COMBINING NATURE-INSPIRED ALGORITHMS FOR OPTIMIZING DRONE POSITIONING

LEE BOON HAO

UNIVERSITI TUNKU ABDUL RAHMAN

COMBINING NATURE-INSPIRED ALGORITHMS FOR OPTIMIZING DRONE POSITIONING

LEE BOON HAO

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

September 2024

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature :

Name : Lee Boon Hao

ID No. : 2106860

Date : 8/3/2024

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"COMBINING NATURE-INSPIRED ALGORITHMS FOR OPTIMIZING DRONE POSITIONING"** was prepared by **LEE BOON HAO** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature :

Supervisor : Dr. Lee Ying Loong

Date : 29 September 2024

Signature : kckhor

Co-Supervisor : Khor Kok Chin

Date : 29-9-2024

ACKNOWLEDGEMENTS

I would like to express sincere gratitude to my research supervisor Dr. Lee Ying Loong and co-supervisor Dr. Khor Kok Chin for their constant support throughout the project. I am grateful for their knowledge, companionship, enthusiasm and openness to communication, which made this research project a very insightful and enjoyable experience.

In addition, I would like to extend my thanks to Dr. Lee Ying Loong and Dr. Khor Kok Chin for their guidance, advice and patience in helping me with my research, as well as for assisting me with writing and submitting a research paper on this project to the International Conference on Information Technology and Digital Applications (ICITDA 2024),which has been conditional accepted as of 31st of August 2024. They have helped me greatly in optimising my research methodology and presenting the results in a clear and concise manner.

ABSTRACT

The crayfish optimization algorithm (COA) and mountain gazelle optimization algorithm (MGO) have emerged as two powerful metaheuristics for global optimization. It has been shown that these two metaheuristics outperform conventional metaheuristics and are capable of finding near-optimal solutions for most of the benchmark functions. However, in certain benchmark functions, COA and MGO do not perform better than the conventional metaheuristics. To this end, a new Hybrid Cooperative COA-MGO algorithm (HCCMGA) is proposed. In the proposed HCCMGA algorithm, the solution vector is split into two solution subvectors of smaller dimensions, one optimized by COA and the other by MGO. The fitness of each search agent in COA and MGA is evaluated in a cooperative manner, whereby the solution subvector in COA is combined with the best solution subvector from MGO and vice versa. The proposed HCCMGA is evaluated using 13 benchmark functions and is compared with several state-of-the-art metaheuristics. Results show that the proposed HCCMGA outperforms state-of-the-arts optimization algorithms in solution quality, search stability and convergence speed, as the HCCMGA achieves solutions closest to the optimum in eight out of the 13 benchmark functions with low standard deviations and less numbers of iterations. The usage of drones has been increased recently due to its flexibility and accessibility. However, the network connection provided by the drone to ground user is often limited as the drone's positioning of the air has not been optimized to provide the maximum network connection to the ground user. Many different metaheuristics have been used to optimize the drones' position. However, the result from these metaheuristics still cannot reach the optimal solution. To this context, the proposed HCCMGA has been applied to drone base station (DBS) positioning scheme. Results show that the proposed HCCMGA outperforms existing COA and MGO algorithms on DBS positioning scheme as the HCCMGA provides more average coverage radius than COA and MGO.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

3D	Three-Dimensional
5G	Fifth-Generation
6G	Sixth-Generation
ACO	Ant Colony Optimization
AHA	Artificial Hummingbird Algorithm
ALA	Ant Lion Optimizer
AWGN	Additive White Gaussian Noise
BS	Base Station
COA	Crayfish Optimization Algorithm
DBS	Drone Base Station
DCA	Department of Civil Aviation Malaysia
DP	Dynamic Programming
E2E	End-to-end
eMBB	Enhanced Mobile Broadband
GA	Genetic Algorithm
GWO	Grey Wolf Optimization
HCCMGA	Hybrid Cooperative Crayfish Optimization Algorithm and Mountain Gazelle Optimizer
IR	Far-infrared
KPI	Key Performance Indicator
LOS	Line-of-sight
M2M	Machine-to-machine Communication
MGO	Mountain Gazelle Optimization
MWA	Maximal Weighted Area
PSO	Particle Swarm Optimisation
QoS	Quality of Service
SA	Simulated Annealing
SCA	Sine Cosine Algorithm
SNR	Signal-to-noise Ratio
VR	Virtual Reality
WBS	Work Breakdown Structure
WOA	Whale Optimization Algorithm

CHAPTER 1

INTRODUCTION

1.1 Background

Drones are aircrafts which can operate without human on the board. Nowadays, drones have been involved in many sectors, including agricultural, military, surveillance and others. The use of drones has risen drastically due to its accessibility, efficiency, and flexibility. For the sixth generation (6G) communications, drones are responsible for transferring mobile base stations (BSs) and providing network connectivity in certain areas, especially in the locations which are impacted due to natural disasters, or locations that are far from fixed network coverage. These drone BSs (DBSs) can provide greater flexibility in the coverage area, since it can travel without any barrier, and the cost of travelling is cheaper compared to the traditional BS networks.

The positioning of the DBSs plays a huge role in providing the connectivity to the users on the ground. Optimal positioning of the DBSs can deliver good network connectivity to the ground users. However, the performance of DBS networks may be affected by several environmental factors, such as path loss, fading, interference from other wireless devices, and shadowing. These factors may decrease the performance of the DBS networks as they may decline the output signal strengths from the DBSs. Figure 1.1 illustrates three-dimensional (3D) positioning of multiple DBSs in an area.

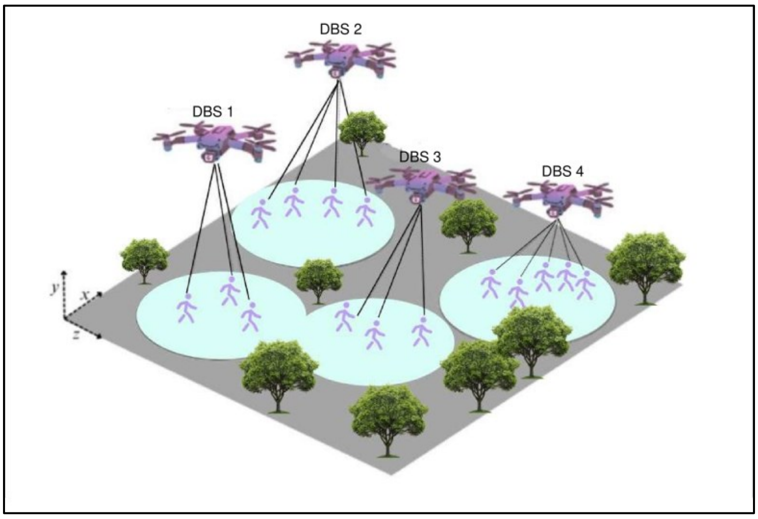

Figure 1.1: 3D positioning of multiple DBSs

In the current literature, different mathematical optimization and heuristic algorithms, such as dynamic programming (DP) (Hadiwardoyo et al., 2020), genetic algorithms (GA) (Sun and Fang, 2023) and particle swarm optimization (PSO) (Zhang and Zhang, 2022), have been proposed to optimize DBS positioning in 6G networks for optimal network coverage. Exhaustive approaches such as brute forcing are not viable solutions for DBS positioning due to their huge complexity which is prohibitively high for practical implementation.

Various metaheuristic optimization algorithms have been developed based on biological evolutions and swarm behaviors for global optimization. Notable metaheuristic algorithms include PSO (Kennedy and Eberhart, 2002), Greywolf Optimization Algorithm (GWO) (Mirjalili, Mirjalili, and Lewis, 2015), Ant Colony Optimization (ACO) (Dorigo et al., 1996), Ant Lion Optimization (ALO) (Mirjalili, 2015), and Genetic Algorithm (GA) (Goldberg, 1989). Recently, the crayfish optimization algorithm (COA) and mountain gazelle optimization (MGO) have emerged as powerful metaheuristics for global optimization. However, these algorithms could not perform well in solving certain problems.

In this project, a DBS positioning scheme based on a new hybrid cooperative combination of crayfish optimization Algorithm (COA) and mountain gazelle optimizer (MGO) algorithm (HCCMGA) will be developed and investigated. This DBS positioning scheme is targeted to not only dynamically determine suitable DBS locations, but also provide average capacity among users. Nevertheless, existing DBS positioning techniques for 6G networks remain far from optimum due to the rapid, random temporal and spatial channel variations.

1.2 Problem Statement

1.2.1 Limitations of Traditional Nature-Inspired Optimization Algorithm

Nature-inspired optimization algorithms are ones belonging to the family of algorithms developed based on behaviours of animals and insects, genetic evolutions and natural phenomena. The main goal of these metaheuristics is to gain the global optimal solution for any given unconstrained optimization problem. In order to achieve this goal, intensive research has been done to improve search diversity of the metaheuristics with a focus on global exploration and local exploitation.

Global exploration is the capability of a metaheuristic in finding good solutions throughout the entire search space of the given optimization problem, while local exploitation is the ability of a metaheuristic in seeking a potentially better solution within a specified region of the search space (Mirjalili and Hashim, 2010). A key design aspect for all metaheuristics is to balance the exploration and exploitation abilities, which is crucial to enhance the ability of the metaheuristics in obtaining optimal solutions within a reasonable amount of time. However, according to Eiben and Schippers (1998), the exploration and exploitation abilities in evolutionary computing often contradict each other, as strengthening the exploration ability may weaken the exploitation ability and vice versa. Many of the nature-inspired optimization algorithms have been proven to be able to solve complex multidimensional problem. However, single nature-inspired optimization algorithm faced difficulties in solving the DBS positioning problem.

According to the "no free lunch" theorem (Wolpert and Macready, 1997), the average performance of a metaheuristic is almost identical to those of other metaheuristics because an algorithm that performs well in solving a set of problems may perform poorly in solving another set of problems. This could be due to varying exploration and exploitation abilities in the design of the metaheuristics. Most of the traditional nature-inspired metaheuristics are developed based on a specific type of natural processes or phenomena, which are used to explore and exploit solutions. Many such metaheuristics are effective and efficient in solving complex optimization problems. However, these metaheuristics, which are often designed solely based on a single specific natural process or phenomenon, are only advantageous for solving certain classes of problems. In view of this, the inherent nature-inspired mechanisms in these metaheuristics could be combined to strengthen both exploration and exploitation of solutions.

1.2.2 High Complexity Solutions

Decreasing complexity plays a huge role in developing effective algorithms for solving the DBS positioning problem. As the algorithms become complex, it will require more processing power and computation time to process these algorithms. However, complex algorithms are not practical for DBS systems, as this system has limited battery life and processing capability. Many existing literature has provided low complexity algorithms to solve the DBS placement problems, such as genetic algorithm (Chen et al., 2018), simulated annealing (Lim et al., 2021), and particle swarm optimization (Li et al., 2018).

However, nowadays many researchers and users try to enhance other algorithms with further reduced complexities, in orders to optimise the DBS battery life duration, processing time and computation resources. Besides that, it is also quite challenging to obtain a good solution for the DBS positioning problem within a satisfactory quality and acceptable complexity. Therefore, developing an algorithm which can be easily implemented in 6G systems, may increases the application flexibility, and decreases the difficulty for future optimisation efforts.

1.2.3 Environmental Challenges for Multi-DBS Positioning

The interference among DBSs is unavoidable when deploying multiple DBSs in a designated area. Besides that, the risk of collisions among the DBSs is also a dominant factor that consititutes to the performance instability for DBS positioning. These issues will affect the network connectivity between the users and the DBSs and may lead to QoS degradation among the users.

In the current literature, many existing DBS positioning schemes do not consider these issues in-depth. As short, many constraints from the DBS 6G network have given a huge challenge to the researchers to find a good nature-inspired optimization algorithm to fit in as the solution for the DBS positioning problem.

1.3 Aim and Objectives

To convey the research gaps mentioned above and secure adequate network connectivity among users, this project aims to develop a multi-DBS positioning scheme for 6G networks with a focal point on the downlink connectivity.

The project objectives are:

1. To develop a hybrid cooperative COA-MGO algorithm (HCCMGA) by using a cooperative learning-based fitness evaluation approach.
2. To enhance the newly developed hybrid nature-inspired optimization algorithm based on the evaluation using standard benchmark functions in terms of solution quality, stability, and convergence.
3. To apply the proposed HCCMGA to solve the DBSs' capacity maximization problem for 6G networks.

1.4 Proposed Methodology

As mentioned previously, a flowchart is developed below to illustrate the steps that will be used to develop a new hybrid optimization algorithm and deploy it for DBS placement. Figure 1.2 shows the flowchart to develop new hybrid optimization algorithm and deploy it for DBS placement.

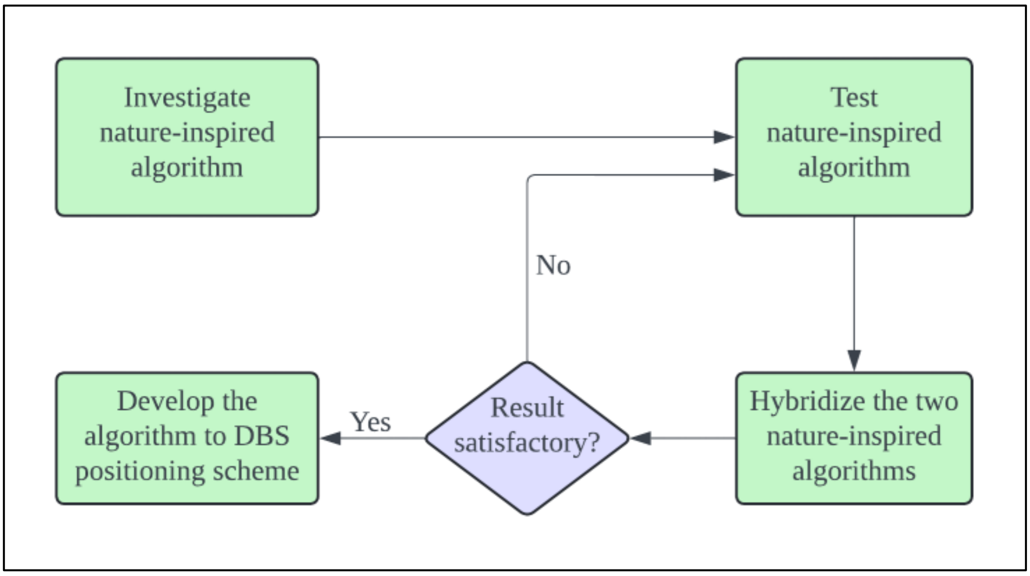

Figure 1.2: Flowchart to develop new hybrid optimization algorithm

Firstly, investigation between different nature-inspired algorithms will be done to identify the nature-inspired algorithms that are suitable to resolve the DBS placement problem. This include understanding the behaviours of different natural, physical phenomena and biological processes in order to learn the different nature-inspired algorithms and to find the best algorithm for hybridization.

Next, several nature-inspired algorithms will be tested with 13 standard benchmark test functions to find the two best nature-inspired algorithms that outperform other algorithms in term of in solution quality, search stability and convergence speed. Later, these two nature-inspired algorithms will be hybridized together to form a new hybrid algorithm. However, in the case that these two nature-inspired algorithms cannot be hybridized together, then, the step will be repeated to choose the next best nature-inspired algorithms until the hybridization process completed.

Lastly, the algorithm will be deployed to the DBS placement problem. When we apply the algorithm to the multiple DBS system model, the path loss, interference, and collision between each DBSs will be considered. To test the effectiveness of the proposed solution, we will conduct simulations within a realistic system model to assess the algorithm's performance. Simulation tools will be used to model the DBS's movements and communication links. The solution will be evaluated based on the average maximum coverage radius received by each DBS.

1.5 Scope and Limitation of the Study

1.5.1 Scope

This project develops a new hybrid algorithm for the 3D positioning of DBS in order to achieve good 6G network coverage. Several points have been highlighted, to outline the project scope.

First of all, the project creates a hybrid nature-inspired optimization algorithm that dynamically optimises the positions of each DBS in a multi-DBS model to provide maximize network capacity among users without affecting the channel allocation for each BS. Secondly, the DBSs' positions must be arranged in such a way that they provide maximum network connectivity and the number of users that can be provided by the DBSs (Mozaffari et al., 2018). The scheme only optimises downlink connection from DBSs to ground users as the project emphasizes on using DBSs to deliver 6G network coverage to users, data transmitted in the uplink direction is typically relatively insignificant. Lastly, the algorithm will be designed and developed assuming that all ground users exist on the same altitude.

1.5.2 Limitations of Study

The implementation of the fairness-aware multi-DBS placement scheme will require multiple simultaneously flying DBSs and a large number of user devices. To obtain the drones and BSs will incur substantial costs, and another challenge is that each drone is required to have a permit issued by the Department of Civil Aviation Malaysia (DCA) before flying can commence. A large airspace is also required to thoroughly test the algorithm, and depending on the location of the airspace, may require further permits before commencing.

Furthermore, obtaining several hundred user devices is also unfeasible due to its high cost. Therefore, testing and implementation of the scheme will be performed in a simulation environment.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

As mentioned in the previous chapter, the DBS positioning scheme in 6G network coverage is one of the popular topics that researchers are investigating nowadays. Many researchers have developed and tested different techniques to find an excellent solution for the DBS positioning problem. Thus, the following literature review provides a brief overview on 6G networks, advantages and challenges of DBSs, and a review on existing DBS positioning schemes.

2.2 6G Networks and Key Technologies

6G mobile networks are the successor to the 5G mobile technology. It will lead to new standards that will achieve performance unreachable by 5G networks. According to the current literature, 6G networks are estimated to perform better than fifth-generation (5G) networks because 6G networks will provide more intelligent networking with lower latency, faster data transmission, and supreme network communication speeds. Mobile communication networks have gone through drastic evolution over the last three decades. Several applications and areas have grown dramatically which include three-dimensional (3D) media, enhanced mobile broadband (eMBB), machine-to-machine (M2M) communications, and virtual reality (VR). The applications and sectors metioned above have grown quickly due to the enormous potential of 5G networks (Banafaa et al., 2023). By 2025, it is expected that approximately 65% of the world's populance will be able to access 5G network. Besides that, it is expected that the amount of wireless data traffic capacity will raise to centuple of equipment in a given cubic meter. Furthermore, spectrum bandwidth is needed for data-hungry applications such as holographic video streaming. This spectrum bandwidth is insufficient in the mm-wave spectrum for 5G networks. These challenges and limitations have inspired the industry to start conceiving the next generation of mobile networks, that is 6G, which is aimed to provide services during 2030s (Alsharif et al., 2020).

Table 2.1: Difference of 6G with 4G and 5G communication system (Chowdhury et al., 2020)

Issue	4G	5G	6G
Per device peak data rate	1 Gbps	10 Gbps	1 Tbps
End-to-end (E2E) latency	100 ms	10 ms	1 ms
Maximum spectral efficiency	15 bps/Hz	30 bps/Hz	100 bps/Hz
Mobility support	Up to 350 km/hr	Up to 500 km/hr	Up to 1000 km/hr
Satellite integration	No	No	Fully
AI	No	Partial	Fully
Maximum frequency	6 GHz	90 GHz	10 THz

Table 2.1 shows the difference between 4G - 6G mobile communications networks. According to this table, there is an extra key performance indicator (KPI) drivers in the 6G mobile communication networks, which is satellite integration. This satellite integration can provide impeccable global coverage in different locations to give user better coverage. Besides that, the per device peak data rate in 6G mobile communication networks will reach 1Tbps, which is 100 times better than the per device peak data rate in 5G networks. Furthermore, the number of mobility support, and the maximum spectral efficiency provided by 6G networks will be better than those of 5G and 4G networks (Chowdhury et al., 2020).

There are several key-enabling technologies for 6G networks. The first important technology for 6G networks is Artificial Intelligence (AI). It is widely known that intelligence is one of the fundamental characteristics for 6G mobile communication networks. From the table 2.1, AI has no involvement for 4G mobile communication networks. However, AI has provided partial support in 5G networks. It is estimated that AI will be a vital component of 6G networks. 6G networks can support the process of conversion from cognitive radio to intelligent radio, and thus provide the full potential of radio signal transmissions (Letaief et al., 2019).

The next crucial technology for 6G networks is TeraHertz communications. To develop ultra high-speed communication networks, it is needed to have high frequencies and broad bandwidths in our mobile communication systems. TeraHertz communications can provide good frequencies and wide bandwidths for the 6G connection systems. The spectrum from the THz bands is between mmWave used by 5G and far-infrared (IR), which can offer broader transmission bandwidth than mmWaves and better propagation than IR. Apart from that, THz communications can be involved in other technological areas, such as indoor wireless mobile networks, and nanoscale communications. There is a limitation from the THz communication systems, that is, the THz antenna design is very challenging. However, despite this limitation, THz still provide various advantages, including high gain and rapid beam scanning (Chen et al., 2020).

Another main technology for 6G networks is quantum communications. Quantum communications and advanced quantum computing can provide attentive security against cyber attacks in 6G networks. It provides high security by appling a quantum key depending on the uncertainty principle and quantum no-cloning theorem. The information received from users is converted to a quantum state by using quantum or photons particles. This information cannot be cloned or accessed without interfering with it, according to the quantum theorem and principles (Huang et al., 2019). Besides that, quantum communications can provide good connectivity for remote communications. Quantum repeaters are important devices in remote global quantum networks, because they can divide the distance of quantum communications into shorter intermediary segments and thus decreasing both photon loss and operation errors (Muralidharan et al., 2016).

Drones are also another essential technology in 6G networks. The main functionalities that drones can provide in 6G networks are delivery, relays and BSs. Firstly, drones can help in delivey especially during the pandemic era. During the Covid-19 pandemic years, drones have been widely used for delivering essential goods and medicines to communities that are located in the lockdown area. Drones can not only minimize human contact, but it can also help people in their basic services. However, the constraint for the drone to provide fast delivery is the consumption of energy from the drone itself (Raja et al., 2023). Next, drone can also act as wireless relays that can provide wireless connections for many users or groups without reliable direct communication links. These wirelee relays are flexible and can oprovide high mobility. Besides that, this wireless relays with the help of drones have become a favourable candidate for producing enhanced wireless connecitivity beyond line-of-sight (LOS) (Zeng et al., 2016). The DBS is another main functionality that drone can provide. The advantages and challenges of DBS will be discussed in the following section.

2.3 DBS advantages and challenges

DBSs are aerial nodes with several BS functionalities and characteristics. Firstly, DBSs can enhance the network range and connectivity for 6G networks. They are reliable and profitable solutions for producing cellular connectivity to inaccessible areas, or terrestrial areas that have finite cellular infrastructure. Furthermore, it is an interesting solution for providing stable, optimal and global wireless connection in unique events, such as natural disasters or concert and sport events. In rural areas, DBSs are estimated to be a continual solution as it can provide cost-effective and long-term connectivity. Moreover, DBSs can also integrate with other physical layer methods which include gigantic MIMO, cognivitive radio, and mmWave. The combination of DBSs with these physical layer methods will be an encouraging solution for data-extensive services (Mozaffari et al., 2019).

However, there are several challenges for DBSs. Firstly, optimal positioning is required to deliver optimal connectivity and network coverage to every ground users in an area (Alzenad et al., 2017). Besides that, interference and risk collision among DBSs in a designated area is also another constraint for DBSs. This is an issue that need to be addressed because the interference and risk collisions among DBSs will decrease the performance as it will degrade the QoS provisioning among users.

2.4 Existing positioning schemes for DBS

Many researchers and academia have use different algorithms to solve the DBS positioning problem. This section will provide comprehensive analysis of the existing positioning schemes for DBS based on non meta-heuristic algorithms and meta-heuristic algorithms. Besides that, this section will identify the similarities, differences and limitations of the existing positioning schemes. The state-of-the-arts in DBS placement will be clearly understood by doing the critical assessment of the existing literature, which will also make it easier to identify any research gaps that need to be filled in the future research.

2.4.1 Existing positioning schemes for DBSs based on non meta-heuristics algorithms

Several existing positioning schemes for DBSs based on non meta-heurtistics algorithm have been developed. In a paper by Akram et al. (2020), the authors seek to solve the essential communication problems in disaster situations with the goal of maximizing the number of covered users with the minimum number of DBSs. The authors have solved this problem by using two methods, which include the Branch and Bound algorithm, and a low complexity heuristics. The authors compare these two methods and the result shows that the simple heuristic solution can achieve better user coverage with a little growth in the number of DBSs. However, this paper only considers optimizing the 2D placement of DBSs and their users' association. Besides that, this paper assumes that the DBS's altitude deployed is constant within the disaster area.

In the study by Hammouti et al. (2021), the DBS positioning problem is solved by using the heuristics greedy algorithm. They try to discover the optimal positions for the DBS and the excellent users association to increase and thus maximize the downlink user rate. Firstly, a binary log-linear learning algorithm is implemented to discover the perfect 3D positioning and users' association that maximize the sum-rate method. Next, they have formulated this problem to become a submodular maximization problem. By doing this step, they will be able to use the greedy approach with a certain performance guarantee. Lastly, a heuristic greedy algorithm has been developed with low information and implementation specifications. According to their simulation, this algorithm can achieve effective results with only a few iterations, and this algorithm is quite simple in implementation. This paper is interesting because it has considered the interference between DBSs and quality of service constraints. However, the DBS's altitude needed to be set in advance before performing the DBS positioning. Furthermore, another problem of this positioning scheme is that, they have no guarantee about the performance this positioning scheme, as the performance is only evaluated based on the simulation, and they are not evaluated based on other benchmark functions. So, it is difficult for us to say that this positioning scheme performs well without adequate evidence.

Meanwhile, in another paper by Alzenad et al. (2018), the aim of this paper is to maximize the number of users with different quality-of-service (QoS) specification. Besides that, this paper attempts to satisfy all the users with different QoS requirements. The authors have tried to use and compare two different algorithms in solving the problem. The first algorithm is the exhaustive search algorithm. In this algorithm, the DBS positioning problem is treated as a multi-circle positioning obstacle and the authors use the exhaustive search method over a 1D parameter in a closed area to find the optimal DBS placement. Then, the authors use another algorithm which is the maximal weighted area (MWA) algorithm to solve the same DBS positioning problem. The result shows that the performances of these two algorithms are very close, however the complexity level for the MWA algorithm is lesser than that of the exhaustive search algorithm. However, this paper do not consider the collision and interference between DBSs which may decrease the performance of the DBS.

Next, in the study by Valiulahi and Masouros (2021), the authors propose a hybrid algorithm to solve the DBS positioning problem. The intention of this paper is to fulfil the QoS specifications of each user with the consideration of co-channel interference and to maximize the system output for all the ground users. The authors first use the mean-shift technique algorithm to find the DBSs' x and y coordinates, which is then connected with the maximum users' density and arrange users to DBSs. Once the x and y coordinates of DBSs are decided, the DBS's altitude is optimized by the second algorithm, which is an iterative algorithm depend on the block coordinate descent method. Then, the performance of this hybrid algorithm is tested with the conventional benchmarks of the circle packing theroem and the concurrent deployment technique with a variable radius in the matter of system sum output and power consumption. Besides that, the hybrid algorithm provides fairness among users in system throughput. However, this paper does not consider the risk of collisions and interference among DBSs.

2.4.2 Existing positioning schemes for DBSs based on meta-heuristic algorithms

Several existing positioning scheme based on meta-heuristic algorithms have been designed and developed. Firstly, in a study by Kalantari et al. (2016), the authors aim to decrease the cost deployment of DBSs by reducing the number of DBSs needed in the system to deliver optimal connectivity for all users in a designated area. The authors first use a meta-heuristic algorithm based on PSO to get the 3D postions of the DBSs for covering all the users with sufficient QoS satisfaction. Then, they reduce the number of needed DBSs by iteratively eliminating the DBS one by one and ensure that the effect of the performance on the system is minimal. The paper has some constraints in which they assume that there are no ground BSs. Besides that, when simulating the algorithm's performance, the authors assume that there are 1000 users in a $100km^2$ region. Lastly, the authors also assume that the maximum altitude of DBSs is 600m.

In another paper of Merwaday et al. (2016), the authors use Genetic Algorithm (GA) to optimize the locations of DBSs. They optimize the locations of DBSs by maximizing the fifth-percentile output of the network for a designated region. However, the authors in this paper have assumed that each DBS will send its location and its users' throughput information to a central server. Furthermore, they use the roulette-wheel selection strategy of GA to determine the best chromosome, which is the candidate solution for the problem. Next, they observe the impact of the optimization deployment based on network performance with several parameters, such as simulation area, DBSs' altitudes, and path-loss exponent. The conclusion that the authors have obtained is that this positioning scheme can act as a quick and reliable alternative solution for the area that suffers from natural calamities or malevolent attacks. However, the authors in this paper do not consider the constraints of the DBS positioning problem which include the collision and interference between DBSs.

Apart from that, in another article by Ouamri et al. (2022), the authors used another meta-heuristic algorithm, which is Grey Wolf Optimizer (GWO) algorithm to find the best placement of DBSs. The authors first analyze the downlink coverage probability according to signal-to-interference plus noise ratio by using stochastic geometry. Then, based on this coverage probability constraint, the authors used the GWO algorithm to find the optimal DBS placement. The ultimate objective of this paper is to maximize the network coverage by raising the number of users covered by DBSs. The authors in this paper do consider the path loss, and interference between DBSs, but they do not consider the risk of collision between each DBS which may lead to the network degradation and thus decrease the network signal quality.

Furthermore, in another paper by Lim et al. (2021), the authors use the simulated annealing (SA) algorithm to develop a DBS positioning scheme with the aim of maximizing the communication coverage. Besides that, this paper has considered the risk of collision between DBSs. The constraint of collisions between DBSs has been added as a penalty function, to ensure that there is a minimum distance exists between each pair of DBSs. During the simulation process, the authors finds out that this positioning scheme achieves better performance than the static DBS positioning scheme in throughput and the number of users with satisfied QoS. Furthermore, it is also found out that the penalty function added for collision avoidance has minimal effect on this positioning scheme. However, the authors do not consider the interference between DBSs. Moreover, this positioning scheme only consider satisfying users with different QoS requirements, and it does not consider the fair QoS provisioning between users.

In addition, a study by Lim et al. (2023) aims to optimize DBS placement by using another meta-heuritic algorithm. The authors first develop a multi-DBS positioning and user association problem that will maximize the fairness utility method that boosts equal QoS provisioning. The authors have considered the collision and limited user capacity that may happen among DBSs. Next, they use an enhanced artificial hummingbird algorithm (AHA) with the greedy user association strategy to perform the DBS positioning and execute user association. Lastly, the authors show that the proposed positioning strategy surpasses the random positioning schemes based on Jain's fairness index, and loss rate. However, the limitation in this study is the lack of consideration for the interference among DBSs that may affect the network performance.

Moreover, in another paper by Jiang et al. (2023), the authors intend to achieve fair load balancing among DBSs with the delay QoS specifications and collision avoidance restrictions. The authors prevent the collision between DBSs by using the collision avoidance algorithm to ensure that the 3D distance between any two DBSs exceed the minimal distance between DBSs. Besides that, the authors make sure that the DBSs' position are within the service boundaries and acceptable altitude by implementing the boundary limitation algorithm. Next, the authors have used a modified grey wolf optimization (GWO) algorithm to optimize DBS placement. The authors have also applied a greedy user association strategy to associate the users with the DBS that delivers the largest effective capacity. Lastly, the authors analyze the performance of this placement scheme in terms of percentage of blocking, total effective capacity, and Jair's fairness index. However, the authors in this paper do not consider the interference among DBSs that may degrade the network performance.

2.4.3 Overview of Related Works

The above sections have provided a high level overview of the existing DBS positioning schemes based on different non-meta-heurtistic and meta-heuristic algorithms. Each algorithm have been designed to achieve a different objective, either to maximize the network coverage for users in a designated area, provide fair QoS provisioning for each users, or to maximize the system throughput. The following table summarizes the main characteristics of each papers discussed above.

Table 2.2: Overview of Related Works

Type	Paper	Algorithms	QoS	Coverage	User rate	Throughput	Limitations
Non-meta-heuristic	(Akram et al., 2020)	Branch and bound		✓			Only optimize the 2D placement for DBS, fixed DBS altitude
	(Hammouti et al., 2021)	Greedy			✓		Fixed DBS altitude and lack of testing with benchmark functions
	(Alzenad et al., 2018)	MWA		✓			No consideration for collision and interference between DBS
	(Valiulahi and Masouros, 2021)	Hybrid				✓	No consideration for collision and interference between DBS
Meta-heuristic	(Kalantari et al., 2016)	PSO	✓				Fixed DBS altitude with no consideration for ground BS
	(Merwaday et al., 2016)	GA				✓	No consideration for collision and interference between DBS
	(Ouamri et al., 2022)	GWO		✓			No consideration

							for collision between DBS
	(Lim et al., 2021)	SA		✓			No consideration for interference between DBS, does not address fair QoS provisioning
	(Lim et al., 2023)	AHA	✓				No consideration for interference between DBS
	(Jiang et al., 2023)	GWO					No consideration for interference between DBS

According to the Table 2.2, there are four existing DBS positioning schemes that are based on the non-meta-heuristic algorithms. After reviewing these four existing positioning schemes, it is found out that there are many constraints that need to be set before the aforementioned algorithms can be applied. For example, the DBS positioning problem which has been transformed into a submodular maximization problem might be better solved by implementing the heuristic greedy algorithm in the next step (Hammouti et al., 2021). Meanwhile, the DBS positioning problem has been optimized in 2D placement by Akram et al. (2020). However, both papers have assumed that the altitudes of DBSs is fixed. Furthermore, both papers do not consider the collisions that might occur between DBSs. Therefore, in short, non meta-heuristics algorithms may not be suitable for the DBS positioning problem because it needs to satisfy many conditions, and the system may need to be modified to become more suitable to implement the solution that the authors intend to apply.

Meanwhile, from the existing positioning schemes based on the meta-heuristic algorithms, it is found out that most of the papers aim to optimize the DBS placement by maximizing the network coverage for each user. In particular, in the studies by Lim et al. (2021), and Merwaday et al. (2016), the authors attempt to satisfy the users with different QoS specifications, while only in the studies by Kalantari et al. (2016), and Lim et al. (2023), the authors aim to provide fair QoS provisioning among users. Besides that, in the paper of Jiang et al. (2023), the authors aim to provide fair load balancing among users. The common limitations among these research papers are that all have not considered the inteferences between DBSs except in the study by Ouamri et al. (2022). This constraint needs to be considered in the DBS placement scheme as the inteference between DBSs may degrade the performance result. Besides that, most papers assume a fixed DBS's altitude value before optimizing the placement of DBSs. Moreover, all papers stated above used meta-heuristic algorithms, which are developed based on a unique single natural phenomenon, to optimize the DBS placement. However, all papers have their own limitations. This is due to the limitations of the single natural phenomenon in those nature-inspired algorithms. Thus, this suggest that a hybrid algorithm may provide a better performance result because by hybridizing multiple different meta-heuristic algorithms, a wider range of the problem can be solved. It is also worth noting that the COA and MGO algorithm that are considered in this project has still yet to be researched for this specific application.

2.5 Summary

The DBS positioning problem is one of the highly researched topic and it has good potential to revolutionise the 6G networks. According to the information gathered and analysis done from the literature review, the development of a new hybrid MGO and COA algorithm-based multi-DBS positioning scheme are considered in this study and are expected to overcome the common limitations highlighted in the previous section.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The research methodologoy and work plan for this project will be presented and discussed in this chapter. Since it is impossible to perform physical testing with DBSs, the problem will formulated mathematically as part of testing in a simulation environment. Figure 3.1 illustrates the block diagram for our research methodology and work plan.

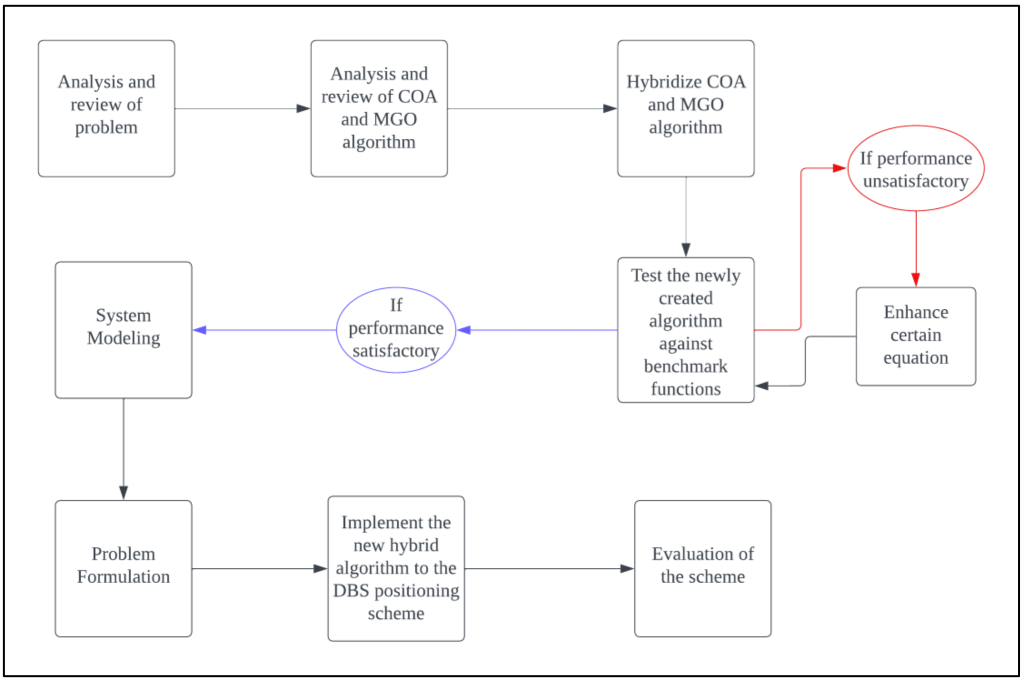

Figure 3.1: Block diagram for research methodology and work plan

3.2 Analysis and Review of the Problem

The objective of this project is to develop a multi-DBS positioning scheme based on the HCCMGA algorithm. This algorithm is expected to support the use of multiple DBSs simultaneously, as well as to ensure that it can satisfy fair DBS positioning and average maximum coverage radius received by each DBS. Since multiple DBS will occupy the same air space, therefore the constraint of possible collision and interference among DBSs should be considered and addressed effectively. Besides that, each ground user will be associated with only one DBS. This is to ensure fair QoS provisioning among the ground users, as well as to prevent QoS oversatisfaction, and thus DBS's transmit power will be utilized more effectively.

3.3 Overview of Crayfish Optimization Algorithm (COA) and Mountain Gazelle (MGO) algorithm

This section describe two algorithm that will be hybridized, which are the Crayfish Optimization Algorithm (COA) and Mountain Gazelle Optimization (MGO) algorithm.

3.3.1 Standard Crayfish Optimization Algorithm

COA is a metaheuristic developed by Jia et al. (2023). It is a swarm optimization algorithm inspired by the crayfish summer resort, competition, and food-hunting habits. The exploration stage for COA is the summer resort phase, while the foraging and competition stages belong to the exploitation stage of COA.

3.3.1.1 Initialize Population

The process of COA starts with the initialization of the population. Equation (3.1) shows the initialization of the COA algorithm:

$$X = [X_1, X_2, \dots, X_N] = \begin{bmatrix} X_{1,1} & \cdots & X_{1,j} & \cdots & X_{1,dim} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ X_{i,1} & \cdots & X_{i,j} & \cdots & X_{i,dim} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ X_{N,1} & \cdots & X_{N,j} & \cdots & X_{N,dim} \end{bmatrix} \quad (3.1)$$

where X is the initial position of the population, N is the population size, dim is the number of dimensions/variables in each position, and $X_{i,j}$ is the position of the i-th crayfish in the j-th dimension. Each crayfish is set as a size $1 \times dim$ matrix in the multidimensional optimization problem. Each row in the matrix is a candidate solution to the problem. Meanwhile, for a group of variables ($X_{i,1}$, $X_{i,2}$, … , $X_{i,dim}$), every variable X_i cannot be positioned out beyond lower and upper boundaries. For initialization, $X_{i,j}$ can be obtained as

$$X_{i,j} = l_j + \left(u_j - l_j\right) \times r \tag{3.2}$$

where u_j is the superior limit of the jth dimension, l_j is the bottom limit of the *j*-th dimension, *r* is the random value within 0 to 1.

3.3.1.2 Definition of temperature and intake of crayfish

After the candidate solution, *X* has been initialized, a temperature needs to be calculated because it can affect the behavior of the crayfish. Equation (3.3) is used to calculate the temperature:

$$t = r \times 15 + 20 \tag{3.3}$$

Note that the feeding area of a crayfish is between 15°C to 30°C. Besides that, the nourishing portion of crayfish is influenced by the temperature. The nourishing intake of crayfish can be calculated with

$$p = C_1 \times \left(\frac{1}{\sqrt{2\ \pi}\ \sigma}\ exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right)\right) \tag{3.4}$$

where *t* is the environment's temperature of the located crayfish, while μ is the most appropriate temperature for crayfish, σ and C_1 = variables that limit the crayfish's intake at different temperatures.

3.3.1.3 Summer Resort Stage (exploration)

If t is higher than 30°C, the crayfish will enter the summer resort phase, the COA's exploration stage. At this stage, the crayfish will find a tunnel for summer vacation. The tunnel X_s is identified as

$$X_s = \left(X_G + X_L\right) / 2 \tag{3.5}$$

where X_G is the optimal placement acquired so far at the present repetition and X_L is the optimal placement of the present populace.

Before entering the cave, the crayfish will fight for caves on a random occasion. If r is lower than 0.5, the crayfish will straightaway move into the cave for summer vacation since no other crayfish competes for the caves. The location of the crayfish is defined in the following equation when it enters the cave.

$$X_{i,j}^{T+1} = X_{i,j}^{T} + C_2 r \left(X_s - X_{i,j}^{T}\right) \tag{3.6}$$

where T is the present repetition index, C_2 is the decreasing curve, calculated as follows:

$$C_2 = 2 - (T/T_{max}) \tag{3.7}$$

where T_{max} is the maximum number of iterations. The aim of crayfish is to go towards the cave, which is the best solution, and it will enhance the exploitation capability of COA, enabling COA to converge quicker.

3.3.1.4 Competition Stage (exploitation)

When t is higher than 30, and r is higher or equal to 0.5, another crayfish is also intended to enter the cave. They then compete for the cave through the following equation:

$$X_{i,j}^{T+1} = X_{i,j}^{T} - X_{z,j}^{T} + X_s \tag{3.8}$$

where z is the random crayfish's individual, computed by

$$z = \mathrm{round}\big(r(N-1)\big) + 1 \tag{3.9}$$

In equation (3.9), the round function is used to round the floating-point value to the nearest integer. In this phase, the crayfish fight among each other to enter a cave and the crayfish X_i will adjust their location depending on the location X_z of another crayfish. Thus, by modifying their positions, the search scope of COA is enlarged, and the exploration capability of the COA algorithm is expanded.

3.3.1.5 Foraging Stage (exploitation)

When t is less than or equal to 30°C, it is appropriate to feed the crayfish. So, the crayfish will approach the food. The crayfish will compute the food size after discovering it. If the food is too huge, the crayfish will shred the food using its claws and eat it using its second and third walking feet. The food position X_f is computed with

$$X_f = X_G \tag{3.10}$$

Meanwhile, the food size Q is explained in the following equation:

$$Q = C_3 r\left(f_i / f_f\right) \tag{3.11}$$

where C_3 is the food factor for the biggest food with a fixed value of 3, f_i is the fitness value of the ith crayfish, f_f is the fitness value of the food position.

The judgment of the food size depends on the biggest food size. If Q is larger than $(C_3 + 1)/2$, it implies that the food size is too big. Thus, the crayfish will shred the food by using its first claw foot. The position of the food is defined as

$$X_f = \exp\left(-\frac{1}{Q}\right) \times X_f. \tag{3.12}$$

After the food is torn and it has become tinier, the second and third feet of the crayfish will grab the food and place it into the mouth alternately. The equation below, which includes the sine and cosine methods, will be used to replicate the alternating process:

$$X_{i,j}^{T+1} = X_{i,j}^{T} + X_f p(cos(2\pi r) - \sin(2\pi r)). \tag{3.13}$$

When $Q \leq (C_3 + 1)/2$, it implies that the food size is small and the crayfish will just approach the food and eat it straight away with the following mathematical function:

$$X_{i,j}^{T+1} = \left(X_{i,j}^{T} - X_f\right)p + prX_{i,j}^{T}, \tag{3.14}$$

In short, the crayfish will use various feeding approaches depending on the food size Q, and the position of the food X_{food} is the optimal solution. If the food size Q is suitable for the crayfish to eat, it will straightaway go towards the food. However, if the food size Q is very big, then there is still a notable gap between the best fitness value and the crayfish. Thus, X_f should be decreased and it should become nearer to the food. Besides that, the uncertainty of the food intake will be controlled to enhance the COA algorithm. Therefore, throughout the foraging stage, COA will aim to move towards the optimum solution and enhance the exploitation capability of the algorithm.

3.3.2 Standard Mountain Gazelle Optimization

MGO is a novel metaheuristics algorithm designed by Abdollahzadeh et al. (2022). This algorithm is motivated by gazelles' hierarchical and social life. This algorithm uses the four key causes in the social living of mountain gazelles to perform optimization operations. These four factors include migration to search for food, bachelor male herds, maternity herds, and solitary and territorial males.

In this algorithm, every gazelle (X_i) is a member of either bachelor male herds, maternity herds, or territorial solitary males. A brand-new gazelle will be originated from either one of these three herds. The mature male gazelle in the herd area is the best solution for the MGO algorithm. Besides that, around one-third of the searched populace in the whole populace is expected to have the lowest price since in the male bachelor group, the gazelles are too youthful and not yet mature enough to oversee a territory.

Additionally, maternity gazelle herds are the alternative solutions available to the populace. The forceful gazelles that have superior solutions will be retained during each iteration. Meanwhile, other results that are added to the overall populace that have lower price are viewed as elderly and not healthy gazelles. These gazelles will be eliminated from the overall populace.

3.3.2.1 Territorial Solitary Males

When the male gazelle become mature and powerful enough, they will make a solid territory. The distance between each territory is great enough. The young male gazelles will aim to gain an uncontrolled land, while the adult male gazelles will try to protect their environment. The following equation will be used to create the adult male region:

$$A = m_g - \left|\left(i_1 B - i_2 X(t)\right) \times F\right| \times Cof_r \tag{3.15}$$

where m_g is the location vector of the optimal universal result for the adult males, i_1 and i_2 are the random integers between 1 to 2, B is the coefficient vector of the young males, calculated using equation (3.16), F is computed using equation (3.17), Cof_r is the randomly chosen coefficient vector revised in each iteration computed using (3.18):

$$B = X_a \times \lfloor r_1 \rfloor + M_p \times \lceil r_2 \rceil, a = \left\{\left\lceil \frac{N}{3} \right\rceil \cdots N\right\} \tag{3.16}$$

$$F = N_1(D) \times \exp\left(2 - T \times \left(\frac{2}{T_{max}}\right)\right) \tag{3.17}$$

$$Cof_i = \begin{cases} (a+1) + r_3, \\ a \times N_2(D), \\ r_4(D), \\ N_3(D) \times N_4(D)^2 \times \cos\left((2r_4) \times N_3(D)\right), \end{cases} \tag{3.18}$$

where X_a is the random result for young gazelle in the interval of a, M_p is the mean number of search agents $\left\lceil \frac{N}{3} \right\rceil$ that were selected randomly, N is the overall number of gazelles, r_1 and r_2 are the random numbers within 0 to 1, N_1 is the random value from the normal distribution, exp() is the exponential function, T_{max} is the maximum number of iterations, T is the present repetition index, a is computed using equation (3.19), r_3, r_4 and r are the random values between 0 and 1, N_2, N_3 and N_4 are the random values in the normal space and the problem dimensions.

$$a = -1 + T \times \left(\frac{-1}{T_{max}}\right), \tag{3.19}$$

where T_{max} is the maximum number of repetitions, and T is the repetition index.

3.3.2.2 Maternity Herds

Maternity herds play a significant part in the wheel of life of mountain gazelles since they can produce solid males. This action can be formulated using

$$M = (B + Cof_{2,r}) + (i_3 \times m_g - ri_4 \times X_r) \times Cof_{3,r}, \quad (3.20)$$

where B is the impact factor vector for young males, computed using equation (3.16), $Cof_{2,r}$ is the randomly chosen coefficient vectors that are computed using equation (3.18), i_3 and i_4 are the random numbers between 1 and 2, m_g is the best adult male universal result in the present iteration, and X_r is the gazelle's vector position selected randomly based on the population.

3.3.2.3 Bachelor Male Herds

When the male gazelles become adults, they will be more likely to make a new colony and grab control over female gazelles. Meanwhile, the youthful males will fight with others over the control and domain of the female gazelles. The following equation is used to express this gazelles' actions mathematically.

$$H = (X(T) - D) + (i_5 m_g - i_6 B) \times Cof_r \quad (3.21)$$

where $X(T)$ is the gazelle's location vector in the present repetition, i_5 and i_6 are the random integers between 1 and 2, m_g is the male gazelle's location vector, which indicates the optimal result. D is computed using

$$D = (|X(T)| + |m_g|) \times (2r_6 - 1) \quad (3.22)$$

where $X(T)$ is the gazelle's location vector in the present repetition, m_g is the male gazelle's location vector, which indicates the optimal result, r_6 is the random integer within 0 to 1.

3.3.2.4 Migration to Search for Food

Gazelles always find for food supplies and will go a long way to get the food and migrate. The following equation formulates the gazelles' behavior mathematically:

$$S = (u - l) \times r_7 + l, \quad (3.23)$$

where u is the upper limits, l is the lower limits, r_7 is the random value between 0 and 1.

These four techniques are implemented in all gazelles to generate a new era of gazelles. A new generation will be included in the whole populace, and every era is equivalent to a new reproduction. Furthermore, each of the gazelles will be sorted according to the ascending order during each era. The best gazelles with positive solutions and excellent quality will be kept in the overall population. On the other hand, weak or old gazelles will be eliminated from the overall populace. Matured male is the best gazelle that possesses the area.

3.4 Hybrid Cooperative COA-MGO Algorithm

Both COA and MGO have been shown to outperform popular metaheuristics such as GA, PSO, GWO and WOA for continuous global optimization. While COA and MGO demonstrate great improvements over conventional metaheuristics through evaluations of a set of test functions, there are several functions in which COA does not yield better performance. MGO exhibits a similar shortcoming in several test functions but some of these functions are able to be optimized via COA. To bridge this gap, the aim of the current study is to leverage and combine the features of these two algorithms to improve search diversity.

3.4.1 Competition and Cooperation

It is worth to remark that both COA and MGO exhibit some form of competitive behaviors. For instance, COA enters a competition stage where crayfish fight among each other to enter a cave when the temperature is too high, whereas MGO allows adult mountain gazelles to compete with each other for territories and possession of female gazelles. While the competitive behaviors usually lead to improved search diversity and optimization performance, these can be further enhanced through cooperation.

In fact, many traditional metaheuristics rely on cooperation to obtain optimal solutions and they have shown substantial search performance improvements. For example, GA exchange solution information between chromosomes in the crossover stage, PSO signals all particles regarding the locations of good solutions, and GWO imitates the cooperative hunting movements of grey wolves. These features essentially enhance exploration and exploitation abilities of the algorithms. In view of this, leveraging both competition and cooperation potentially enhance the search diversity.

The underlying optimization principles of COA and MGO is vastly different, especially their respective competition mechanisms. A mix of such mechanisms is definitely challenging and may alter their inherent advantages in search exploration and exploitation. Instead of mixing the mechanisms to form a single algorithm, these two algorithms can be mix together to work cooperatively by exchanging solution information among each other. This is actually similar to the crossover mechanism of GA, where chromosomes exchange a part of its solution with each other to form new candidate solutions. This mechanism has also been adapted in PSO to develop a cooperative multi-swarm optimization technique (vandenBergh and Engelbrecht, 2004), and it has been shown to achieve substantial enhancements in exploration and exploitation. Nevertheless, the cooperative multi-swarm PSO in the study by vandenBergh and Engelbrecht (2004) lacks the competitive behaviors that could be synergized with cooperation to further enhance search diversity. To this end, a HCCMGA algorithm is developed which can leverage the optimization mechanisms of both COA and MGO with cooperative information exchanges.

3.4.2 Splitting Factor and Cooperative Fitness Evaluation

In the proposed HCCMGA, the *dim*-dimensional solution space is divided into two lower-dimensional solution subspaces, with one being optimized by COA and the other by MGO. Here, a splitting factor has been introduced which is $S= m:n$, where m and n are the numbers of dimensions of the solution subspace of COA and MGO respectively. To understand the splitting factor, we use the following example: Assuming a 10-dimensional solution space, we can split it into an 8-dimensional solution subspace and a 2-dimensional solution subspace. If the 8-dimensional solution subspace is to be optimized by COA whereas the other by MGO, this indicates $S = 8:2$ where $m = 8$ and $n = 2$, as shown in Figure 3.2.

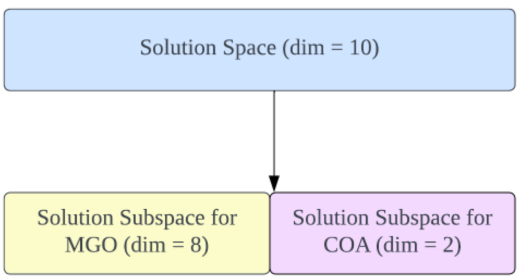

Figure 3.2: Example of splitting the solution vector into two different solution subvectors

Based on this subspace setting, the solution vector of each crayfish in COA consists of 8 variables whereas that of each mountain gazelle in MGO consists of 2 variables. The use of the splitting factor is to provide flexibility to determine the degree of dominance between COA and MGO in the optimization process. With the example of S = 8:2, COA makes a larger contribution to the optimization process compared to MGO. In other words, the optimization process is dominantly affected by the optimization mechanism of COA. To exert equal dominance between COA and MGO, $m = n$ can be set for the splitting factor.

After splitting the solution space between COA and MGO, the candidate solutions obtained by COA and MGO can be assessed through a cooperative fitness evaluation approach. P_M and P_C can be firstly denoted as the MGO and COA populations respectively. Next, $P_M.\hat{y}$ and $P_C.\hat{y}$ are referred as the best candidate solutions found by MGO and COA respectively. Next, the following functions are defined:

$$Q_{COA}(P_C.\mathbf{X}_i) = (P_M.\hat{y}, P_C.\mathbf{X}_i) \tag{3.24}$$

$$Q_{MGO}(P_M.\mathbf{X}_i) = (P_M.\mathbf{X}_i, P_C.\hat{y}) \tag{3.25}$$

where $P_C.\mathbf{X}_i$ is the candidate solution of the i-th crayfish, and $P_M.\mathbf{X}_i$ is the candidate solution of the i-th mountain gazelle.

In equation (3.24), $Q_{COA}(P_C.\mathbf{X}_i)$ returns the dim-dimensional solution vector where the candidate solution of the i-th crayfish is merged with the best solution found by MGO. For example, when dim = 10 and S = 8:2, $Q_{COA}(P_C.\mathbf{X}_i)$ returns a 10-dimensional solution vector consisting of the 8-dimensional $P_C.\mathbf{X}_i$ and 2-dimensional $P_M.\hat{y}$. Similarly, $Q_{MGO}(P_M.\mathbf{X}_i)$ in equation (3.25) returns the dim-dimensional solution vector where the candidate solution of the i-th mountain gazelle is combined with the best solution found by COA.

Using equations (3.24) and (3.25), the fitness of each search agent in COA and MGO populations can be evaluated. In fact, this exhibits a form of cooperation between COA and MGO, whereby COA determines the best solution in its solution subspace, given the best solution information from MGO; whereas MGO identifies its best solution in its subspace, given the best solution information obtained by COA. Such a cooperative information exchanges can greatly improve the search diversity of both algorithms, while retaining their respective optimization mechanisms.

3.4.3 Hybrid Cooperative Framework

Algorithm 1: HCCMGA

Randomly initialize n-dimensional $P_M.\mathbf{X}_i$ for all $i = 1, \dots, N_1 = \left\lceil \frac{mN}{m+n} \right\rceil$ and m-dimensional $P_C.\mathbf{X}_j$ for all $j = 1, \dots, N_2 = N - N_1$.
Evaluate $f\left(\left(P_M.\mathbf{X}_i, P_C.\mathbf{X}_j\right)\right)$ for all $i = 1, \dots, N_1$ and $j = 1, \dots, N_2$.
Set $P_M.\hat{y} = P_M.\mathbf{X}_{i^*}$, $P_C.\hat{y} = P_C.\mathbf{X}_{j^*}$ and $f_o = f\left(\left(P_M.\mathbf{X}_{i^*}, P_C.\mathbf{X}_{j^*}\right)\right)$ where $(i^*, j^*) = \arg\max_{(i,j)} \left\{ f\left(\left(P_M.\mathbf{X}_i, P_C.\mathbf{X}_j\right)\right) \right\}$.
Evaluate $f(Q_{MGO}(P_M.\mathbf{X}_i))$ for all $i = 1, \dots, N_1$ and $f(Q_{COA}(\, P_C.\mathbf{X}_j))$ for all $j = 1, \dots, N_2$.
Initialize T_{max} and set $T = 0$, $P_M.\hat{y}_o = P_M.\hat{y}$ and $P_C.\hat{y}_o = P_C.\hat{y}$.
while $T < T_{max}$
  Calculate temperature t using (3.3).
  for each crayfish $i = 1{:}\, N_2$
    if $t > 30$
      Calculate cave X_s using (3.5) and generate a random number $r \sim U(0, 1)$.
      if $r < 0.5$
        Perform the summer resort phase with (3.6) for the i-th crayfish.
      Else
        Calculate (3.8) for the i-th crayfish to compete for caves.
      End
    Else
      Obtain food intake p with (3.4) and food size Q with (3.11).
      if $Q > 2$
        Calculate (3.12) for the i-th crayfish to shreds food.
        Calculate (3.13) to perform foraging for the i-th crayfish.
      Else
        Calculate (3.14) to perform foraging for the i-th crayfish.
      End
    End
  End
  Set $P_C.\hat{y} = P_C.\mathbf{X}_{i^*}$ where $i^* = \arg\max_i \{ f(Q_{COA}(P_C.\mathbf{X}_i)) \}$.
  Evaluate $f(Q_{MGO}(P_M.\mathbf{X}_i))$ for all $i = 1, \dots, N_1$.
  for each gazelle $i = 1{:}N_1$
    Generate new gazelles A, M, H and S using (3.15), (3.20), (3.21) and (3.23) respectively in the same order.
    Evaluate $f(Q_{MGO}(A))$, $f(Q_{MGO}(M))$, $f(Q_{MGO}(H))$ and $f(Q_{MGO}(S))$, and include the new gazelles A, M, H and S to population P_M.
  end
  Sort population P_M from the gazelle with the highest fitness to the one with the lowest fitness.
  Remove the gazelles with lowest fitness by remaining the first N_1 gazelles with the highest fitness in the population.
  Set $P_M.\hat{y} = P_M.\mathbf{X}_{i^*}$ where $i^* = \arg\max_i \{ f(Q_{MGO}(P_M.\mathbf{X}_i)) \}$.
  Set $n_1 = \left\lceil \frac{mN_1}{m+n} \right\rceil$ and $n_2 = \left\lceil \frac{nN_2}{m+n} \right\rceil$.
  Sort the populations P_M and P_C from the lowest fitness to the highest fitness.
  if $m > n$
    Replace the first n variables in $P_M.\mathbf{X}_i$ with $P_C.\hat{y}$ for all $i = 1, \dots, n_1$ and the first n variables in $P_C.\mathbf{X}_j$ with $P_M.\hat{y}$ for $j = 1, \dots, n_2$.
  Else

40. Replace the first m variables in $P_M.\mathbf{X}_i$ with $P_C.\hat{y}$ for all $i = 1, \dots, n_1$ and the first m variables in $P_C.\mathbf{X}_j$ with $P_M.\hat{y}$ for $j = 1, \dots, n_2$.
41. **End**
42. **for** $i = 1{:}\,N_1$
43. **for** $j = 1{:}\,N_2$
44. **if** $f_o < f((P_M.\mathbf{X}_i, P_C.\mathbf{X}_j))$
45. Set $P_M.\hat{y}_o = P_M.\mathbf{X}_i$ and $P_C.\hat{y}_o = P_C.\mathbf{X}_j$.
46. **End**
47. **End**
48. **End**
49. Set $P_M.\hat{y} = P_M.\hat{y}_o$ and $P_C.\hat{y} = P_C.\hat{y}_o$.
50. $T \leftarrow T + 1$.
51. **End**
52. Return $(P_M.\hat{y}, P_C.\hat{y})$.

Figure 3.3: Pseudocode of HCCMGA algorithm

The current study designs the proposed hybrid cooperative COA-MGO algorithm as shown in Figure 3.3. Firstly, a population of N search agents is created, with $N_1 = \left\lceil \frac{mN}{m+n} \right\rceil$ agents serving as the MGO population P_M whereas $N - N_1$ agents serving as the COA population P_C. It is noteworthy that the size of the populations is determined by the splitting factor. Then, the algorithm identifies $P_M.\hat{y}$ and $P_C.\hat{y}$ through iterative fitness evaluations of $f((P_M.\mathbf{X}_i, P_C.\mathbf{X}_j))$ where $f(.)$ is the fitness function and $(P_M.\mathbf{X}_i, P_C.\mathbf{X}_j)$ is the complete solution vector composed of MGO solution subvector $P_M.\mathbf{X}_i$ and COA solution subvector $P_C.\mathbf{X}_j$ (see step 2).

The iterative optimization process of the proposed HCCMGA begins with the COA optimization mechanism on the assigned n-dimensional solution subspace (see steps 7 – 26), followed by the MGO optimization mechanism on the assigned m-dimensional solution space (see steps 27 – 34). Next, we introduce solution exchange between P_M and P_C, where the best solution subvector from the MGO population replaces some of the candidate solution subvectors with the lowest fitness in the COA population, and vice versa (see steps 35 – 41). This solution exchange provides a low degree of search diversity to both the COA and MGO optimization mechanisms and information regarding the best solutions obtained from other dimensions. After that, the algorithm identifies $P_M.\hat{y}$ and $P_C.\hat{y}$ through iterative fitness evaluations of $f((P_M.\mathbf{X}_i, P_C.\mathbf{X}_j))$ (see steps 42 – 49). The aforementioned iterative optimization process of the proposed HCCMGA is repeated until $T = T_{max}$, where T is the iteration index and T_{max} is the allowed maximum number of iterations. At the end of the optimization process, the output solution of $(P_M.\hat{y}, P_C.\hat{y})$ is returned by the proposed algorithm.

In each iteration, the proposed HCCMGA requires computation of $N_1N_2 = \left\lceil\frac{mN}{m+n}\right\rceil\left(N - \left\lceil\frac{mN}{m+n}\right\rceil\right)$ fitness evaluations. Thus, the computational complexity of one iteration in the proposed algorithm is of $O(N^2)$. As such, the asymptotic computational complexity of the proposed HCCMGA can be obtained as $O(T_{max}\, N^2)$. Compared to COA and MGO, the computational complexity of the proposed HCCMGA is higher because both COA and MGO incurs an asymptotic computational complexity of $O(T_{max}N)$ for each iteration. Nevertheless, in the next chapter, the proposed HCCMGA can achieve convergence quicker than the state-of-the-arts in many of the standard benchmark test function, especially those with multimodal natures. In such a case, the computational complexity required to achieve convergence to near-optimal solutions could be lower than or equivalent to that of MGO and COA.

3.5 Application of HCCMGA on drone positioning

This section investigates the application of the proposed HCCMGA for DBS placement in aerial communication networks.

3.5.1 System Model

A multi-DBS network shown in Figure 1.1, which is based on an air-to-ground communication model similar to the study by Chen et al. (2020), is considered. The set of DBSs is denoted by D and the set of user is denoted by U. Therefore, the total numbers of DBSs and the users can be indicated as $|D|$ and $|U|$ respectively. Meanwhile, the 3D Cartesian points of DBS $d \in D$ and user $u \in U$ are indicated as (x_d, y_d, h_d) and $(x_u, y_u, 0)$ individually in the DBS positioning. Additionally, we designate $p_d = (x_d, y_d)$ and $q_u = (x_u, y_u)$ as the horizontal 2D coordinates of the DBS d and user u respectively.

It is assumed that the channel between DBSs and users is LoS-dominated, and the free-space path loss model is considered for the multi-DBS network. Therefore, the channel gain between DBS d and user u can be modeled as follows:

$$g_{d,u} = \frac{\beta_0}{D_{d,u}} = \frac{\beta_0}{||p_d - q_u||^2 + {h_d}^2} \tag{3.26}$$

where β_0 is the channel gain of the reference distance of 1 m, and $D_{d,u}$ denotes square of the distance from DBS d to user u.

Next, the capacity achieved by user u from DBS d can be obtained as

$$R_{d,u} = B \log_2\left(1 + \frac{P g_{d,u}}{\sigma^2}\right) \tag{3.27}$$

where B is the allocated bandwidth, P denotes the transmit power of each DBS, σ^2 is the additive white Gaussian noise (AWGN). A minimum required signal-to-noise ratio (SNR) of δ_{req} needs to be guaranteed for each user to satisfy the sensitivity of information receiver. With this, the DBS maximum coverage radius projected on the ground can be estimated as follows:

$$d_0 = \sqrt{\frac{\alpha}{\delta_{req}} - h_d} \tag{3.28}$$

where α is the quality of service (QoS) requirement and it can be computed as follows:

$$\alpha = \frac{P\beta_0}{\sigma^2} \tag{3.29}$$

3.5.2 Problem Formulation

The main goal of the DBS placement is to maximize the average capacity of the network, which can be formulated in the following optimization problem:

$$P1: \max_{\{p_d\},\{h_d\},\{a_{d,u}\}} \frac{\sum_{d \in D} \sum_{u \in U} a_{d,u} R_{d,u}}{|D|} \tag{3.30}$$

subject to

$$x_{min} \leq x_d \leq x_{max} \; \forall d \in D \tag{3.31}$$

$$y_{min} \leq y_d \leq y_{max} \; \forall d \in D \tag{3.32}$$

$$h_{min} \leq h_d \leq h_{max} \; \forall d \in D \tag{3.33}$$

$$a_{d,u} ||p_d - q_u||^2 \leq {d_0}^2 \; \forall d \in D, \forall u \in U \tag{3.34}$$

$$\sum_{u \in U} a_{d,u} \leq N_{max} \; \forall d \in D \tag{3.35}$$

$$\sum_{d \in D} a_{d,u} = 1 \tag{3.36}$$

where

$$a_{d,u} = \begin{cases} 1 & \text{if DBS } d \text{ associates with user } u \\ 0 & \text{otherwise.} \end{cases}$$

Constraints (3.31) – (3.33) defines the ground area at which users are distributed and the space where each DBS can be placed. Here, $x_{min,}$ y_{min} and z_{min} are the minimum boundaries whereas $x_{max,}$ y_{max} and z_{max} are the maximum boundaries of the 3D cartesian space. Constraint (3.34) ensures that the distance between each DBS and its associated ground user is within the maximum coverage radius of the DBS, denoted by d_0 (Chen et al., 2020). Constraint (3.35) enforces that that the number of users served by each DBS does not exceed its maximum user capacity, denoted by N_{max}. Lastly, each ground user can only be served by or associated with one DBS, as enforced by constraint (3.36).

As the problem $P1$ is a mixed-integer programming problem which is usually NP-hard, a greedy algorithm is designed and shown in Algorithm 2 to obtain the solutions of $a_{d,u}$, and to serve as the fitness function for the implementation of the proposed HCCMGA.

Algorithm 2: Fitness function $f_{\mathrm{DBS}}(\{p_d\},\{h_d\})$	
1.	Input: $\{p_d\}$,$\{h_d\}$.
2.	Initialize $a_{d,u} = 0$ for all $d \in D$ and $u \in U$.
3.	**for** user $u \in U$
4.	Set $d^* = \arg\max_d\{R_{d,u}\}$.
5.	**if** $\|p_{d^*} - q_u\|^2 \leq d_0{}^2$ and $\sum_{i \in U} a_{d^*,i} \leq N_{max}$
6.	Set $a_{d,u} = 1$.
7.	**End**
8.	**End**
9.	Calculate $f_{\mathrm{DBS}}(\{p_d\},\{h_d\})$ using (3.30).
10.	Return $f_{\mathrm{DBS}}(\{p_d\},\{h_d\})$.

Figure 3.4: Pseudocode for calculation of fitness function

With Algorithm 2, problem $P2$ can be simplified as the following unconstrained optimization problem:

$$P2: \max_{\{p_d\},\{h_d\}} f_{\mathrm{DBS}}(\{p_d\},\{h_d\}), \tag{3.37}$$

subject to constraint (3.31) – (3.33). Here, problem $P2$ is solved using the proposed HCCMGA.

3.6 Gantt Chart

The project's duration is expected to span over two trimesters. The general work plan for the first and second trimesters are illustrated in Figure 3.5 and Figure 3.6 seperately, in the form of Work Breakdown Structures (WBS) and Gantt Chart. Note that the duration in the figure is calculated in days. Besides that, the "predecessors" in the work breakdown structure indicates that the tasks in the "predecessors" needs to be completed before the current task.

WBS	Description	Start	End	Duration	Predecessors	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
1	**FYP title registration**	29/1/2024	30/1/2024	2														
2	**Develop work plan for FYP1 and FYP2**	31/1/2024	3/2/2024	4	1													
3	**Understanding the problem**	4/2/2024	2/3/2024	28														
3.1	Conduct literature review	4/2/2024	24/2/2024	21														
3.1.1	Literature review about 6G network	4/2/2024	10/2/2024	7	2													
3.1.2	Literature review about DBS	11/2/2024	17/2/2024	7	3.1.1													
3.1.3	Literature review about exisiting positioning scheme	18/2/2024	24/2/2024	7	3.1.2													
3.2	Testing code for different algorithm	11/2/2024	17/2/2024	7	2													
3.3	Study COA and MGO	18/2/2024	2/3/2024	14	3.2													
4	**Progress report**	3/3/2024	16/3/2024	14														
4.1	Identify problem statements and objectives	3/3/2024	7/3/2024	5	3.1													
4.2	Identify project scope and limitations	8/3/2024	12/3/2024	5	4.1													
4.3	Write and submit progress report	13/3/2024	16/3/2024	4	4.2													
5	**Project proposal**	17/3/2024	19/4/2024	35														
5.1	Compile information about literature review	17/3/2024	30/3/2024	14	4.3													
5.2	Identify project methodology	31/3/2024	6/4/2024	7	5.1													
5.3	Establish preliminary system modeling	7/4/2024	10/4/2024	4	5.2													
5.4	Preliminary problem formulation	11/4/2024	13/4/2024	3	5.3													
5.5	Obtain preliminary results from code of different algorithm	14/4/2024	17/4/2024	4	5.4													
5.6	Submission of project proposal	18/4/2024	19/4/2024	2	5.5													
6	**Presentation**	14/3/2024	24/3/2024	11														
6.1	Prepare presentation slides	14/3/2024	20/3/2024	7	5.5													
6.2	Conduct mock presentation	21/3/2024	23/3/2024	3	6.1													
6.3	Conduct presentation	24/3/2024	24/3/2024	1	6.2													

Figure 3.5: WBS and Gantt Chart for FYP1

In the first trimester, several main tasks are done, which include the title registration, work plan development, problem understanding, progress report writing, project proposal development, and presentation. The goal on this trimester is to develop a clear, detailed and relevant project proposal. Most of the time in this trimester has been spent on literature review, which include literature review on 6G network, DBS, and existing DBS positioning scheme. The project proposal includes studying relevant literature, identifying project methodology, and obtaining preliminary results after testing different meta-heuristic algorithm with the standard benchmark functions.

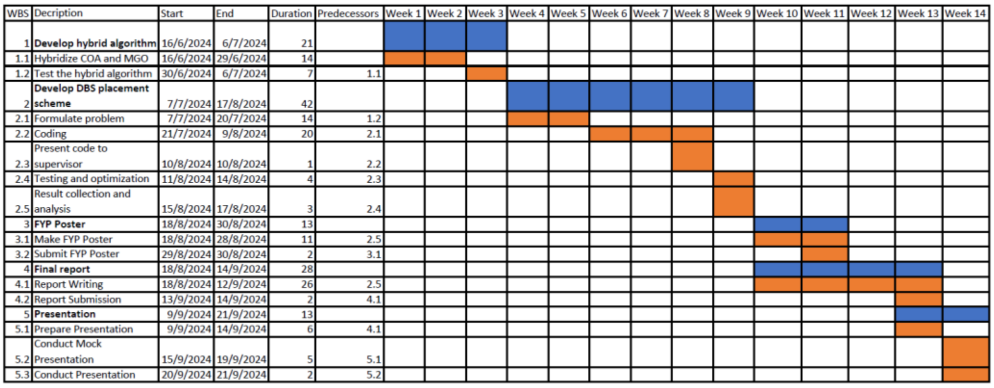

Figure 3.6: WBS and Gantt Chart for FYP2

During the second trimester, the goal moves fully toward the development of project components that were proposed in the first trimester. The tasks include developing hybrid algorithm based on COA and MGO, testing the hybrid algorithm based on standard benchmark functions, developing an effective DBS positioning scheme based on the hybrid algorithm, testing and optimizing the DBS positioning scheme, and analysis of the result from the DBS positioning scheme. Besides that, some time has been allocated to develop an FYP poster, finishing up the whole project report, and prepare for the presentation.

3.7 Summary

In short, the research methodology and work plan has been proposed in this chapter which includes the studing of the COA and MGO algorithm, development of HCCMGA algorithm, system modelling, and problem formulation. Although there has been no research conducted on using COA or MGO algorithm for solving the DBS positioning problem, the existing literature on these two algorithms suggest that these two algorithms has its strength in solving different optimization problem, which may be suitable to provide an effective DBS placement scheme.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this chapter, two different experiment are conducted. The first experiment is to test the HCCMGA on standard benchmark test functions. This experiment is conducted to access whether HCCMGA outperforms state-of-the-arts optimization algorithms in solution quality, search stability and convergence speed. The second experiment is to apply the HCCMGA on DBS positioning scheme and test it against other state-of-the-arts optimization algorithms to check whether HCCMGA provides the highest average network coverage radius among all algorithms. The simulation tool that has been used for these two experiments is MATLAB.

4.2 Experiment result of HCCMGA on standard benchmark

In this section, two different simulations are carried out. The first simulation is to assess the proposed HCCMGA with different splitting factors. Meanwhile, the second simulation is to compare the performance of the proposed HCCMGA against five other metaheuristics. For fair performance comparisons, $N = 40$, $T_{\max} = 500$, and $dim = 30$ are set for all the simulations as refer to (Jia et al., 2023). Table 4.1 lists the 13 benchmark functions (Jia et al., 2023), their search space ranges, their dimensions and their minimum values. From these 13 benchmark function, F1 - F7 are unimodal functions (with a single global optimum) while F8 - F13 are multimodal functions (with a single global optimum and many local optima). Unimodal functions assess the performance of a metaheuristic in optimizing functions with only a single optimum, whereas multimodal functions assess the performance of a metaheuristic in optimizing functions with many local optima that could cause premature convergence and trap the algorithm in local optima.

Table 4.1: Benchmark functions

Function	Dim	Range	F_{min}

$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	[-100, 100]	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	[-10, 10]	0
$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	30	[-100, 100]	0
$F_4(x) = max\{\lvert x_i \rvert, 1 \leq i \leq n\}$	30	[-100, 100]	0
$F_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-30, 30]	0
$F_6(x) = \sum_{i=1}^{n} (x_i + 0.5)^2$	30	[-100, 100]	0
$F_7(x) = \sum_{i=1}^{n} i \times x_i^4 + random[0, 1)$	30	[-1.28, 1.28]	0
$F_8(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	30	[-500, 500]	-418.9829 x *dim*
$F_9(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	[-5.12, 5.12]	0
$F_{10}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i)\right) + 20 + e\right)$	30	[-32, 32]	0
$F_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	[-600, 600]	0
$F_{12}(x) = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$, where $y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a^m), & x_i \end{cases}$	30	[-50, 50]	0
$F_{13}(x) = 0.1(\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]) + \sum_{i=1}^{n} u(x_i, 5,100,4)$	30	[-50, 50]	0

4.2.1 Performance of HCCMGA with Different Splitting Factors

Here, the HCCMGA has been evaluated based on five different splitting factors, namely S = 15:15, 10:20, 20:10, 5:25, and 25:5.

Table 4.2: Minimization Result of Benchmark Functions for HCCMGA with Different Splitting Factors

F	S = 15:15	S = 10:20	S = 20:10	S = 5:25	S = 25:5
F1	$\mathbf{0 \pm 0}$	$2E^{-92} \pm 7.8E^{-92}$	$\mathbf{0 \pm 0}$	$1.89E^{-75} \pm 1.04E^{-74}$	$\mathbf{0 \pm 0}$
F2	$\mathbf{0 \pm 0}$	$1.06E^{-52} \pm 3.85E^{-52}$	$1.64E^{-286} \pm \mathbf{0}$	$1.47E^{-44} \pm 3.06E^{-44}$	$5.27E^{-288} \pm \mathbf{0}$
F3	$\mathbf{1.19E^{-245} \pm 0}$	$2.18E^{-12} \pm 9.40E^{-12}$	$3.49E^{-30} \pm 1.91E^{-29}$	$4.77E^{-9} \pm 1.94E^{-8}$	$3.74E^{-208} \pm \mathbf{0}$
F4	$7.21E^{-1} \pm 1.49$	$3.7E^{-1} \pm 1.06$	$\mathbf{8.46E^{-12} \pm 4.61E^{-11}}$	$1.95E^{-1} \pm 6.13E^{-1}$	$1.78E^{-2} \pm 7.18E^{-2}$
F5	$8.61 \pm 1.34E^{1}$	$1.74E^{-28} \pm 6.5E^{-28}$	$3.71E^{1} \pm 2.54E^{1}$	$\mathbf{3.09E^{-29} \pm 8.89E^{-29}}$	$2.89E^{1} \pm 2.17E^{-2}$
F6	$2.47E^{-7} \pm 6.45E^{-7}$	$4.9E^{-6} \pm 1.03E^{-5}$	$1.01 \pm 3.48E^{-1}$	$\mathbf{3.74E^{-8} \pm 1.32E^{-7}}$	$3.17 \pm 4.31E^{-1}$
F7	$\mathbf{5.2E^{-1} \pm 3.21E^{-1}}$	$\mathbf{5.2E^{-1} \pm 3.21E^{-1}}$	$\mathbf{5.2E^{-1} \pm 3.21E^{-1}}$	$\mathbf{5.2E^{-1} \pm 3.21E^{-1}}$	$\mathbf{5.2E^{-1} \pm 3.21E^{-1}}$
F8	$\mathbf{-1.26E^{4}} \pm 3.36E^{-5}$	$\mathbf{-1.26E^{4}} \pm 2.09E^{-5}$	$-1.07E^{4} \pm 6.11E^{2}$	$\mathbf{-1.26E^{4} \pm 1.17E^{-6}}$	$-8.26E^{3} \pm 1.21E^{3}$
F9	$1.89E^{-15} \pm 1.04E^{-14}$	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$1.89E^{-15} \pm 1.04E^{-14}$	$\mathbf{0 \pm 0}$
F10	$2.19E^{-15} \pm 1.74E^{-15}$	$3.38E^{-15} \pm 1.66E^{-15}$	$\mathbf{1.36E^{-15} \pm 1.23E^{-15}}$	$3.73E^{-15} \pm 1.45E^{-15}$	$1.36E^{-15} \pm 1.54E^{-15}$
F11	$1.04E^{-16} \pm 2.08E^{-16}$	$3.07E^{-16} \pm 4.22E^{-16}$	$\mathbf{0 \pm 0}$	$5.18E^{-17} \pm 1.99E^{-16}$	$3.85E^{-1} \pm 1.47E^{-1}$
F12	$\mathbf{1.57E^{-32} \pm 5.57E^{-48}}$	$3.27E^{-32} \pm 3.38E^{-32}$	$4.19E^{-2} \pm 3.28E^{-2}$	$1.72E^{-28} \pm 4.83E^{-28}$	$2.3E^{-1} \pm 7.42E^{-2}$
F13	$\mathbf{1.35E^{-32} \pm 5.57E^{-48}}$	$\mathbf{1.35E^{-32} \pm 5.57E^{-48}}$	$7.97E^{-1} \pm 2.25E^{-1}$	$1.36E^{-32} \pm 4.96E^{-34}$	$1.94 \pm 7.09E^{-2}$

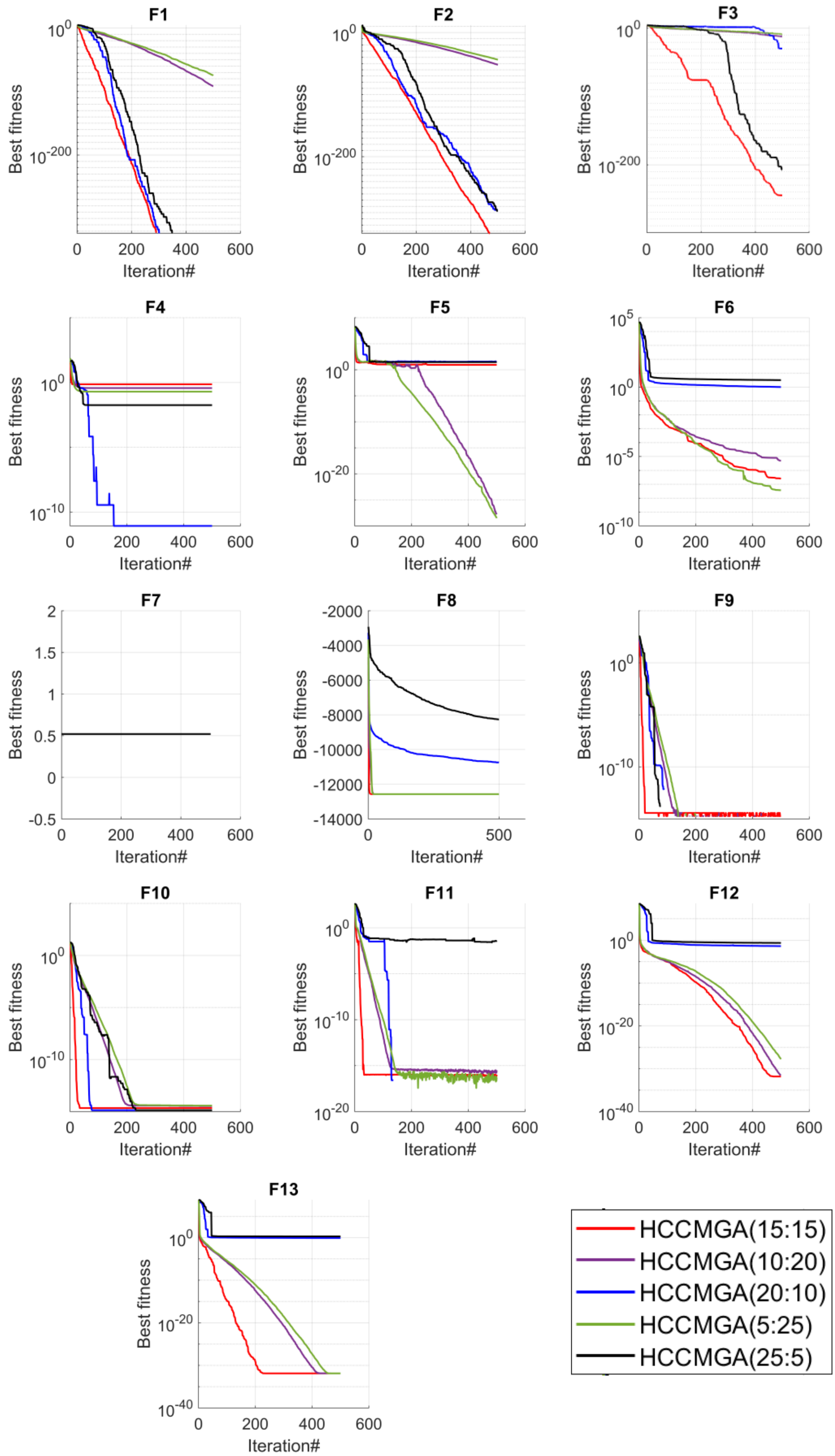

Figure 4.1: Convergence performance of HCCMGA with different splitting factor for F1 – F13

The statistical performance results for each of the functions are presented in Table 4.2 while the convergence curve for each of the functions is presented by Figure 4.1 The results in Table 4.2 are presented in the form of *mean*$\pm$*standard deviation* and they are based on the average results over 30 runs. From the table, the values in bold indicate the best results.

From Table 4.2, HCCMGA with S = 15:15 is generally the best and most balanced performer in terms of solution quality by achieving the best mean fitness value in seven test functions, with four of them being unimodal and the rest being multimodal. HCCMGA with S = 20:10 places second by excelling in six test functions. Meanwhile, HCCMGA with S = 10:20 and S = 5:25 achieves the best fitness in four test functions. HCCMGA with S = 25:5 only performs well in three out of the 13 test functions.

HCCMGA with S = 15:15 and S = 20:10 are the generally the most stable, as they achieve the lowest standard deviations in six out of the 13 test functions, while those with other splitting factors only excel in less than six functions. Interestingly, we can observe that the standard deviations achieved for F4, F5 and F8 are quite large, and only the HCCMGA with a certain splitting factor can excel in these test functions. This implies that with certain splitting factors, the HCCMGA can achieve better stability and solution quality.

Figure 4.1 shows the convergence performance of the proposed HCCMGA with different splitting factors. Overall, HCCMGA with S = 15:15 achieves the fastest convergence in majority of the 13 test functions, except F4 and F5. For F1 and F2, HCCMGA with S = 15:15, 20:10 and 25:5 converge much faster than others, with HCCMGA that sets S = 15:15 places first among them. Next, for F3, HCCMGA with S = 15:15 and 25:5 converge faster than others to the global optimum. F4 is a special case as the HCCMGA with S = 20:10 converges ahead of others. Meanwhile, for F5, HCCMGA with S = 10:20 and 5:25 yield better performance than others, with S = 5:25 places first. For F6, HCCMGA with S = 15:15 and 5:25 have similar convergence curves. Besides that, since HCCMGA with any splitting factor achieves equal mean and standard deviation value in F7, they also yield similar convergence performance. From the convergence curve of F8, we can see that the convergence performance from the HCCMGA with S = 15:15, 10:20, and 5:25 has exhibit similar convergence trends. For F9, F10 and F11, HCCMGA with S = 15:15 converges to the minimum value at around 40 iterations, which is much better than others. Lastly, for F12 and F13, HCCMGA with S = 15:15, 10:20, and 5:25 perform well, with S = 15:15 places first among them.

In short, the HCCMGA with a splitting factor of S = 15:15 is overall the best performer in terms of solution quality, stability and convergence speed. In the subsequent subsection, we consider S = 15:15 for comparisons with the other metaheuristics.

4.2.2 Performance of HCCMGA against Other Metaheuristics

In this section, five state-of-the-art metaheuristics are compared with the proposed HCCMGA using the same 13 test functions in Table 4.1. The five metaheuristics selected are COA, MGO, GWO, WOA and SCA. The source code of COA, MGO, GWO, WOA, and SCA is referred from the source code of the author of (Jia et al., 2023). S = 15:15 have been chosen for the HCCMGA.

Table 4.3: Minimization Result of Benchmark Functions for Different Algorithms

F	HCCMGA	COA	MGO	GWO	WOA	SCA
F1	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$2.43E^{-77} \pm 1.26E^{-77}$	$8.08E^{-31} \pm 9.98E^{-31}$	$2.77E^{-79} \pm 1.35E^{-78}$	$1.11E^{1} \pm 3.19E^{1}$
F2	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$1.88E^{-45} \pm 7.28E^{-45}$	$1.51E^{-18} \pm 1.09E^{-18}$	$5.23E^{-52} \pm 5.88E^{-52}$	$9.5E^{-3} \pm 1.15E^{-2}$
F3	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$2.87E^{-10} \pm 1.16E^{-9}$	$1.01E^{-6} \pm 2.70E^{-6}$	$3.62E^{4} \pm 1.21E^{4}$	$6.5E^{3} \pm 4.61E^{3}$
F4	$8.82E^{-1} \pm 1.74$	$\mathbf{0 \pm 0}$	$3.36E^{-27} \pm 1.21E^{-26}$	$1.1E^{-7} \pm 1.08E^{-7}$	$4.28E^{1} \pm 2.5E^{1}$	$3.31E^{1} \pm 1.13E^{1}$
F5	$8.61 \pm 1.34E^{1}$	$2.64E^{1} \pm 8.2E^{-1}$	$\mathbf{4.54E^{-21} \pm 1.76E^{-20}}$	$2.68E^{1} \pm 6.71E^{-1}$	$2.77E^{1} \pm 4.2E^{-1}$	$7.27E^{3} \pm 1.53E^{4}$
F6	$3.54E^{-6} \pm 9.68E^{-6}$	$2.92E^{-1} \pm 2.32E^{-1}$	$\mathbf{3.62E^{-9} \pm 1.96E^{-8}}$	$5.27E^{-1} \pm 3.48E^{-1}$	$1.5E^{-1} \pm 1E^{-1}$	$1.58E^{1} \pm 2.93E^{1}$
F7	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$	$\mathbf{4.9E^{-1} \pm 2.78E^{-1}}$
F8	$\mathbf{-1.26E^{4}} \pm 3.15E^{-5}$	$-8.65E^{3} \pm 7.25E^{2}$	$\mathbf{-1.26E^{4} \pm 8.57E^{-10}}$	$-6.06E^{3} \pm 1.2E^{3}$	$-1.09E^{4} \pm 1.65E^{3}$	$-3.78E^{3} \pm 2.44E^{2}$
F9	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	1.83 ± 2.26	0 ± 0	$3.31E^{1} \pm 4.15E^{1}$
F10	$2.43E^{-15} \pm 2.22E^{-15}$	$\mathbf{8.88E^{-16} \pm 0}$	$1.24E^{-15} \pm 1.08E^{-15}$	$6.07E^{-14} \pm 7.75E^{-15}$	$3.61E^{-15} \pm 2.26E^{-15}$	$1.25E^{1} \pm 9.36$
F11	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$\mathbf{0 \pm 0}$	$3.63E^{-2} \pm 8.28E^{-2}$	$1E^{-2} \pm 3E^{-2}$	$7.7E^{-1} \pm 3.5E^{-1}$
F12	$\mathbf{1.57E^{-32} \pm 5.57E^{-48}}$	$6.33E^{-3} \pm 5.94E^{-3}$	$2.78E^{-28} \pm 8.35E^{-28}$	$3.74E^{-2} \pm 1.71E^{-2}$	$1E^{-1} \pm 4.7E^{-1}$	$1.89E^{4} \pm 1E^{5}$
F13	$\mathbf{1.35E^{-32} \pm 5.57E^{-48}}$	$2.13 \pm 2.57E^{-1}$	$1.36E^{-32} \pm 3.13E^{-34}$	$5.29E^{-1} \pm 1.69E^{-1}$	$2.6E^{-1} \pm 1.3E^{-1}$	$8.06E^{5} \pm 2.79E^{5}$

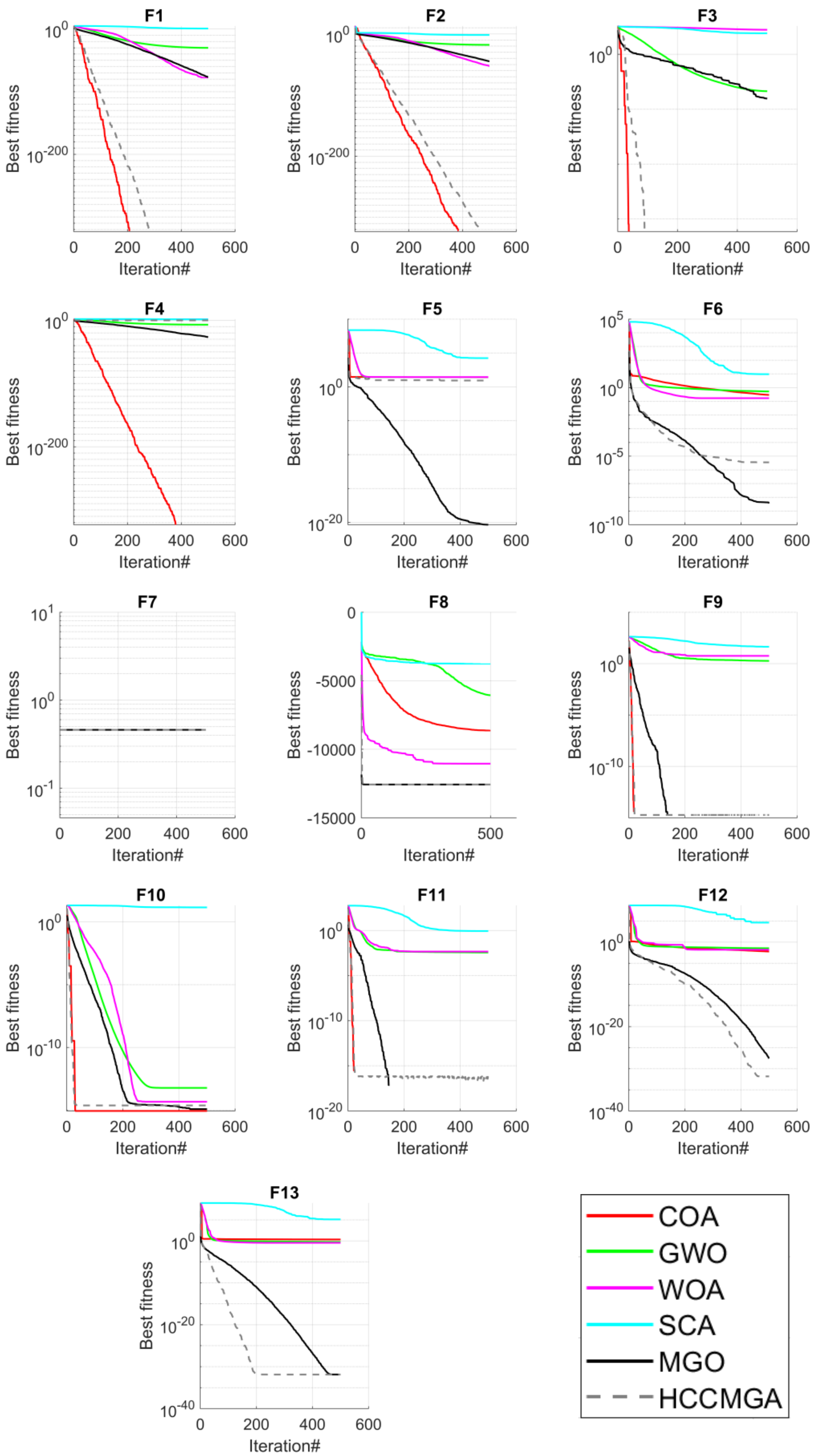

Figure 4.2: Convergence performance of different algorithms for F1 – F13

The experimental statistical results for each of the functions are presented in Table 4.3, while the convergence speed for each of the algorithm based on each function from F1 to F13 is presented in Figure 4.2. The results are presented in the form of *mean*$\pm$*standard deviation* and they are based on the average results over 30 runs. We can observe that the proposed HCCMGA achieves the lowest mean values in nine out of the 13 benchmark functions. Meanwhile, COA reaches the lowest mean value in eight test functions whereas MGO excels in six test functions.

For unimodal test functions (i.e., F1 – F7), HCCMGA and COA reach the best optimum mean values in F1, F2, and F3. These two algorithms have also achieved the lowest standard deviations, which implies a high level of stability. For F4, only COA achieves the optimum mean value. Meanwhile, for F5, HCCMGA places second behind MGO, which yields a better mean and standard deviation values. Next, for F6, MGO attains the lowest mean value, followed by HCCMGA. Lastly, all algorithms obtain similar mean and standard deviation values for F7. In sum, HCCMGA is comparable with COA for unimodal functions, with the latter being slightly better.

For multimodal test functions (i.e., F8 – F13), HCCMGA achieves the lowest mean value in five out of the six test functions, while COA and MGO have only excelled in three out of the six functions. In F8, both HCCMGA and MGO reach the global average minimum mean value. However, MGO has a slightly better stability than HCCMGA due to the fact that the former achieves a lower standard deviation. Besides that, HCCMGA, COA, and MGO achieve the global minimum for F9 and F11 with the lowest standard deviations. Furthermore, for F10, COA achieves the lowest mean and standard deviation values. For F12 and F13, HCCMGA is the best performer, followed by MGO.

Figure 4.2 shows the convergence graphs of the different metaheuristics for F1 – F13. We can see that the proposed HCCMGA converges well in most of the test functions, except F4 and F5. For F1, COA converges faster than HCCMGA, as COA yields the minimum fitness value at around 200 iterations while HCCMGA obtains the minimum fitness value at around 250 iterations. This trend can also be found for functions F2 and F3. For function

F4, COA performs well than others algorithms as it is the only algorithm that can find the global minimum in less than 500 iterations. For function F5, MGO converges faster than other algorithms to the minimum fitness value. Meanwhile, for F6, HCCMGA and MGO have similar convergence curve in the first 250 iterations. After that, MGO performs better since it can converges into lower result. For F7, we can see that the convergence of all algorithms is similar since they can find the similar minimum mean value. For F8, both HCCMGA and MGO exhibit the best convergence curves. For F9, COA and HCCMGA outperform MGO in convergence speed. Next, for F10, MGO converges to the solution after 200 iterations, while COA and HCCMGA converge to the optimal solution before 50 iterations, with HCCMGA being slightly faster. Furthermore, for F11, HCCMGA shows the highest convergence speed, followed by COA. For F12 and F13, HCCMGA outperforms other algorithms in convergence speed, followed by MGO.

4.2.3 Overview of the Overall Result

In overall, based on the two simulations that are conducted, it is clear that the HCCMGA algorithm that has been designed has equal reliance on COA and MGO algorithm since the HCCMGA with splitting factor S of 15:15 is the best performer among all HCCMGA with different splitting factor based on solution quality, stability, and convergence. Next, it is also clear that the proposed HCCMGA algorithm outperforms state-of-the-arts optimization algorithms in solution quality, search stability and convergence speed, as the HCCMGA achieves solutions closest to the optimum in eight out of the 13 benchmark functions with low standard deviations and less numbers of iterations.

4.3 Experimental result of applying HCCMGA on DBS positioning scheme

The current study considers that the ground users are distributed randomly in the area of 1500 m × 1500 m (Chen et al., 2020). β_0 = -30 dB, B = 1 Hz, P = 40dBm and σ^2 = -90 dBm are set for the network (Chen et al., 2020). The maximum number of ground users that can be served by each DBS is set to

N_{max} = 100. For comparison, COA and MGO are adopted, and all algorithms are evaluated over 100 runs.

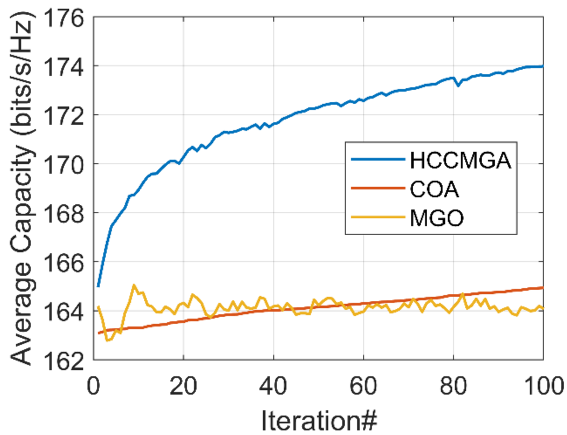

Figure 4.3: Convergence performance for the scenario with |*D*| = 10 DBSs and |*U*| = 100 users.

From Figure 4.3, it can be observed that the average capacity achieved by HCCMGA is around 75 bits/s/Hz, while the those achieved by COA and MGO are around 65 bits/s/Hz and 64 bits/s/Hz respectively, after converging within 100 iterations.

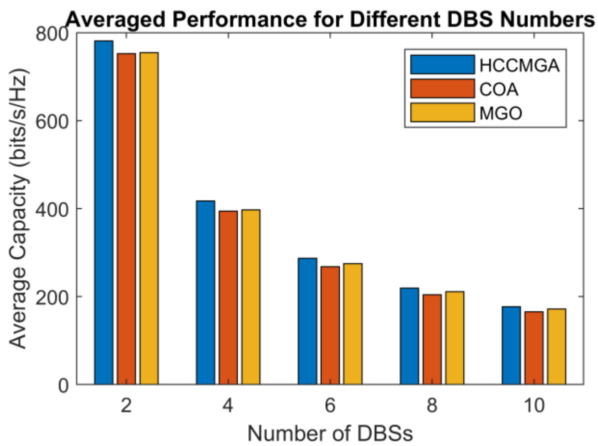

Figure 4.4: Average capacity performance for the scenario with |*U*| = 100 users.

Figure 4.4 shows that the average capacity achieved by HCCMGA is higher than those by COA and MGO across different numbers of DBSs. Moreover, there is an decreasing trend on average capacity for HCCMGA from

$|D|$ = 2 DBSs to $|D|$ = 10 DBSs, which is from around 800 bits/s/Hz to 200 bits/s/Hz, respectively.

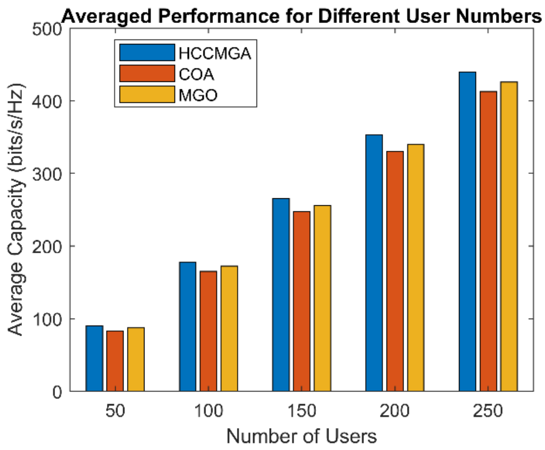

Figure 4.5: Average capacity performance for the scenario with $|D|$ = 10 DBSs.

Lastly, from Figure 4.5, it can be observed that the HCCMGA outperforms COA and MGO across different numbers of users. The average capacity achieved by HCCMGA increases from around 100 bits/s/Hz to 450 bits/s/Hz as the number of users increases from 50 to 250 users.

4.4 Summary

In short, this chapter clearly stated the result of testing the proposed HCCMGA with 13 standard benchmark test functions and applying the HCCMGA on DBS positioning scheme. Results show that the proposed HCCMGA outperforms state-of-the-arts optimization algorithms in solution quality, search stability and convergence speed, as the HCCMGA achieves solutions closest to the optimum in eight out of the 13 benchmark functions with low standard deviations and less numbers of iterations. Next, for the DBS positioning scheme, the result shows that the proposed HCCMGA outperforms existing COA and MGO algorithms on DBS positioning scheme as the HCCMGA provides more average coverage radius than COA and MGO.

CHAPTER 5

CONCLUSION

5.1 Conclusion

In this project, a hybrid cooperative COA-MGO algorithm, namely HCCMGA, has been proposed, which is developed based on cooperative solution information exchange between COA and MGO during the optimization process. The performance of the proposed HCCMGA is compared with the standard COA, MGO, GWO, WOA and SCA algorithms using 13 benchmark test functions consisting of unimodal and multimodal ones. Simulation results show that the proposed HCCMGA has achieved the best mean and standard deviation values for majority of the test functions, with sufficiently high convergence speeds. The HCCMGA has also been applied to solve a DBS placement problem, and it is shown to achieve substantial improvements in average capacity maximization for DBS communication networks.

5.2 Limitations and Recommendations

5.2.1 Discrete Optimization

In this project, the objective function that has been created is only be able to solve the problem which include the variable of continuous value. It cannot solve the problem that have the variable of discrete value. Therefore, in section 3.5.2, we have used a greedy algorithm to obtain the solutions of $a_{d,u}$ for the fitness function. In the future, we may need to modify the equation in the system model and problem formulation section so that it can also solve the discrete optimization problem.

5.2.2 Limited Channel Bandwidth

The DBS positioning scheme for this project does not address limited channel bandwidth of drones. In future works, to reflect the real-world capabilities of DBSs, the DBS positioning scheme that we have developed should be modified so that the DBSs will have unlimited channel bandwidth and can serve any number of users. Besides that, a one-way association between user and a specific DBS has been used to reach the fairness provision in the scheme. In future, a two-way user-DBS association can be formulated to the objective function to limit the numbers of users served by a DBS.

5.2.3 Mobile Users

In this project, the proposed schme assumed that the users in the ground area are static. However, in real-world, this is impossible since the users will move in the ground area to obtain better network connectivity. In the future, this problem can be solved by recalculateing the position of DBSs at fixed time intervals based on ground users' position that is changing continuously. Moreover, instead of collecting real-time data from the ground users' position, machine learning approaches can also be implemented to forecast how users may move around the service area.

REFERENCES

Abdollahzadeh, B., Gharehchopogh, F.S., Khodadadi, N. and Mirjalili, S., 2022. Mountain Gazelle optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems. *Advances in Engineering Software*, 174, p.103282. http://dx.doi.org/10.1016/j.advengsoft.2022.103282.

Akram, T., Awais, M., Naqvi, R., Ahmed, A. and Naeem, M., 2020. Multicriteria UAV base stations placement for Disaster Management. *IEEE Systems Journal*, 14(3), pp.3475–3482. http://dx.doi.org/10.1109/JSYST.2020.2970157.

Al-Hourani, A., Kandeepan, S. and Lardner, S., 2014. Optimal lap altitude for maximum coverage. *IEEE Wireless Communications Letters*, 3(6), pp.569–572. https://doi.org/10.1109/LWC.2014.2342736

Alsharif, M.H., Kelechi, A.H., Albreem, M.A., Chaudhry, S.A., Zia, M.S. and Kim, S., 2020. Sixth generation (6G) wireless networks: Vision, research activities, challenges and potential solutions. *Symmetry*, 12(4), p.676. http://dx.doi.org/10.3390/sym12040676.

Alzenad, M., El-Keyi, A., Lagum, F. and Yanikomeroglu, H., 2017. 3-D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage. *IEEE Wireless Communications Letters*, 6(4), pp.434–437. https://doi.org/10.1109/LWC.2017.2700840.

Alzenad, M., El-Keyi, A. and Yanikomeroglu, H., 2018. 3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements. IEEE Wireless Communications Letters, 7(1), pp.38–41. https://doi.org/10.1109/LWC.2017.2752161.

Banafaa, M., Shayea, I., Din, J., Hadri Azmi, M., Alashbi, A., Ibrahim Daradkeh, Y. and Alhammadi, A., 2023. 6G Mobile Communication Technology: Requirements, targets, applications, challenges, advantages, and opportunities. *Alexandria Engineering Journal*, 64, pp.245–274. http://dx.doi.org/10.1016/j.aej.2022.08.017

Chen, R., Li, X., Sun, Y., Li, S. and Sun, Z., 2020. Multi-uav coverage scheme for average capacity maximization. *IEEE Communications Letters*, 24(3), pp.653–657. https://doi.org/10.1109/LCOMM.2019.2962774

Chen, S., Liang, Y.-C., Sun, S., Kang, S., Cheng, W. and Peng, M., 2020. Vision, requirements, and technology trend of 6G: How to tackle the challenges of system coverage, capacity, user data-rate and movement speed. *IEEE Wireless Communications*, 27(2), pp.218–228. https://doi.org/10.1109/MWC.001.1900333

Chen, Y., Li, N., Wang, C., Xie, W. and Xv, J., 2018. A 3D placement of unmanned aerial vehicle base station based on multi-population genetic algorithm for maximizing users with different QoS requirements. *2018 IEEE 18th International Conference on Communication Technology (ICCT)*. https://doi.org/10.1109/ICCT.2018.8600206

Chowdhury, M.Z., Shahjalal, Md., Ahmed, S. and Jang, Y.M., 2020. 6G wireless communication systems: Applications, requirements, technologies, challenges, and Research Directions. *IEEE Open Journal of the Communications Society*, 1, pp.957–975. https://doi.org/10.1109/OJCOMS.2020.3010270

Dorigo, M., Maniezzo, V. and Colorni, A., 1996. Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1), pp.29–41. https://doi.org/10.1109/3477.484436

Du, W., Ying, W., Yang, P., Cao, X., Yan, G., Tang, K. and Wu, D., 2020. Network-based heterogeneous particle swarm optimization and its application in UAV communication coverage. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(3), pp.312–323. https://doi.org/10.1109/TETCI.2019.2899604

Eiben, A.E. and Schippers, C.A., 1998. On evolutionary exploration and Exploitation. *Fundamenta Informaticae*, 35(1–4), pp.35–50. http://dx.doi.org/10.3233/FI-1998-35123403

Goldberg, D.E., 1989. Genetic algorithms in search, optimization, and machine learning. *Choice Reviews Online*, 27(02). https://doi.org/10.5860/choice.27-0936

Hadiwardoyo, S.A., Calafate, C.T., Cano, J.-C., Krinkin, K., Klionskiy, D., Hernández-Orallo, E. and Manzoni, P., 2020. Three dimensional UAV positioning for dynamic UAV-to-car communications. *Sensors*, 20(2), p.356. https://doi.org/10.3390/s20020356

Hammouti, H.E., Hamza, D., Shihada, B., Alouini, M.-S. and Shamma, J.S., 2021. The optimal and the greedy: Drone association and positioning schemes for internet of UAVs. IEEE Internet of Things Journal, 8(18), pp.14066–14079. http://dx.doi.org/10.1109/JIOT.2021.3070209

Huang, T., Yang, W., Wu, J., Ma, J., Zhang, X. and Zhang, D., 2019. A survey on Green 6G network: Architecture and Technologies. *IEEE Access*, 7, pp.175758–175768. https://doi.org/10.1109/ACCESS.2019.2957648

Jia, H., Rao, H., Wen, C. and Mirjalili, S., 2023. Crayfish optimization algorithm. *Artificial Intelligence Review*, 56(S2), pp.1919–1979. http://dx.doi.org/10.1007/s10462-023-10567-4

Jiang, S., Lee, Y.L., Tham, M.L., Qin, D., Chang, Y.C. and Sim, A.G., 2023. Proportionally fair load balancing with statistical quality of service provisioning for Aerial Base Stations. *ETRI Journal*, 45(5), pp.887–898. http://dx.doi.org/10.4218/etrij.2023-0035

Kalantari, E., Yanikomeroglu, H. and Yongacoglu, A., 2016. On the number and 3D placement of drone base stations in Wireless Cellular Networks. *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. https://doi.org/10.1109/VTCFall.2016.7881122

Kennedy, J. and Eberhart, R., 2002. Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4, pp.1942–1948. https://doi.org/10.1109/ICNN.1995.488968

Lee, Y.L., Chuah, T.C., El-Saleh, A.A. and Loo, J., 2018. User association for backhaul load balancing with quality of service provisioning for Heterogeneous Networks. *IEEE Communications Letters*, 22(11), pp.2338–2341. https://doi.org/10.1109/LCOMM.2018.2867181

Lee, Y.L., Tan, W.L., Lau, S.B., Chuah, T.C., El-Saleh, A.A. and Qin, D., 2020. Joint Cell Activation and User Association for Backhaul load balancing in Green Hetnets. *IEEE Wireless Communications Letters*, 9(9), pp.1486–1490. https://doi.org/10.1109/LWC.2020.2994643

Letaief, K.B., Chen, W., Shi, Y., Zhang, J. and Zhang, Y.-J.A., 2019. The roadmap to 6G: Ai Empowered Wireless Networks. *IEEE Communications Magazine*, 57(8), pp.84–90. https://doi.org/10.1109/MCOM.2019.1900271

Li, B., Chen, C., Zhang, R., Jiang, H. and Guo, X., 2018. The energy-efficient UAV-based BS coverage in air-to-ground communications. *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. https://doi.org/10.1109/SAM.2018.8448770

Lim, N.H., Lee, Y.L., Tham, M.L., Chang, Y.C., Sim, A.G. and Qin, D., 2021. Coverage optimization for UAV base stations using simulated annealing. *2021 IEEE 15th Malaysia International Conference on Communication (MICC)*. https://doi.org/10.1109/MICC53484.2021.9642136

Lim, Y.K., Jiang, S., Khor, K.C., Lee, Y.L. and Ke, F., 2023. Fairness-aware unmanned aerial vehicle-mounted base station placement with Quality of Service provisioning. 2023 International Conference on Computer, Control, Informatics and its Applications (IC3INA). http://dx.doi.org/10.1109/IC3INA60834.2023.10285774

Lin, X.-H., Bi, S.-Z., Cheng, N., Dai, M.-J. and Wang, H., 2022. An α-fairness approach to balancing the energy consumption among sensors for UAV–IOT systems. *IEEE Internet of Things Journal*, 9(18), pp.17965–17978. https://doi.org/10.1109/JIOT.2022.3164403

Merwaday, A., Tuncer, A., Kumbhar, A. and Guvenc, I., 2016. Improved throughput coverage in natural disasters: Unmanned Aerial Base Stations for public-safety communications. *IEEE Vehicular Technology Magazine*, 11(4), pp.53–60. https://doi.org/10.1109/MVT.2016.2589970

Mirjalili, S., 2015. The Ant Lion optimizer. *Advances in Engineering Software*, 83, pp.80–98. https://doi.org/10.1016/j.advengsoft.2015.01.010

Mirjalili, S., 2016. SCA: A sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems*, 96, pp.120–133. https://doi.org/10.1016/j.knosys.2015.12.022

Mirjalili, S. and Hashim, S.Z., 2010. A new hybrid PSOGSA algorithm for function optimization. *2010 International Conference on Computer and Information Application*.

https://doi.org/10.1109/ICCIA.2010.6141614

Mirjalili, S. and Lewis, A., 2016. The whale optimization algorithm. *Advances in Engineering Software*, 95, pp.51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Mirjalili, S., Mirjalili, S.M., and Lewis, A., 2015. Grey wolf optimizer. *Advances in Engineering Software*, 69, pp.46-61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Mozaffari, M., Saad, W., Bennis, M., Nam, Y.-H. and Debbah, M., 2019. A tutorial on uavs for wireless networks: Applications, challenges, and open problems. *IEEE Communications Surveys & Tutorials*, 21(3), pp.2334–2360. https://doi.org/10.1109/COMST.2019.2902862

Muralidharan, S., Li, L., Kim, J., Lütkenhaus, N., Lukin, M.D. and Jiang, L., 2016. Optimal architectures for long distance quantum communication. *Scientific Reports*, 6(1).

https://doi.org/10.1038/srep20463

Ouamri, M.A., Oteşteanu, M.-E., Barb, G. and Gueguen, C., 2022. Coverage analysis and efficient placement of Drone-BSS in 5G Networks. *ICCEIS 2021*. https://www.mdpi.com/2673-4591/14/1/18#

Raja, G., Saravanan, G. and Dev, K., 2023. 6G-assisted UAV-truck networks: Toward Efficient Essential Services Delivery. *IEEE Communications Standards Magazine*, 7(3), pp.4–9. http://dx.doi.org/10.1109/MCOMSTD.0003.2200003

Sun, Y. and Fang, L., 2023. Method for passive location of UAV conical formation based on genetic algorithm. *2023 3rd International Conference on Electronic Information Engineering and Computer (EIECT)*. https://doi.org/10.1109/EIECT60552.2023.10442525

Valiulahi, I. and Masouros, C., 2021. Multi-uav deployment for throughput maximization in the presence of co-channel interference. *IEEE Internet of Things Journal*, 8(5), pp.3605–3618. http://dx.doi.org/10.1109/JIOT.2020.3023010

vandenBergh, F. and Engelbrecht, A.P., 2004. A cooperative approach to particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3), pp.225–239. https://doi.org/10.1109/TEVC.2004.826069

Wolpert, D.H. and Macready, W.G., 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), pp.67–82. https://doi.org/10.1109/4235.585893

Zeng, Y., Zhang, R. and Lim, T.J., 2016. Wireless Communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Communications Magazine*, 54(5), pp.36–42. http://dx.doi.org/10.1109/MCOM.2016.7470933

Zhang, W. and Zhang, W., 2022. Efficient UAV localization based on modified particle swarm optimization. *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*.

https://doi.org/10.1109/ICCWorkshops53468.2022.9814686

LIST OF PUBLICATIONS

1. Lee B. H., Lee Y. L., and Khor K. C., 2024. Hybrid Cooperative Crayfish and Mountain Gazelle Algorithm for Global Optimization. *ICITDA 2024*. (Conference Paper, Status: Accepted)
2. Lee B. H., Lee Y. L., Khor K. C., and Lee J. B., 2024. Hybrid Cooperative Crayfish and Mountain Gazelle Algorithm for Global Continuous Optimization with Applications on Drone Base Station Placement (Journal, Status: To be submitted)