garcia-Tabares. Dimensioning of point absorbers for wave energy conversion by means of differential evolutionary algorithms. *IEEE Transactions on Sustainable Energy*, 2018. doi: 10.1109/TSTE.2018.2860462.

[111] Markos I Bonovas and Ioannis S Anagnostopoulos. Modelling of operation and optimum design of a wave power take-off system with energy storage. *Renewable Energy*, 147:502–514, 2020. URL https://doi.org/10.1016/j.renene.2019.08.101.

[112] Australian wave energy atlas. http://awavea.csiro.au/, 2016.

[113] F Fàbregas Flavià, Aurélien Babarit, and Alain H Clément. On the numerical modeling and optimization of a bottom-referenced heave-buoy array of wave energy converters. *International Journal of Marine Energy*, 19:1–15, 2017. URL https://doi.org/10.1016/j.ijome.2017.05.004.

[114] Nataliia Sergiienko. Wave energy converter (wec) array simulator. https://au.mathworks.com/matlabcentral/fileexchange/71840-wave-energy-converter-wec-array-simulator?s_tid=prof_contriblnk, February 2020.

[115] Mehdi Neshat. Improved smart local search&nelder-mead for optimizing wecs. https://au.mathworks.com/matlabcentral/fileexchange/73597-improved-smart-local-search-nelder-mead-for-optimizing-wecs?s_tid=prof_contriblnk, October 2019.

[116] Hari Mohan Dubey, Manjaree Pandit, and Bijaya K Panigrahi. An overview and comparative analysis of recent bio-inspired optimization techniques for wind integrated multi-objective power dispatch. *Swarm and Evolutionary Computation*, 38: 12–34, 2018. URL https://doi.org/10.1016/j.swevo.2017.07.012.

[117] S. Mostapha Kalami Heris. Differential evolution (DE) Matlab source code. https://www.mathworks.com/matlabcentral/fileexchange/52897-differential-evolution-de, April 2018.

[118] Noor H Awad, Mostafa Z Ali, Ponnuthurai N Suganthan, and Robert G Reynolds. An ensemble sinusoidal parameter adaptation incorporated with l-shade for solving cec2014 benchmark problems. In *Proceedings of the IEEE congress on evolutionary computation*, pages 2958–2965. IEEE, 2016.

[119] Anne Auger and Nikolaus Hansen. A restart cma evolution strategy with increasing population size. In *Proceedings of the IEEE congress on evolutionary computation*, volume 2, pages 1769–1776. IEEE, 2005.

[120] Nandar Lynn and Ponnuthurai Nagaratnam Suganthan. Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm and Evolutionary Computation*, 24:11–24, 2015. URL https://doi.org/10.1016/j.swevo.2015.05.002.

[121] Nandar Lynn and Ponnuthurai Nagaratnam Suganthan. Heterogeneous comprehensive learning PSO (HCLPSO) algorithm Matlab source code. http://www.ntu.edu.sg/home/epnsugan/, June 2015.

[122] Noor H Awad, Mostafa Z Ali, Ponnuthurai N Suganthan, and Robert G Reynolds. LSHADE-Ep-Sin algorithm Matlab source code. http://www.ntu.edu.sg/home/epnsugan/, July 2016.

[123] Anne Auger and Nikolaus Hansen. A restart CMA-ES with increasing population size (IPOP-CMA-ES) Matlab source code. http://cma.gforge.inria.fr/cmaes_sourcecode_page.html#matlab, April 2012.

[124] Ágoston E Eiben, Robert Hinterding, and Zbigniew Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, 1999. doi: 10.1109/4235.771166.

[125] Jorge Nocedal and Stephen J Wright. *Nonlinear Equations*. Springer, 2006.

[126] Cheng-Long Xiao and Han-Xiong Huang. Optimal design of heating system for rapid thermal cycling mold using particle swarm optimization and finite element method. *Applied Thermal Engineering*, 64(1-2):462–470, 2014. URL https://doi.org/10.1016/j.applthermaleng.2013.12.062.

[127] Mevludin Glavic and Louis Wehenkel. Interior point methods: A survey, short survey of applications to power systems, and research opportunities. *University of Liège, Belgium*, 2004.

[128] Seyedali Mirjalili, Ibrahim Aljarah, Majdi Mafarja, Ali Asghar Heidari, and Hossam Faris. Grey wolf optimizer: Theory, literature review, and application in computational fluid dynamics problems. In *Nature-Inspired Optimizers*, pages 87–105. Springer, 2020.

[129] B Bhattacharya, DL Shrestha, and DP Solomatine. Neural networks in reconstructing missing wave data in sedimentation modelling. In *Proceedings of the International Association of Hydraulic Engineering and Research's XXXth congress*, volume 500, pages 770–778. Thessaloniki Greece, 2003.

[130] E Alexandre, L Cuadra, JC Nieto-Borge, G Candil-Garcia, M Del Pino, and S Salcedo-Sanz. A hybrid genetic algorithm—extreme learning machine approach

for accurate significant wave height reconstruction. *Ocean Modelling*, 92:115–123, 2015. URL https://doi.org/10.1016/j.ocemod.2015.06.010.

[131] Laura Cornejo-Bueno, Adrián Aybar-Ruíz, Silvia Jiménez-Fernández, Enrique Alexandre, Jose Carlos Nieto-Borge, and Sancho Salcedo-Sanz. A grouping genetic algorithm—extreme learning machine approach for optimal wave energy prediction. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, pages 3817–3823. IEEE, 2016.

[132] Laura Cornejo-Bueno, Eduardo C Garrido-Merchán, Daniel Hernández-Lobato, and Sancho Salcedo-Sanz. Bayesian optimization of a hybrid system for robust ocean wave features prediction. *Neurocomputing*, 275:818–828, 2018. URL https://doi.org/10.1016/j.neucom.2017.09.025.

[133] Dripta Sarkar, Emile Contal, Nicolas Vayatis, and Frederic Dias. Prediction and optimization of wave energy converter arrays using a machine learning approach. *Renewable Energy*, 97:504–517, 2016. URL https://doi.org/10.1016/j.renene.2016.05.083.

[134] Scott C James, Yushan Zhang, and Fearghal O'Donncha. A machine learning framework to forecast wave conditions. *Coastal Engineering*, 137:1–10, 2018. URL https://doi.org/10.1016/j.coastaleng.2018.03.004.

[135] Liang Li, Zhiming Yuan, and Yan Gao. Maximization of energy absorption for a wave energy converter using the deep machine learning. *Energy*, 165:340–349, 2018. URL https://doi.org/10.1016/j.energy.2018.09.093.

[136] Kai-Hung Lu, Chih-Ming Hong, and Qiangqiang Xu. Recurrent wavelet-based elman neural network with modified gravitational search algorithm control for integrated offshore wind and wave power generation systems. *Energy*, 170:40–52, 2019. URL https://doi.org/10.1016/j.energy.2018.12.084.

[137] Nataliia Y Sergiienko. *Three-tether wave energy converter: hydrodynamic modelling, performance assessment and control*. PhD thesis, Ph. D. Thesis, University of Adelaide, Adelaide, Australia, 2018.

[138] Mehdi Neshat, Bradley Alexander, Nataliia Y. Sergiienko, and Markus Wagner. A hybrid evolutionary algorithm framework for optimising power take off and placements of wave energy converters. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '19, pages 1293–1301, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6111-8.

[139] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[140] Stefan Droste, Thomas Jansen, and Ingo Wegener. On the analysis of the (1+ 1) evolutionary algorithm. *Theoretical Computer Science*, 276(1-2):51–81, 2002. URL https://doi.org/10.1016/S0304-3975(01)00182-7.

[141] Matthias Kimmel Bryony Collins Albert Cheung Lisa Becker Rohan Boyle, David Strahan et al. Global trends in renewable energy investment 2019. *United Nations Environment Programme*, 2019.

[142] Diego Vicinanza, Jørgen Harck Nørgaard, Pasquale Contestabile, and Thomas Lykke Andersen. Wave loadings acting on overtopping breakwater for energy conversion. *Journal of Coastal Research*, 65(sp2):1669–1674, 2013. URL https://doi.org/10.2112/SI65-282.1.

[143] L Cuadra, S Salcedo-Sanz, JC Nieto-Borge, E Alexandre, and G Rodríguez. Computational intelligence in wave energy: Comprehensive review and case study. *Renewable and Sustainable Energy Reviews*, 58:1223–1246, 2016.

[144] N. Y. Sergiienko. Wave energy converter (wec) array simulator (https://www.mathworks.com/matlabcentral/fileexchange/71840-wave-energy-converter-wec-array-simulator), 2020. Retrieved February 6, 2020.

[145] Nikolaus Hansen. Cma-es: Evolution strategy with covariance matrix adaptation for nonlinear function minimization (http://www.lri.fr/ hansen/purecmaes.m), 2009. Retrieved March 20, 2018.

[146] Ezgi Zorarpacı and Selma Ayşe Özel. A hybrid approach of differential evolution and artificial bee colony for feature selection. *Expert Systems with Applications*, 62:91–103, 2016. URL https://doi.org/10.1016/j.eswa.2016.06.004.

[147] Seyedali Mirjalili and Andrew Lewis. S-shaped versus v-shaped transfer functions for binary particle swarm optimization. *Swarm and Evolutionary Computation*, 9: 1–14, 2013. URL https://doi.org/10.1016/j.swevo.2012.09.002.

[148] Seyedali Mirjalili. Enhanced binary particle swarm optimization (bpso) with 6 new transfer functions (https://www.mathworks.com/matlabcentral/fileexchange/42448-enhanced-binary-particle-swarm-optimization-bpso-with-6-new-transfer-functions), 2020. Retrieved March 21, 2020.

[149] Myles Hollander, Douglas A Wolfe, and Eric Chicken. *Nonparametric statistical methods*, volume 751. John Wiley & Sons, 2013.

[150] GECCO 2019, best paper awards, 2019. URL https://gecco-2019.sigevo.org/index.html/Best+Paper+Awards. Accessed 31 May 2020.

[151] L. D. Mann. Application of ocean observations & analysis: The ceto wave energy project. In *Operational Oceanography in the 21st Century*, pages 721–729. Springer, 2011.

[152] J. Falnes. *Ocean waves and oscillating systems: linear interactions including wave-energy extraction*. Cambridge University Press, 2002.

[153] B. Ding, B. S. Cazzolato, M. Arjomandi, P. Hardy, and B. Mills. Sea-state based maximum power point tracking damping control of a fully submerged oscillating buoy. *Ocean Engineering*, 126:299–312, 2016. URL https://doi.org/10.1016/j.oceaneng.2016.09.020.

[154] J. C. Bezdek and R. J. Hathaway. Convergence of alternating optimization. *Neural, Parallel & Scientific Computations*, 11(4):351–368, 2003.

[155] Ran Cheng and Yaochu Jin. A social learning particle swarm optimization algorithm for scalable optimization. *Information Sciences*, 291:43–60, 2015. URL https://doi.org/10.1016/j.ins.2014.08.039.

[156] Zhenyu Yang, Ke Tang, and Xin Yao. Self-adaptive differential evolution with neighborhood search. In *Proceedings of the IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 1110–1116. IEEE, 2008.

[157] Michael Liebreich. Bloomberg new energy finance summit. *London: Bloomberg New Energy Finance*, pages 1–10, 2013.

[158] Yan Yaobao. Principle and device of the ocean wave energy conversion generation. *Shanghai Scientific and Technical Publishers*, 1(1):21–24, 2013.

[159] Mehdi Neshat, Bradley Alexander, Nataliia Y. Sergiienko, and Markus Wagner. A new insight into the position optimization of wave energy converters by a hybrid local search. *CoRR*, abs/1904.09599:1–23, 2019. URL http://arxiv.org/abs/1904.09599.

[160] Simon Thomas, Mikael Eriksson, Malin Göteman, Martyn Hann, Jan Isberg, and Jens Engström. Experimental and numerical collaborative latching control of wave energy converter arrays. *Energies*, 11(11):1–16, 2018.

[161] S Ribeiro e Silva, RPF Gomes, and AFO Falcao. Hydrodynamic optimization of the ugen: Wave energy converter with u-shaped interior oscillating water column. *International Journal of Marine Energy*, 15:112–126, 2016. URL https://doi.org/10.1016/j.ijome.2016.04.013.

[162] Michael JD Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in optimization and numerical analysis*, pages 51–67. Springer, 1994. doi: 10.1007/978-94-015-8330-5_4.

[163] Aurelien Babarit, Jørgen Hals, M Muliawan, A Kurniawan, Torgeir Moan, and Jorgen Krokstad. Numerical estimation of energy delivery from a selection of wave energy converters–final report. Technical report, Ecole Centrale de Nantes & Norges Teknisk-Naturvitenskapelige Universitet, 2011.

[164] David IM Forehand, Aristides E Kiprakis, Anup J Nambiar, and A Robin Wallace. A fully coupled wave-to-wire model of an array of wave energy converters. *IEEE Transactions on Sustainable Energy*, 7(1):118–128, 2015. URL 10.1109/TSTE.2015.2476960.

[165] NY Sergiienko, BS Cazzolato, M Arjomandi, B Ding, and LSP da Silva. Considerations on the control design for a three-tether wave energy converter. *Ocean Engineering*, 183:469–477, 2019. URL https://doi.org/10.1016/j.oceaneng.2019.04.053.

[166] Yuan Sun, Michael Kirley, and Xiaodong Li. Cooperative co-evolution with online optimizer selection for large-scale optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1079–1086. ACM, 2018.

[167] Wen Long, Ximing Liang, Shaohong Cai, Jianjun Jiao, and Wenzhuan Zhang. A modified augmented lagrangian with improved grey wolf optimization to constrained optimization problems. *Neural Computing & Applications*, 28(1):421–438, 2017. doi: https://doi.org/10.1007/s00521-016-2357-x.

[168] Akash Saxena, Rajesh Kumar, and Swagatam Das. β-chaotic map enabled grey wolf optimizer. *Applied Soft Computing*, 75:84–105, 2019. URL https://doi.org/10.1016/j.asoc.2018.10.044.

[169] Gaganpreet Kaur and Sankalap Arora. Chaotic whale optimization algorithm. *Journal of Computational Design and Engineering*, 5(3):275–284, 2018. URL https://doi.org/10.1016/j.jcde.2017.12.006.

[170] Mohammad Nabi Omidvar, Xiaodong Li, and Ke Tang. Designing benchmark problems for large-scale continuous optimization. *Information Sciences*, 316:419–436, 2015.

[171] Kay Chen Tan, YJ Yang, and Chi Keong Goh. A distributed cooperative coevolutionary algorithm for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 10(5):527–549, 2006.

[172] R Paul Wiegand, William C Liles, and Kenneth A De Jong. The effects of representational bias on collaboration methods in cooperative coevolution. In *Proceedings of the 2002 International Conference on Parallel Problem Solving from Nature*, pages 257–268. Springer, 2002.

[173] Antony W Iorio and Xiaodong Li. A cooperative coevolutionary multiobjective algorithm using non-dominated sorting. In *Proceedings of the 2004 Genetic and Evolutionary Computation Conference*, pages 537–548. Springer, 2004.

[174] Mohammad Nabi Omidvar, Xiaodong Li, and Xin Yao. Smart use of computational resources based on contribution for cooperative co-evolutionary algorithms. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 1115–1122, 2011. URL https://doi.org/10.1145/2001576.2001727.

[175] Yuan Sun, Michael Kirley, and Saman K Halgamuge. A recursive decomposition method for large scale continuous optimization. *IEEE Transactions on Evolutionary Computation*, 22(5):647–661, 2017. doi: 10.1109/TEVC.2017.2778089.

[176] Zhenyu Yang, Ke Tang, and Xin Yao. Large scale evolutionary optimization using cooperative coevolution. *Information Sciences*, 178(15):2985–2999, 2008. URL https://doi.org/10.1016/j.ins.2008.02.017.

[177] Bruno Borgarino, Aurélien Babarit, and Pierre Ferrant. Impact of wave interactions effects on energy absorption in large arrays of wave energy converters. *Ocean Engineering*, 41:79–88, 2012.

[178] Johannes Falnes and Jorgen Hals. Heaving buoys, point absorbers and arrays. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1959):246–277, 2012. ISSN 1364-503X.

[179] Jørgen Hals Todalshaug. Practical limits to the power that can be captured from ocean waves by oscillating bodies. *International Journal of Marine Energy*, 3-4(2013):e70–e81, 2013. ISSN 2214-1669. URL https://doi.org/10.1016/j.ijome.2013.11.012.

[180] Jørgen Hals Todalshaug. *Hydrodynamics of WECs*, pages 139–158. Springer International Publishing, Cham, 2017. ISBN 978-3-319-39889-1.

[181] Paula B Garcia-Rosa, Giorgio Bacelli, and John V Ringwood. Control-informed geometric optimization of wave energy converters: The impact of device motion and force constraints. *Energies*, 8(12):13672–13687, 2015.

[182] Yadong Wen, Weijun Wang, Hua Liu, Longbo Mao, Hongju Mi, Wenqiang Wang, and Guoping Zhang. A shape optimization method of a specified point absorber

wave energy converter for the south china sea. *Energies*, 11(10):2645, 2018. ISSN 1996-1073. URL https://www.mdpi.com/1996-1073/11/10/2645.

[183] Rezvan Alamian, Rouzbeh Shafaghat, and Mohammad Reza Safaei. Multi-objective optimization of a pitch point absorber wave energy converter. *Water*, 11(5):969, 2019. ISSN 2073-4441. URL https://www.mdpi.com/2073-4441/11/5/969.

[184] LiGuo Wang and John V Ringwood. Geometric optimization of a hinge-barge wave energy converter. In *Proceedings of the 13th European Wave and Tidal Energy Conference*, page 1389, 2019.

[185] Arthur Pecher and Jens Peter Kofoed. *Handbook of Ocean Wave Energy*. Ocean Engineering & Oceanography. Springer International Publishing, 2017. ISBN 978-3-319-39888-4.

[186] AP McCabe. Constrained optimization of the shape of a wave energy collector by genetic algorithm. *Renewable Energy*, 51:274–284, 2013. ISSN 0960-1481.

[187] A Garcia-Teruel, D. I. M. Forehand, and Henry Jeffrey. Metrics for wave energy converter hull geometry optimisation. In *Proceedings of the 13th European Wave and Tidal Energy Conference*. EWTEC, 2019.

[188] Caitlyn E. Clark and Bryony DuPont. Reliability-based design optimization in offshore renewable energy systems. *Renewable and Sustainable Energy Reviews*, 97: 390–400, 2018. ISSN 1364-0321. URL https://doi.org/10.1016/j.rser.2018.08.030.

[189] Caitlyn Clark, Anna Garcia-Teruel, Bryony DuPont, and David Forehand. Towards reliability-based geometry optimization of a point-absorber with pto reliability objectives. In *Proceedings of the 13th Europian Wave and Tidal Energy Conference*, pages 1365–1 – 1365–10, 2019.

[190] A Garcia-Teruel, D. I. M. Forehand, and Henry Jeffrey. Wave energy converter hull design for manufacturability and reduced lcoe. In *Proceedings of the 7th International Conference on Ocean Energy*, pages 1–9. ICOE2018, 2019.

[191] N. Y. Sergiienko, B. S. Cazzolato, B. Ding, and M. Arjomandi. An optimal arrangement of mooring lines for the three-tether submerged point-absorbing wave energy converter. *Renewable Energy*, 93:27–37, 2016. ISSN 0960-1481. URL https://doi.org/10.1016/j.renene.2016.02.048.

[192] J. T. Scruggs, S. M. Lattanzio, A. A. Taflanidis, and I. L. Cassidy. Optimal causal control of a wave energy converter in a random sea. *Applied Ocean Research*, 42 (2013):1–15, 2013. ISSN 0141-1187.

[193] N. Y. Sergiienko, A. Rafiee, B. S. Cazzolato, B. Ding, and M. Arjomandi. Feasibility study of the three-tether axisymmetric wave energy converter. *Ocean Engineering*, 150:221–233, 2018. URL https://doi.org/10.1016/j.oceaneng.2017.12.055.

[194] N.Y. Sergiienko, B.S. Cazzolato, M. Arjomandi, B. Ding, and L.S.P. da Silva. Considerations on the control design for a three-tether wave energy converter. *Ocean Engineering*, 183:469 – 477, 2019. ISSN 0029-8018. URL https://doi.org/10.1016/j.oceaneng.2019.04.053.

[195] Matt Folley. *Numerical modelling of wave energy converters: State-of-the-art techniques for single devices and arrays*. Elsevier Science, Saint Louis, 2016.

[196] L.S.P. Silva, N.Y. Sergiienko, C.P. Pesce, B. Ding, B. Cazzolato, and H.M. Morishita. Stochastic analysis of nonlinear wave energy converters via statistical linearization. *Applied Ocean Research*, 95:102023, 2020. ISSN 0141-1187. URL https://doi.org/10.1016/j.apor.2019.102023.

[197] S.F. Hoerner. *Fluid-dynamic drag: Practical information on aerodynamic drag and hydrodynamic resistance*. Hoerner Fluid Dynamics, 1965.

[198] Adrian De Andres, Jéromine Maillet, Jørgen Hals Todalshaug, Patrik Möller, David Bould, and Henry Jeffrey. Techno-economic related metrics for a wave energy converters feasibility assessment. *Sustainability*, 8(11):1109, 2016. URL https://doi.org/10.3390/su8111109.

[199] A Kai Qin, Vicky Ling Huang, and Ponnuthurai N Suganthan. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on Evolutionary Computation*, 13(2):398–417, 2008. doi: 10.1109/TEVC.2008.927706.

[200] A Babarit, J Hals, M Muliawan, A Kurniawan, T Moan, and J Krokstad. Numerical estimation of energy delivery from a selection of wave energy converters – final report. Report, Ecole Centrale de Nantes & Norges Teknisk-Naturvitenskapelige Universitet, 2011.

[201] M Majidi Nezhad, D Groppi, P Marzialetti, L Fusilli, G Laneve, F Cumo, and D Astiaso Garcia. Wind energy potential analysis using sentinel-1 satellite: A review and a case study on mediterranean islands. *Renewable and Sustainable Energy Reviews*, 109:499–513, 2019.

[202] M Majidi Nezhad, D Groppi, F Rosa, G Piras, F Cumo, and D Astiaso Garcia. Nearshore wave energy converters comparison and mediterranean small island grid integration. *Sustainable Energy Technologies and Assessments*, 30:68–76, 2018.

[203] Jeff Lerner, Michael Grundmeyer, and Matt Garvert. The importance of wind forecasting. *Renewable Energy Focus*, 10(2):64–66, 2009. URL https://doi.org/10.1016/S1755-0084(09)70092-4.

[204] Jürgen Schmidhuber and Sepp Hochreiter. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[205] Nikolaus Hansen and Stefan Kern. Evaluating the cma evolution strategy on multimodal test functions. In *Proceedings of the International Conference on Parallel Problem Solving from Nature*, pages 282–291. Springer, 2004.

[206] Guoliang Luo, Erli Dan, Xiaochun Zhang, and Yiwei Guo. Why the wind curtailment of northwest china remains high. *Sustainability*, 10(2):570, 2018. URL https://doi.org/10.3390/su10020570.

[207] Kurian J Vachaparambil, Eric Aby Philips, Karan Issar, Vaidehi Parab, Divya Bahadur, and Sat Gosh. Optimal siting considerations for proposed and extant wind farms in india. *Energy Procedia*, 52:48–58, 2014.

[208] US Fish, Wildlife Service, et al. Biological opinion on the alta east wind project, kern county, california.(3031 (p), caca-052537, cad000. 06)(8-8-13-f-19). memorandum to supervisory projects manager. *Renewable Energy*, 8, 2013.

[209] Vinodh K Natarajan and Jebagnanam Cyril Kanmony. Sustainability in india through wind: a case study of muppandal wind farm in india. *International Journal of Green Economics*, 8(1):19–36, 2014.

[210] Shiwei Xia, KW Chan, Xiao Luo, Siqi Bu, Zhaohao Ding, and Bin Zhou. Optimal sizing of energy storage system and its cost-benefit analysis for power grid planning with intermittent wind generation. *Renewable Energy*, 122:472–486, 2018. URL https://doi.org/10.1016/j.renene.2018.02.010.

[211] Hüseyin Akçay and Tansu Filik. Short-term wind speed forecasting by spectral analysis from long-term observations with missing values. *Applied Energy*, 191: 653–662, 2017.

[212] Ergin Erdem and Jing Shi. Arma based approaches for forecasting the tuple of wind speed and direction. *Applied Energy*, 88(4):1405–1414, 2011. URL https://doi.org/10.1016/j.apenergy.2010.10.031.

[213] Chuanjin Yu, Yongle Li, Huoyue Xiang, and Mingjin Zhang. Data mining-assisted short-term wind speed forecasting by wavelet packet decomposition and elman neural network. *Journal of Wind Engineering and Industrial Aerodynamics*, 175: 136–143, 2018.

[214] Abdullah Gani, Kasra Mohammadi, Shahaboddin Shamshirband, Torki A Altameem, Dalibor Petković, and Sudheer Ch. A combined method to estimate wind speed distribution based on integrating the support vector machine with firefly algorithm. *Environmental Progress & Sustainable Energy*, 35(3):867–875, 2016. URL https://doi.org/10.1002/ep.12262.

[215] Chuanjin Yu, Yongle Li, and Mingjin Zhang. An improved wavelet transform using singular spectrum analysis for wind speed forecasting based on elman neural network. *Energy Conversion and Management*, 148:895–904, 2017.

[216] Ladislav Zjavka. Wind speed forecast correction models using polynomial neural networks. *Renewable Energy*, 83:998–1006, 2015. URL https://doi.org/10.1016/j.renene.2015.04.054.

[217] Hasan Masrur, Meas Nimol, Mohammad Faisal, and Sk Md Golam Mostafa. Short term wind speed forecasting using artificial neural network: A case study. In *Proceedings of the International Conference on Innovations in Science, Engineering and Technology*, pages 1–5. IEEE, 2016.

[218] Shouxiang Wang, Na Zhang, Lei Wu, and Yamin Wang. Wind speed forecasting based on the hybrid ensemble empirical mode decomposition and ga-bp neural network method. *Renewable Energy*, 94:629–636, 2016. URL https://doi.org/10.1016/j.renene.2016.03.103.

[219] Qinghua Hu, Rujia Zhang, and Yucan Zhou. Transfer learning for short-term wind speed prediction with deep neural networks. *Renewable Energy*, 85:83–95, 2016. URL https://doi.org/10.1016/j.renene.2015.06.034.

[220] HZ Wang, GB Wang, GQ Li, JC Peng, and YT Liu. Deep belief network based deterministic and probabilistic wind speed forecasting approach. *Applied Energy*, 182:80–93, 2016.

[221] Hui Liu, Xi-wei Mi, and Yan-fei Li. Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and elman neural network. *Energy Conversion and Management*, 156:498–514, 2018.

[222] Hui Liu, Xiwei Mi, and Yanfei Li. Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, lstm network and elm. *Energy Conversion and Management*, 159: 54–64, 2018.

[223] Jie Chen, Guo-Qiang Zeng, Wuneng Zhou, Wei Du, and Kang-Di Lu. Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction

and extremal optimization. *Energy Conversion and Management*, 165:681–695, 2018.

[224] Wenyu Zhang, Jujie Wang, Jianzhou Wang, Zengbao Zhao, and Meng Tian. Short-term wind speed forecasting based on a hybrid model. *Applied Soft Computing*, 13 (7):3225–3233, 2013. URL https://doi.org/10.1016/j.asoc.2013.02.016.

[225] Mahdi Khodayar and Jianhui Wang. Spatio-temporal graph deep neural network for short-term wind speed forecasting. *IEEE Transactions on Sustainable Energy*, 10(2):670–681, 2018.

[226] You Lin, Ming Yang, Can Wan, Jianhui Wang, and Yonghua Song. A multi-model combination approach for probabilistic wind power forecasting. *IEEE Transactions on Sustainable Energy*, 10(1):226–237, 2018. doi: 10.1109/TSTE.2018.2831238.

[227] Ekaterina Vladislavleva, Tobias Friedrich, Frank Neumann, and Markus Wagner. Predicting the energy output of wind farms based on weather data: Important variables and their correlation. *Renewable Energy*, 50:236–243, 2013.

[228] Sinvaldo Rodrigues Moreno and Leandro dos Santos Coelho. Wind speed forecasting approach based on singular spectrum analysis and adaptive neuro fuzzy inference system. *Renewable Energy*, 126:736–754, 2018.

[229] S Belaid and A Mellit. Prediction of daily and mean monthly global solar radiation using support vector machine in an arid climate. *Energy Conversion and Management*, 118:105–118, 2016.

[230] Kenneth O Stanley, Jeff Clune, Joel Lehman, and Risto Miikkulainen. Designing neural networks through neuroevolution. *Nature Machine Intelligence*, 1(1):24–35, 2019.

[231] Nikolaus Hansen and Anne Auger. Principled design of continuous stochastic search: From theory to practice. In *Theory and principled methods for the design of metaheuristics*, pages 145–180. Springer, 2014.

[232] Hugo Miguel Inácio Pousinho, Víctor Manuel Fernandes Mendes, and João Paulo da Silva Catalão. A hybrid pso–anfis approach for short-term wind power prediction in portugal. *Energy Conversion and Management*, 52(1):397–402, 2011.

[233] M Lydia, S Suresh Kumar, A Immanuel Selvakumar, and G Edwin Prem Kumar. Linear and non-linear autoregressive models for short-term wind speed forecasting. *Energy Conversion and Management*, 112:115–124, 2016. URL https://doi.org/10.1016/j.enconman.2016.01.007.

[234] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[235] Bahareh Nakisa, Mohammad Naim Rastgoo, Andry Rakotonirainy, Frederic Maire, and Vinod Chandran. Long short term memory hyperparameter optimization for a neural network based emotion recognition framework. *IEEE Access*, 6:49325–49338, 2018.

[236] Aowabin Rahman, Vivek Srikumar, and Amanda D Smith. Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks. *Applied Energy*, 212:372–385, 2018.

[237] Chu Zhang, Jianzhong Zhou, Chaoshun Li, Wenlong Fu, and Tian Peng. A compound structure of elm based on feature selection and parameter optimization using hybrid backtracking search algorithm for wind speed forecasting. *Energy Conversion and Management*, 143:360–376, 2017. URL https://doi.org/10.1016/j.enconman.2017.04.007.

[238] Tuhfe Göçmen and Gregor Giebel. Estimation of turbulence intensity using rotor effective wind speed in lillgrund and horns rev-i offshore wind farms. *Renewable Energy*, 99:524–532, 2016.

[239] J.Å. Dahlberg. Assessment of the lillgrund wind farm: Power performance wake effects, 2009. URL http://www.vattenfall.se/sv/file/15_Assessment_of_the_Lillgrund_W.pdf_16596737.pdf.

[240] Tarmo Soomere. Anisotropy of wind and wave regimes in the baltic proper. *Journal of Sea Research*, 49(4):305–316, 2003.

[241] Leslie N. Smith. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay, 2018.

[242] Ivan Komusanac, Daniel Fraile, and Guy Brindley. Wind energy in europe in 2018, trends and statistics. *Wind Europe*, 2019.

[243] M Lydia, S Suresh Kumar, A Immanuel Selvakumar, and G Edwin Prem Kumar. A comprehensive review on wind turbine power curve modeling techniques. *Renewable and Sustainable Energy Reviews*, 30:452–460, 2014.

[244] Ahmed E Saleh, Mohamed S Moustafa, Khaled M Abo-Al-Ez, and Ahmed A Abdullah. A hybrid neuro-fuzzy power prediction system for wind energy generation. *International Journal of Electrical Power & Energy Systems*, 74:384–395, 2016.

[245] Majid Morshedizadeh, Mojtaba Kordestani, Rupp Carriveau, David SK Ting, and Mehrdad Saif. Improved power curve monitoring of wind turbines. *Wind engineering*, 41(4):260–271, 2017.

[246] A Kai Qin and Ponnuthurai N Suganthan. Self-adaptive differential evolution algorithm for numerical optimization. In *Proceedings of the IEEE congress on evolutionary computation*, volume 2, pages 1785–1791. IEEE, 2005.

[247] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1-14, pages 281–297. Oakland, CA, USA, 1967.

[248] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.

[249] Matthias Lange and Ulrich Focken. *Physical approach to short-term wind power prediction*, volume 208. Springer, 2006.

[250] Christoph Bergmeir and José M Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213, 2012. URL https://doi.org/10.1016/j.ins.2011.12.028.

[251] Rajesh G Kavasseri and Krithika Seetharaman. Day-ahead wind speed forecasting using f-arima models. *Renewable Energy*, 34(5):1388–1393, 2009.

[252] Kanna Bhaskar and SN Singh. Awnn-assisted wind power forecasting using feed-forward neural network. *IEEE Transactions on Sustainable Energy*, 3(2):306–315, 2012.

[253] Yaoyao He and Haiyan Li. Probability density forecasting of wind power using quantile regression neural network and kernel density estimation. *Energy Conversion and Management*, 164:374–384, 2018.

[254] Barbara G Brown, Richard W Katz, and Allan H Murphy. Time series models to simulate and forecast wind speed and wind power. *Journal of Climate and Applied Meteorology*, 23(8):1184–1195, 1984.

[255] Carlos D Zuluaga, Mauricio A Alvarez, and Eduardo Giraldo. Short-term wind speed prediction based on robust kalman filtering: An experimental comparison. *Applied Energy*, 156:321–330, 2015.

[256] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[257] Qiuyi Huang, Yue Cui, Lina Bertling Tjernberg, and Pramod Bangalore. Wind turbine health assessment framework based on power analysis using machine learning method. In *Proceedings of the IEEE Innovative Smart Grid Technologies Europe.*, pages 1–5. IEEE, 2019.

[258] Mon-Fong Jiang, Shian-Shyong Tseng, and Chih-Ming Su. Two-phase clustering process for outliers detection. *Pattern Recognition Letters*, 22(6-7):691–700, 2001.

[259] Qiuyi Huang, Yue Cui, Lina Bertling Tjernberg, and Pramod Bangalore. Wind turbine health assessment framework based on power analysis using machine learning method. In *Proceedings of the IEEE Innovative Smart Grid Technologies Europe.*, pages 1–5, 09 2019.

[260] Yue Cui, Pramod Bangalore, and Lina Bertling Tjernberg. An anomaly detection approach based on machine learning and scada data for condition monitoring of wind turbines. In *Proceedings of the IEEE International Conference on Probabilistic Methods Applied to Power Systems.*, pages 1–6. IEEE, 2018.

[261] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1), 2015.

[262] Hamid Moeini and Farhad Mohammad Torab. Comparing compositional multivariate outliers with autoencoder networks in anomaly detection at hamich exploration area, east of iran. *Journal of Geochemical Exploration*, 180:15–23, 2017.

[263] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. doi: 10.1109/TPAMI.2013.50.

[264] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, pages 1019–1027, 2016.

[265] Josef Tvrdík, L Misik, and I Krivy. Competing heuristics in evolutionary algorithms. *Intelligent Technologies: Theory and Applications*, pages 159–165, 2002.

[266] Shengchao Zhou, Lining Xing, Xu Zheng, Ni Du, Ling Wang, and Qingfu Zhang. A self-adaptive differential evolution algorithm for scheduling a single batch-processing machine with arbitrary job sizes and release times. *IEEE Transactions on Cybernetics*, 2019.

[267] Seyedali Mirjalili. How effective is the grey wolf optimizer in training multi-layer perceptrons. *Applied Intelligence*, 43(1):150–161, 2015. URL https://doi.org/10.1007/s10489-014-0645-7.

[268] Todd K Leen and Genevieve B Orr. Optimal stochastic search and adaptive momentum. In *Advances in neural information processing systems*, pages 477–484, 1994.

[269] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[270] O Fujiwara, B Jenchaimahakoon, and NCP Edirishinghe. A modified linear programming gradient method for optimal design of looped water distribution networks. *Water Resources Research*, 23(6):977–982, 1987.

[271] Okitsugu Fujiwara and Do Ba Khang. A two-phase decomposition method for optimal design of looped water distribution networks. *Water Resources Research*, 26(4):539–549, 1990.

[272] Zheng Y Wu, Paul F Boulos, Chun Hou Orr, and Jun Je Ro. Using genetic algorithms to rehabilitate distribution systems. *Journal-American Water Works Association*, 93(11):74–85, 2001.

[273] AWM Ng and BJC Perera. Selection of genetic algorithm operators for river water quality model calibration. *Engineering Applications of Artificial Intelligence*, 16 (5-6):529–541, 2003. doi: https://doi.org/10.1016/j.engappai.2003.09.001.

[274] Edward Keedwell and Soon-Thiam Khu. A hybrid genetic algorithm for the design of water distribution networks. *Engineering Applications of Artificial Intelligence*, 18(4):461–472, 2005.

[275] Weiwei Bi, Graeme C Dandy, and Holger R Maier. Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. *Environmental Modelling & Software*, 69:370–381, 2015.

[276] Zong Woo Geem. Particle-swarm harmony search for water network design. *Engineering Optimization*, 41(4):297–311, 2009.

[277] Feifei Zheng, Aaron C Zecchin, Jeffery P Newman, Holger R Maier, and Graeme C Dandy. An adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems. *IEEE Transactions on Evolutionary Computation*, 21(5):773–791, 2017.

[278] Min-Der Lin, Yu-Hsin Liu, Gee-Fon Liu, and Chien-Wei Chu. Scatter search heuristic for least-cost design of water distribution networks. *Engineering Optimization*, 39(7):857–876, 2007.

[279] Idel Montalvo, Joaquín Izquierdo, Rafael Pérez-García, and Manuel Herrera. Improved performance of pso with self-adaptive parameters for computing the optimal design of water supply systems. *Engineering Applications of Artificial Intelligence*, 23(5):727–735, 2010.

[280] Kazem Mohammadi Aghdam, Iraj Mirzaee, Nader Pourmahmood, and Mohammad Pourmahmood Aghababa. Design of water distribution networks using accelerated momentum particle swarm optimisation technique. *Journal of Experimental & Theoretical Artificial Intelligence*, 26(4):459–475, 2014. URL https://doi.org/10.1080/0952813X.2013.863227.

[281] Feifei Zheng, Angus R Simpson, and Aaron Zecchin. A performance comparison of differential evolution and genetic algorithm variants applied to water distribution system optimization. In *Proceedings of the World Environmental and Water Resources Congress.*, pages 2954–2963, 2012.

[282] Feifei Zheng, Angus R Simpson, and Aaron C Zecchin. Coupled binary linear programming–differential evolution algorithm approach for water distribution system optimization. *Journal of Water Resources Planning and Management*, 140(5): 585–597, 2013.

[283] A Sedki and Driss Ouazar. Hybrid particle swarm optimization and differential evolution for optimal design of water distribution systems. *Advanced Engineering Informatics*, 26(3):582–591, 2012.

[284] Juan Reca and Juan Martínez. Genetic algorithms for the design of looped irrigation water distribution networks. *Water Resources Research*, 42(5):1–9, 2006.

[285] Bryan A Tolson, Masoud Asadzadeh, Holger R Maier, and Aaron Zecchin. Hybrid discrete dynamically dimensioned search (hd-dds) algorithm for water distribution system design optimization. *Water Resources Research*, 45(12):1–15, 2009.

[286] Bryan A Tolson and Christine A Shoemaker. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resources Research*, 43(1):1–16, 2007.

[287] Urban Škvorc, Tome Eftimov, and Peter Korošec. Gecco black-box optimization competitions: Progress from 2009 to 2018. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 275–276, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6748-6.

[288] Claudius M Bürger, Peter Bayer, and Michael Finkel. Algorithmic funnel-and-gate system design optimization. *Water Resources Research*, 43(8):1–19, 2007. URL https://doi.org/10.1029/2006WR005058.

[289] Peter Bayer, Emre Duran, Rainer Baumann, and Michael Finkel. Optimized groundwater drawdown in a subsiding urban mining area. *Journal of Hydrology*, 365(1-2):95–104, 2009.

[290] S Belaqziz, Sylvain Mangiarotti, M Le Page, S Khabba, S Er-Raki, T Agouti, Laurent Drapeau, MH Kharrou, M El Adnani, and Lionel Jarlan. Irrigation scheduling of a classical gravity network based on the covariance matrix adaptation–evolutionary strategy algorithm. *Computers and Electronics in Agriculture*, 102: 64–72, 2014.

[291] Akinola Ikudayisi and Josiah Adeyemo. Irrigation water optimization using evolutionary algorithms. *Environmental Economics*, 6(1):200–205, 2015.

[292] Jens Grundmann, Niels Schütze, Gerd H Schmitz, and Saif Al-Shaqsi. Towards an integrated arid zone water management using simulation-based optimisation. *Environmental Earth Sciences*, 65(5):1381–1394, 2012.

[293] G Romero-Tapia, MJ Fuente, and Vicenç Puig. Leak localization in water distribution networks using fisher discriminant analysis. *IFAC-Papers OnLine*, 51(24): 929–934, 2018.

[294] Uli Maier, Cecilia DeBiase, Oliver Baeder-Bederski, and Peter Bayer. Calibration of hydraulic parameters for large-scale vertical flow constructed wetlands. *Journal of Hydrology*, 369(3-4):260–273, 2009. URL https://doi.org/10.1016/j.jhydrol.2009.02.032.

[295] Frank Neumann and Ingo Wegener. Randomized local search, evolutionary algorithms, and the minimum spanning tree problem. *Theoretical Computer Science*, 378(1):32–40, 2007.

[296] Feifei Zheng, Angus R Simpson, and Aaron C Zecchin. Optimal rehabilitation for large water distribution systems using genetic algorithms. *Australia Water Association*, 2011.

[297] Gideon Eiger, Uri Shamir, and Aharon Ben-Tal. Optimal design of water distribution networks. *Water Resources Research*, 30(9):2637–2646, 1994.

[298] Andrea Bolognesi, Cristiana Bragalli, Angela Marchi, and Sandro Artina. Genetic heritage evolution by stochastic transmission in the optimal design of water distribution networks. *Advances in Engineering Software*, 41(5):792–801, 2010.

[299] Razi Sheikholeslami, Aaron C Zecchin, Feifei Zheng, and Siamak Talatahari. A hybrid cuckoo–harmony search algorithm for optimal design of water distribution systems. *Journal of Hydroinformatics*, 18(3):544–563, 2016.