Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa

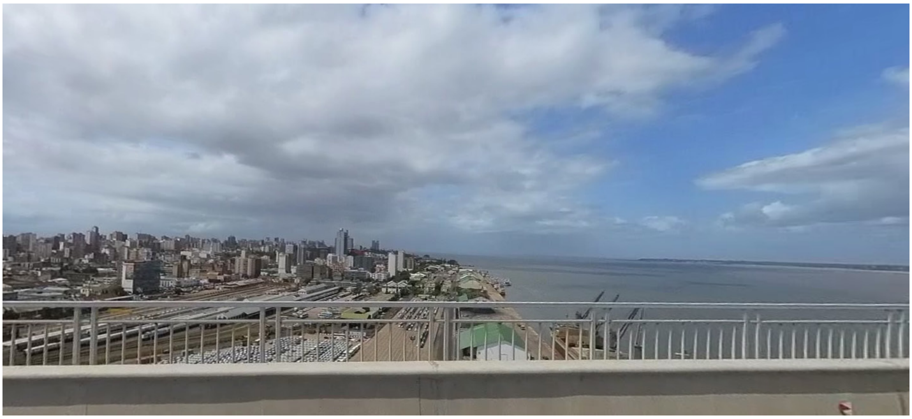
Sicuaio, Tomé Eduardo

2024

Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa

TOMÉ EDUARDO SICUAIO

DEPARTMENT OF PHYSICAL GEOGRAPHY AND ECOSYSTEM SCIENCE | LUND UNIVERSITY



Department of Physical Geography
and Ecosystem Science
Faculty of Science

ISBN 978-91-89187-39-9

Multi-objective Optimization and GIS to Improve Climate Change
Induced Disaster Risk Management in Africa

Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa

Tomé Eduardo Sicuaio

LUND
UNIVERSITY

DOCTORAL DISSERTATION

Doctoral dissertation for the degree of Doctor of Philosophy (PhD) at the Faculty of Science at Lund University to be publicly defended on Tuesday, June 11th 2024, at 10.00 in Varlden Hall, Department of Physical Geography and Ecosystem Science, Geocentrum I, Sölveggatan 10, Lund

Faculty opponent
Professor Qiming Zhuo

Organization: LUND UNIVERSITY

Document name: DOCTORAL DISSERTATION

Date of issue: 2024-06-11

Author(s): Tomé Eduardo Sicuaio

Sponsoring organization: Swedish International Development Agency (SIDA)

Title: Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa

Abstract:

We live in an era that suffers from climate change issues, not least disaster risks induced by climate change. While measures, such as proper evacuation plans, are required to reduce the negative impacts of disasters when they hit, actions should also be taken to reduce climate change effects by e.g. increasing the use of renewable energies or proper urban land-use allocation, where climate change factors are considered among other criteria. Planning and decision-making on these issues are usually complex and complicated, since several criteria, usually conflicting with each other, should be taken into consideration.

Multi-objective optimization (MOO) has proven to be a proper technique for solving multi-criteria decision analysis problems, where criteria are conflicting. Metaheuristic algorithms, inspired from nature, has been developing and showing a proper performance for solving complex MOO problems. Meanwhile, these algorithms yet need to be modified and adjusted to perform well, for each specific case study project. This research aims to improve metaheuristic algorithms to make them suitable for solving some spatial problems related to disaster risk management.

The research started by making a comparative study between well-known multi-objective optimization algorithms in order to not only learn about MOO, but also get an insight about the performance of algorithms and how they could be enhanced (Paper 1). Then the study continued by proposing a modified multi-objective cuckoo search algorithm for evacuation planning (Paper 2). The third study was in the context of the impact of urban land-uses on climate change, and hence modified and applied the Non-dominant Sorting Genetic Algorithm – III (NSGA-III) for urban land-use planning in Mozambique (Paper 3). The last study was about modifying NGSA-II for solar farm site selection in Mozambique (Paper 4).

The results of the above studies demonstrated the high potential of metaheuristic algorithms and multi-objective optimization to solve complex spatial problems that in turn can facilitate planning and decision-making to prevent or respond climate change induced disaster risks. The performances of the modified and original algorithms were compared. The evaluation showed improved performance for each of the selected case studies.

Key words: Urban planning, multi-objective optimization, Evacuation planning, Land use allocation, Geographical Information Science, Metaheuristic algorithms.

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language: English

Number of pages: 108

ISSN and key title:

ISBN (Print): 978-91-89187-39-9

ISBN (PDF) : 978-91-89187-40-5

Recipient's notes

Price

Security classification

Signature

Date 2024-04-19

Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa

Tomé Eduardo Sicuaio

LUND UNIVERSITY

Coverphoto by Julia Grahl, CC BY-SA 4.0, via Wikimedia Commons and back cover by Tomé Eduardo Sicuaio

Faculty of Science

Department of Physical Geography and Ecosystem Science

ISBN (Print): 978-91-89187-39-9

ISBN (PDF): 978-91-89187-40-5

Printed in Sweden by Media-Tryck, Lund University

Lund 2024

NORDIC SWAN ECOLABEL

Printed matter 3041 0903

Media-Tryck is a Nordic Swan Ecolabel certified provider of printed material. Read more about our environmental work at www.mediatryck.lu.se

MADE IN SWEDEN

To my wonderful family

Table of Contents

Acknowledgments

I would like to express my deepest gratitude to my advisor, Prof. Ali Mansourian, whose unwavering support, guidance, and expertise have been invaluable throughout this doctoral journey. His mentorship has shaped not only the academic content of this thesis but also my growth as a researcher. I extend my sincere thanks to the members of my doctoral committee Prof. Petter Pilesjö, Prof. Andrey Shyndyapin, Prof. Manuel Joaquim Álves, Dr. Pengxiang Zhao, and Dr. Andreas Persson, for their insightful feedback and constructive criticism. Your collective expertise has enriched the quality of this research.

I am grateful to the Department of Physical Geography and Ecosystem Science for providing the resources and environment conducive to scholarly pursuits. Special thanks to Prof. Petter Pilesjö for fostering an atmosphere of academic excellence. Heartfelt appreciation goes to my colleagues, Olive Niyomubyeyi, Marcio Mathe, and Amida Mussa, and fellow graduate students Inês Alfredo and Sandra Sambo who have been companions on this intellectual voyage. The camaraderie and shared experiences have made this challenging endeavor more rewarding.

I owe a debt of gratitude to my wife Dora Sicuaio, my sons Eduardo, Elvis, and Tome, and my daughter Mariana for their unwavering support, understanding, and encouragement. Your belief in my abilities has been a constant source of motivation. I owe a debt of gratitude to my family for their unwavering support, understanding, and encouragement. Your belief in my abilities has been a constant source of motivation.

I am sincerely grateful for the generous financial support provided by the Swedish International Development Cooperation Agency (SIDA) through the Eduardo Mondlane University Mozambique-Sweden Programme for Research, Higher Education, and Institutional Advancement in collaboration with Lund University. This funding has been instrumental in facilitating my academic pursuits and ensuring the successful completion of my doctoral studies. I extend my heartfelt thanks to the Eduardo Mondlane University-Sweden Coordination Office for their efficient administrative support, which has been pivotal in navigating the various aspects of my research journey. Special appreciation is reserved for the Team Leader of the GIS Sub-programme, Prof. Andrey Shyndyapin, whose unwavering support and valuable advice have been invaluable throughout my studies. His guidance played a crucial role in shaping the trajectory of my research.

Last but not least, I want to express my appreciation to all the participants who generously contributed their time and insights to this study. Your willingness to share your experiences has been crucial to the success of this research. This journey would not have been possible without the support of all these individuals, and for that, I am truly thankful.

List of Papers

I. Niyomubyeyi, O., Sicuaio, T.E., Diaz Gonzalez, J.I., Pilesjö, P., Mansourian, A. A Comparative Study of Four Metaheuristic Algorithms, AMOSA, MOABC, MSPSO, and NSGA-II for Evacuation Planning. Algorithms 2020,13, 16. https://doi.org/10.3390/a13010016

II. Sicuaio, T., Niyomubyeyi, O., Shyndyapin, A., Pilesjö, P., Mansourian, A. Multi-Objective Optimization Using Evolutionary Cuckoo Search Algorithm for Evacuation Planning. Geomatics 2022, 2, 53–75. https://doi.org/10.3390/geomatics2010005

III. Sicuaio, T., Zhao, P., Pilesjö, P., Shindyapin, A., Mansourian, A. Sustainable and resilient land use planning: A multi-objective optimization approach. ISPRS Int. J. Geo-Inf. 2024, 13(3), 99; https://doi.org/10.3390/ijgi13030099

IV. Sicuaio, T., Zhao, P., Pilesjö, P., Shindyapin, A., Mansourian, A. A multi-objective optimization approach for solar farm site selection: Case study in Maputo, Mozambique. Submitted (under review).

Author Contributions

Paper I: Formal analysis of the results and process of editing the paper;

Paper II: Selection of the methodology, python implementation of the algorithm using DEAP framework, analysis, visualization of the results and writing;

Paper III: Conceptualization, methodology design, python implementation of the algorithm using pymoo framework, testing and analysis of the results, and writing;

Paper IV: Conceptualization, methodology design, python implementation of the algorithm using pymoo framework, analysis of the results and writing.

Abstract

We live in an era that suffers from climate change issues, not least disaster risks induced by climate change. While measures, such as proper evacuation plans, are required to reduce the negative impacts of disasters when they hit, actions should also be taken to reduce climate change effects by e.g. increasing the use of renewable energies or proper urban land-use allocation, where climate change factors are considered among other criteria. Planning and decision-making on these issues are usually complex and complicated, since several criteria, usually conflicting with each other, should be taken into consideration.

Multi-objective optimization (MOO) has proven to be a proper technique for solving multi-criteria decision analysis problems, where criteria are conflicting. Metaheuristic algorithms, inspired from nature, has been developing and showing a proper performance for solving complex MOO problems. Meanwhile, these algorithms yet need to be modified and adjusted to perform well, for each specific case study project. This research aims to improve metaheuristic algorithms to make them suitable for solving some spatial problems related to disaster risk management.

The research started by making a comparative study between well-known multi-objective optimization algorithms in order to not only learn about MOO, but also get an insight about the performance of algorithms and how they could be enhanced (Paper 1). Then the study continued by proposing a modified multi-objective cuckoo search algorithm for evacuation planning (Paper 2). The third study was in the context of the impact of urban land-uses on climate change, and hence modified and applied the Non-dominant Sorting Genetic Algorithm – III (NSGA-III) for urban land-use planning in Mozambique (Paper 3). The last study was about modifying NGSA-II for solar farm site selection in Mozambique (Paper 4).

The results of the above studies demonstrated the high potential of metaheuristic algorithms and multi-objective optimization to solve complex spatial problems that in turn can facilitate planning and decision-making to prevent or respond climate change induced disaster risks. The performances of the modified and original algorithms were compared. The evaluation showed improved performance for each of the selected case studies.

Abbreviations

ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
AIS	Artificial Immune Systems
AMOSA	Archive-based Multi-Objective Simulated Annealing
BA	Bat Algorithm
BB	Branch and Bound
CSA	Cuckoo Search Algorithm
IMOCS	Improved Multi-Objective Cuckoo Search
DP	Dynamic Programming
DRM	Disaster Risk Management
EOA	Elephant Optimization Algorithm
GA	Genetic Algorithm
GIS	Geographical Information Systems
GOA	Grasshopper Optimization Algorithm
IP	Integer Programming
LUA	Land Use Allocation
LP	Linear Programming
ML	Machine Learning
MOABC	Multi-Objective Artificial Bee Colony
MOCSA	Multi-Objective Cuckoo Search Algorithm
MOO	Multi-Objective Optimization
MSPSO	Multi-Objective Particle Swarm Optimization
NFP	Network Flow Programming
NSGA	Non-dominated Sorting Genetic Algorithm
PSO	Particle Swarm Optimization
SA	Simulated Annealing
SS	Scatter Search
TS	Tabu Search
TSP	Travelling Salesman Problem

1 Introduction

In recent years, the world has been facing an increase in climate change induced natural disasters, such as extreme heat waves, hurricanes, cyclones, floods, mega-fires, disease outbreaks, and drought that are influencing the whole world, not least Mozambique. In this context, there is a need to minimize the impact of the disaster risks on the environment and societies, by proper disaster risk management measures. Disaster risk management is a function of different variables, which are usually conflicting. This makes planning for disaster risk management a multi-criteria decision analysis (MCDA) problem. There are different techniques to solve MCDA problems, among them multi-objective optimization (MOO) that could be based on metaheuristic algorithms.

A heuristic technique is a problem-solving strategy that uses a practical approach to find solutions quickly and efficiently, even if they are not guaranteed to be optimal or perfect. Heuristics are often used when faced with complex or uncertain situations where it may be very difficult or sometimes impossible to find an exact solution. A metaheuristic technique is a higher-level problem-solving strategy or algorithmic framework used to find approximate solutions to difficult optimization and search problems [1]. Metaheuristics are designed to work with a wide variety of problem types and are especially useful when traditional optimization techniques, such as exact algorithms, become impractical due to the complexity or size of the problem.

Mozambique is a risk-prone country. It is ranked third among African countries exposed to risks from multiple weather-related hazards such as flooding, epidemics, cyclones and droughts [2]. Examples of such sever disasters can be listed as follows. Mozambique experienced flooding in many parts of the country in February and March 2000, causing the death of about 800 people, the loss of 20 000 cattle, and the inundation of 1400 km2 of arable land. There was another flood in December 2006, when the water overflowed the Cahora Bassa Dam leading to the deaths of 29 people and displacement of 121 000 more [3]. The tropical cyclone Eloise hit Mozambique on January 23, 2021, bringing devastating winds and extreme rainfall to Beira and neighboring districts. The cyclone caused severe flooding that claimed ten lives and caused massive damage. The norm of Emergency appeal has been revised to include the impact of the tropical storm Ana, which made landfall on January 24, 2022 and destroyed thousands of homes, as well as dozens of schools and hospitals. An estimated 125 000 people were affected, with many people already highly vulnerable from Eloise and other disasters in recent years [4]. There is a consensus among experts that climate change has a meaningful contribution in increased frequency and severity of these natural disasters.

Considering above, there is an urgent need to minimize the impact of natural disaster risks and make risk areas resilient to natural disaster risks. Examples of actions to be taken to achieve the aim are:

- In order to cope with climate change, among other actions, the use of renewable energies should be increased. Site selection for construction of renewable energy sites, e.g. solar farms, is a spatial problem to be solved in this respect.

- Vulnerability to natural disasters is associated with weak urban planning, which do not consider the effect of climate change. So, there is an immediate need to improve urban planning, in Mozambique, to reduce the effect of disaster risks and mitigate them in the long run.

- While the risks associated to climate change and urban structure is in place, proper disaster plans, among them evacuation plans, should be at hand to be able to keep people safe, when disasters hit.

These construct the three main case studies for this PhD thesis.

A variety of factors should be considered for evacuation planning, urban planning, and renewable energy site selection. These factors are usually conflicting with each other that makes it challenging and complicated to achieve a proper solution where all factors are optimized to their best. This calls for using MCDA to solve the problems, where there is trade-off between influencing factors in the final possible solutions. This thesis focuses on using multi-objective optimization (MOO) techniques for problem solving. However, these techniques need to be improved and adapted to fit the specific problems at hand that constructs the research aim and objectives of the thesis.

1.1 Research Aim and objectives

This research aims to improve multi-objective optimization (MOO) techniques to make them suitable for solving spatial problems. Evacuation planning, land-use planning, and site selection for renewable energies are three case studies of the research to achieve the aim.

The specific objectives of the research are:

- To review well-known MOO algorithms and their performances for evacuation planning.
- To improve and adopt a multi-objective cuckoo search algorithm for evacuation planning. The focus is on developing a multi-objective

optimization model designed to efficiently allocate victims to safe areas, and to guide them to emergency shelters through safe evacuation routes.

- To enhance and adopt the NSGA-III algorithm for Land Use Allocation (LUA) in urban areas, as part of sustainability and resilience urban planning.
- To improve and adopt the NSGA-II algorithm to solve solar farm site selection, as a contribution to the global shift towards sustainable energy practices.

1.2 Research Questions

The thesis helps us to get a better insight about the below questions:

- How are the performances of different MOO algorithms for a spatial problem?
- How can MOO algorithms be improved to perform better when solving spatial problems?
- How can MOO be useful to solve complex spatial problems?

1.3 Structure of the Thesis

After this introductory chapter, where the scope aim, and objectives of the thesis have been described, Chapter 2 delves into reviewing theories and applications of multi-objective optimization. In this chapter different techniques, ranging from classical to metaheuristic approaches have been reviewed. Similar studies related to the applications of MOO in evacuation planning, land use allocation, and site selection for renewable energy have been reviewed. In Chapter 3, the methodology employed for the implementation of the case studies (the four papers) have been described. Chapter 4 unveils the findings derived from each case study, providing an in-depth analysis of the results obtained. The thesis concludes with Chapter 5, where the comprehensive conclusions are presented. These conclusions are categorized into practical contributions, methodological advancements, and conceptual contributions. Finally, the chapter concludes by outlining potential avenues for future research.

2 Theories and Applications of MOO

Spatial analysis is a set of techniques used to analyze, interpret, and understand patterns and relationships in geographic data. It involves examining the spatial distribution of features, exploring spatial relationships, and deriving meaningful insights from geographic information. Spatial analysis is widely applied in various fields, including geography, environmental science, urban planning, epidemiology, and business intelligence. Spatial analysis plays a critical role in understanding complex spatial relationships, making informed decisions, and addressing spatial challenges in diverse fields. It continues to evolve with advancements in technology and data, contributing to improved spatial modeling, visualization, and problem-solving. Geographic Information System (GIS) is a foundational technology for spatial analysis, providing tools to capture, store, manipulate, analyze, and visualize spatial data. GIS is used for mapping, spatial querying, overlay analysis, and modeling to solve spatial problems.

2.1 Spatial Multi-criteria Decision Analysis (Spatial MCDA)

Spatial MCDA is an integration of GIS and decision-making techniques, where different variables influence the decision. Traditionally, multi-attribute optimization has been widely used for Spatial MCDA. Multi-attribute optimization is an approach used to address problems with multiple attributes or criteria, where these criteria may not necessarily conflict with each other. In multi-attribute optimization, the goal is to find solutions that are optimal with respect to the various attributes or criteria considered. This could involve techniques such as weighted sum methods, goal programming, or other aggregation approaches to identify a single optimal solution that balances the different criteria. However, in real world problems, variables are usually conflicting, which turned the attention of researchers on using multi-objective optimization (MOO) for Spatial MCDA. MOO focuses on optimizing solutions when there are multiple conflicting objectives that need simultaneous consideration. The aim of MOO is to find a set of solutions that represent the trade-offs between these objectives, known as the Pareto-optimal front. MOO algorithms seek to identify the best compromise solutions that offer the most favorable trade-offs across the multiple objectives.

The integration of spatial analysis and multi-objective optimization can be seen in applications such as:

- Urban Planning, optimizing land use while considering factors like environmental impact, accessibility, and social equity;
- Environmental Management, identifying optimal locations for conservation areas considering ecological diversity and human impact;
- Transportation Planning, finding optimal routes that minimize travel time, cost, and environmental impact.

The integration involves considering spatial constraints and relationships within the optimization process, ensuring that the resulting solutions make sense in the spatial context. While spatial analysis focuses on understanding and interpreting spatial data, multi-objective optimization deals with finding optimal solutions when facing conflicting objectives. Their integration is crucial in addressing complex real-world problems that involve both spatial considerations and multiple conflicting goals. A research trend, in this line, is to modify and improve MOO algorithms to make them suitable for spatial problems in hand.

The MOO techniques are classified into classical and heuristic methods. The classical techniques comprise linear programming, integer programming, Branch and Bound, Dynamic programming, and Network flow programming, while the heuristic techniques include Evolutionary algorithms, Swarm intelligence algorithms, and Neighborhood based algorithms (see Figure 1).

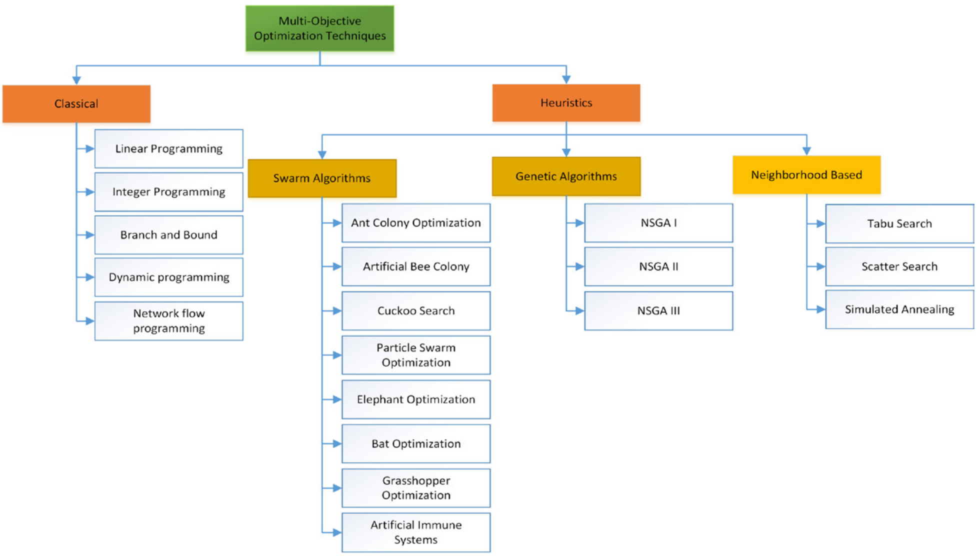


Figure 1: Flowchart of different Multi-objective optimization algorithms

Figure 1 provides a categorical sample of multi-objective optimization algorithms. The first category comprises heuristic techniques and classical techniques. The classical methods include Linear programming (LP), Integer programming (IP), Branch and Bound (BB), Dynamic Programming (DP), and Network Flow Programming (NFP). Heuristic methods include Swarm algorithms, Genetic algorithms, and Neighborhood based algorithms. Swarm algorithms are nature-inspired algorithms developed based on the interaction between living organisms such as flocks of birds, ants, and fish. This class of algorithms comprises Particle Swarm Optimization (PSO), and Artificial Bee Colony (ABC) algorithms, which are of the most popular algorithms in this category. Other recently proposed algorithms in this family are Cuckoo Search Algorithm (CS), Bat Algorithm (BA), Ant Colony Optimization (ACO), Elephant Optimization Algorithm (EOA), Artificial Immune Systems (AIS), and Grasshopper Optimization Algorithm (GOA). Genetic algorithms are search heuristics inspired by Charles Darwin's theory of natural evolution. These algorithms reflect the process of natural selection where the fittest individuals are selected for reproduction in order to produce the offspring of the next generation. Genetics algorithms comprise Non-dominated sorting genetic algorithm I (NSGA-I), NSGA-II, and NSGA-III. The Neighborhood based technique is a heuristic method for solving a set of combinatorial optimization and global optimization problems. Neighborhood based technique includes Tabu Search (TS), Scatter Search (SS), and Simulated Annealing (SA).

2.2 Classical Techniques

This section deals with classical techniques, comprising linear programming, which is an optimization method with both objective function and constraints modelled as a series of linear expressions with non-negative decision variables, integer programming (IP) is a mathematical optimization technique used to solve problems where decision variables are required to take integer values. It is a type of mathematical programming where the objective is to optimize a linear or nonlinear function subject to a set of constraints, while also imposing the additional requirement that some or all of the decision variables must be integers. Another classical technique is branch and bound, which consists of searching an optimal solution over a finite set of alternatives. Dynamic programming method is another classical technique, which solves certain types of sequential optimization problems by breaking them down into simpler problems. The Network Flow Programming method deals with the solution of problems modeled in terms of the flow of a commodity through a network.

2.2.1 Linear Programming

A linear programming (LP) problem is an optimization problem in which both the objective function and the constraints on the solution can be expressed as a series of linear expressions in the decision non-negative variables [5]. In an LP problem, the solution methods fall into two categories. One of the categories are the Simplex methods, which search the extreme points of the feasible region of the problem until the optimality conditions are satisfied. Any LP problem (primal) has its dual form [6]. The important point about an LP model is that the feasible region is a convex space and the objective function is a convex function. So, the optimal solution can be found at an extreme point of the feasible region. The Simplex procedures are good in finding the optimal solution, but they have poor worst-case time performance.

2.2.2 Integer Programming

An integer programming (IP) problem is a mathematical optimization or feasibility model in which some or all variables are restricted to be integers. The term IP refers to integer linear programming (ILP), in which the objective function and the constraints are linear [7]. Many important real-world problems can be formulated as integer programming problems such as production planning, scheduling, territorial partitioning, telecommunications networks, and cellular networks. The subject is so important that several monographs are devoted entirely to it [8]. Williams [9] concluded that IP models are generally much more difficult to solve than LP models.

2.2.3 Branch and Bound

The branch and bound (BB) method consists of finding an optimal solution over a finite set of alternatives. An obvious approach is to enumerate all the alternatives, based on a rooted tree, and then select the best [10]. The BB method uses the depth-first search and breadth-first search strategies to search the optimal solution in the tree in an ordered fashion. The depth-first search moves straight down a sequence of branches until it reaches a terminal node before backtracking up to the nearest junction. In contrast, breadth-first search enumerates all the branches at one level before moving on to the next level. Depth-first search finds a feasible solution early and it does not have the vast storage requirements of breadth-first search. In many implementations of the BB algorithm, the optimal solution is quickly found and it then spend most of the search time in proving that this is in fact the optimal solution. Thus, if there is insufficient time to complete the search, the best solution to date can be taken as a heuristic solution to the problem [11].

2.2.4 Dynamic Programming

The dynamic programming (DP) is a procedure that solves certain types of sequential optimization problems by breaking them down into simpler problems [12]. It solves the problem in stages, dealing with all options at a particular stage before moving on to the next. In this sense it can often be represented as a breadth-first search [13, 14]. Sniedovich [15] stated that the design of a DP algorithm for a particular problem involves three tasks; the definition of the stages and states, the derivation of a simple formula for the cost/value of the starting stage/state(s) and the derivation of a recursive relationship for all states at stage k in terms of previous stages and states. One of the main disadvantages of a DP approach is that the number of sub-problems that needs to be solved is dependent not only on the stages but also on the states. While the number of stages is usually related to the size of the problem in the traditional sense, the number of states is frequently related to the size of the constants in the problem. What is extremely important is the fact that these techniques in many cases do not require as much advanced mathematical training as the other classical methods such as IP or BB do [16].

2.2.5 Network Flow Programming

Network flow programming (NFP) deals with the solution of problems that can be modeled in terms of the flow of a commodity through a network. The NFP solves a wide range of areas such as the flow of current in electrical networks, the flow of fluids in pipeline networks, information flow in communications networks, traffic flow in road or rail networks, and problems such as shortest path, spanning tree, matching and location problems, as well as, scheduling and allocation problems to the analysis of medical x-rays [17]. The NFP algorithms for the maximum flow and minimum cost flow problems are relatively simple. The out-of-kilter algorithm also has the advantage that it is easy to find an initial solution as the upper and lower bounds do not need to be satisfied. However, these are not necessarily the most efficient algorithms in each case. The Ford–Fulkerson algorithm [18]is only bounded by the capacity on the arcs. There are also inherent inefficiencies in the algorithms, e.g. that subsequent iterations may need to recalculate labels already calculated in previous iterations. The network simplex algorithm, featured in the comprehensive study on network analysis, stands out as a notably more efficient approach [19].

2.3 Heuristic Algorithms

Heuristic algorithms comprise population based, neighborhood based, and other algorithms. The population-based algorithms include evolutionary algorithms and swarm intelligence algorithms. Evolutionary Algorithms are efficient heuristic search methods based on Darwinian evolution with powerful characteristics of robustness and flexibility to capture global solutions of complex optimization problems. The Swarm Intelligence algorithms are inspired from simple behaviors and self-organizing interaction among agents, such as ant colonies foraging, bird flocking, animal herding, bacterial growth, honey bees, fish schooling, and so on. An evolutionary algorithm (EA) is a subset of evolutionary computation, a generic population-based metaheuristic optimization algorithm. An EA uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection. Candidate solutions to the optimization problem play the role of individuals in a population, and the fitness function determines the quality of the solutions (see also loss function). Evolution of the population then takes place after the repeated application of the above operators. Below, are described EAs, which include Genetic algorithms, Genetic programming, Evolutionary Programming, Differential Evolution, and Evolutionary Strategy.

2.3.1 Evolutionary Algorithms

Evolutionary algorithms (EAs) are computational techniques inspired by the principles of biological evolution, such as natural selection and genetic inheritance. These algorithms are used to solve optimization and search problems by mimicking the process of natural selection to evolve solutions over successive generations. Evolutionary algorithms are versatile and have been applied to a wide range of optimization problems in various fields, including engineering, economics, bioinformatics, and machine learning. Common variants of EAs include genetic algorithms (GA), evolutionary strategies (ES), and differential evolution (DE), each with its own characteristics and applications.

2.3.1.1 Genetic Algorithms (GAs)

Genetic Algorithms (GAs) were invented by John Holland who developed this idea in his book "Adaptation in natural and artificial systems" in the year of 1975. A Genetic algorithm (GA) is a metaheuristic technique inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms [20]. A few applications of GA are as follows: Nonlinear dynamical systems–predicting, data analysis, robot trajectory planning, finding shape of protein molecules, TSP and sequence scheduling. In addition, functions for creating images, control–gas pipeline, pole balancing, missile evasion pursuit, design–semiconductor layout, aircraft design, keyboard configuration, communication networks, scheduling–manufacturing, facility scheduling, resource allocation, and machine learning–designing neural networks.. The advantages of genetic algorithms include; solution space is wider, easy to discover global optimum, only uses function evaluations, easily modified for different problems, very robust to difficulties in the evaluation of the objective function, and they are resistant to becoming trapped in local optima. The limitations of genetic algorithms include the problem of identifying fitness function, the problem of choosing the various parameters like the size of the population, mutation rate, cross over rate, the selection method and its strength, no effective terminator, and having trouble finding the exact global optimum.

Multi-objective genetic algorithms comprise non-dominated sorting genetic algorithm I, II, and III. The non-dominated sorting genetic algorithm (NSGA) proposed in [21] was one of the first such evolutionary algorithms. Over the years, the main criticisms of the NSGA approach have been as follows: high computational complexity of non-dominated sorting $O((MN)^3)$, lack of elitism, and need for specifying the sharing parameter [22]. Non-dominated Sorting Genetic Algorithm II alleviates all of the above three difficulties. Specifically, a fast non-dominated sorting approach with $O((MN)^2)$ computational complexity is present, and also, a selection operator is present that creates a mating pool by combining the parent and offspring populations and selecting the best N solutions (with respect to fitness and spread). NSGA-II is an evolutionary multi-objective optimization algorithm that has been applied to a wide variety of search and optimization problems since its publication in 2000 [23]. NSGA-III, developed by Deb and Jain [24], is similar to the original NSGA-II algorithm with significant changes in its selection operator. But, unlike in NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying and adaptively updating a number of well-spread reference points. NSGA-III uses a predefined set of reference points to ensure diversity in obtained solutions. A key aspect of NSGA-III is that it does not require any additional parameters.

2.3.1.2 Differential evolution (DE)

Differential evolution (DE), introduced by Storn and Price in the 1990s [25], is a method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality [25]. Such methods are commonly known as metaheuristics as they make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. DE optimizes a problem by maintaining a population of candidate solutions and creating new candidate solutions by combining existing ones according to its simple formulae, and then keeping whichever candidate solution that has the best score or fitness of the optimization problem at hand. In this way, the optimization problem is treated as a black box that merely provides a measure of quality given a candidate solution, and the gradient is therefore not needed. Strengths of DE are e.g.: (1)

Ability to handle non-differentiable, nonlinear and multimodal cost functions, (2) Parallelizability to cope with computation intensive cost functions, and (3) Ease of use, i.e. few control variables to steer the minimization. These variables should also be robust and easy to choose. A forth strength is (4) Good convergence properties, i.e. consistent convergence to the global minimum in consecutive independent trials.

2.3.1.3 Evolutionary Strategy (ES)

Evolutionary Strategy (ES) is a sub-class of nature-inspired direct search (and optimization) methods belonging to the class of Evolutionary Algorithms (EAs) which use mutation, recombination, and selection applied to a population of individuals containing candidate solutions in order to evolve iteratively better and better solutions [26, 27]. The Evolution Strategies (ES) algorithm boasts several strengths [28, 29] that contribute to its appeal and effectiveness in optimization tasks. Firstly, its implementation is remarkably straightforward and fast, as it does not require back-propagation. This simplicity facilitates quick deployment and execution. Secondly, ES does not rely on a differentiable policy, allowing for the use of various function approximations, including binary ones, without the need for complex gradient calculations. Thirdly, ES does not necessitate storing all episodes for future updates, eliminating the need for massive memory resources such as those required for experience replay. Fourthly, only one network for the policy needs to be defined, omitting the requirement for a value function. Fifthly, ES demonstrates superior exploration behavior compared to other policy search techniques, as it can generate more random behavior by directly tweaking the weights. Sixthly, its simplicity and minimal internal data exchange enable scalability, facilitating easy parallelization and execution across a large number of CPUs. Lastly, ES is invariant to the sampling time of observations, meaning it remains effective regardless of how often actions are performed or rewards calculated. These strengths collectively contribute to ES's versatility, efficiency, and suitability for a wide range of optimization tasks.

2.3.2 Swarm Intelligence

Swarm intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviors of insects and other animals. Swarm intelligence comprises Ant Colony Optimization, Artificial Immune Systems Particle Swarm Optimization, Cuckoo Search, The Elephant Optimization, Bat Optimization, Grasshopper Optimization, and Honey Bee Optimization.

2.3.2.1 Ant colony optimization (ACO)

Ant colony optimization (ACO) is a population-based metaheuristic technique that can be used to find approximate solutions to difficult optimization problems [30, 31]. The ACO, inspired from the foraging behavior of ant species, is a swarm intelligence algorithm for solving hard combinatorial optimization problems. Marco Dorigo proposed ABO in 1992 in his PhD thesis [32], and it was initially used on the well-known Traveling Salesman Problem. The basic idea of the ACO is imitating the behavior of real ants searching for food. It was found that real ants are able to communicate information concerning food sources via an aromatic essence called pheromone. While they move along, real ants lay down pheromone in a quantity that depends on the quality of the food source discovered. Other ants, observing the pheromone trail, are attracted to follow it. Thus, the path will be enhanced and will therefore attract more ants. ACO algorithms have several strengths, such as parallel processing and adaptability, but they also face challenges related to convergence speed and parameter sensitivity. Opportunities lie in hybridization and exploring new application areas, while threats include competition from other metaheuristics and the complexity of implementation.

2.3.2.2 Artificial immune systems

Artificial immune systems (AIS) are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving [33]. AIS emerged in the mid-1980s with articles authored by Farmer [34]. The human immune system can be used as inspiration when developing algorithms to solve difficult computational problems. Artificial immune systems offer several advantages that enhance the reliability and effectiveness of the system for detecting and responding to threats. Firstly, the presence of a large number of detectors contributes to the permanency and reliability of the system by providing extensive coverage and detection capabilities. This ensures that malicious activities are more likely to be identified, enhancing overall security. Secondly, the absence of a single point of rejection strengthens the system's resilience against attacks, reducing the risk of system compromise. Thirdly, the distributed nature of the computing environment, with an increasing number of nodes, further bolsters security by distributing the workload and reducing vulnerability to targeted attacks. Additionally, storing all collisions of detectors with malicious objects in memory enables the training of detectors, facilitating continuous improvement and adaptation to evolving threats. However, AIS also present certain disadvantages that need to be addressed. Firstly, there is a risk of autoimmune reactions, where the system may mistakenly identify benign activities as malicious, potentially leading to unnecessary interventions or disruptions. Secondly, in a distributed computing environment with a small number of nodes, immunodeficiency becomes a concern, as the system may lack the necessary resources and redundancy to effectively detect and respond to threats. These drawbacks underscore the importance of carefully managing the system's behavior and resources to mitigate potential risks and optimize performance [35]. The biological immune system is a robust, complex, adaptive system that defends the body from foreign pathogens. It is able to categorize all cells (or molecules) within the body as self or non-self-substances [36]. It does this with the help of a distributed task force that has the intelligence to take action from a local and also a global perspective using its network of chemical messengers for communication. Artificial Immune Systems exhibit strengths in adaptability, robustness, and parallel processing. However, challenges include complexity, limited theoretical understanding, and sensitivity to parameters. Opportunities lie in hybridization and biomedical applications, while threats include competition from other algorithms and limited adoption due to complexity and ethical concerns.

2.3.2.3 Particle swarm optimization

Particle swarm optimization (PSO) is a population-based stochastic evolutionary computation technique algorithm for optimization, which is based on social–psychological principles, developed by Kennedy and Eberhart [1, 2] in 1995. PSO offers numerous advantages that make it a popular choice for optimization tasks. Firstly, its simple implementation and intuitive concept make it accessible to both beginners and experts. Secondly, as a gradient-free algorithm, PSO does not require gradient information, making it suitable for problems where gradients are challenging or expensive to compute. Thirdly, PSO possesses global search capability, allowing it to explore the solution space effectively by exchanging information among particles. Fourthly, its versatility enables application to various optimization problems, including continuous, discrete, and combinatorial domains. Additionally, PSO is inherently parallelizable, leading to faster convergence in certain cases, and it requires only a few parameters, simplifying parameter tuning.

However, PSO also presents several disadvantages. Firstly, it may converge to local optima instead of the global optimum, particularly in complex and multimodal problems. Secondly, its performance can be sensitive to parameter choices, posing challenges for optimal parameter selection. Thirdly, PSO lacks memory of past solutions, limiting its adaptability to dynamic environments. Fourthly, the algorithm lacks a solid theoretical foundation, making it difficult to analyze convergence properties rigorously. Fifthly, handling constraints in PSO can be challenging and may require additional mechanisms. Lastly, for large-scale problems, the computational cost of evaluating fitness functions for each particle can become prohibitive, impacting algorithm efficiency. While PSO offers several advantages, it also has limitations that need to be considered when selecting it for optimization tasks. The choice of optimization algorithm should be based on the specific characteristics of the problem at hand, weighing the advantages and disadvantages of each algorithm accordingly.

2.3.2.4 The Bees algorithm

The Bees algorithm, developed by Pham et al.[39] in 2005, is a population-based search algorithm which mimics the foraging strategy of honey bees to look for the best solution to an optimization problem. Each candidate solution is thought of as a food source (flower), and a population (colony) of n agents (bees) is used to search the solution space. Each time an artificial bee visits a flower (lands on a solution), it evaluates its profitability (fitness).

The Bee Algorithm presents several advantages stemming from its nature-inspired approach and design. Firstly, it draws inspiration from the foraging behavior of honeybees, replicating their communication and decision-making processes to inform effective optimization strategies. Secondly, the algorithm is adept at global optimization, efficiently exploring solution spaces to uncover global optima. Thirdly, its adaptability allows it to tackle a wide range of optimization problems, spanning both continuous and discrete domains, and accommodating various types of constraints. Fourthly, similar to other swarm intelligence algorithms, the Bee Algorithm can be parallelized, enabling distributed computing and potentially accelerating convergence. Lastly, its gradient-free nature renders it suitable for optimization tasks where gradients are challenging to compute, akin to Particle Swarm Optimization (PSO).

However, the Bee Algorithm also has its drawbacks. Firstly, it exhibits sensitivity to parameter choices, which can impact performance, necessitating careful parameter tuning. Secondly, its implementation and understanding can be complex due to the intricacies of honeybee communication and decision-making processes. Thirdly, like PSO, it may lack a solid theoretical foundation, hindering rigorous analysis of convergence properties and behavior. Fourthly, the algorithm may lack explicit memory mechanisms, affecting its adaptability to dynamic environments. Fifthly, its performance can be dependent on the initial population of solutions, with poor initialization potentially leading to suboptimal results. Lastly, scalability may pose a challenge for large-scale optimization problems due to computational costs. While the Bee Algorithm offers advantages such as global search capability and adaptability, its sensitivity to parameters and potential complexity in implementation are significant considerations. The suitability of the Bee Algorithm, like any optimization algorithm, depends on the specific characteristics of the problem being addressed.

2.3.2.5 Elephant herding optimization

Elephant herding optimization (EHO), proposed by Wang et al. in 2015, is a nature-inspired metaheuristic optimization algorithm based on the herding behavior of elephants [40]. EHO uses a clan operator to update the distance of the elephants in each clan with respect to the position of a matriarch elephant. The design of the elephant search algorithm is inspired by the habitual features of elephant herds, hence the name Elephant Search Algorithm, which separates the search agents into two gender groups representing the dual (local and global) search patterns. The Elephant Search Algorithm is distinguished by three key properties essential for an effective metaheuristic search algorithm. Firstly, it emphasizes the iterative refinement of solutions not only through regular update cycles but also through major reforms represented by the lifetimes of searching elephants. This dynamic approach allows for both incremental improvements and significant adjustments to the search process, enhancing the algorithm's adaptability and exploration capabilities. Secondly, the algorithm prioritizes intensive local searches guided by chief female elephants, recognizing that these efforts are more likely to yield optimal results due to their focused and informed direction. This targeted approach enables efficient exploitation of promising regions in the search space, maximizing the likelihood of finding high-quality solutions. Thirdly, male elephants serve as rangers, exploring distant areas of the search space to prevent the algorithm from getting trapped in local optima. By venturing into uncharted territory, these elephants facilitate exploration and diversification, guiding the entire elephant clan towards global optima. Together, these unique properties equip the Elephant Search Algorithm with the versatility, efficiency, and robustness necessary for effective optimization in diverse problem domains.

2.3.2.6 The Bat algorithm

The Bat algorithm, proposed and developed by Yang [41] in 2010, is a swarm-based metaheuristic algorithm for global optimization. It was inspired by the echolocation behavior of micro bats, with varying pulse rates of emission and loudness. All bats use echolocation to sense distance, and they also “know” the difference between food/prey and background barriers in some magical way. Bats fly randomly with certain velocity at given positions with a fixed frequency, varying wavelength and loudness to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission r in the range of [0, 1], depending on the proximity of their target. Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) to a minimum constant value. The Bat algorithm is praised for its strengths, including its ability to amalgamate features from various existing methods, its simplicity in both concept and structure, its effective exploitation capabilities, its maintenance of solution diversity within the population, and its rapid convergence facilitated by automatically focusing on promising solution areas. Additionally, it employs parameter control for iterative updates and serves as both a global and local optimizer. However, the algorithm also exhibits weaknesses, such as limited exploration capabilities, reliance on parameter tuning for optimal performance, the necessity for a refined strategy to balance exploration and exploitation, and the need for enhanced techniques to accelerate convergence for improved performance.

2.3.2.7 The Grasshopper optimization algorithm

The Grasshopper optimization algorithm (GOA), proposed by Saremi et al. [42] and developed by Mirjalili et al. [43], simulates locust swarm behavior in the wild. In GOA, the position of the locusts in the swarm represents a possible solution to a given optimization problem. Grasshoppers are insects well-known as a dangerous pest that affects and damages crop production and agriculture. Their life cycle includes two phases called nymph and adulthood. The nymph phase is characterized by small steps and slow movements, while the adulthood phase is characterized by long-range and abrupt movements. The movements of nymph and adulthood constitute the intensification and diversification phases of GOA.

In terms of advantages, GOA boasts simplicity in its design and implementation, rendering it accessible to a broad spectrum of users. Its flexibility allows it to tackle various optimization problems, spanning from continuous to discrete and mixed-integer domains. GOA has demonstrated efficiency in terms of convergence speed and solution quality for certain problem types. Additionally, the algorithm requires fewer parameters to be tuned, streamlining the optimization process. Moreover, its capability for global exploration of the search space enhances the likelihood of discovering optimal solutions.

However, despite its merits, GOA has notable drawbacks. It may encounter challenges in scaling up to handle very large-scale optimization problems, especially those characterized by high-dimensional search spaces or intricate constraints. Unlike some other algorithms, GOA lacks a robust theoretical foundation, which may limit its predictability and generalizability. Sensitivity to parameter settings is another concern, requiring careful calibration to achieve optimal performance. Additionally, the balance between exploration and exploitation in GOA may not always be optimal, potentially leading to premature convergence or insufficient exploration of the search space. Lastly, its performance may lack robustness across different problem domains or instances, making it less dependable in diverse scenarios.

2.3.2.8 Cuckoo search

Cuckoo search is an optimization algorithm developed by Yang and Deb [44]. They were inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of host birds of other species. Some host birds can engage direct conflict with the intruding cuckoos. For example, if a host bird discovers the eggs that are not their own, it will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere. Some cuckoo species, such as the New World brood-parasitic Tapera, have evolved in such a way that female parasitic cuckoos are often very specialized in the mimicry in colors and pattern of the eggs of a few chosen host species. Cuckoo search idealizes such breeding behavior, and thus can be applied for various optimization problems.

The cuckoo search algorithm (CSA) offers several advantages: it possesses global search capability, allowing it to explore the entire solution space effectively for global optimization; it maintains simplicity in its basic concept, making it accessible for implementation and application; it does not require gradient information, rendering it suitable for problems where gradients are difficult to compute; it naturally maintains population diversity, which aids in avoiding premature convergence and effectively exploring the search space; and it can be parallelized, facilitating distributed computing and potentially faster convergence, particularly for large-scale problems.

However, CSA also exhibits some disadvantages: it can be sensitive to parameters, necessitating careful tuning for optimal performance; it may lack a strong theoretical foundation, making rigorous analysis of its convergence properties and behavior challenging; and it lacks an explicit memory mechanism to remember past solutions, potentially limiting its ability to leverage historical information during the optimization process. This could affect its capacity to adjust to evolving environments or dynamic optimization challenges. CSA relies on Levy flights for exploration, and the algorithm's efficacy hinges on the precise execution of these flights. Inadequate implementation of Levy flights can hamper the algorithm's performance. Levy flights are a type of random walk characterized by occasional long jumps interspersed with shorter steps. They are named after French mathematician Paul Lévy [45], who first introduced them in the context of probability theory. Moreover, grappling with constraints poses a difficulty; integrating constraints into CSA can prove challenging. Ensuring the algorithm conforms to and respects optimization problem constraints might necessitate supplementary mechanisms. Limited applicability, CSA may not be universally applicable and may not outperform other optimization algorithms in all problem scenarios. Its effectiveness can depend on the characteristics of the problem. Cuckoo Search Algorithm has advantages such as global search capability and simplicity, but it also has limitations, including sensitivity to parameters and a potential lack of theoretical understanding. The choice of CSA depends on the specific optimization problem and the trade-offs between its strengths and weaknesses.

2.3.3 Neighborhood based Algorithm

A neighborhood-based algorithm is a type of algorithm used in machine learning and data analysis that relies on the idea of similarity or proximity among data points. The concept is particularly prevalent in collaborative filtering, a technique commonly used in recommendation systems. One challenge with neighborhood-based algorithms is the "cold start" problem, where it is difficult to make accurate recommendations for new users or items with limited interaction history. It's important to note that while neighborhood-based algorithms have been popular in recommendation systems, other approaches such as matrix factorization and deep learning-based methods have gained prominence due to their ability to handle larger datasets and capture complex patterns. Tabu Search and Scatter Search are among the neighborhood-based algorithms.

2.3.3.1 Tabu search

Tabu search (TS) is a "higher level" heuristic procedure for solving optimization problems, first introduced by Fred W. Glover [46] in 1985. Then Willard formalized the implementation in 1989 [47], designed to guide other methods (or their component processes) to escape the trap of local optimality. Tabu Search is widely used in solving combinatorial optimization problems, scheduling problems, and other complex optimization tasks. Its effectiveness lies in its ability to balance exploration and exploitation through the use of the tabu list and aspiration criteria. However, successful implementation often requires careful parameter tuning and adaptation to the characteristics of the specific optimization problem.

The TS algorithm presents numerous advantages that make it a versatile choice for optimization problems. First and foremost, its capability for global optimization ensures it can effectively locate global optima, making it applicable to a diverse array of optimization challenges. Moreover, its flexibility allows it to handle various problem types, spanning combinatorial, continuous, and discrete optimization domains. Unlike some other optimization methods, TS operates without the need for derivative information, making it well-suited for scenarios where obtaining such information is impractical or costly. The inclusion of a memory mechanism, particularly the tabu list, prevents the algorithm from revisiting previously explored solutions, thereby fostering solution diversity and preventing repetitive cycles. Additionally, its relatively simple implementation compared to other metaheuristic algorithms renders it accessible to practitioners with varying levels of expertise. TS also demonstrates proficiency in handling constraint-laden optimization problems, thanks to the tabu list's ability to steer clear of infeasible solutions. However, TS does come with its set of drawbacks. One significant challenge lies in the sensitivity of its parameters, where suboptimal choices can greatly impact its performance. Determining an appropriate tabu tenure, in particular, can be a complex and problem-specific endeavor. The size of the tabu list represents another critical parameter; setting it too small may lead to premature solution revisitations, while setting it too large may hinder exploration efforts. Moreover, the computational intensity required for maintaining and updating the tabu list can pose efficiency concerns, depending on the complexity of the problem. Like many optimization algorithms, TS is susceptible to getting trapped in local optima, particularly in intricate and multimodal optimization scenarios. Adapting TS for Multi-Objective Optimization (MOO) also presents challenges, as balancing conflicting objectives demands careful consideration. While TS offers compelling advantages such as global optimization capability and adaptability to various problem types, its effectiveness hinges on meticulous parameter tuning and careful consideration of the specific characteristics of the problem. Nonetheless, with proper parameterization and problem understanding, TS remains a powerful tool for tackling optimization challenges across diverse domains.

2.3.3.2 Scatter search

Scatter search (SS), introduced by Glover [48] in 1977, is an evolutionary approach for optimization. It has been applied to problems with continuous and discrete variables and with one or multiple objectives [49]. SS consists of five methods: (1) Diversification generation, (2) Improvement, (3) Reference set update, (4) Subset generation, and (5) Solution combination [50]. The SS algorithm improves other algorithms further by maintaining a balance between intensification and diversification during search [51]. Scatter search has shown merit in applications where the optimization horizon (represented by a number of objective function evaluations) is severely limited. The Scatter Search (SS) algorithm offers several advantages that make it a versatile choice for optimization problems. Firstly, it is designed for global optimization, making it suitable for a wide range of optimization challenges. Secondly, the algorithm emphasizes diversity preservation, ensuring that a variety of solutions are generated and maintained to prevent premature convergence and thoroughly explore the solution space. Additionally, SS incorporates both intensification and diversification strategies, striking a balance between exploiting promising regions and exploring the entire solution space. Its adaptability allows customization for different types of optimization problems, including combinatorial, continuous, and mixed-variable problems. Furthermore, SS uses a memory mechanism to store promising solutions and historical information, aiding in decision-making during the search process. It can also be extended for multi-objective optimization problems, addressing trade-offs between conflicting objectives.

However, SS also has its disadvantages. Firstly, it can be computationally intensive, especially for large solution spaces, as generating diverse solutions and combining them may require significant computational resources. Secondly, like many optimization algorithms, SS is sensitive to parameter settings, necessitating proper tuning for optimal performance. Its implementation can also be more complex compared to simpler optimization algorithms, requiring a deeper understanding of the algorithm. Moreover, its effectiveness may decrease for very large-scale optimization problems due to increased computational costs and the need for managing a large set of solutions. Additionally, SS may not be suitable for dynamic optimization problems where the problem characteristics change over time, as adapting to dynamic environments can be challenging. Lastly, incorporating constraints into SS can be difficult, requiring additional mechanisms to ensure adherence to constraints in the optimization problem. While Scatter Search offers advantages such as global optimization, diversity preservation, and adaptability, its computational intensity and sensitivity to parameters should be carefully considered. The suitability of SS depends on the specific characteristics of the optimization problem and the trade-offs between its strengths and weaknesses.

2.3.3.3 Simulated annealing

Simulated annealing (SA), introduced by Kirkpatrick et al. in 1983 inspired by the annealing procedure of metal working [52], is a meta-heuristic technique for optimization that consists of a probabilistic local search technique, and is based on an analogy with thermodynamics. SA provides a clear and simple approach of finding near optimal solutions of difficult combinatorial optimizations where there are many local minima. Although not fast in absolute terms it is still comparatively fast in relation to other approaches to combinatorial optimization, and this speed can be improved though careful parallelization [53]. Simulated Annealing (SA) offers several advantages that make it an attractive option for optimization problems. Firstly, it is renowned for its ease of implementation and utilization, making it accessible even to those without extensive optimization expertise. Additionally, SA has been proven effective in providing optimal solutions to a diverse array of problems, showcasing its versatility and reliability.

However, SA does come with its set of disadvantages. One notable drawback is its potential for lengthy execution times, particularly if the annealing schedule is set to run for an extended duration. This can result in prolonged optimization processes, which may not be feasible for time-sensitive applications or large-scale problems. Furthermore, SA involves several tunable parameters, including the annealing schedule itself, which can require meticulous parameter tuning for optimal performance. Managing these parameters effectively can be challenging and may require a significant investment of time and computational resources [54]. While Simulated Annealing offers advantages such as ease of implementation and the ability to provide optimal solutions across various problem domains, its potential for long runtimes and the presence of numerous tunable parameters are important considerations when utilizing this algorithm. These factors should be carefully weighed against the specific requirements and constraints of the optimization problem at hand.

2.4 Multi-objective optimization

Multi-objective optimization (MOO) is a mathematical and computational approach used to solve problems with multiple conflicting objectives. In many real-world scenarios, decision-makers face situations where they need to optimize more than one criterion simultaneously, and these objectives may be conflicting or mutually exclusive. MOO addresses the challenge of finding a set of solutions that represents a trade-off among these objectives, known as the Pareto front or Pareto set. Equation (1) defines the general form of an MOO problem [55, 56]:

$$\left.\begin{array}{lll} \min/\max & f_k(\mathbf{x}), & k = 1,2,3,\dots,K; \\ \text{s.t.} & g_j(\mathbf{x}) \geq 0, & j = 1,2,3,\dots,J; \\ & h_p(\mathbf{x}) = 0, & p = 1,2,3,\dots,H; \\ & x_i^{(L)} \leq x_i \leq x_i^{(U)}, & i = 1,2,3,\dots,n. \end{array}\right\} \tag{1}$$

where a solution $\mathbf{x}$ is a vector of n decision variables: $\mathbf{x} = (x_1, x_2, x_3, \dots, x_n)^T$. K is the number of objective functions; J and H represent the number of inequalities and equalities constraints, respectively. $x_i^{(L)}$ and $x_i^{(U)}$ are the lower bound and upper bound of the decision variable x_i. The set of lower and upper bounds defines the decision variables space and for MOO the objective space is a multi-dimensional space as shown in Figure 2 [3].

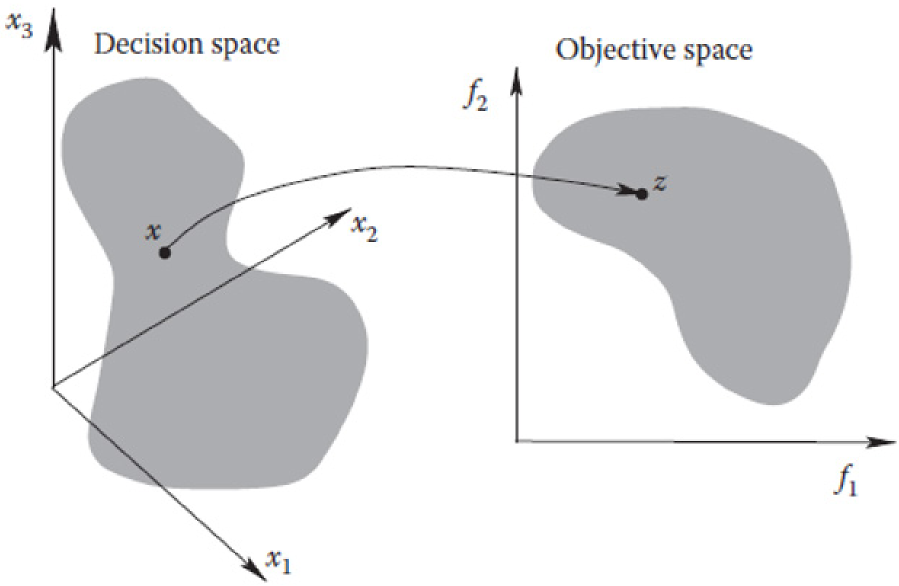


Figure 2: Tree-dimensional decision variable space and two-dimensional objective space

The key concepts in MOO comprise Objective functions, which involves optimizing two or more objective functions that capture different aspects of the problem. These objectives may have conflicting requirements, and optimizing one may compromise the others. Pareto front, the solutions that represent the best trade-offs among conflicting objectives are found on the objective space. A solution is Pareto optimal if there is no other solution that improves one objective without degrading another. Decision variables are the parameters or variables that decision-makers can adjust to find optimal solutions. The values of these variables determine the performance with respect to the objective functions. MOO problems often involve constraints or limitations that solutions must satisfy. Feasible solutions are those that meet these constraints. The primary goal of MOO is to provide decision-makers with a set of solutions that represent trade-offs among conflicting objectives. The decision-maker can then choose a solution based on their preferences and priorities. Various optimization algorithms are employed for solving multi-objective problems. Evolutionary algorithms, such as the Non-dominated Sorting Genetic Algorithm II (NSGA-II), are popular choices due to their ability to explore the Pareto front efficiently. Hypervolume Indicator is a metric used to evaluate the quality of a set of Pareto optimal solutions. It measures the volume of the objective space covered by the solutions, providing insights into the diversity and distribution of the Pareto front.

MOO is crucial for addressing the complexities of real-world problems by providing decision-makers with efficient, balanced, and comprehensive solutions. Its application spans across various domains, contributing to improved decision-making processes, resource allocation, risk management, and sustainability.

The resolution of a Multi-Objective Optimization (MOO) problem can be categorized into two main approaches: the Pareto method and scalarization [57]. These methods differ in their approach to handling desired solutions and performance indicators. The Pareto method is employed when distinct solutions and performance indicators exist, leading to a compromise solution (tradeoff) that can be visualized through a Pareto optimal front (POF). In contrast, the scalarization method involves incorporating performance indicators as components in a scalar function, which is then integrated into the fitness function.

The Pareto method, specifically in the context of solving Multi-Objective Optimization (MOO) problems, is based on the principles of Pareto dominance. Named after Vilfredo Pareto, this approach is used to identify and characterize the set of Pareto-optimal solutions in a multi-objective optimization problem. Pareto optimality represents a state where no solution in the set can be improved in one objective without compromising another [58, 59]. The Pareto method is particularly powerful for problems where there is no clear single optimal solution, and a set of trade-off solutions is preferred. It allows decision-makers to explore the trade-off space and make informed decisions based on their preferences and priorities for each objective.

The Scalarization Method is an approach used in Multi-Objective Optimization (MOO) to convert multiple conflicting objectives into a single scalar function. This scalar function, often referred to as an aggregated or weighted sum, allows the optimization algorithm to treat the multi-objective problem as a single-objective optimization problem. The Scalarization Method is particularly useful when the decision-maker has a clear preference for the importance of each objective. However, it may oversimplify the trade-off relationships between objectives and may not capture the full Pareto front in the solution space. The choice of weights can significantly influence the results, and sensitivity analysis is often conducted to assess the robustness of the solution concerning different weight configurations.

2.5 Applications of MOO

Multi-objective optimization (MOO) finds applications in various fields where decision-makers need to consider and balance multiple conflicting objectives. Some notable applications of multi-objective optimization include Urban planning, Evacuation planning and Renewable energies. In Urban planning, MOO aids in optimizing land use allocation, transportation networks, and infrastructure planning in urban areas, considering objectives such as economic development, environmental sustainability, and social equity. In Evacuation planning, MOO plays a crucial role in enhancing the effectiveness and efficiency of evacuation planning, especially in scenarios where conflicting objectives need to be considered. In Renewable energies, MOO is widely applied in determining optimal locations for renewable energy projects, balancing various objectives to achieve sustainable and efficient solutions. MOO is widely used in engineering for designing systems and products that need to satisfy multiple performance criteria, such as maximizing efficiency while minimizing cost and environmental impact. In financial portfolio optimization, investors often have conflicting goals like maximizing returns while minimizing risks [60, 61]. MOO helps in finding portfolios that strike a balance between these objectives. Balancing conflicting objectives like minimizing costs, maximizing efficiency, and reducing environmental impact is crucial in optimizing supply chain operations, and MOO can assist in achieving these goals [62]. In designing and operating energy systems, MOO can help optimize the trade-offs between cost, environmental impact, and energy efficiency [63, 64]. MOO is employed to find solutions that optimize resource allocation and conservation strategies while minimizing ecological impact in environmental management [65, 66]. In aircraft design, objectives such as fuel efficiency, speed, and safety can be conflicting [67, 68]. MOO helps in finding optimal designs that balance these objectives. In managing water resources, MOO can be used to optimize dam operations, water distribution, and irrigation strategies while considering conflicting objectives like maximizing water availability and minimizing environmental impact [65, 66]. MOO is applied in drug formulation, treatment planning, and medical device design, considering multiple conflicting objectives like efficacy, safety, and cost [69, 70]. Optimizing manufacturing processes involves balancing objectives like production speed, quality, and cost, and MOO aids in finding optimal solutions

[71, 72]. In network design and optimization, MOO helps in achieving trade-offs between factors such as data transfer speed, network reliability, and cost [73, 74]. MOO is utilized in robotics for motion planning, trajectory optimization, and robot design, considering conflicting objectives such as energy efficiency and task completion time [75, 76].

2.5.1 Evacuation Planning

Evacuation is an important disaster management tool. According to Bish [77], the process of evacuation planning consists of three phases: 1) determination of the safe areas, 2) selecting the optimum path between risk zones and safe areas, and 3) joining risk zones associated with each safe area. The current study focuses on the second and third phase of evacuation planning. Several researchers have worked on evacuation planning issues and proposed various multi-objective optimization approaches to handle the problem. For instance, research by Baou [78] triggered a study about evacuation planning in earthquake disasters considering two conflicting objective functions, using Remote Sensing and Geographical Information System (GIS). Their study used a multi-objective technique to solve the problem and search for the optimal distribution of people from risk zones to safe areas. Saadatseresht [79] carried out similar researches, in Iran, using multi-objective evolutionary algorithms, with two objective functions, in conjunction with GIS to minimize evacuation costs from risk zones to safe areas.

On the other hand, Stepanov and Smith [80] solved the transportation networks for evacuation routing of production goods using multi-objective functions based on integer programming for the best route assignment. The model consists of the minimization of the total travel distance and the excess clearance time. The authors also evaluated the performance measures of the evacuation plan such as clearance time, the total distance, and blocking probabilities.

Coutinho-Rodrigues [81] introduced, in Portugal Coimbra, a multi-objective approach to identify evacuation paths and the location of shelters for urban evacuation planning. To address the evacuation-planning problem, six conflicting objective functions, in a mixed-integer linear programming model, were considered. These objective functions comprised the minimization of total travel distance for people to reach their shelter, the minimization of the total risk of the evacuation paths, the minimization of total travel distance associated with backup paths, the minimization of total risk at the shelters (i.e., risks associated with the shelter site), the minimization of the total time required to transfer people from their evacuation shelter to an external hospital when necessary, and the minimization of the total number of shelters.

In Chaoyang District, Beijing, China, Zhao et al. [82] carried out a scenario-based model using a multi-objective optimum allocation for earthquake emergency shelters based on a modified Particle Swarm Optimization algorithm. This model minimized the total weighted evacuation time from residential areas to a specified shelter, and also reduced the total area of all the shelters. The model was demonstrated to be convenient for the optimization of shelter allocation.

Ikeda [83] developed an evacuation route planning for a safety route guidance system after a natural disaster using a multi-objective genetic algorithm. The proposed system has three objective functions, which are: evacuation distance, evacuation time, and safety of evacuation route. Gai [84] built up a model for assessing the risks associated with the evacuation process in response to potential chemical accidents, where a multi-objective evacuation routing model for toxic cloud releases is proposed, taking into account that the travel speed in each path will be affected by disaster extension. The developed model minimizes travel time and individual evacuation risk along a path.

Ghasemi [85] carried out an uncertain multi-objective multi-commodity multi-period multi-vehicle location-allocation mixed-integer mathematical programing model and proposed it for the response phase of earthquakes. The proposed model includes five echelons, namely affected areas, distribution centers, hospitals, temporary accommodation centers, and temporary care centers. Two objective functions minimizing the total cost of the location-allocation of facilities and minimizing the amount of the shortage of relief supplies are considered. The uncertainty is modeled using a scenario-based probability approach. Niyomubyeyi [86] developed a model for evacuation planning assessed in Kigali, Rwanda. The model determined the minimum optimal distribution of evacuees to shelters using the multi-objective Artificial Bee Colony algorithm.

Although several studies have been carried out on evacuation planning using multi-objective optimization, assessed with different population-based metaheuristics, so far, none of them has solved the evacuation problem using the multi-objective cuckoo search algorithm. Furthermore, the challenges of evacuation management increase substantially with the size of the risk area and population. Thus, the development of more practical approaches is still strongly needed to address the emergency evacuation concerns. In practice, shelter location-allocation planning and the risk management on evacuation route are always entangled with one another. Therefore, this study proposes a multi-objective optimization approach that selects the optimum path from risk zone to safe area and determines appropriate allocation of shelters to the population in danger.

2.5.2 Urban Planning

Urban planning is a technical and political process that focuses on the development and design of land use and the built environment, such as transportation, infrastructures, green spaces, and accessibility. The lack of urban planning affects the transportation system, infrastructure, layout, prescribed density of residences, commerce, and industrial areas [87]. Multi-objective optimization has been commonly used in urban planning. For instance, Deb et. al. [87] developed a Geographical Information System (GIS) based on MOEA and applied it to the Mediterranean landscape of Southern Portugal for land-use management, with three objective functions: maximization of economic return, maximization of carbon sequestration, and minimization of soil erosion. In Baboldasht, a district of Isfahan in Iran, Sahebgharani [88] developed a novel meta-heuristic algorithm named parallel particle swarm to allocate seven land types (residential, commercial, cultural, educational, medical, sportive, and green space) in order to maximize compactness, compatibility, and suitability objective functions. Compared with the results performed by GA in other studies, it was found that both the quality and convergence time of the parallel particle swarm optimization are better than GA. Liu and Xi [89] investigated multi-objective optimization of the spatial structure and layout of a protected area using the NSGA-II algorithm and four objective functions maximizing the value of ecosystem services: provisioning, supporting, regulating, and cultural. García et al. [90] proposed a multi-objective optimization model for sustainable land use allocation in the Plains of San Juan, Puebla, Mexico, and searched for the optimal solution using NSGA-II. Zhao et al. [91] developed a gray multi-objective dynamic programming (GMDP) model and the ant colony optimization (ACO) algorithm for land-use optimization in Lancang County, China. The maximization of social, economic, and ecological benefits is used as the optimization objective in the model. Caparros and Dawson [92] developed a spatial optimization framework to optimize the location of future residential development against several sustainability objectives, and conducted a case study of Middlesbrough in the northeast United Kingdom. Shifa et al. [93] carried out a study to optimize the allocation of land resources, including the optimization of quantity and space, to bring forward the land-use space optimization model based on the particle swarm evolutionary algorithm. The results showed that the model could analyze the data of multi-dimensional discrete decision space with good space search features and high accuracy in parallel.

Overall, various multi-objective optimization methods are applied in urban planning, such as GA, particle swarm optimization, simulated annealing, ant colony optimization, etc. Each of these optimization methods presents a set of reasons [94] behind its use. On the one hand, such reasons aggregate robustness and efficiency, intelligent ranking of the Pareto solutions, and less computational time. On the other hand, they guarantee an optimal solution, better performance in spatial data, and low computational cost.

2.5.3 Optimal location for renewable energy

Transitioning to renewable energy is inevitable in order to meet the world's growing energy demands sustainably, mitigate climate change, and address environmental and health concerns. Many studies and technologies are developed to address the sustainability of renewable energies, from different perspectives, and among them studies which have been based on MOO. Holloway et al. [95] underscored the detrimental impacts of coal and gas energy production on the environment, highlighting the pressing need for Australia to transit towards renewable energy sources to combat climate change. Their study focuses on identifying optimal locations for deploying a distributed hybrid renewable energy generation system in rural regions of Western Australia. The researchers employed a data mining approach, utilizing K-Means and K-Medoids clustering algorithms to partition the dataset into clusters. From an initial dataset of 69 locations, they proceeded with a filtering process to refine the selection. Visual representations of the cluster data were then mapped onto Western Australia to aid in decision-making. Evaluation of the clustering algorithms was conducted using the Dunn index, revealing that K-Means outperformed K-Medoids given the nature of the dataset. Subsequently, the researchers utilized HOMER software to assess the potential wind and solar energy output for each cluster centroid. Interestingly, while K-Medoids identified locations with higher average solar and wind energy potential, its reduced internal validation and inability to cluster data points effectively raised concerns about its overall utility. Aisaba et al. [96] addressed the challenge posed by the intermittent and uncertain nature of wind turbines and solar photovoltaic (PV) systems in the context of grid planning operations. Their study introduced a novel multi-objective distributional robust optimization model aimed at determining optimal locations for these renewable energy sources. The objective is twofold: to minimize the variance of renewable energy sources while maximizing power production. They evaluated the performance of different forecasting models—Autoregressive Moving Average (ARMA), Deep Learning Gated Recurrent unit (GRU), and Deep Learning Long Short-Term Memory (LSTM)—for predicting wind speed and solar irradiation. They compare the root mean square errors (RMSE) of these models to gauge their accuracy. Using the forecasting error information, uncertain variables are characterized within an ambiguity set, which incorporates bounds, means, and covariance values. Additionally, the authors propose a modified multi-objective non-dominated sorting genetic algorithm (NSGA-II) to obtain a tractable Pareto front solution. To validate the effectiveness of their model, actual candidate sites for wind turbines and solar PV systems in Saudi Arabia are utilized. The results indicated that the proposed model offers an attractive and less conservative solution compared to a multi-objective robust optimization model when accounting for forecasting uncertainties.

Durmaz et al. [97] addressed a practical problem in the field of biomass supply chain management and demonstrated the development of a systematic approach to optimize the design and planning of such networks, with a specific focus on the poultry industry in Turkey. They used integration of GIS, Analytic Hierarchy Processes, and multi-objective mixed integer linear programming optimization to achieve the aim, which is to determine the optimal number, locations, and sizes of biogas facilities, as well as the network flow and electricity generation potential. The model considers two primary objectives: maximizing profit and minimizing the total distance between poultry farms and biogas facilities. To assess the impact of various parameters on the outcomes of the model, they performed a sensitivity analysis and this analysis revealed that both maximum distance parameters and purchasing prices significantly influence decision-making and financial performance.

Hai et al. [98] proposed a hybrid electricity-generation system that combined solid oxide fuel cells (SOFC), biomass gasification, and wind energy to enhance power generation efficiency and reduce environmental impact compared to traditional biomass-driven SOFC systems [99]. The integration of renewable energy sources and the production of pure hydrogen played a key role in achieving these objectives. In addition, it demonstrated the potential of such system to contribute to cleaner and more sustainable electricity generation. Delgarm et al. [100] presented a methodology that combines simulation-based optimization with building energy performance analysis to achieve better energy-efficient building designs using of a mono- and multi-objective particle swarm optimization (MOPSO) algorithm. The building design offered valuable insights for architects and engineers working on building projects in various climatic regions, helping them make informed decisions to reduce energy consumption and improve overall performance [101, 102]. This included minimizing annual cooling, heating, and lighting electricity consumption, as well as the total annual building electricity demand.

Deveci and Guler [103] presented a comprehensive approach to renewable energy planning in Turkey, considering multiple objective functions that consisted of minimization of the levelized cost of electricity generation and maximization of short-term electricity generation from renewable energy resources and factors. It offers recommendations for optimizing renewable energy investments, which can be instrumental in guiding energy policy decisions and enhancing the country's transition to a more sustainable and cost-effective energy mix. A Competitive Multi-Objective Particle Swarm Optimizer (CMOPSO), which is a state-of-the-art metaheuristic optimization algorithm, was used for the study.

In another study [104] GIS and AHP were used to identify optimal locations for a solar photovoltaic (PV) power plant in the Malatya Province, Turkey. Various factors, such as solar energy potential, infrastructure, topography, and environmental considerations, were considered for optimum site selection. The study presented a map highlighting the most suitable sites for solar energy plants, providing decision-makers with an empirical basis for optimal site selection in comparison to existing solar PV power plants.

Another study for solar power farm site selection was carry out by Uyan [101, 102] in Turkey and he took in consideration factors like terrain quality, local weather conditions, proximity to transmission lines, agricultural facilities, and environmental conservation. The study employed integrated GIS and AHP to determine a final index model categorized into four classes ("low suitable," "moderate," "suitable," and "best suitable"), using an equal interval classification method. Mapping the indexes decision-makers could visualize the suitability map for creating solar farms.

The input data used in these studies range from environmental and topographic variables as well as resource availability (solar irradiance, wind speed) to equipment costs, maintenance requirements, and policy-related variables. Objective functions in optimization models often include minimizing the overall cost, maximizing energy output, or achieving a balance between economic and environmental considerations. Some studies incorporate multi-objective optimization to address conflicting objectives.

The present thesis will utilize distinct datasets comprising geographical information about the study area, data on renewable resources such as solar and wind patterns, economic parameters encompassing cost data and investment incentives, as well as environmental data including emission factors.

3 Methodology

This chapter summarizes software libraries, techniques, algorithms, case studies, and evaluation mechanisms used for the implementation of this PhD thesis.

3.1 Software libraries

Distributed Evolutionary Algorithms in Python (DEAP) [106] framework and a Python library for multi-objective optimization (PYMOO) [107] were the major libraries used for the implementation of the algorithms in different subprojects of the thesis. DEAP is a popular and flexible open-source library for building and experimenting with evolutionary algorithms. DEAP provides tools for implementing various types of evolutionary algorithms, including genetic algorithms, genetic programming, and other evolutionary strategies. Multi-objective optimization involves optimizing multiple conflicting objectives simultaneously. PYMOO provides a set of tools and algorithms specifically designed for solving problems with multiple objectives.

To evaluate the quality of the optimal Pareto front the Hypervolume indicator is usually used [108]. The Hypervolume is a metric commonly used in multi-objective optimization to assess the quality of a set of solutions. It provides a measure of how well a set of solutions covers the objective space. Specifically, it quantifies the volume of the objective space that is dominated by a given set of solutions. We used an R software package called Hypervolume [109] and a Python package denominated Pygmo [110] to compute the Hypervolume measure. Pygmo (Python Global Multi-objective Optimizer) is an open-source Python library designed for solving optimization problems, particularly those involving multiple objectives. Pygmo is built on top of the Pagmo (Parallel Global Multi-objective Optimizer) C++ library. It provides a convenient interface for defining and solving optimization problems with multiple objectives, constraints, and decision variables. The R Hypervolume package assesses the configuration and size of datasets in high-dimensional spaces, enabling various set operations such as intersection, union, identification of unique components, inclusion testing, and hole detection. It employs a stochastic geometry methodology for high-dimensional kernel density estimation, delineation through support vector machines, and generation of convex hulls. This approach finds applications in modeling Hypervolumes for traits and niches, as well as in species distribution modeling. The Running metric was employed to assess the convergence of the algorithm [111]. Repeatability analysis [112] was used assessing the consistency and reliability of optimal Pareto front set and the goal was to determine how much variability exists in the outcomes and whether the observed results are consistent across different runs or trials. To understand the stability and consistency of the optimal results Variability analysis [113] was used, which involved examining the degree of variation or dispersion in a set of optimal Pareto fronts.

For finding the shortest paths between risk zones and shelters Dijkstra's algorithm from the ArcGIS Network Analyst extension was used, that is, a popular algorithm in computer science and graph theory [114].

In this thesis, spatial analysis is widely used across in the case studies, related to urban planning, evacuation planning, and solar power farm site selection. The design objectives, study area, data set, frameworks, and methods used for each case study (paper) are detailed in Table 1.

Table 1: Overview of research, data, and methodology for the four papers

	Paper I	Paper II	Paper III	Paper IV
Research Applications	Evacuation planning	Evacuation planning	Land use allocation	Solar power farms
Spatial Objectives	Total traveling distance, Total shelters overload	Total traveling distance, Total shelters overload, Total risk in the evacuation routes	Total economic income, Total carbon emission, accessibility, space syntax integration, and compactness	Total solar radiation, Total distance to grid, Total distance to street, Total distance to city, Total slope, and Total aspect
Study Area	Kigali, Rwanda	Maputo City, Mozambique	KaMavota district, Maputo City, Mozambique	KaMavota district, Maputo City, Mozambique
Methods	Network Analysis and Four metaheuristics: AMOSA, MOABC, NSGA-II, MSPSO	Network Analysis and MOCSA	Network Analysis, Space syntax Analysis, Raster Analysis, NSGA-III	Network Analysis, Space syntax Analysis, Raster Analysis, NSGA-II
Datasets	Road networks, location of shelters, Location of risk zones, Population data, DEM	Road networks, location of shelters, Location of risk zones, Population data, DEM	Road networks, Land cover, DEM, Average Carbon emission, Average value of Land use	Road networks, Power lines, Land cover, DEM, building map.
Strategies for reliability	Documentation of search strategy and results	Detailed documentation of research process	Documentation of data sources And calculations	Documentation of data sources and calculations
Strategies for internal Validity	Discussion with supervisors and academic peers	Collection of feedback during tool development	Discussions with co-authors On interpretation of results	Discussions with coauthors On interpretation of results
Contributions	Contribute to the understanding of suitable optimization methods for addressing complex spatial problems in disaster management.	Contribute to the field of disaster management and emergency response by offering a more efficient approach to resource allocation during evacuation events.	Contributions by addressing the complexity of LUA through a novel MOO approach, utilizing an improved NSGA-III algorithm.	Contributes valuable insights for decision-makers involved in the global transition to renewable energies, offering a systematic approach to optimize land use for solar farms.
Research questions	How do the multi-objective versions of classical metaheuristic algorithms (AMOSA, MOABC, NSGA-II, and MSPSO) perform in an urban evacuation problem in Rwanda, and how do they compare in terms of effectiveness, efficiency, repeatability, and computational time? Additionally, what modifications to the MOABC algorithm could enhance its effectiveness for evacuation planning?	How can a multi-objective optimization model, specifically utilizing Improved Multi-Objective Cuckoo Search (IMOCS), enhance the efficiency of emergency evacuation planning by minimizing total travel distance, mitigating risk along evacuation routes, and preventing shelter overload in comparison to the standard Multi-Objective Cuckoo Search (MOCS)?	How can a multi-objective optimization (MOO) approach, specifically utilizing an improved Non-dominated Sorting Genetic Algorithm III (NSGA-III), contribute to addressing the complex challenges of Land Use Allocation (LUA) in urban planning, with a focus on achieving maximum economy, minimum carbon emissions, maximum accessibility, maximum integration, and maximum compactness as optimal objectives?	How can a multi-objective optimization (MOO) model for land use selection in solar farms contribute to the global transition to renewable energies, and what is the efficiency and quality of solutions achieved through the adaptation and optimization of the NSGA-II algorithm?
Year	2020	2022	2024	2024

3.2 Cases studies

In this subsection, we provide three case studies applied to urban evacuation planning, urban land-use allocation, and site selection for renewable energies. Urban evacuation planning is a crucial aspect of emergency management aimed at efficiently and safely moving people from areas at risk to safer locations during various types of emergencies or disasters. This planning process involves the coordination of resources, infrastructure, and communication strategies to ensure the orderly and effective evacuation of residents from urban areas. Urban land use allocation is a critical aspect of urban planning that involves the systematic allocation of different types of land uses within an urban area to achieve specific planning objectives. This process aims to optimize the use of available land while addressing the social, economic, and environmental needs of the community. Site selection for renewable energy projects is a crucial step in optimizing energy production, minimizing environmental impact, and ensuring the overall success of the project. Different types of renewable energy sources have distinct requirements and considerations for site selection. Below, we describe the used data set, methodology, and the aims for each research.

3.2.1 Urban Evacuation Planning (Paper 1 and Paper 2)

The studies focus on urban evacuation planning in Kigali, Rwanda, and Maputo, Mozambique, two sub-Saharan African capitals facing increasing challenges from extreme weather events and rapid urbanization. To address the pressing need for effective urban evacuation planning, the study leverages spatial data, population information, and advanced algorithms to develop comprehensive evacuation plans.

The methodology begins by utilizing spatial data provided by city authorities, including road networks, administrative boundaries, and bridge locations. Population data sourced from national statistics agencies informs the selection of suitable shelters adhering to global standards outlined in The Sphere Project [115]. Additionally, a digital elevation model (DEM) aids in slope analysis for optimal shelter placement, considering topography and infrastructure.

Evacuation routes are determined using Dijkstra's algorithm on road networks extracted from Open Street Map[116]. This algorithm calculates the shortest path and generates a distance matrix, crucial for computing total travel distance. A Risk function assesses overall risk, utilizing datasets on roads, bridges, shelters, and residential areas to estimate risk along evacuation paths.

The studies propose a multi-objective optimization model for developing efficient evacuation plans, aiming to minimize total travel distance, reduce risk on evacuation routes, and prevent shelter overload. Evaluation of the approach compares the performance of Improved Multi-Objective Cuckoo Search (IMOCS) against standard Multi-Objective Cuckoo Search (MOCS), with IMOCS generally outperforming MOCS in execution time. Evaluation metrics such as the Hypervolume indicator and convergence assessment are employed to assess the quality of the Pareto front and algorithm performance [117].

Moreover, four classical metaheuristic algorithms (AMOSA, MOABC, NSGA-II, and MSPSO) are evaluated for their performance in solving urban evacuation problems in Rwanda, with results indicating AMOSA and MOABC achieving good quality solutions. NSGA-II exhibits faster execution time and convergence speed, while AMOSA, MOABC, and MSPSO demonstrate higher repeatability compared to NSGA-II. The study suggests that further modifications to MOABC could enhance its effectiveness for evacuation planning [118].

Overall, the methodology provides a comprehensive approach to address the challenges of urban evacuation planning through multi-objective optimization and the evaluation of various metaheuristic algorithms. It underscores the importance of leveraging spatial data, population information, and advanced algorithms to develop effective urban evacuation plans, aiming to enhance the safety and resilience of cities in the face of extreme weather events and rapid urbanization.

Below, we describe each objective function used in papers 1 and 2 to model the problem of evacuations planning.

3.2.1.1 Total travelling distance

In the realm of evacuation planning, minimizing total traveling distance is of paramount importance due to several critical reasons. Evacuation scenarios typically involve the rapid movement of large numbers of people away from a danger zone, such as during natural disasters like hurricanes, floods, or earthquakes, or in response to human-made crises like terrorist attacks or industrial accidents. Minimizing total traveling distance in evacuation planning is essential for optimizing the efficiency, safety, and effectiveness of evacuation operations. By reducing travel time, alleviating congestion, minimizing exposure to hazards, optimizing resource allocation, and enhancing accessibility, planners can help to ensure that evacuees reach safety swiftly and securely, saving lives and mitigating the impact of disasters on affected communities.

In Equation (2) [117, 118], higher population building blocks are given precedence for allocation to the closest safe area, ensuring that more individuals can reach safety in the shortest timeframe possible. Here, d_{ij} represents the distance along the optimal path between the ith building block and the jth safe area. P_{ij} represents the population of the ith building block that is designated to be evacuated to the jth safe area. m denotes the count of building blocks slated for evacuation to the safe area, while n signifies the number of shelters designated to accommodate the evacuees.

$$f_1 = \sum_j^n \sum_i^m d_{ij} \cdot P_{ij} \tag{2}$$

3.2.1.2 Total shelter overload

In the realm of evacuation planning, minimizing total shelter overload plays a crucial role in ensuring the safety and well-being of evacuees during emergencies. Minimizing total shelter overload in evacuation planning is essential for promoting the safety, health, well-being, and resilience of evacuees during emergencies. By optimizing resource allocation, maintaining hygienic conditions, supporting psychological well-being, enhancing accessibility and comfort, and fostering resilience and sustainability, planners can help ensure that shelters remain effective and supportive environments for individuals seeking refuge during times of crisis.

Equation (3) [117, 118] represents the total population allocated to each safe area, ensuring it does not exceed its capacity. The absolute sign for f_2 determines if the total population of evacuees surpasses the safe areas' combined capacity. In such cases, the excess should be distributed among the safe areas while minimizing the overload capacity for each. C_j denotes the capacity of the jth safe area and is greater than zero.

$$f_2 = \sum_j^n \left| \frac{\sum_i^m P_{ij}}{C_j} - 1 \right| \tag{3}$$

3.2.1.3 Total risk in the evacuation routes

In the domain of evacuation planning, minimizing total risk along evacuation paths is a critical aspect that directly impacts the safety and success of evacuation operations. Minimizing total risk along evacuation paths is essential for ensuring the safety, efficiency, and effectiveness of evacuation planning and operations. By prioritizing the safety of evacuees, optimizing evacuation routes, mitigating secondary hazards, enhancing community resilience, and fostering public confidence and trust, planners can help to minimize the impact of disasters and protect the lives and livelihoods of those at risk.

Equation (4) [4] is designed to model risks inherent in evacuation planning. Subsequently, we choose the most appropriate risk model for our specific scenario. In this study, we examine evacuation routes, which may encompass numerous bridges or flooded areas.

$$f_3 = \sum_{j}^{n} \sum_{i}^{m} R_{ij} \tag{4}$$

Let r_k represent the probability of a bridge k collapsing or a flooded area k being impassable. Conversely, the probability of successfully crossing a bridge is given by $1 - r_k$, where r_k denotes the collapse probability of bridge k. Consequently, the probability of successfully traversing all bridges is $\prod_k^N (1 - r_k)$, and the cumulative risk of traversing the entire route (denoted as R_{ij}) is calculated as:

$$R_{ij} = 1 - \prod_{k=1}^{N} (1 - r_k) \tag{5}$$

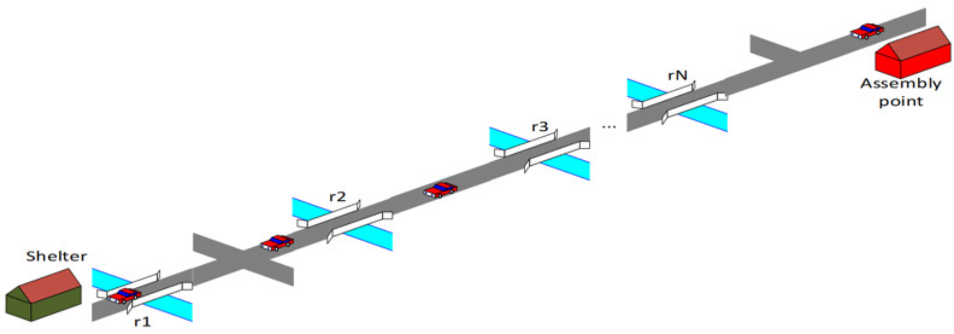


Figure 3: Risk Modelling in an evacuation path

Each bridge is assigned a risk factor representing the likelihood of crossing without incident, while flooded areas are assigned a risk factor indicating the probability of crossing without damaging the means of transportation. Figure 3 illustrates an evacuation route featuring multiple bridges.

3.2.2 Urban Land Use Allocation (Paper 3)

The study addresses the numerous challenges and complexities associated with urban land use allocation, highlighting the need for a comprehensive and collaborative approach involving various stakeholders. These challenges include limited space and high demand, inadequate infrastructure, affordability and housing shortages, environmental degradation, zoning and regulatory issues, gentrification and displacement, lack of public participation, climate change considerations, brownfield redevelopment, and data management and technology integration challenges.

To address these challenges, the study emphasizes the importance of rational land use in achieving sustainability and resilience, particularly in urban areas like Maputo, Mozambique, experiencing rapid urbanization and economic development. It introduces land use allocation as a multi-objective optimization problem, recognizing conflicting objectives such as environmental protection, economic development, and resource utilization.

The study proposes the Non-dominated Sorting Genetic Algorithm III (NSGA-III) as a suitable approach for addressing the complexities of land use allocation. NSGA-III is highlighted for its ability to find multiple Pareto-optimal solutions, addressing convergence and diversity issues inherent in multi-objective optimization problems.

Applied to the Kamavota district in Maputo City, Mozambique, the methodology demonstrates the effectiveness of NSGA-III in generating optimal land use allocation plans. Results indicate that the improved NSGA-III outperforms the standard NSGA-III, producing solutions that effectively balance conflicting objectives.

The optimal land use allocation plans generated by the proposed approach offer valuable insights for policymakers and city planners, providing alternative strategies for enhancing urban sustainability and resilience. By considering multiple objectives and constraints simultaneously, the methodology contributes to more informed decision-making in urban planning processes, aligning with the broader goals of sustainable development.

Overall, the study underscores the importance of addressing the complex challenges of urban land use allocation through innovative methodologies like multi-objective optimization, aiming to build sustainable and resilient cities for the future.

Table 2 provides detailed information about different land use classes in the study area. The table consists of eight attributes: (1) the land use types, (2) the area of one unit in square meters for each land use type, (3) the weight of the land use that represents the fraction of each land use type, (4) the recommended minimum travel distance between residential and other land use types by Minister of Environment, Science and Technology [119] and United Nations Human Settlements Programme [120], (5) the weight (w_j) of the land use distance that represents the fraction of the distance between the residential and other land use types, (6) the average value for each land use type, (7) the maximum capacity representing the maximum number of persons that can be allocated to each land use type, and (8) the average carbon emission that represents the amount of carbon emitted in each land use type

Table 2: Information about different land use types in Maputo

Land use types	Area for each unit(m^2)	Weight of the land use (μ_j)	Recommended Minimum travel distance	Weight of the land use distance (w_j)	Average value (Thousand MZM)	Maximum Capacity (person)	Average carbon emission (million tonnes)
Residential	500	0.0028	0m	0	859	5	0.0210
Nursery	5000	0.0283	800m	0.0298	5689	1000	0.0012
Primary School	12140	0.0686	800m	0.0298	13.813	1500	0.0018
Secondary school	24300	0.1373	1250m	0.0466	27640	5000	0.0260
Urban Health Center	50000	0.2826	2500m	0.0931	58889	25000	0.0410
Public Facilities	25000	0.1413	3000m	0.1117	1889	500	0.0320
Fire service	10000	0.0565	7500m	0.2793	2889	50	0.0220
waste burning center	50000	0.2826	11000m	0.4097	40819	40	0.4000
Total	**176940**	**1**	**65950m**	**1**	**138687.81**	**33095**	**0.7340**

The objective functions for paper three are presented below.

3.2.2.1 Maximization of the economic objective

In the realm of land use allocation, maximizing the economic objective holds significant importance due to its wide-ranging impacts on various aspects of society, development, and sustainability. Maximizing the economic objective in land use allocation is essential for promoting sustainable economic development, job creation, revenue generation, infrastructure development, and long-term prosperity. By prioritizing land uses that maximize economic returns while considering social and environmental factors, planners can support vibrant, resilient, and inclusive communities that thrive economically while safeguarding natural resources and enhancing quality of life for present and future generations.

The economic objective seeks to maximize the overall land use value, which encompasses the total worth of the land excluding any structures erected upon it. This valuation includes the intrinsic value of the raw land as well as additional values such as enhanced accessibility. When demand surpasses supply, the land's value typically rises. Similarly, land may possess inherent worth, such as containing oil reserves, which can elevate its value. It's important to note that the assessed value of the land and any structures may not necessarily align with the current market price, as the selling price is contingent upon prevailing market conditions.

In the Kamavota district, the land use composition consists of 5% fourth-floor buildings, 10% third-floor buildings, 15% second-floor buildings, 40% first-floor buildings, and 30% open space. Each floor of the buildings holds its own economic value. The economic value of each building is calculated by summing the economic value of each floor, multiplied by the proportion of the respective land use type. To derive the total economic land use value for the entire study area, the economic value for each cell is aggregated according to Equation 6.

$$f_1(\boldsymbol{x}) = \max\left(\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=0}^{K} E_{ij}^{(k)} \cdot \mu_{ij}\right), \tag{6}$$

In Equation (6), $E_{ij}^{(k)}$ represents the annual land value of the property located in cell (i, j), while μ_{ij} denotes the weight assigned to the land use type (as detailed in Table 2). Here, i signifies the ith row within the grid cell of the study area, with N indicating the total number of rows, and j representing the jth column within the grid, and M being the total number of columns. The variable k pertains to the number of floors present in the building situated on the land, while K signifies the maximum number of floors (or maximum height) permitted to be constructed, which, in this study, is set at five. Given that various land use configurations yield varying economic benefits, optimizing the design and arrangement of diverse land uses becomes imperative to maximize economic returns.

3.2.2.2 Minimization of the carbon emission objective

Minimizing the carbon emission objective in land use allocation is increasingly recognized as essential due to its profound implications for addressing climate change, promoting environmental sustainability, and ensuring the well-being of present and future generations. Minimizing the carbon emission objective in land use allocation is critical for addressing climate change, protecting ecosystems, improving public health, enhancing resilience, and promoting sustainable development. By prioritizing low-carbon land uses and reducing emissions associated with land use activities, planners can contribute to global efforts to mitigate climate change and build a more sustainable and resilient future for all.

The extent of human activity now wields a profound influence on the climate system. Human-induced alterations in greenhouse gas emissions disrupt the energy balance within the climate system, exerting a significant forcing effect. A critical aspect in estimating future emissions from infrastructures in developing nations and formulating effective mitigation strategies involves comprehending the emissions embedded within existing infrastructure stocks. This understanding is crucial for reconciling human development with climate change mitigation efforts, necessitating a clear grasp of how infrastructures contribute to both well-being and greenhouse gas emissions. While indirect emissions stemming from infrastructure usage are well-documented, information regarding indirect emissions originating from their construction remains fragmented.

To estimate the carbon emissions of each building, the carbon emission of each floor can be calculated as the sum of the carbon emitted by each floor multiplied by the proportion of the respective land use type. To determine the total carbon emissions within the study area, the carbon emissions for each cell are aggregated according to equation 7.

$$f_2(\boldsymbol{x}) = \min\left(\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=0}^{K} C_{ij}^{(k)} \cdot \mu_{ij}\right), \tag{7}$$

In this context, N denotes the total number of rows, where i signifies the ith row, M represents the total number of columns, and j represents the jth column. The variable k pertains to the floor level within a building, while K denotes the total number of floors. $C_{ij}^{(k)}$ represents the carbon emissions attributed to the building located in cell (i, j) on the kth floor, and μ_{ij} signifies the weight assigned to the land use type.

3.2.2.3 Maximization of the accessibility objective

Maximizing the Accessibility objective in land use allocation is crucial for promoting equitable access to essential services, opportunities, and amenities for all members of society. Maximizing the Accessibility objective in land use allocation is essential for promoting social equity, enhancing mobility and connectivity, driving economic development, improving health and well-being, and fostering environmental sustainability. By prioritizing accessible and inclusive land use planning strategies, policymakers can create vibrant, resilient, and livable communities that offer opportunities and amenities for all residents, now and in the future.

Ensuring accessibility within the built environment is essential for its intended and desired use. This objective function prioritizes the seamless access of residential areas to various non-residential public spaces, including parks, schools, and hospitals[5]. Accessibility, a commonly used metric for gauging ease of access to a location, fosters diverse use of public spaces, a key advantage in contemporary urban planning practices. In this study, we evaluate the accessibility of each residential cell to other land-use cells.

For every residential cell i and all other land uses cells j, we compute the weighted distances between them by multiplying the Euclidean distances between the cells by the weight of the respective land use j. Consequently, the total distance between each residential cell and cells of other land use types is calculated as the cumulative sum of distances from each residential cell i to other land use cells. Therefore, this objective function seeks to maximize accessibility by minimizing the total distance between residential cells and other land-use cells, as depicted in equation 8.

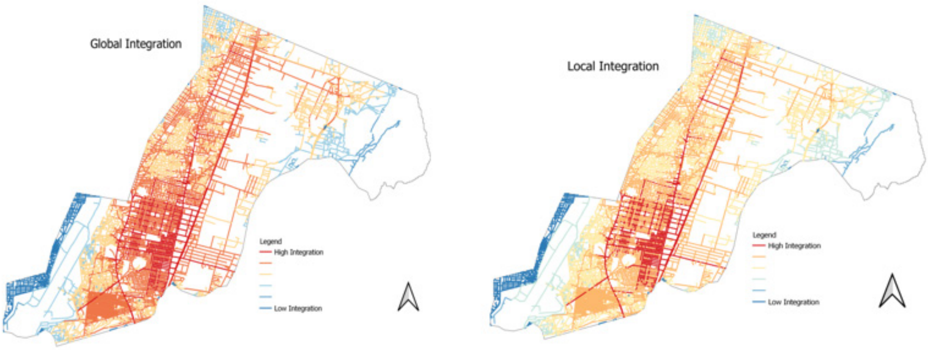
$$f_3(\boldsymbol{x}) = \min\left(\sum_{i=1}^{N}\sum_{j=1}^{M} d_{ij} \cdot w_j\right), \tag{8}$$

In this context, the variable i denotes residential cells, where N stands for the total number of residential cells. The variable j represents cells of other land uses, with M indicating the total number of cells pertaining to other land uses. The term d_{ij} signifies the minimum distance between residential cell i and land use cell j. Additionally, w_j represents the weight assigned to the distance to the facility for the respective land use (see Table 2).

3.2.2.4 Maximization of the space syntax integration objective

By prioritizing well-connected and accessible urban spaces, planners can create vibrant, livable communities that offer opportunities and amenities for all residents while ensuring long-term resilience and adaptability to future challenges.

Space syntax is an evidence-based approach centered on understanding the relationships between various spaces and the interactions between space and society. In space syntax analysis, axial lines are frequently utilized to represent urban structures, portraying the longest lines of sight within two-dimensional urban spaces. Global and local integration are two key morphological parameters often employed in axial line-based representations to analyze urban structure. Figure 4 illustrates global and local integration in the study area. Integration quantifies the number of transitions required from one street segment to reach all other street segments in the network via the shortest paths. Ensuring the proximity of public facilities to streets is crucial in urban land use planning.



(a) Global integration (b) Local integration

Figure 4: Space Syntaxe integration Measurement. Global integration represents the average integration value across all spaces in the configuration. It provides a holistic measure of the overall connectivity of the spatial network. Local integration quantifies the average segment depth of a particular space across its immediate neighbors or adjacent spaces. It provides a localized measure of connectivity within the spatial network.

This objective function is designed to enhance the proximity between cells representing non-residential public facilities (such as markets and shops) and streets exhibiting high local integration values. To achieve this objective, we calculate the Euclidean distance between streets with high integration values and the nearest non-residential land use types for each axial line segment i with high integration and each cell j representing a public facility. The distance between i and j corresponds to the Euclidean distance between the high-integration line segment and the public facility. Thus, the total distance between each high-integration line segment and each public facility cell is determined by summing the distances from each high-integration line segment i to all public facility cells. This process is outlined in equation 9.

$$f_4(\boldsymbol{x}) = \min\left(\sum_{i=1}^{S}\sum_{j=1}^{N} d_{ij}\right), \tag{9}$$

In this context, the variable i denotes the axial line segment with high integration, where S represents the total number of segments with high integration. The variable j represents a cell targeted for land use, specifically comprising public facilities, with N representing the total number of cells for this land use target. The term d_{ij} signifies the minimum distance from segment i to the nearest cell j.

3.2.2.5 Maximization of the compactness objective

Maximizing the Compactness objective in land use allocation is crucial for promoting efficient land utilization, sustainable development, and vibrant urban environments. Maximizing the Compactness objective in land use allocation is essential for promoting efficient land use, sustainable development, and vibrant urban environments. By prioritizing compact development patterns, planners can optimize land use efficiency, reduce infrastructure costs, promote sustainable transportation options, create vibrant urban centers, and preserve natural resources and ecosystems, fostering resilient and livable communities for current and future generations.

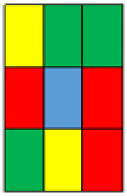
The global trend of urbanization has led to environmental degradation, prompting the emergence of the compact city concept as a potential solution. Compact cities have been shown to offer significant advantages in terms of suitable urban form and sustainability. A fundamental principle of urban sustainability is that more compact urban forms are more efficient in their overall use of space and energy. This often translates into the construction of high-rise buildings with a focus on energy management in their outer envelopes. Compact cities also provide residents with all necessary amenities within one community, including work opportunities. Citizens employed in compact cities can walk or bike short distances to work instead of driving, thereby reducing fossil fuel consumption, emissions, pollutants, and traffic density. In essence, compactness suggests efficient land planning, high density of the built environment, and intensification of activities.

The objective of this study is to maximize compactness for the sustainability of the city. We employ the basic eight-neighbor method to measure compactness. For each cell in the land use grid, we compute its ratio δ_{ij}, which represents the quotient between the number of neighboring cells with the same type and the total number of neighboring cells. Figure 5 illustrates examples of compactness calculation for different cells. The objective compactness is expressed as equation (10):

$$f_5(\boldsymbol{x}) = \frac{1}{T}\sum_{i=1}^{N}\sum_{j=1}^{M}\delta_{ij}, \tag{10}$$

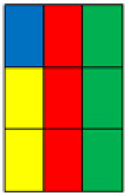
In this context, δ_{ij} represents the ratio of cells allocated for the same land-use type in each cell's eight neighboring cells. Here, j denotes the jth row, where N is the total number of rows. Similarly, i signifies the ith column, where M represents the total number of columns. Additionally, T denotes the total number of cells in the study area.

Grid x — δ_{ij} — $f_5(x)$



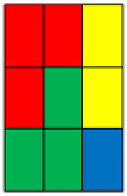
0,00	0,20	0,33
0,00	0,00	0,20
0,00	0,00	0,33

0,12



0,00	0,20	0,33
0,20	0,25	0,40
0,33	0,20	0,33

0,25



0,67	0,40	0,33
0,40	0,25	0,20
0,67	0,40	0,00

0,37

Figure 5: Example of Compactness Calculation

3.2.3 Site Selection for Solar Power Farms (Paper 4)

Renewable energies hold significant importance in addressing a variety of global challenges and contributing to the transition to a more sustainable and resilient energy system. The relevance of renewable energies extends across environmental, economic, social, and geopolitical dimensions, making these energies a crucial component of the global energy landscape as the world seeks sustainable and clean energy solutions.

In this study, the focus lies on the critical role of renewable energies in addressing global challenges such as climate change, environmental pollution, energy security, and sustainable economic development. With society's increasing shift away from fossil fuels towards cleaner and more sustainable energy sources, renewable energies, e.g. solar power, continue to gain momentum.

The study proposes a multi-objective optimization (MOO) model for the site selection of solar farms, aiming to maximize solar irradiation absorption while minimizing various factors such as distance to the electric grid, road networks, urban areas, slope, and hill shadow aspect values. By optimizing these objectives, the goal is to identify optimal locations for solar farms.

To achieve this, the study adapts the NSGA-II algorithm to fit the MOO model and optimize its performance. The improved algorithm is then utilized to generate optimal location maps for solar farms, providing decision-makers with a range of choices based on specific objective functions.

The results demonstrate the efficiency and effectiveness of the improved algorithm, as evidenced by the high-quality Pareto front-set solutions obtained. The Hypervolume indicator is used to assess the quality of the solutions, indicating that the algorithm performs well in finding optimal solutions that balance multiple objectives.

Overall, the methodology presented in the study offers a systematic approach to address the complex task of solar farm site selection through multi-objective optimization, providing valuable insights for decision-making in the renewable energy sector.

Below we present the summary of objective functions for paper 4.

3.2.3.1 Maximization of the Solar Radiation Objective

Maximizing solar radiation in solar power farms is critical for optimizing energy production efficiency, economic viability, and environmental benefits. Solar radiation directly impacts the amount of energy solar panels can generate, making it essential to ensure optimal sunlight exposure. By maximizing solar radiation, solar power farms can increase energy output, leading to higher revenues and improved economic feasibility. Additionally, harnessing maximum solar radiation reduces reliance on fossil fuels, contributing to environmental sustainability by reducing greenhouse gas emissions and air pollution. Moreover, maximizing solar radiation enhances grid stability and reliability by diversifying the energy mix and promoting energy security by utilizing locally available renewable resources. Overall, prioritizing the maximization of solar radiation in solar power farms is vital for achieving sustainable energy production and mitigating the impacts of climate change.

Solar radiation plays a crucial role in determining the efficiency and output of a solar farm, as it represents the sunlight that reaches the Earth's surface and can be converted into electricity through solar panels. Let R denote a set of non-negative real numbers, while n and m represent sets of positive integer numbers. Define $R^{n\times m}$ as the set of grid cells receiving solar irradiation over Maputo, and let $E \subseteq R^{n\times m}$ be a subset of cells irradiating over a photovoltaic power plant. The objective function, denoted by $f_1: E \rightarrow R$, defines the total amount of direct solar irradiation utilized in the photovoltaic power plant. Consider $x \in E$, where $x = (x_{ij})_{n\times m}$, and x_{ij} represents the amount of direct solar irradiation over the ith and jth cell within the photovoltaic power plant. If the cell (i, j) does not contain a photovoltaic power plant, then the value of x_{ij} is zero. The total amount of direct solar irradiation can be expressed as equation (11):

$$f_1(x) = \max\left(\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}\right). \tag{11}$$

3.2.3.2 Minimization of the Distance to Electric Grid Objective

Minimizing the distance from solar power farms to the electric grid holds significant importance for several reasons. Firstly, reducing this distance decreases transmission losses, as electricity traveling over long distances experiences greater energy loss. By minimizing the distance to the electric grid, solar power farms can maximize the efficiency of electricity transmission, ensuring that more of the energy generated reaches consumers. Secondly, minimizing the distance to the electric grid enhances grid stability and reliability. Solar power farms located closer to the grid can respond more quickly to fluctuations in electricity demand or supply, helping to balance the grid and prevent power outages. Thirdly, reducing the distance to the electric grid reduces the need for extensive infrastructure development, such as building new transmission lines or substations. This can lower the overall cost of solar power projects and accelerate their deployment, making renewable energy more accessible and affordable. Additionally, minimizing the distance to the electric grid can facilitate the integration of solar power into existing energy systems. Solar farms located nearby can more easily connect to the grid, allowing for smoother integration of renewable energy sources into the overall energy mix. Overall, minimizing the distance from solar power farms to the electric grid is essential for optimizing the efficiency, reliability, and cost-effectiveness of solar energy generation, ultimately advancing the transition to a more sustainable and resilient energy infrastructure.

Consider $E_1 \subseteq R^{n\times m}$, representing a set comprising all suitable locations for solar energy sites, where $x_{ij}^{\alpha} \in E_1$ denotes the suitable location for solar energy site α. Let $E_2 \subseteq R^{n\times m}$ be a set representing electric grids in Maputo, with $x_{ij}^{\beta} \in E_2$ denoting electric line β. Here, $E_1 \cap E_2 = \phi$ and $E = E_1 \cup E_2$. Define $f_2: E \rightarrow R$ as the objective function, which determines the total distance between suitable locations for solar energy and electric grid lines, where d_1 represents the minimum distance between the electric grid and the solar power plant. Thus, the total distance function to the electric grid is expressed as Equation (12):

$$f_2(x) = \min \sum_i \sum_j \sum_k \sum_l \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right|, \tag{12}$$

where $\left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right|$ is the Euclidian distance between x_{ij}^{α} and x_{kl}^{β}, and $\min \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right| \geq d_1$, for all $x_{ij}^{\alpha} \in E_1$ and all $x_{kl}^{\beta} \in E_2$. d_1 is the minimal distance allowed between the power solar farms and the electric grid.

3.2.3.3 Minimization of the Distance to Road Objective

Minimizing the distance from solar power farms to road networks is crucial for several reasons. Firstly, reducing this distance streamlines transportation logistics during the construction phase of solar power projects. Shorter distances to road networks minimize transportation costs and facilitate the delivery of equipment, materials, and manpower to the construction site, thus expediting project completion and reducing overall construction expenses. Secondly, minimizing the distance to road networks enhances operational efficiency and maintenance activities. Closer proximity to roads allows for easier access to solar arrays for routine inspections, repairs, and equipment replacement. This accessibility reduces downtime and ensures uninterrupted energy production, maximizing the reliability and performance of solar power farms. Thirdly, shorter distances to road networks facilitate grid connection and electricity transmission. Solar power farms located near roads can more efficiently connect to the electrical grid, reducing the need for extensive infrastructure development and associated costs. Additionally, closer proximity to roads simplifies the installation of power lines and substations, streamlining the process of delivering solar-generated electricity to consumers. Moreover, minimizing the distance from solar power farms to road networks enhances emergency response capabilities. In the event of unforeseen incidents such as equipment failures or natural disasters, quick access to road networks enables rapid deployment of emergency personnel and equipment to the site, minimizing downtime and mitigating potential disruptions to energy production. Overall, minimizing the distance from solar power farms to road networks is essential for optimizing construction efficiency, operational effectiveness, grid integration, and emergency response readiness. By strategically locating solar facilities close to road infrastructure, stakeholders can enhance the overall performance and resilience of solar energy projects, advancing the transition to a more sustainable and reliable energy future.

Consider $E_1 \subseteq R^{n \times m}$ representing a set comprising all suitable locations for solar energy sites, where $x_{ij}^{\alpha} \in E_1$ denotes the suitable location for solar energy site α. Let $E_2 \subseteq R^{n \times m}$ be a set representing road networks in Maputo, with $x_{ij}^{\beta} \in E_2$ denoting road line β. Here, $E_1 \cap E_2 = \phi$ and $E = E_1 \cup E_2$. Define $f_3: E \rightarrow R$ as the objective function, which determines the total distance between suitable locations for solar power farms and roads. Thus, the total distance function is expressed as equation (13):

$$f_3(x) = \min \sum_i \sum_j \sum_k \sum_l \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right|, \tag{13}$$

where $\left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right|$ is the Euclidian distance between x_{ij}^{α} and x_{kl}^{β}, and $\min \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right| \geq d_2$, for all $x_{ij}^{\alpha} \in E_1$and all $x_{kl}^{\beta} \in E_2$. d_2 is the minimum distance allowed between the power solar farms and the closest road.

3.2.3.4 Minimization of the Distance to Urban area

Minimizing the distance between solar power farms and residential areas carries significant importance for several reasons. Firstly, reducing this distance helps mitigate potential concerns related to land use conflicts and aesthetic impacts. By locating solar farms closer to residential areas, stakeholders can minimize the visual and environmental impacts associated with large-scale solar installations, preserving the aesthetic appeal of the surrounding landscape and minimizing community opposition to renewable energy projects. Secondly, minimizing the distance to residential areas enhances the economic feasibility and attractiveness of solar energy adoption. Proximity to residential communities reduces transmission losses and infrastructure costs associated with delivering electricity from solar farms to consumers, making solar energy more cost-effective and accessible to local residents. Additionally, closer proximity to residential areas encourages community engagement and participation in renewable energy initiatives, fostering a sense of ownership and support for sustainable energy development. Thirdly, reducing the distance to residential areas promotes distributed generation and energy self-sufficiency. Locating solar power farms near residential communities enables residents to directly benefit from clean energy generation, potentially reducing their dependence on centralized power grids and fossil fuel-based electricity sources. This decentralization of energy production enhances energy resilience, reduces vulnerability to grid outages, and empowers communities to take control of their energy future. Moreover, minimizing the distance from solar power farms to residential areas can stimulate local economic development and job creation. Solar energy projects located near residential communities provide opportunities for local employment, subcontracting, and procurement, stimulating economic activity and generating revenue for local businesses and municipalities. Overall, minimizing the distance from solar power farms to residential areas is essential for enhancing community acceptance, economic viability, energy resilience, and environmental sustainability. By strategically locating solar installations closer to residential communities, stakeholders can maximize the benefits of solar energy while minimizing potential drawbacks, accelerating the transition to a cleaner, more sustainable, energy future.

Solar power plants can generate noise, such as that from inverters, and may have visual impacts. To mitigate these effects on nearby residences, minimum distances can be established. Consider $E_1 \subseteq R^{n\times m}$ as a set encompassing all suitable locations for solar energy sites, where $x_{ij}^{\alpha} \in E_1$ denotes the suitable location for solar energy site α. Let $E_2 \subseteq R^{n\times m}$ represent all urban areas in Maputo, with $x_{ij}^{\beta} \in E_2$ being the suitable location for solar energy site β. Here, $E_1 \cap E_2 = \phi$ and $E = E_1 \cup E_2$. We define $f_4: E \rightarrow R$ as the objective function, which determines the total distance between suitable locations for solar energy and urban areas, where d_3 represents the minimum distance between the residential area and the solar power plant. Thus, the total distance function is expressed as equation (14):

$$f_4(x) = \min \sum_i \sum_j \sum_k \sum_l \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right| \tag{14}$$

where $\left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right|$ is the Euclidian distance between x_{ij}^{α} and x_{kl}^{β}, and $\min \left| x_{ij}^{\alpha} - x_{kl}^{\beta} \right| \geq d_3$, for all $x_{ij}^{\alpha} \in E_1$and all $x_{kl}^{\beta} \in E_2$. d_3 is the minimal distance allowed between the power solar farm to the urban area (habitational infrastructures).

3.2.3.5 Minimization of the Slope Objective

Minimizing slope in a location for solar power farms is crucial for optimizing energy production efficiency and reducing operational challenges. Firstly, a flat terrain allows for optimal orientation and positioning of solar panels, maximizing their exposure to sunlight throughout the day. By minimizing slope, solar panels can capture sunlight more effectively, resulting in higher energy yields and increased electricity generation. Secondly, flat terrain simplifies the installation and maintenance of solar arrays. Solar panels are typically mounted on fixed structures or tracking systems that require level ground for proper installation and operation. Minimizing slope reduces the need for costly site preparation and grading, streamlining the construction process and minimizing project expenses. Thirdly, flat terrain facilitates the deployment of equipment and machinery required for construction and maintenance activities. Access roads, construction vehicles, and heavy machinery can navigate flat terrain more easily, reducing transportation costs and minimizing logistical challenges associated with rugged or uneven terrain. Moreover, minimizing slope enhances the stability and durability of solar installations. Flat terrain provides a stable foundation for mounting structures, reducing the risk of structural failure or damage due to uneven ground conditions, erosion, or landslides. This enhances the reliability and longevity of solar power farms, ensuring consistent energy production over the lifetime of the project. Overall, minimizing slope in a location for solar power farms is essential for optimizing energy production, reducing construction and maintenance costs, facilitating project logistics, and enhancing the reliability and durability of solar installations. By selecting sites with minimal slope, stakeholders can maximize the efficiency and effectiveness of solar energy projects, contributing to the transition to a cleaner, more sustainable, energy future.

A solar farm's location should prioritize minimal shading and optimal sunlight exposure for efficient energy capture. In this context, the Digital Elevation Model (DEM) slope becomes relevant. Slope indicates the terrain's steepness or incline at a specific site, typically measured in degrees or as a percentage (rise over run). For solar power plant placement, areas with relatively flat or low slopes are preferred, as they facilitate easier installation of solar panels and maximize sun exposure. Conversely, high slopes can pose challenges during panel installation, potentially necessitating additional structural support. Moreover, steeper slopes may impede solar panel efficiency by obstructing direct sunlight for prolonged periods. Consider R as a set of non-negative real numbers, and n and m as sets of positive integers. Let $R^{n\times m}$ represent the set of grid cells covering Maputo's slope, and $E \subseteq R^{n\times m}$ denote a set of cells containing suitable land for a solar site. Define $f_5 : E \to R$ as the objective function determining the total amount of slope utilized in a photovoltaic power plant. Here, $x \in E$ and $x = (x_{ij})_{n\times m}$, where x_{ij} represents the slope over the cell (i, j) within the photovoltaic power plant. If the ith and jth cells do not contain suitable land for a solar site, x_{ij} is set to zero. The total slope utilized across all photovoltaic power plants is expressed as equation (15):

$$f_5(x) = \min\left(\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}\right). \tag{15}$$

3.2.3.6 Maximization of the Aspect Objective

Maximizing aspect in a location for solar power farms is vital for optimizing energy production and maximizing the efficiency of solar panels. Aspect refers to the direction that a slope faces, which directly influences the amount of sunlight received by solar panels throughout the day. Selecting sites with favorable orientations, ensures that solar panels receive maximum sunlight exposure, leading to higher energy yields and increased electricity generation. By maximizing aspect, solar power farms can capitalize on the sun's movement across the sky, maximizing the amount of sunlight captured by solar panels at different times of the day. This results in more consistent and reliable energy production, reducing variability and enhancing the predictability of solar energy generation. Furthermore, selecting sites with favorable aspect can improve the performance of solar panels and optimize their efficiency. Solar panels operate most efficiently when they are directly facing the sun, known as being "south-facing" in the northern hemisphere and "north-facing" in the southern hemisphere. Maximizing aspect ensures that solar panels are oriented towards the sun for a greater portion of the day, maximizing their energy output and overall performance. Additionally, maximizing aspect can help mitigate shading effects caused by nearby obstructions such as trees, buildings, or terrain features. By selecting sites with favorable aspect, solar power farms can minimize shading and ensure uniform sunlight exposure across the entire solar array, maximizing energy production and reducing the risk of efficiency losses. Overall, maximizing aspect in a location for solar power farms is essential for optimizing energy production, maximizing efficiency, and enhancing the reliability and performance of solar installations. By selecting sites with favorable orientations and maximizing aspect, stakeholders can maximize the benefits of solar energy and contribute to the transition to a cleaner, more sustainable, energy future.

Aspect refers to the compass direction that a slope faces, typically measured in degrees from north (e.g., 0° for north, 90° for east, 180° for south, and 270° for west) [37]. It significantly influences the amount of sunlight received by a location throughout the day. Solar panels are most efficient when facing south (in the Northern Hemisphere) or north (in the Southern Hemisphere) to capture the maximum direct sunlight. Optimizing a solar power plant's aspect involves aligning solar panels to maximize exposure to the sun. Consider R as a set of non-negative real numbers, and n and m as sets of positive integers, let $R^{n\times m}$ represent the set of grid cells covering Maputo's aspect, and $E \subseteq R^{n\times m}$ denote a set of cells irradiating over a photovoltaic power plant. Define $f_6: E \to R$ as the objective function determining the total amount of direct solar irradiation utilized in a photovoltaic power plant. Here, $x \in E$ and $x = (x_{ij})_{n\times m}$, where x_{ij} represents the amount of direct solar irradiation over the cell (i, j) for a solar energy site. If the cell (i, j) is not suitable for solar energy sites, x_{ij} is set to zero. The total aspect is expressed as Equation (16):

$$f_6(x) = \max\left(\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}\right). \tag{16}$$

3.3 Metaheuristic Methods on MOO

In this subsection, we give detailed descriptions of the methods used in this thesis, from MOO techniques to the tools for analyzing the quality of solutions in spatial problems. Integrating multi-objective optimization with spatial analysis in Geographic Information Systems (GIS) involves combining the capabilities of optimization algorithms with spatial data to find optimal solutions considering multiple conflicting objectives in a spatial context. When integrating multi-objective optimization with spatial analysis in GIS, it is essential to carefully define the objectives, constraints, and spatial relationships to ensure the relevance and feasibility of the solutions obtained. The choice of method depends on the specific problem domain and the nature of the spatial data involved.

The integration of the Multi-Objective Cuckoo Search Algorithm (MOCSA), Archive Multi-Objective Simulated Annealing (AMOSA), Multi-Objective Artificial Bee Colony (MOABC), Multi-Objective Particle Swarm Optimization (MOPSO), Non-Dominated Sorting Genetic Algorithm-II (NSGA-II), and Non-Dominated Sorting Genetic Algorithm-III (NSGA-III) with spatial analysis in GIS results in synergizing the optimization capabilities of each algorithm with spatial data and analyses. These integrated approaches enable a comprehensive analysis, taking into account both the multi-objective optimization goals and the spatial context. This integration facilitates more informed decision-making across various domains where spatial considerations play a pivotal role.

Figure 6 provides a general architecture for solving Spatial MOO problems using metaheuristic algorithms involves considerations specific to spatial domains. The architecture consists of three domains, the intelligence domain, the design domain and post-processing domain. The intelligence domain consists of three steps. The first step comprises the Spatial Problem formulation, which defines the spatial MOO problem, incorporating location-based decision variables and spatial constraints. In addition, it specifies spatial objectives that capture the spatial distribution or arrangement of solutions. The second step comprehends the algorithm selection for spatial problem that choose a metaheuristic algorithm suitable for spatial optimization. Algorithms like Spatial Genetic Algorithms (SGA), Spatial Particle Swarm Optimization (SPSO), or Spatial Differential Evolution (SDE) may be appropriate. The third step of the intelligence domain comprises the spatial encoding, which develops an encoding scheme that considers the spatial nature of the decision variables. Spatial coordinates, distances, or neighborhood relationships might be encoded based on the problem requirements.

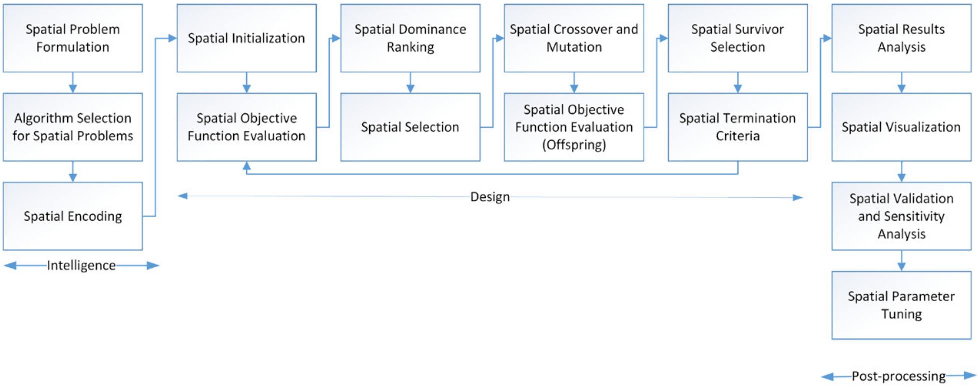


Figure 6: A General Architecture for Spatial MOO problems using metaheuristics algorithms, adapted from [122]

The design domain consists of eight steps. The first step comprises the spatial initialization, which generates an initial population of spatial solutions. It considers the spatial relationships and constraints during the initialization process. The second step comprehends the spatial objective function evaluation that evaluates spatial objective functions considering spatial characteristics. This may involve analyzing patterns, distributions, or connectivity in addition to traditional optimization metrics. The third step is the spatial dominance ranking, this applies Pareto dominance criteria with consideration of spatial relationships. It identifies spatially non-dominated solutions that form the Pareto front. The spatial selection step selects solutions for reproduction based on dominance ranking and spatial considerations. It gives preference to solutions that exhibit favorable spatial characteristics. The spatial crossover and mutation steps implement spatially-aware genetic operators to generate new solutions. These operators should respect spatial constraints and relationships. The spatial objective-function evaluation step evaluates the spatial objective functions for the newly generated offspring, considering spatial patterns and distributions. The spatial survivor selection step combines parent and offspring populations, and selects solutions for the next generation considering spatial dominance and spatial diversity. The spatial termination criteria step defines stopping criteria, such as a maximum number of generations or reaching a specific spatial convergence level.

The post-processing domain consists of four steps. The first step comprises the spatial results analysis, which analyzes the spatial characteristics of the obtained Pareto front. It evaluates the spatial trade-offs between conflicting objectives and identifies spatially efficient solutions. The second step is the spatial visualization, which visualizes the Pareto front and spatial characteristics to facilitate decision-making. Spatial plots, maps, or graphs may be used for effective visualization. The spatial validation and sensitivity analysis step validates results considering spatial aspects and performs sensitivity analysis to assess robustness concerning variations in spatial parameters and the spatial parameter tuning step, and fine-tunes algorithm parameters, especially those related to spatial considerations, to improve performance on specific spatial MOO instances.

The described architecture highlights the spatial aspects that should be integrated into each stage of solving spatial MOO problems using metaheuristic algorithms. The specific details may vary based on the chosen algorithm and the characteristics of the spatial optimization problem.

3.3.1 Multi-objective Cuckoo Search Algorithm

The Multi-Objective Cuckoo Search Algorithm (MOCSA) was developed as an extension of the original Cuckoo Search Algorithm. MOCSA is tailored to tackle optimization problems with multiple conflicting objectives simultaneously. It offers a robust solution approach, generating a diverse set of solutions that capture trade-offs among different criteria. At its core, MOCSA operates by iteratively updating cuckoo solutions' positions using Levy flights, a stochastic movement pattern akin to the foraging behavior of cuckoo birds. This enables exploration and exploitation of the solution space while maintaining diversity. Solutions undergo dynamic selection, replacement, and adaptation processes, mirroring cuckoos' nesting behavior. A crucial aspect of MOCSA is its fitness evaluation, where solutions are assessed based on multiple conflicting objectives. The algorithm aims to identify Pareto-optimal solutions that represent the best trade-offs among these criteria. Through iterative exploration, MOCSA dynamically adjusts the solution population, favoring solutions contributing to the Pareto front—a set of non-dominated solutions where no solution is superior in all objectives. MOCSA strikes a balance between exploration and exploitation, offering decision-makers a comprehensive view of trade-offs in multi-objective optimization problems. Its versatility and effectiveness in navigating complex landscapes make it valuable across various domains, where balancing multiple objectives is paramount for informed decision-making.

The standard MOCS algorithm was formulated for tackling continuous optimization problems characterized by multi-objective functions [6]. In order to address discrete multi-objective optimization problems, certain adjustments were introduced to the original MOCS framework. Previous research endeavors have applied MOCS with the specific aim of identifying optimal Pareto solutions [7, 8]. Additionally, some studies have explored the hybridization of MOCS with other optimization algorithms, aiming to enhance the overall performance of MOCS [9–11]. The MOCS optimization algorithms incorporate three essential parameters: Probability to abandon the worst nest(p_a), this parameter determines the likelihood of abandoning the least promising solution; Non-negative step size (α), represents a non-negative step size that needs to be tailored to the scale of the problem. In most cases, it exceeds one. For evacuation planning problems, the step size is specifically related to the current solution x_t; Random step length (λ), is a crucial parameter in the optimization process [12]. During the generation of new solutions $x_i^{(t+1)}$ for a given cuckoo, a Lévy flight is employed. A Lévy flight characterizes a random walk, where the steps are defined in terms of a step length. This length is distributed according to a heavy probability distribution, and the direction of the steps is both isotropic and random. The procedure is executed as presented in Equation 17:

$$x_i^{(t+1)} = \left\lfloor x_i^{(t)} + \alpha \oplus Levy(\lambda) \right\rfloor \%(1+5), \tag{17}$$

where $\oplus$ means entry wise multiplication, and % is the modulus arithmetic operator, and this returns the remainder of the division of each vector component by (1 + 5) to guarantee that every entry wise is between zero and five. Moreover, there is a probability p_a that the worst nests may be abandoned, allowing for the construction of new nests at different locations through a process of random walks and mixing. This can be achieved by randomly permuting the solutions based on their similarity or difference to the host egg.

3.3.2 Archive Multi-Objective Simulated Annealing Algorithm

AMOSA finds applications in diverse domains like engineering, finance and environmental management, where decision-makers must navigate trade-offs among conflicting objectives. At its core, AMOSA begins by generating an initial solution within the search space and evaluating it using multiple objective functions. An archive is then initialized to store non-dominated solutions, representing those not surpassed by any other solutions across all objectives. Through simulated annealing iterations, AMOSA explores neighboring solutions, evaluating their fitness based on the multi-objective criteria and accepting or rejecting them using the Metropolis criterion. The algorithm dynamically adjusts its exploration-exploitation balance by gradually reducing the temperature parameter, controlling the likelihood of accepting worse solutions. This temperature reduction encourages exploration of the solution space early on and focuses on promising regions as the algorithm progresses. AMOSA maintains an archive of non-dominated solutions throughout the process, ensuring a diverse representation of the Pareto front. The algorithm iterates until a termination criterion is met, such as reaching a maximum number of iterations or a predefined temperature threshold. By orchestrating a dynamic interplay between exploration and exploitation, AMOSA effectively navigates complex optimization landscapes, offering decision-makers a comprehensive view of trade-offs among conflicting objectives.

In Paper 1, the AMOSA algorithm underwent an extension from the principles of simulated annealing to address problems with multiple objectives. This extension primarily revolves around the methodology for calculating the probability of accepting an individual x_0 when $f(x_0)$ is dominated concerning $f(x)$. The acceptance of novel solutions hinges on the probability derived from assessing the degree of dominance between two solutions, denoted as a and b, in the following manner, expressed in Equation (18):

$$\Delta dom_{a,b} = \prod_{i=1, f_i(a) \neq f_i(b)}^{M} (|f_i(a) - f_i(b)|/R_i) \tag{18}$$

where M = number of objectives and R_i is the range of the ith objective. A new solution is selected based on the probability computed with the following equation (19):

$$p_{qs} = \frac{1}{1 + e^{-\frac{E(q,T)-E(s,T)}{T}}}, \tag{19}$$

where q is the current state and $E(s,T)$ and $E(q,T)$ are the corresponding energy values of s and q, respectively [13]. Equations (18) and (19) were employed for the selection and sorting of non-dominated solutions within the archive. The algorithm concludes when the cooling process reaches the pre-defined low temperature and the maximum number of iterations is achieved.

3.3.3 Multi-Objective Artificial Bee Colony Algorithm

The Multi-Objective Artificial Bee Colony Algorithm (MOABC) is a bio-inspired optimization technique modeled after the foraging behavior of honeybee colonies. Developed to tackle problems with multiple conflicting objectives, MOABC orchestrates a collaborative and dynamic search for Pareto-optimal solutions. At its core, MOABC initializes a population of artificial bees, each representing a potential solution within the problem's search space. These bees embody candidate trade-offs among the multiple objectives under consideration. The algorithm employs three types of artificial bees: employed bees, onlooker bees, and scout bees. Employed bees explore the search space by exploiting their current solutions and iteratively improving them through local search mechanisms. Onlooker bees evaluate solutions communicated by employed bees, favoring those that contribute to the Pareto front. Scout bees ensure diversity in the search space by discovering new solutions in uncharted regions, preventing premature convergence to suboptimal solutions. Fitness evaluation is pivotal in MOABC, involving assessing each solution's performance based on multiple conflicting objectives. The algorithm aims to discover a set of solutions collectively forming the Pareto front, where no solution is superior in all aspects. Through iterative cycles, the artificial bees dynamically explore, share information, and adapt their solutions. MOABC strikes a balance between exploration and exploitation, enabling it to navigate complex landscapes with diverse trade-offs. The algorithm continues this dynamic exploration until meeting a termination criterion, such as reaching a maximum number of iterations or achieving a satisfactory Pareto front. MOABC emerges as a robust optimization tool, leveraging the collective intelligence observed in nature to address challenges posed by multi-objective optimization problems. Its collaborative and adaptive approach suits domains where decision-makers must navigate trade-offs among conflicting objectives, such as engineering design, finance, and resource allocation.

In MOABC, the representation of the bee population is analogous to the depiction in Figure 7. In the initial phase of the algorithm, a set of scout bees is initialized, where each bee represents a food source in the form of an array.

Initial population [[2, 2, 1, 1, 1, 2, 2, 3, 2, 1], [3, 2, 2, 2, 1, 2, 2, 2, 1, 1],
[3, 3, 2, 3, 3, 3, 2, 1, 3, 2], [1, 3, 1, 2, 1, 1, 1, 1, 3, 2]]

Figure 7: An example of coding of a solution. The indices of the list represent the number of 10 building blocks while elements from 1 to 3 represent shelters. In this example, a population of 4 solutions is randomly generated.

The array size corresponds to the number of building blocks and consists of 10 repeated indices, each representing one of the 10 shelters. Following the initialization and fitness evaluation, the superior solutions are stored in an external archive (a new list). Since this archive houses the best solutions identified thus far, each employed bee x_{id} selects a solution randomly from the archive to update itself, becoming v_{id}. . The update of the solution is carried out through the following equations (20 and 21):

$$v_{id} = x_{id} + w \cdot rand[0,1](x_{id} - x_{kd}), \tag{20}$$

$$p_i = \frac{f(X_i)}{\sum_{j=1}^{n} f(X_j)} \tag{21}$$

where i represents the food source which is going to be updated, $k \in \{1, 2, \dots, bee\}$, and $d \in \{1, 2, \dots, D\}$ are randomly chosen indexes. The coefficient w is used to control the influence of the food source k in the production of the new food source.

Following the assessment of the fitness of employee bees and the update of the archive with the best solutions, a roulette wheel selection method is employed to choose onlooker bees for the subsequent generation. The roulette wheel method selects an individual based on the probability p_i, determined by calculating the ratio of individual fitness $f(x_i)$ to the total fitness of the population n, as outlined in Equation (20). Both employed bees and onlooker bees execute a neighborhood search using the expression in Equation (19) [14].

3.3.4 Multi-Objective Particle Swarm Optimization Algorithm

The Multi-Objective Particle Swarm Optimization (MOPSO) algorithm is a captivating narrative inspired by the collective behavior observed in nature, particularly in the swarming patterns of birds and fish. It extends the traditional Particle Swarm Optimization (PSO) to address problems with multiple conflicting objectives, belonging to the class of population-based optimization algorithms. Envision a bustling swarm of particles navigating a vast multidimensional search space, driven by the collective quest for optimal solutions that balance conflicting objectives. The algorithm begins with the genesis of a vibrant particle swarm, with each member representing a potential solution within the landscape. As the algorithm unfolds, each particle's fitness is evaluated against multiple conflicting objectives. In this dynamic swarm, particles celebrate personal achievements through personal best solutions (pbest) while collectively striving for excellence represented by the global best solution (gbest). The heart of the MOPSO algorithm lies in the fluid and adaptive movement of particles, informed by past directions, personal insights, and global achievements. Pareto dominance adds complexity to the evaluation process, where solutions are judged not only on individual merits but also in relation to each other. The algorithm maintains exploration vitality, preventing premature convergence and fostering a broad representation of potential solutions. Iteratively, the swarm dances through the solution space until a predetermined termination criterion signals the conclusion. The journey's end reveals an ensemble of non-dominated solutions forming the Pareto front, providing decision-makers with a nuanced palette of alternatives to consider. MOPSO transcends computational processes, embodying collaboration, exploration, and discovery. Through its bio-inspired design, it navigates decision-making scenarios, offering harmonious solutions where conflicting objectives find resolution in the intricate dance of the swarm.

In the MOPSO algorithm applied in this research, the search space comprises every conceivable arrangement of all building blocks assigned to any potential shelter, with each arrangement considered as a potential particle. The MOPSO algorithm seeks a particle location that satisfies the two defined objective functions related to evacuation planning. Here, a particle is synonymous with a solution and is initially set randomly (see Figure 8). It is worth noting that the SPSO algorithm was originally devised for continuous spaces and real numbers, whereas our problem space is discrete. To address this, a rounded value method was implemented to map between the discrete problem space and the continuous space. Specifically, the 10 shelters are randomly associated with integer values ranging from 1 to 10. The real values generated from updating the positions of particles (particle movements) are rounded to obtain integer values within the range of 1 to 10. Figure 6 illustrates an example of an initial particle in continuous space transformed into discrete space after a particle position update.

Particle in continuous space	2.014	1.277	2.020	1.682	1.454	2.263	1.162	1.245	2.571	1.869
Particle in discrete space	2	1	2	2	1	2	1	1	3	2

Figure 8: An example of MSPSO particle mapped in continuous space and remapped in discrete space.

3.3.5 Non-Dominated Sorting Genetic Algorithm-II

The Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) is a powerful and widely used optimization algorithm designed to address multi-objective optimization problems. It belongs to the family of evolutionary algorithms and is an extension of the original NSGA algorithm. NSGA-II was introduced by Deb et al. [15] in 2002. NSGA-II has proven to be effective in solving complex optimization problems with multiple conflicting objectives and is widely applied in various fields, including engineering, finance, and operations research.

In manuscript 4, we elucidate the adaptation of our model for implementation in NSGA-II. This evolutionary algorithm is specifically designed for addressing multi-objective optimization problems, wherein the task involves optimizing multiple conflicting objectives simultaneously. NSGA-II is a Python-based framework, taking advantage of the object-oriented programming paradigm [16]. In object-oriented programming, a class serves as an extensible program-code-template for creating objects. It provides initial values for state (member variables) and implementations of behavior (member functions or methods) [17]. To tailor our model for integration with NSGA-II, we define a class called "Model," inheriting the properties and methods [18] of the NSGA-II class named "ElementWise." The "Model" class encapsulates the logical definition of all objective functions and constraints [19]. To generate a sample of random individuals, we introduce a class named "RandomIntegerSample," inheriting from the NSGA-II class "PermutationRandomSampling." We override the method in "PermutationRandomSampling" with our own logic to generate individuals that are in proximity to the solution. The process of generating the initial population holds significant importance as it accelerates the performance of the algorithm and ensures the creation of individuals that are not entirely random but rather close to the solution. In our approach, two crossover operators are employed: the one-point crossover and the two-point crossover operators. These operators play a crucial role in combining genetic information during the evolution of the population.

Following are the key steps for the implementation:

Step 1. Initialization: Generate a population of 50 individuals, and potential solutions using the class "RandomIntegerSample."

Step 2. Evaluation: Assess the fitness of each individual concerning multiple objectives. In multi-objective optimization, the goal is to identify a set of solutions representing a trade-off between conflicting objectives.

Step 3. Non-dominated sorting: Categorize individuals into different fronts based on their dominance relationship. An individual dominates another if it is equal or superior in all objectives and strictly superior in at least one objective. Higher fronts consist of individuals that are non-dominated by those in lower fronts.

Step 4. Crowding distance assignment: Assign a crowding distance to individuals within each front. The crowding distance measures how crowded an individual is within its front, aiding in maintaining diversity in the population.

Step 5. Sorting and selection: Sort individuals based on their front and crowding distance. Individuals in less crowded regions of higher fronts are given preference for the next generation.

Step 6. Crossover and mutation: Apply genetic operators such as crossover and mutation to create offspring from the selected individuals. Utilize the one-point crossover and two-point crossover operators.

Step 7. Replacement: Combine the offspring with the parent population, creating a new population. Maintain the population size by selecting individuals based on non-dominated sorting and crowding distance.

Step 8. Termination criteria: Iterate through these steps until a termination criterion is met. The predefined criterion is a specified number of generations, set at 50 in this case.

3.3.6 Non-Dominated Sorting Genetic Algorithm-III

The Non-Dominated Sorting Genetic Algorithm-III (NSGA-III) is an extension of the NSGA-II and is designed to address many-objective optimization problems, where the number of objectives is significantly larger than 2. Developed by Kalyanmoy Deb and Himanshu Jain [24], NSGA-III maintains the principles of non-dominated sorting, diversity preservation, and Pareto dominance while introducing enhancements to handle a larger number of objectives more efficiently.

NSGA-III includes innovation and adaptation, where solutions evolve through environmental selection, guided by reference points, to unveil a Pareto front that captures the intricate nuances of many-objective optimization problems. Its contribution to the optimization narrative lies in its ability to effectively handle scenarios with a multitude of conflicting objectives, providing decision-makers with valuable insights into trade-offs and compromises.

NSGA-III stands as a robust algorithm designed for addressing Multi-Objective Optimization (MOO) problems with efficiency, navigating and approximating the Pareto front through non-dominated sorting and reference points. This algorithm yields a diverse set of high-quality solutions, empowering decision-makers to make well-informed choices aligned with their preferences across various objectives. In paper 3, the procedural steps of NSGA-III, depicted in Figure 9, commence with population initialization based on the specified problem range and constraints.

Step 1	**Initialization**
➤	Initialize the population by generating a set of individuals randomly or using a specified initialization strategy.
➤	Evaluate the objective functions for each individual in the population.
➤	Perform non-dominated sorting to classify individuals into different Pareto fronts.
➤	Calculate the crowding distance for each individual within each Pareto front.

Step 2	**Evolutionary Operators**
➤	Perform selection to choose individuals for reproduction. NSGA-III typically uses binary tournament selection based on dominance and crowding distance.
➤	Apply genetic operators such as crossover and mutation to create offspring individuals from the selected parents.
➤	Evaluate the objective functions for the offspring individuals.
➤	Merge the parent and offspring populations to form a combined population.

Step 3	**Selection**
➤	Combine the parent and offspring populations to create a combined population.
➤	Perform non-dominated sorting on the combined population.
➤	Select individuals for the next generation based on non-dominated sorting and crowding distance. NSGA-III introduces a reference point-based selection mechanism for this purpose.

Step 4	**Reference Point Update**
➤	Update reference points used in environmental selection based on the obtained solutions in the current generation. This helps guide the search towards the desired regions of the objective space.

Step 5	**Termination Criteria**
➤	Check for termination conditions such as reaching a maximum number of generations, convergence criteria, or a predefined computational budget.
➤	If termination conditions are not met, repeat steps 2-4 until termination.

Step 6	**Output**
➤	Once the algorithm terminates, the final population consists of a diverse set of non-dominated solutions approximating the Pareto front.

Figure 9: The procedure of NSGA III

Subsequently, a sorting process based on non-domination criteria is applied to the initialized population. Upon completion of the sorting, crowding distance values are assigned front-wise. The selection of individuals from the population is then executed based on their rank and crowding distance, employing a binary tournament selection mechanism with a crowded-comparison operator. The real-coded genetic algorithm integrates simulated binary crossover and polynomial mutation as genetic operators [136].

NSGA-III amalgamates the offspring population with the existing generation population, determining the individuals of the subsequent generation through a selection process. This cycle continues until the population size surpasses the current population size. In the context of this study, enhancements are made to NSGA-III, specifically focusing on mutation and crossover operations while adhering to constraints associated with the sizes of distinct land use types. The land use types, outlined in Table 1, exhibit varying sizes—ranging from one cell for residential areas to four cells for fire services and twenty cells for urban areas. Notably, the space syntax analysis guides the recommendation to establish public facilities in cells with high integration.

Despite public facilities having ten cells each, this study emphasizes that there is no mandatory requirement to group these cells based on the distribution of cells with high integration. A significant challenge in the optimization of land use allocation lies in ensuring the completeness of land use types comprising more than one cell throughout the optimization process.

3.3.6.1 Constraint-preserved mutation operation

Mutation involves altering the structure of a gene, resulting in a variant form that may be passed down to offspring. This alteration can be attributed to changes in the gene sequence, or the deletion, insertion, or rearrangement of larger sections of genes or chromosomes [20]. The purpose of mutations is to introduce diversity into solutions by modifying genes within the chromosome. Mutation operators play a crucial role in generating offspring with favorable characteristics. In the realm of spatial mutations, various operators, such as the two-step spatial mutation, are employed to foster solution diversification and prevent entrapment in local optima, ultimately promoting the compactness of land use. In consideration of the specified constraints, a constraint-preserved mutation operation approach has been devised. Given the existence of eight land use types in this study, each land use type is assigned a probability of 0.125 for selection. Notably, since the residential land use type comprises a single cell per unit, all mutation operations are exclusively conducted between the residential land use type and the remaining seven types. When a land use type other than residential is selected, it is reallocated to the nearest available space containing residential cells. This meticulous approach ensures that each mutation operation strictly adheres to the predefined constraints regarding the sizes of various land use types.

3.3.6.2 Constraint-preserved crossover operation

Crossover, a key process in genetic algorithms, involves exchanging segments between paired homologous chromosomes [21]. During the execution of a genetic algorithm, a crossover operation generates a novel combination of genes by swapping genes between different chromosomes, guided by a specific or adaptive probability. Given that land use types, excluding residential, consist of more than one cell, the conventional practice of swapping genes poses a risk of disrupting the completeness of certain land use types. In paper 3, we enhance the traditional crossover operator to safeguard against this constraint. Specifically, our improved crossover operator adopts a strategy to maintain the integrity of land use types. In each crossover operation, units of land use types are randomly selected from both parent chromosomes. If the selected units share the same land use type, the new offspring is generated through the crossover. However, if the units represent different land use types, the neighboring cells of the chosen land use unit from parent 1 are considered until the completion of the crossover operation. This meticulous approach ensures that the crossover process respects the predefined constraint related to the completeness of land use types.

3.4 Hypervolume

Hypervolume is a performance metric commonly used in multi-objective optimization to assess the quality of a set of solutions. It provides a quantitative measure of the volume of the objective space that is dominated by a particular set of solutions, often referred to as a Pareto front. The concept of Hypervolume is particularly useful for comparing different Pareto fronts and evaluating the effectiveness of various algorithms in generating diverse and well-distributed solutions. Hypervolume serves as a valuable tool in multi-objective optimization, offering a quantitative measure of the quality and diversity of solutions on the Pareto front. It provides decision-makers with insights into the trade-offs among conflicting objectives and aids in algorithm selection and performance comparison. The strengths of Hypervolume lie in its quantitative measure, global assessment capabilities, and objective comparison. However, challenges include sensitivity to reference points and a potential lack of intuitiveness. Opportunities include integration with other metrics and algorithm benchmarking, while threats involve algorithm-specific performance variations and the complexity of interpretation.

The methodology presented by Zitzler et al. [129] focuses on designing quality measures for approximations of the Pareto-optimal set, crucial for assessing performance and developing multi-objective optimizers. They highlight the Hypervolume measure for its desirability, noting its tendency to favor convex, inner portions of the objective space. To address this, they introduce a methodology for designing quality measures based on the Hypervolume measure, effectively accommodating various decision-maker preferences. Guerreiro et al. [104] delve into the significance of the Hypervolume indicator as a widely used metric for assessing stochastic multi-objective optimizers and guiding evolutionary algorithms. They emphasize its strong theoretical properties, particularly its strict monotonicity concerning set dominance. The study extensively explores the computation of Hypervolume-related problems, providing insights into their interrelations and evaluating primary algorithms based on computational efficiency. Blonder et al. [130] highlight the pivotal role of the Hutchinsonian Hypervolume [22] across ecological and evolutionary studies but note challenges with existing methods in handling high-dimensional or holey datasets. They introduce a novel multivariate kernel density estimation method to overcome these hurdles, demonstrating its performance through comparative analysis. This practical solution enables the quantification of high-dimensional ecological Hypervolumes, complementing theoretical discussions with a concrete computational approach. Collectively, these methodologies contribute to advancing our understanding and utilization of Hypervolume-based metrics in multi-objective optimization and ecological studies. They cover conceptual, computational, and practical aspects, offering a comprehensive approach to address.

3.5 Repeatability analysis

Repeatability analysis, often referred to as test-retest reliability, is a statistical method used to assess the consistency or stability of measurements or experimental results over repeated trials. This type of analysis is crucial in various fields, including psychology, medicine, engineering, and social sciences, to ensure that measurements or observations are reliable and can be trusted for making inferences or decisions. Repeatability analysis is a valuable tool for assessing the consistency of measurements or experimental results, providing insights into the reliability of data and the robustness of measurement instruments or experimental procedures. It is valuable for benchmarking, identifying trends, and optimizing processes. However, limitations include potential sensitivity to initial conditions and the need for careful interpretation of variability. Opportunities lie in process optimization and quality assurance, while threats involve the risk of misinterpreting variability and resource-intensive implementation [23].

3.6 Sensitivity Analysis

Sensitivity analysis is a systematic study conducted to evaluate how the variation or uncertainty in the input parameters of a system or model influences the output or outcomes. It is a crucial tool in decision-making processes, risk assessment, and the optimization of models across various domains, including finance, engineering, environmental science, and healthcare. The primary goal is to identify which input parameters have the most significant impact on the model's results and to understand the robustness of the model and reliability. Sensitivity analysis is a versatile and powerful tool that provides valuable insights into the behavior of models and systems under different scenarios. It is widely used to enhance decision-making processes, improve model reliability, and manage uncertainties in complex systems. Sensitivity analysis is a powerful tool for identifying critical factors, assessing risk, and guiding optimization efforts. It provides opportunities for scenario planning and model improvement. However, its dependency on assumptions and limitations in accounting for unknown factors pose weaknesses. The threats involve the risk of overlooking interactions and uncertainties in input data.

In their respective studies, both Seo et al. [24] and Yang et al. [25] address optimization challenges in electromagnetic analysis and design methodologies. Seo et al. [24]focus on design sensitivity in the context of the multi-objective benchmark testing electromagnetic analysis methods (TEAMs) problem. They employ the material derivative concept of shape sensitivity analysis and a gradient-based algorithm, replacing original objective functions with definite integral forms to handle minimax problems. Their approach is validated through comparison with finite-difference numerical sensitivities and single-objective optimization results, demonstrating the validity of modified objectives for the benchmark problem.

On the other hand, Yang et al. [25] propose a novel implementation of sensitivity analysis alongside an enhanced Tabu search algorithm for optimal design in electromagnetic devices using the finite-element method (FEM). They introduce a direct sensitivity formulation, facilitating calculation of sensitivity versus design variables from FEM results, which is expected to be well-received by industrial users due to its efficiency. Additionally, sensitivity analysis guides search performance during optimization, with further enhancements such as a transition criterion for different current states and the use of a Tabu list with a binary space partitioning tree to accelerate the discovery of the global optimum. They provide optimal design examples to illustrate the effectiveness and advantages of their algorithm in practical engineering applications.

3.7 The Kruskal-Wallis test

The Kruskal-Wallis test is a non-parametric statistical test used to determine whether there are statistically significant differences between the medians of two or more independent groups. It is an extension of the Mann-Whitney U test, which is used for comparing two independent groups. The Kruskal-Wallis test is applicable when the data do not meet the assumptions required for parametric tests, such as normality or homogeneity of variances. Instead of analyzing the raw data values, the Kruskal-Wallis test ranks the data from all groups combined and compares the distributions of these ranks across the different groups.

To conduct the Kruskal-Wallis test, we follow a series of steps to analyze the data and determine if there are significant differences between the central tendencies of multiple groups.

First, we start by ranking the data. This involves combining the data from all groups and arranging the values from lowest to highest. In cases where there are ties (i.e., two or more data points with the same value), we assign them the average rank. This ranking process ensures that each data point is represented fairly within the combined dataset. Next, we proceed to calculate the rank sums for each group. By summing the ranks of the data points within each group, we obtain a measure of the central tendency for that particular group. This step provides insight into the overall distribution of values within each group and helps us understand their relative positions compared to one another. With the rank sums calculated, we move on to the calculation of the test statistic (H). This statistic, known as H, is computed based on the rank sums and the sample sizes of the groups. Essentially, H measures the degree of difference between the groups' rank sums. Larger values of H indicate greater disparities in the central tendencies among the groups. Once we have calculated H, we proceed to determine the critical value or p-value. This involves comparing the calculated test statistic (H) to a critical value from the Kruskal-Wallis distribution or converting it into a p-value. If the calculated test statistic exceeds the critical value or if the p-value is less than the chosen significance level (typically $\alpha = 0.05$), then there is evidence to reject the null hypothesis. This suggests that at least one group differs significantly from the others in terms of their central tendencies. Finally, we arrive at the interpretation stage. If the null hypothesis is rejected, it signifies that there are significant differences between the central tendencies of the groups. However, the Kruskal-Wallis test does not specify which specific groups exhibit these differences. To identify such pairwise distinctions, additional post-hoc tests may be necessary.

The Kruskal-Wallis test provides a rigorous framework for assessing differences in central tendencies across multiple groups. By following these steps, researchers can gain valuable insights into the variability within their data and make informed decisions about group comparisons. The Kruskal-Wallis test is a non-parametric alternative to the analysis of variance (ANOVA) for comparing the central tendencies of multiple groups. It is robust against violations of normality and homogeneity assumptions and is widely used in situations where these assumptions cannot be met.

4 Findings and Discussion

The findings and discussion chapter in "Multi-objective Optimization and GIS to Improve Climate Change Induced Disaster Risk Management in Africa" presents a comprehensive analysis of the research outcomes and their implications for enhancing disaster risk management in the face of climate change challenges across the African continent. Through the integration of multi-objective optimization techniques and Geographic Information Systems (GIS), this study aims to address the complex and dynamic nature of climate change-induced disasters, providing valuable insights for policymakers, practitioners, and stakeholders involved in disaster risk reduction efforts. In this chapter, we delve into the key findings derived from the application of multi-objective optimization algorithms within the GIS framework, highlighting the effectiveness of these approaches in identifying optimal strategies for mitigating disaster risks. Furthermore, we discuss the implications of these findings in the context of African countries, considering the unique socio-economic, environmental, and institutional contexts prevalent in the region. Through a critical examination of the results, we explore potential pathways for integrating these innovative approaches into existing disaster risk management practices, fostering resilience and sustainability in the face of escalating climate-related challenges. This chapter serves as a platform for dialogue and knowledge exchange, facilitating the development of evidence-based policies and strategies to safeguard communities and ecosystems vulnerable to climate change-induced disasters across Africa.

This thesis cover evacuation planning, urban land use allocation, and site selection for renewable energies. The first two papers deal with evacuation planning; the third paper encompasses urban land use allocation, and the fourth paper cover the site selection for renewable energies.

Multi-Objective Optimization (MOO) stands as a linchpin in urban planning, fostering sustainable development and bolstering urban resilience across various domains. In urban evacuation planning, MOO solutions are instrumental in Optimal Route Planning, where they discern evacuation routes that minimize time, maximize evacuee numbers, and mitigate shelter congestion. Similarly, in Urban land use allocation, MOO facilitates the Balancing of Land Use Objectives, optimizing space allocation amidst conflicting aims like maximizing residential area while minimizing environmental impact. Moreover, in urban site selection for renewable energies, MOO guides Optimizing Resource Utilization, aiding in selecting sites that maximize energy output while minimizing environmental impact. Across these domains, MOO offers a comprehensive decision-making approach, adeptly considering multiple conflicting objectives concurrently. This enables urban planners to make informed choices, striking a balance between various factors and fostering the creation of resilient, sustainable, and livable urban environments.

4.1 Comparison of metaheuristic algorithms (AMOSA, MOABC, NSGA-II, and MSPSO)

It is often recommended to conduct experiments and comparisons using benchmark problems or a representative subset of actual problem to observe how each algorithm performs under specific conditions [142, 143]. Selecting the best metaheuristic among AMOSA, MOABC, MSPSO, and NSGA-II depends on the specific characteristics of optimization problem, as well as the goals and requirements we have for the optimization process. Each metaheuristic has its strengths and weaknesses, and their performance can vary based on the nature of the problem at hand. In Paper I, we applied the metaheuristic algorithms in an emergency evacuation-planning problem, whose aim was to minimize both the total traveled evacuation distance and the total shelters overload. The efficiency of the four algorithms was evaluated in terms of convergence speed and execution time using the Kruskal–Wallis test. Convergence speed, measured by fitness variation, illustrates how the algorithm approaches the optimum solution over iterations, while execution time indicates the algorithm's running speed. The p-values obtained from the Kruskal–Wallis test indicated a highly significant difference in convergence speed (fitness variation rate) among the algorithms for both objective functions [118].

In Figure 10, a boxplot illustrates the average execution time across the four algorithms. A careful examination of the box plots reveals that NSGA-II emerges as the fastest algorithm in terms of execution time when compared to the remaining three. Zavala et al. [144], in a comparative study of metaheuristic algorithm evaluation, employed the multi-box plot chart to assess the quality of the Optimal Pareto front set. They utilized the multi-box plot chart of the Hypervolume indicator, which simultaneously evaluates both convergence and maximum spread. Additionally, they found that NSGA-II outperformed other algorithms in certain test problems.

This observation highlights a crucial insight: while AMOSA excels in terms of convergence speed, as evidenced by its superior fitness variation, it does not necessarily translate to the shortest execution time. The boxplot underscores that the algorithm with a high convergence speed may not always be the one with the swiftest execution time. Notably, the execution time is primarily influenced by factors such as the size of the population and the number of iterations. The research conducted by Ceja-Cruz et al. [145] demonstrates that AMOSA surpasses NSGA-II in both efficiency and the quality of the Pareto front. This highlights the dependence of the performance of an algorithm on the specific characteristics of the problem being addressed.

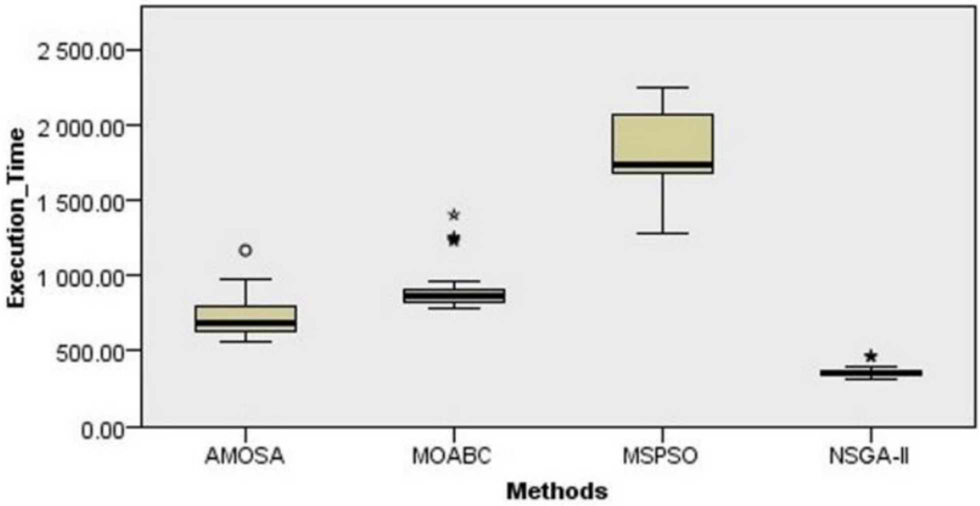


Figure 10: Variation mean of the execution time of 30 runs. The outlier symbols (°, *) represent extreme values in data set of each method.

These findings emphasize the importance of considering multiple performance metrics when evaluating the efficiency of optimization algorithms. The trade-off between convergence speed and execution time is a critical aspect in algorithm selection, and the specific requirements of the optimization problem should guide the choice of the most suitable algorithm for a given scenario. In addition, these findings indicate that the algorithms significantly differ in their convergence speeds, particularly when optimizing the capacity function. The observed distinctions underscore the importance of selecting an algorithm that aligns with the specific characteristics and requirements of the optimization problem at hand. The convergence speed and execution time analyses provide valuable insights into the relative performance of the algorithms, aiding in informed decision-making during the optimization process.

Studies showed that AMOSA is effective for multi-objective optimization problems, especially in continuous spaces. It can handle complex and non-linear objective functions [146, 147]. MOABC is inspired by the foraging behavior of bees and is efficient for multi-objective optimization with good exploration capabilities [148, 149]. Both AMOSA and MOABC are sensitive to parameter settings, and the performance may depend on the problem characteristics [150, 151]. MSPSO is efficient for continuous and discrete optimization problems and provides a good balance between exploration and exploitation. NSGA-II is widely used and well-established algorithm for multi-objective optimization [152].

4.2 IMOCS for Emergency Evacuation Planning

The Multi-Objective Cuckoo Search (MOCS) algorithm, like any optimization algorithm, comes with its own set of advantages in addressing multi-objective optimization problems and is inspired by efficient natural processes, it is important to be mindful of its sensitivity to parameters and potential challenges with certain problem types. Studies performed by Ab Wahab et al. and Weise et al. [153, 154] recommended to conduct thorough experimentation and benchmarking to assess its suitability for specific optimization tasks. In Paper II, we adapted the MOCS, originally tailored for continuous problems, to tackle integer problems, specifically discrete problems. To address these discrete problems, we employed an improved version of the MOCS, wherein necessary modifications were introduced to tailor it to the emergency evacuation-planning problem.

In this research, the implementation of the Multi-Objective Cuckoo Search (MOCS) utilized the Distributed Evolutionary Algorithm in Python (DEAP) framework. The primary objective was to determine the optimal number of shelters to be established. The mathematical model underlying this approach aims to minimize the number of shelters, and additional objectives pertain to path lengths as well as the risks associated with paths and shelter locations. In the context of a multi-objective problem, the absence of a single best solution is acknowledged, and the focus shifts to optimal solutions, specifically non-dominated (efficient, or Pareto-optimal) solutions. This paradigm allows for a comprehensive exploration of trade-offs among multiple conflicting objectives.

Figure 11 illustrates a comparison between the standard Multi-Objective Cuckoo Search (MOCS) depicted in dark red and the improved MOCS represented in steel blue. Both algorithms underwent 500 generations, revealing that the computation time (measured in minutes) for the improved MOCS is shorter than that of the standard MOCS. This efficiency gain in computation time can be attributed to the improved MOCS's approach of applying crossover and mutation operators to enhance the selected best solution, resulting in the generation of new individuals for the subsequent generation. Additionally, the utilization of Levi's flights for generating individuals contributes to the improved computation time. While, Meng et al.[155] introduced the Improved Multi-Objective Cuckoo Search (IMOCS) algorithm to address the limitations of the existing MOCS algorithm. They incorporated novel strategies such as constraint-based population initialization using the Individual Constraints and Group Constraints technique (ICGC) and dynamic adaptive probability (DAP) to enhance search efficiency and solution quality. Additionally, a Flock Search Strategy (FSS) was introduced to expedite convergence and improve the quality of non-dominated solutions. The study compared IMOCS with MOCS and NSGA-II. The results in Meng et al. [155] revealed that IMOCS outperformed other algorithms in terms of convergence speed, convergence property, and solution diversity. These findings highlight the efficacy of IMOCS in overcoming the limitations of existing algorithms and its suitability for solving complex optimization problems.

Our convergence analysis suggests that the IMOCS achieves better solutions more efficiently, further supporting its effectiveness in optimizing the underlying multi-objective problem. Similar results, about efficiency and effectiveness of the IMOCS, were found by Othman et al. [156].

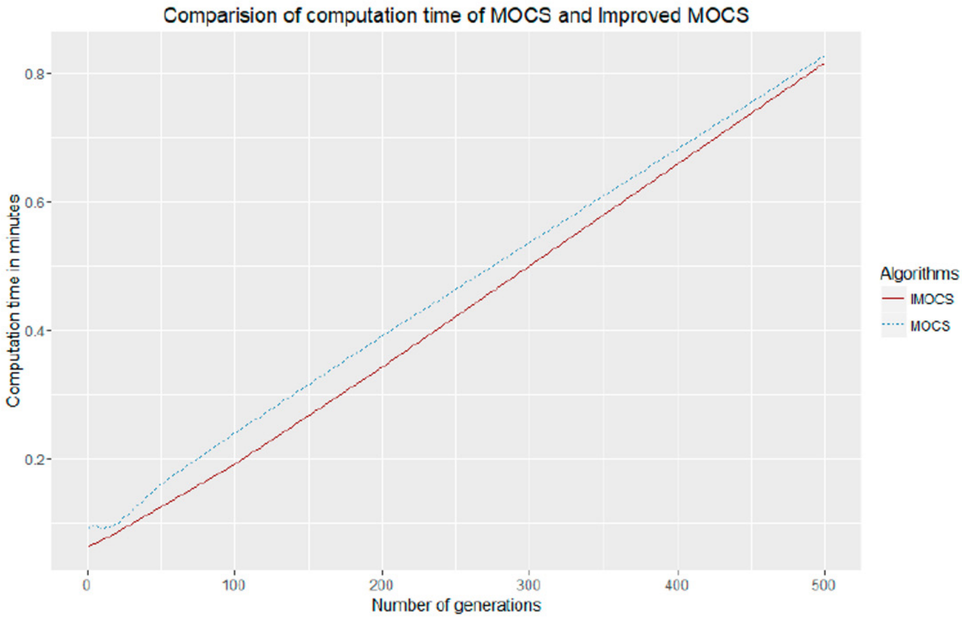


Figure 11: Comparison of computation time between the standard MOCS and the improved MOCS.

In this study, the Hypervolume indicator was used to evaluate the performance of the IMOCS against the standard Multi-Objective Cuckoo Search (MOCS) in terms of Pareto front quality [157]. As mentioned in the previous chapter, the Hypervolume indicator is a widely used measure that assesses the quality of an approximated Pareto front, by calculating the size of the space enclosed by all solutions on the Pareto front concerning a user-defined reference point. Liang et al. [158] devised an indicator-based MOCS algorithm by incorporating enhanced diversity enhancement (IDE) and adaptive scaling factor (ASF) techniques tailored for MOPs and carried out a similar use of the Hypervolume indicator. Their algorithm utilized Hypervolume as the indicator to enhance convergence and population dispersion. IDE strategically selects areas with significant Hypervolume to reconstruct the parent population, mitigating issues related to population diversity[158]. Comparative evaluations against state-of-the-art multi-objective evolutionary algorithms confirmed the efficacy and efficiency of the proposed approach.

For the Hypervolume indicator, only a reference point needs to be provided, and it measures the volume dominated by the given set of solutions concerning this reference point [159]. Unlike other performance indicators that require a target set, Hypervolume only needs a reference point. The comparison of Hypervolume values, as presented in Figure 12, indicates that the Improved MOCS algorithm achieves a higher Hypervolume than the standard MOCS algorithm. This result suggests that IMOCS exhibits superior performance compared to MOCS, particularly in terms of Pareto front quality.

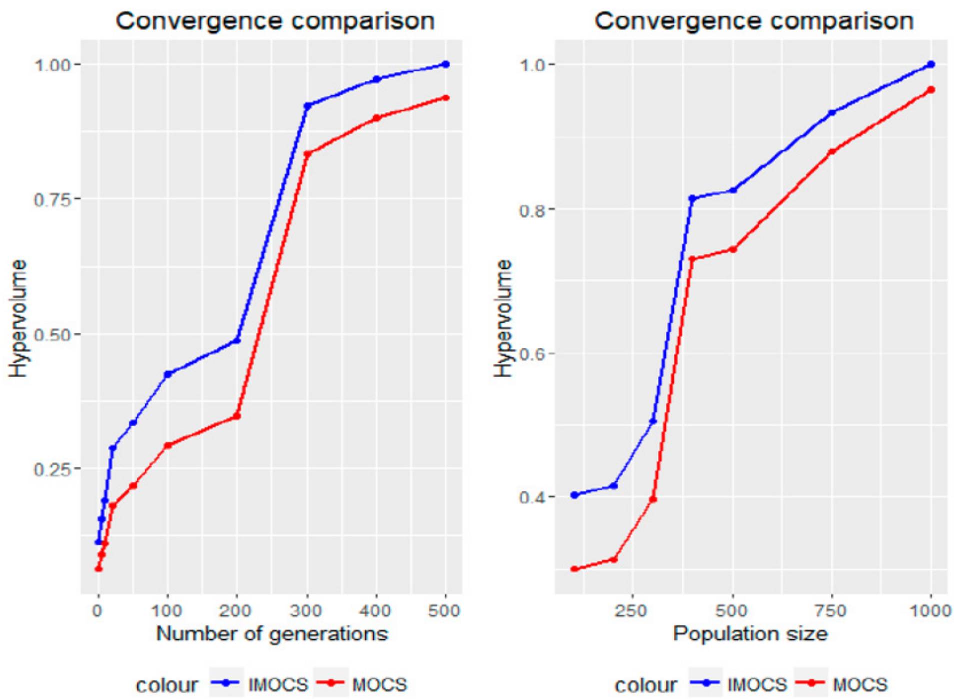


Figure 12: Hypervolume convergence analysis, (left) for different number of generations, (right) for different population size. The blue plot depicts the Hypervolume of the IMOCS, while the red plot illustrates the Hypervolume of the Standard MOCS. Both plots demonstrate an increase corresponding to the growth in both the number of generations and the population size.

4.3 Improved NSGA-III for Urban Land Use Allocation

Utilizing an improved Non-dominated Sorting Genetic Algorithm III (NSGA-III) for Land Use Allocation (LUA) in a Multi-Objective Optimization (MOO) framework can provide a robust and informed approach to urban planning, taking into account the diverse and often conflicting objectives associated with land use in urban areas. A MOO approach employing an improved version of NSGA-III for LUA involves optimizing multiple conflicting objectives to make informed decisions about the allocation of land for different uses in an urban area.

In this thesis, we employed the Python Multi-Objective Optimization (PYMOO) framework to implement NSGA-III. PYMOO, a dedicated Python library for multi-objective optimization, offers comprehensive tools for visualizing the Pareto front and assessing trade-offs between objectives using matplotlib [160] and other plotting libraries. The outcome of our analysis involves determining the optimal allocation of the eight land use types in the study area by considering five objective functions: minimizing carbon emissions, maximizing population capacity, maximizing total income, ensuring high accessibility, and achieving high compactness. It is important to note that in multi-objective optimization problems, there is not a singular "best" solution; instead, we obtain a set of optimal solutions. An optimal solution is characterized as non-dominated (efficient or Pareto-optimal), where no other solution is superior in all objectives. Each run in the multi-objective optimization process yields a non-dominated set of solutions, facilitating comparisons between different sets.

For the analysis of convergence, we utilized the widely recognized performance indicator known as Hypervolume [108]. So, to conduct the analysis, we employed the Hypervolume package from R software. This package constructs the Hypervolume using various methods, including box-kernel density estimation, Gaussian kernel density estimation, or one-class support vector machine, after performing error-checking on input data [161]. Additionally, the dominated portion of the objective space can be utilized to gauge the quality of non-dominated solutions [162].

Figure 13 presents a comparison between the standard and improved NSGA-III in terms of the Hypervolume indicator. The Hypervolume values are normalized for both the standard and improved NSGA-III to ensure a range between 0 and 1 for a more effective comparison. The observation from the figure reveals that the Pareto front generated by the improved NSGA-III exhibits a higher Hypervolume, indicating superior trade-offs among conflicting objectives. Moreover, the quality of solutions consistently improves with the progression of generation numbers, and the improved NSGA-III maintains its superiority in terms of solution quality in the land-use allocation problem studied here.

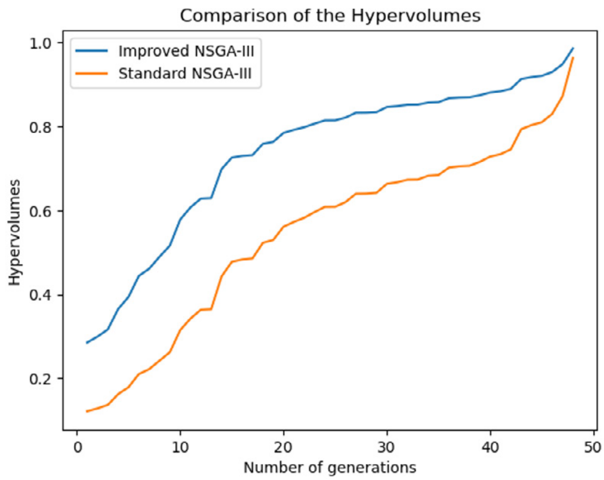


Figure 13: Comparison of the Hypervolume indicators of the improved NSGA-III (Blue) and standard NSGA-III (Orange) in a land-use allocation problem.

In this study we also performed a performance comparison between the standard NSGA-III (represented by the orange line) and the improved NSGA-III (depicted by the blue line) concerning computation time, as illustrated in Figure 14. Both algorithms were executed for 500 generations. It is evident from the graph that the computation time (measured in hours) for the improved NSGA-III is consistently lower than that of the standard NSGA-III across different generations. Furthermore, as the number of generations increases, the growth rate of computation time for the improved NSGA-III is notably slower compared to the standard NSGA-III.

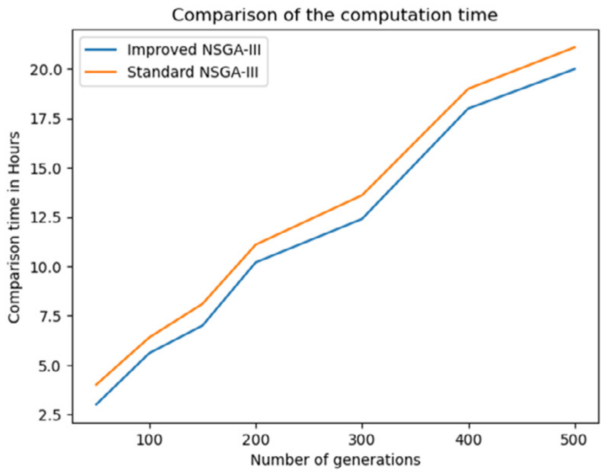


Figure 14: Comparison between the improved (blue) and standard (orange) NSGAIII in terms of computation time in a land-use allocation problem.

The land use allocation study of the Kamavota district utilizes an optimization approach to generate planning support scenarios [163]. This approach allows for quantitative trade-off analysis and accommodates diverse user preferences, resulting in various optimized solutions. The optimal solution reflects a balanced consideration of multiple objectives, contributing to an inclusive and multi-modal accessible district with transformed transportation options. Figure 15 shows the optimal land use allocation according to the objective function preference.

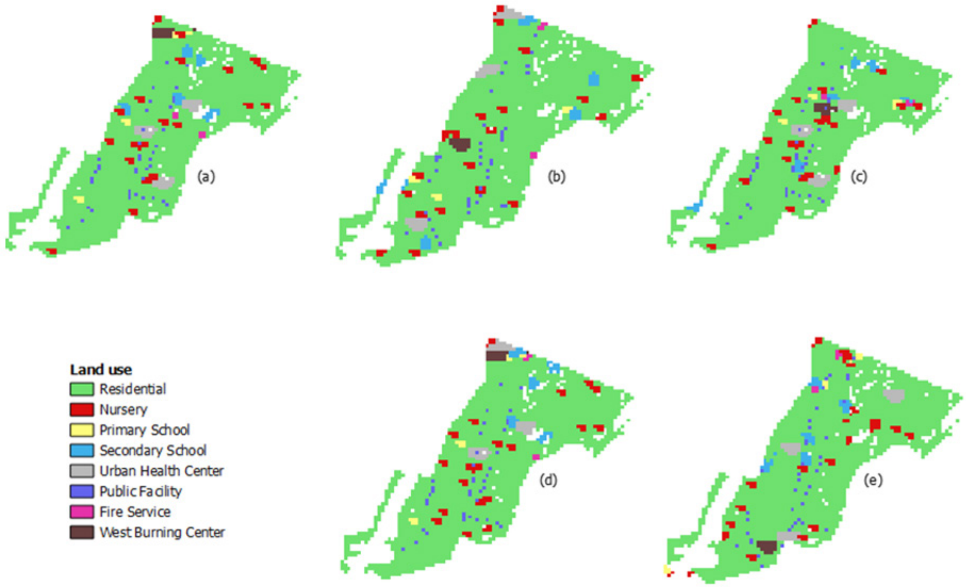


Figure 15: Optimal maps in a land-use allocation problem preferring one of the objective functions.. (a) preferred economic income, (b) preferred carbon emission, (c) preferred accessibility, (d) preferred space syntax integration, and (e) preferred compactness.

4.4 NSGA-II solar farms site selection

In optimization problems of site selection for power solar farms using multiple objectives, the Pareto front comprises solutions that are not dominated by any other solution in all objectives. Dominance occurs when a solution is equal to or better than another solution in all objectives and strictly better in at least one objective. In site selection for power solar farms, the objective is to find solutions that strike a balance between conflicting objectives, as enhancing one may lead to a decline in another [59]. Before, the analysis of the optimal pareto front set, it is important to note that pymoo framework by default, only deals with minimization problems. So, the maximization problems to be solved using pymoo need to be transformed to minimization problem. In this context, e.g. Figure 16(S16) provides the lowest values of the total direct solar radiation and the total aspect. This apparent contradiction is due to the fact of solving the site selection solar power farms problem as minimization problem while some objective functions are to be maximized.

For example, if decision-makers prioritize maximizing total solar radiation, they may opt for the solution depicted in Figure 16(S3), which offers the highest value for total solar radiation while maintaining an optimal total aspect value. Conversely, if minimizing total distance and total slope are the main concerns, the solution illustrated in Figure 16(S4) might be favored, as it prioritizes these objectives.

Solutions that exhibit similar preference levels across all objective functions can also be identified, such as those shown in Figure 16(S41), Figure 16(S43), and Figure 16(S49). These solutions allocate nearly equal importance to each objective function and may be suitable when decision-makers have balanced priorities across all objectives. Figure 16 displays the optimal Pareto front using Petal diagrams, also known as radial plots or flower diagrams. These diagrams visualize values in a circular manner, featuring "petals" or "arms" radiating from a central point. Each petal corresponds to a different category or variable, with its length or area proportionate to the value of that category. Petal diagrams are effective for presenting multivariate data, allowing easy comparison of variable magnitudes across different categories. The circular layout provides a comprehensive view of the data, enabling the observation of patterns by examining petal lengths or areas. Widely used in diverse fields, such as biology and market research, Petal diagrams offer a visually engaging means to convey complex information and relationships [164].

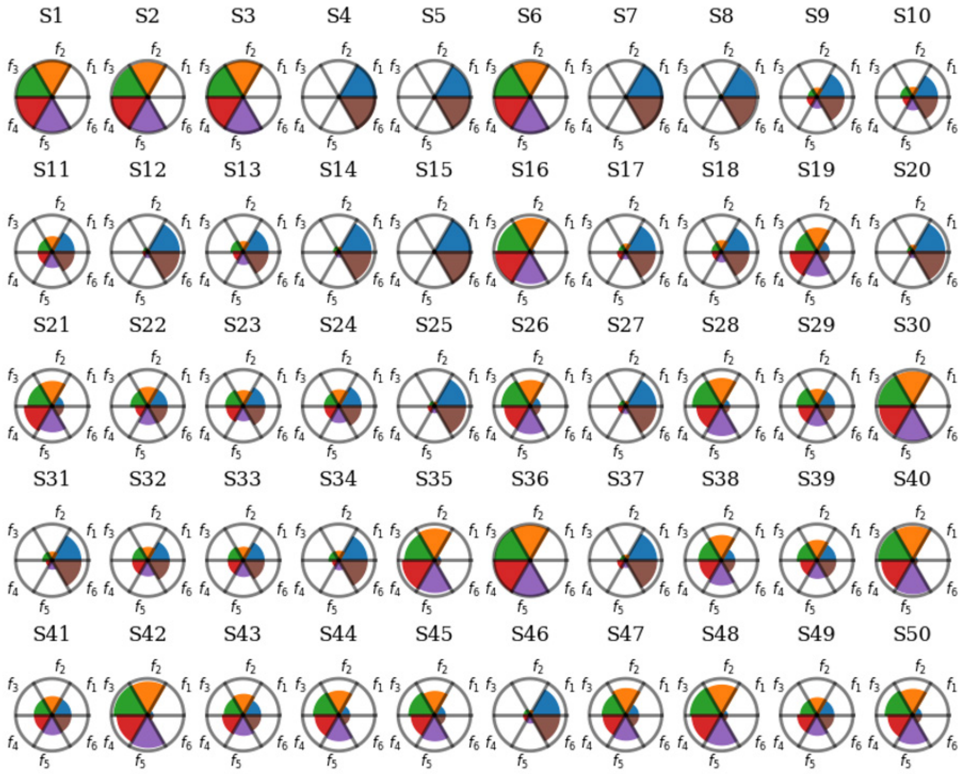


Figure 16: Optimal Pareto front set for site selection for solar power farms problem, where f_1 – total solar radiation, f_2 – Total distance to grid, f_3 – total distance to street, f_4 – total distance to city, f_5 – total slope, and f_6 – total aspect. $Si,\ i = 1,2,3, \dots ,50$ represents each optimal solution into optimal Pareto front set. Each colour represents the normalized value of the corresponding objective function.

Similar studies represented the optimal Pareto front using Radar plots and Parallel Coordinate plots. El-Shorbagy et al. [165] utilized radar plots, also known as spider charts or web charts, to visualize multi-objective optimization problems and the Pareto front. These plots represent optimal solutions, with each axis corresponding to an objective, and points closer to the outer edge indicating better performance. Bi and Wang [166] employed Parallel Coordinate Plots to represent the Optimal Pareto front, offering another effective visualization method. These plots analyze the distribution of solutions across different objective ranges, providing insights into solution density and trade-offs.

Optimal solution map offers the most effective spatial arrangement for locating solar power farms. However, in multi-objective optimization, the optimal solution, known as Pareto Optimal, entails conflicting objectives. This implies that prioritizing certain objective functions comes at the expense of others. In Figure 17, the map illustrates the spatial distribution of optimal locations for solar power farms.

In this scenario, we assume that decision-makers prioritize maximizing total solar radiation while optimizing the overall aspect, as depicted in optimal Pareto solution A. This prioritization entails less emphasis on total distance to the grid, street, residential infrastructures, and slope. The map not only highlights the optimal locations for constructing solar power farms but also incorporates other land use resources such as protected areas. These protected areas designate land not designated for solar farms.

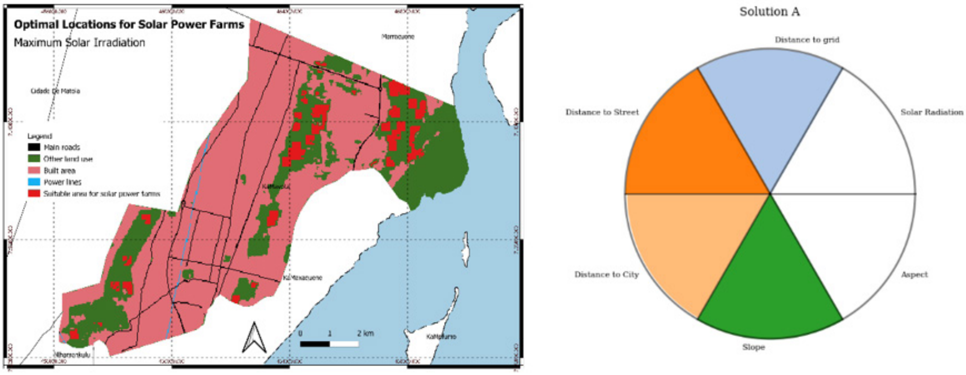


Figure 17: Optimal map and its corresponding preferred objective. The red color in the map represents the optimal location for solar power farms preferring the maximum total solar irradiation. The Petal diagram represents the optimal solution giving importance to the total solar radiation and the total Aspect.

Figure 18 showcases a map delineating the optimal placement of sites for solar power farms. Each red area denotes a location identified as optimal for such farms. These selections are made according to optimization criteria that prioritize minimizing the total distance to the grid, street, residential areas, and slope. The red pixels representing optimal Pareto solution B indicate a reduced emphasis on maximizing total solar radiation and aspect compared to other considerations. This suggests that while solar radiation and aspect are taken into account, they are not given primary importance in the decision-making process. In summary, the map visually presents the most suitable locations for solar power farms, factoring in various elements such as proximity to infrastructure and topographical characteristics, as delineated in optimal Pareto solution B.

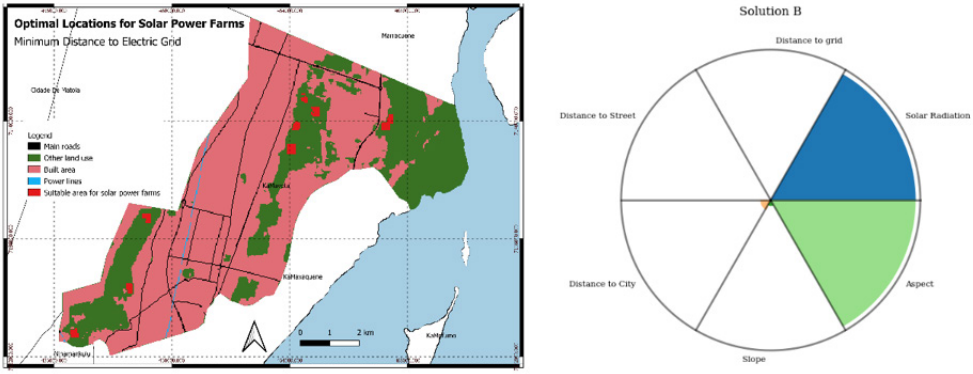


Figure 18: Optimal map and its corresponding preferred objective. The red color in the map represents the optimal location for solar power farms preferring the minimum total distance to electric grid. The Petal diagram represents the optimal solution giving high importance to the total distance to the electric grid and to the total slope.

5 Conclusions

In this chapter, we summarize and provide the conclusions of the thesis and they are divided into three sections. The first section covers conclusions regarding practical contribution to society or humanity. The second section encompasses contributions to the science in the field of evacuation planning and urban planning with methodological developments and conceptual contributions, and the last section, of the conclusions, covers the future direction of the research.

In Paper I, the first research objective has been satisfied by assessing the efficacy of four standard metaheuristic algorithms to solve an optimization problem with two objective functions, for evacuation planning. This research was further developed in Paper II by incorporating three objective functions, and solving the problem using an enhanced MOCS algorithm. So paper 2 has contribution in responding both objectives 1 and 2. Paper III proposes an approach for determining a sustainable and resilient land use allocation, using an improved NSGA-III algorithm that is in line with objective 3. Paper 4 proposes an improved NSGA-II for optimum solar farms site selection that satisfies objective 4.

5.1 Practical Contributions

The metaheuristic methods, including the presented ones, are designed to discover not a 'single perfect solution' but a set of 'good enough' solutions where all satisfy a trade-off between the desired objectives. It is important for decision-makers to recognize this characteristic when assessing the benefits and limitations of these techniques. The improved Multi-Objective Cuckoo Search (IMOCS) algorithm emerged as a more promising solution compared to the standard Multi-Objective Cuckoo Search (MOCS) for problem instances of varying sizes of the initial population or number of generations. Notably, IMOCS demonstrated a favorable balance between solution quality and computational time.

The research findings offer valuable insights into optimizing emergency evacuation routes and shelter utilization while minimizing the total traveling risk for evacuees. The proposed mathematical model and heuristic algorithms can serve as practical tools for the National Institute for Disaster Management, enhancing the efficiency of emergency evacuation routes and shelters, mitigating risks during evacuations, and reducing travel distances. These decision support tools are anticipated to enhance the overall effectiveness of emergency evacuation processes, ensuring the safety of evacuees, including vulnerable population groups.

The focus of Paper III is to develop a Multi-Objective Optimization (MOO) model aimed at obtaining optimized solutions for land use allocation, particularly in response to climate change and natural disasters. The study considers eight simplified land use types: residential, nursery, primary school, secondary school, urban health center, public facility, fire service, and waste burning center. The MOO approach involves defining fitness functions based on five key objectives: Maximization of economic objective, Minimization of carbon emission, Maximization of the accessibility objective, Maximization of space syntax integration, and Maximization of compactness objective.

While the proposed approach has been applied to Maputo as a case study, it holds potential for application in other developing cities in Africa and beyond, provided the necessary data are available. This research highlights the significance of optimizing land resource layouts for achieving sustainability and resilience in the face of climate change and natural disasters.

5.2 Methodological development and conceptual contributions

This subsection provides a detailed methodological development and succinct contribution of the thesis from the provided four studies.

5.2.1 Methodological Development

The Paper I aimed to compare the performance of four multi-objective optimization algorithms (AMOSA, MOABC, MSPSO, NSGA-II) in the context of evacuation planning, focusing on minimizing the accumulated distance from high-risk zones to shelters and the total capacity overload cost of shelters. The optimization criteria were to achieve higher minimum fitness values for both capacity and distance, indicating better alternatives for assigning people to shelters. The evaluation of algorithm performance revealed consistent optimization results, with no evidence of local minimum entrapment.

In terms of convergence speed, the fitness variation analysis showed that AMOSA and NSGA-II, followed by MOABC, exhibited faster and smoother convergence towards optimal solutions. This observation emphasizes the competence of NSGA-II, a widely used algorithm in the literature, and highlights the effectiveness of

AMOSA and MOABC in solving multi-objective optimization problems, including evacuation planning.

The Improved Multi-Objective Cuckoo Search (IMOCS) algorithm emerged as a more promising solution for problem instances of any size when compared to the standard MOCS. Additionally, the IMOCS algorithm demonstrated an effective tradeoff between solution quality and computational time. The research findings offer valuable insights into emergency evacuation routes, shelter utilization, and the overall traveling risk for evacuees along these routes. The proposed mathematical model and heuristic algorithms are positioned as practical tools for the National Institute for Disaster Management to enhance the utilization of emergency evacuation routes and shelters. These tools aim to mitigate risks on roadways during emergencies, reduce travel distance, and ultimately improve the safety of evacuees, including vulnerable population groups.

The Paper III focuses on the critical task of optimizing land resource allocation for enhanced sustainability and resilience, particularly in response to climate change and natural disasters within land use planning. A Multi-Objective Optimization (MOO) model is developed to derive optimized solutions, considering eight simplified land use types: residential, nursery, primary school, secondary school, urban health center, public facility, fire service, and waste burning center. The fitness functions are specified through a multi-objective approach, addressing five key objectives aligned with the principles of sustainability and resilience: Maximization of economic objectives, Minimization of carbon emissions, Maximization of accessibility, Maximization of space syntax integration, and Maximization of compactness. The study further contributes to the improvement of the standard Non-dominated Sorting Genetic Algorithm III (NSGA-III) and utilizes the enhanced NSGA-III to solve land-use allocation problems specifically in the Kamavota district, Maputo city, Mozambique. The evaluation involves convergence analysis and performance comparison, revealing that the improved NSGA-III outperforms the standard NSGA-III in terms of effectiveness. Although the proposed approach is demonstrated through a case study in Maputo, its applicability extends to land-use planning in other cities across Africa and beyond, particularly those in developmental phases, provided the necessary data are available.

The Paper IV addresses the critical decision of site selection for a solar power farm, emphasizing its profound impact on project performance, efficiency, and overall success. The methodological approach involves a comprehensive assessment of various factors, including energy production, environmental responsibility, regulatory compliance, and economic viability. Multi-objective optimization (MOO) is employed as a suitable framework, focusing on simultaneously optimizing multiple conflicting objectives. Unlike traditional optimization problems with a single objective, MOO aims to find a set of solutions that represent a trade-off among diverse criteria. The application of MOO is justified by the need to balance multiple objectives related to site strengths and weaknesses, such as improving solar radiation and enhancing consumer satisfaction. Quantitative measures are assigned to each objective, facilitating optimization through metrics like minimum total distance to road, minimum total distance to infrastructures, or minimum total slope. Multi-objective optimization techniques are utilized to find solutions that simultaneously enhance strengths and mitigate weaknesses, even involving trade-offs. The study presents a multi-objective mathematical model for optimal solar power farm location selection, employing the non-dominated sorting genetic algorithm (NSGA-II) to determine optimal solutions. The Hypervolume indicator is employed to evaluate solution quality, highlighting the impact of different crossover operators on performance and solution quality trade-offs. The proposed model is deemed suitable for small-scale applications and can be extended to various districts or counties.

5.2.2 Conceptual Contributions

The conceptual contribution lies in demonstrating the competence of multiple multi-objective optimization algorithms, namely AMOSA, MOABC, MSPSO, and NSGA-II, in addressing evacuation planning problems. The study showcases their ability to consistently generate optimal solutions without getting trapped in local minima. This provides insights into the versatility of these algorithms and their applicability to complex spatial problems, contributing to the broader understanding of their performance across various optimization scenarios.

The conceptual contribution is centered around the effectiveness of the IMOCS algorithm in addressing emergency evacuation planning, showcasing its superior performance in terms of solution quality and computational time. The study provides practical implications for disaster management authorities by offering decision support tools that can significantly enhance the emergency evacuation process. The emphasis on safety, risk reduction, and efficiency contributes to the overall effectiveness of emergency response strategies. Additionally, the research suggests potential future directions, such as a comparative study of IMOCS with other metaheuristic algorithms, to further expand the understanding of its capabilities and limitations.

The conceptual contribution lies in the development of a robust MOO model tailored for optimizing land use allocation, with a specific focus on sustainability and resilience in the face of climate change and natural disasters. The articulation of key fitness objectives reflects a comprehensive consideration of economic, environmental, and accessibility factors. The improvement of the NSGA-III algorithm enhances its performance in solving complex land-use allocation challenges. The applicability of the study to Maputo serves as a demonstration, opening avenues for its potential application to other cities undergoing development when relevant data become accessible. Overall, the study provides a conceptual framework and a methodological approach that can guide efficient and sustainable land-use planning in diverse urban contexts.

The conceptual contribution lies in the development of a robust multi-objective optimization framework for solar power farm site selection, considering a holistic set of criteria. The study highlights the importance of simultaneously addressing conflicting objectives to achieve a balanced and optimal solution. The integration of quantitative measures and multi-objective optimization techniques provides a systematic approach to decision-making in solar power farm planning. The findings contribute to understanding the impact of different crossover operators on solution quality and performance trade-offs within the NSGA-II framework. Overall, the study provides a valuable model applicable to small-scale scenarios and adaptable to diverse geographic locations.

5.3 Future research

Addressing the challenges posed by the increase in natural disasters due to extreme weather events and mitigating the impacts of climate change are crucial tasks that require multidisciplinary efforts, including advancements in science, technology, and policy. Efficient algorithms play a significant role in providing timely and accurate information for decision-makers to take proactive measures. Collaboration between scientists, engineers, data scientists, policymakers, and other stakeholders is essential for addressing the challenges posed by climate change and extreme weather events. The development and application of advanced algorithms can significantly contribute to improving our understanding of climate dynamics and enhancing our ability to respond effectively to climate-related challenges in real-time.

The hybridization of metaheuristic algorithms is a significant and evolving area in optimization and decision support systems. This approach involves combining two or more metaheuristic algorithms or integrating metaheuristics with other optimization techniques to enhance their performance and robustness. This hybridization is particularly relevant in addressing complex problems related to land use planning, emergency evacuation planning, and various other knowledge domains. Examples of hybrid metaheuristic approaches include combining genetic algorithms with simulated annealing, particle swarm optimization with ant colony optimization, or differential evolution with tabu search. These combinations leverage the strengths of individual algorithms to create powerful hybrid frameworks suitable for addressing the challenges posed by complex optimization problems in diverse domains.

References

1. Sorensen K, Sevaux M, Glover F A History of Metaheuristics.
2. Baez JE, Caruso G, Niu C Extreme Weather and Poverty Risk: Evidence from Multiple Shocks in Mozambique. 34
3. Philander SG (ed) (2008) Encyclopedia of global warming and climate change. SAGE, Los Angeles
4. IFRC (2022) Revised Operational Strategy. Mozambique, Africa 2021-22 Floods and Cyclones.
5. Kuhn HW, Tucker AW NONLINEAR PROGRAMMING. NONLINEAR PROGRAMMING 12
6. Burke EK, Kendall G (eds) (2014) Search Methodologies. https://doi.org/10.1007/978-1-4614-6940-7
7. Vanderbei RJ (2014) Linear Programming: Foundations and Extensions. https://doi.org/10.1007/978-1-4614-7630-6
8. Karlof JK (ed) (2006) Integer programming: theory and practice. Taylor&Francis, Boca Raton, FL
9. (2009) Logic and Integer Programming. https://doi.org/10.1007/978-0-387-92280-5
10. Land AH, Doig AG (1960) An Automatic Method of Solving Discrete Programming Problems. Econometrica 28:497
11. Morrison DR, Jacobson SH, Sauppe JJ, Sewell EC (2016) Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning. Discrete Optimization 19:79–102
12. Bellman R (1954) Some Problems in the Theory of Dynamic Programming. Econometrica 22:37
13. Bellman R (1957) A Markovian Decision Process. Indiana Univ Math J 6:679–684
14. Rust J New Palgrave Dictionary of Economics. 39
15. Sniedovich M Dynamic Programming. 616
16. Bellman R (1966) Dynamic Programming. Science 153:34–37
17. Johnsonbaugh R (2018) Discrete mathematics, Eighth edition. Pearson, New York, NY
18. Ford LR, Fulkerson DR (1957) A Simple Algorithm for Finding Maximal Network Flows and an Application to the Hitchcock Problem. Can j math 9:210–218
19. Ahuja RK, Magnanti TL, Orlin JB (1993) Network Flow. Theory, Algorithms, andApplications.

20. Holland JH (1992) Genetic Algorithms. SCIENTIFIC AMERICAN 9
21. Srinivas N, Deb K (1995) Multiobjective function optimization using nondominated sorting genetic algorithms.
22. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Computat 6:182–197
23. Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. In: Schoenauer M, Deb K, Rudolph G, Yao X, Lutton E, Merelo JJ, Schwefel H-P (eds) Parallel Problem Solving from Nature PPSN VI. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 849–858
24. Deb K, Jain H (2014) An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. IEEE Trans Evol Computat 18:577–601
25. Storn R Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. DIFFERENTIAL EVOLUTION
26. Fogel DB (1994) An introduction to simulated evolutionary optimization. IEEE Trans Neural Netw 5:3–14
27. Lee KY, Yang FF (1998) Optimal reactive power planning using evolutionary algorithms: a comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming. IEEE Trans Power Syst 13:101–108
28. Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans Evol Computat 3:257–271
29. Arnold DV A Comparison of Evolution Strategies with Other Direct Search Methods in the Presence of Noise.
30. Ostfeld A (2011) Ant colony optimization: methods and applications. Intech, Rijeka, Croatia
31. Ding Q, Hu X, Sun L, Wang Y (2012) An improved ant colony optimization and its application to vehicle routing problem with time windows. Neurocomputing 98:101–107
32. Colorni A, Dorigo M, Maniezzo V Distributed Optimization by Ant Colonies. 6
33. Sotiropoulos DN, Tsihrintzis GA (2017) Machine Learning Paradigms. https://doi.org/10.1007/978-3-319-47194-5
34. Dasgupta D (ed) (1999) Artificial Immune Systems and Their Applications. https://doi.org/10.1007/978-3-642-59901-9
35. Mahmudova S (2021) Developing an algorithm for the application of Bayesian method to software using artificial immune systems. Soft Comput 25:11837–11843
36. Gendreau M, Potvin J-Y (eds) (2010) Handbook of Metaheuristics. https://doi.org/10.1007/978-1-4419-1665-5
37. Pereira G, Salvo P Particle Swarm Optimization. 8
38. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks. IEEE, Perth, WA, Australia, pp 1942–1948

39. Pham DT, Ghanbarzadeh A, Koc E, Otri S, Rahim S, Zaidi M Bee Algorithm – A Novel Approach to Function Optimisation.
40. Wang G-G, Deb S, Coelho L dos S (2015) Elephant Herding Optimization.
41. Yang X, Hossein Gandomi A (2012) Bat algorithm: a novel approach for global engineering optimization. Engineering Computations 29:464–483
42. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper Optimisation Algorithm: Theory and application. Advances in Engineering Software 105:30–47
43. Mirjalili SZ, Mirjalili S, Saremi S, Faris H, Aljarah I (2018) Grasshopper optimization algorithm for multi-objective optimization problems. Appl Intell 48:805–820
44. Yang X-S, Deb S (2009) Cuckoo Search with Levy Flights. World Congress on Nature and Biologically Inspired Computing (NaBIC) 210–214
45. Neyman J, Kac M (1962) *Mathematical Statistics and Probability*. Physics Today 15:64–68
46. Glover F, McMILLAN C Interactive decision software and computer graphics for architectural and space planning.
47. Glover F (1986) Future paths for integer programming and links to artificial intelligence. Computers & Operations Research 13:533–549
48. Glover F (1977) HEURISTICS FOR INTEGER PROGRAMMING USING SURROGATE CONSTRAINTS. Decision Sciences 8:156–166
49. Martí R, Laguna M, Glover F (2006) Principles of scatter search. European Journal of Operational Research 169:359–372
50. Laguna M, Martí R (eds) (2003) Scatter Search: Methodology and Implementations in C. https://doi.org/10.1007/978-1-4615-0337-8
51. Riahi V, Hakim Newton MA, Sattar A (2021) A scatter search algorithm for time-dependent prize-collecting arc routing problems. Computers & Operations Research 134:105392
52. (2022) Optimization by Simulated Annealing. 11
53. Eglese RW (1990) Simulated annealing: A tool for operational research. European Journal of Operational Research 46:271–281
54. Bertsimas D, Tsitsiklis J (1993) Simulated Annealing.
55. Matthias E (2005) Multicriteria optimization. Springer Science & Business Media
56. Deb K Multi-Objective Optimization Using Evolutionary Algorithms: An Introduction.
57. Gunantara N (2018) A review of multi-objective optimization: Methods and its applications. Cogent Engineering 5:1502242
58. Censor Y (1977) Pareto optimality in multiobjective problems. Appl Math Optim 4:41–59
59. Stiglitz JE (1981) Pareto Optimality and Competition. The Journal of Finance 36:235–251
60. Alexandre M, Michalak K, Silva TC, Rodrigues FA (2023) Efficiency-stability trade-off in financial systems: A multi-objective optimization approach. Physica A: Statistical Mechanics and its Applications 629:129213

61. El-Abbasy MS, Elazouni A, Zayed T (2020) Finance-based scheduling multi-objective optimization: Benchmarking of evolutionary algorithms. Automation in Construction 120:103392
62. Aslam T, Ng AHC Multi-objective optimization for supply chain management: A literature review and new development.
63. Grond MOW, Luong NH, Morren J, Slootweg JG (2012) Multi-objective optimization techniques and applications in electric power systems. In: 2012 47th International Universities Power Engineering Conference (UPEC). IEEE, Uxbridge, Middlesex, United Kingdom, pp 1–6
64. Koroneos C, Michailidis M, Moussiopoulos N (2004) Multi-objective optimization in energy systems: the case study of Lesvos Island, Greece. Renewable and Sustainable Energy Reviews 8:91–100
65. Wang Y, Chen C, Tao Y, Wen Z, Chen B, Zhang H (2019) A many-objective optimization of industrial environmental management using NSGA-III: A case of China's iron and steel industry. Applied Energy 242:46–56
66. Guillén-Gosálbez G (2011) A novel MILP-based objective reduction method for multi-objective optimization: Application to environmental problems. Computers & Chemical Engineering 35:1469–1477
67. Jimenez H, Mavris D (2017) Pareto-Optimal Aircraft Technology Study for Environmental Benefits with Multi-Objective Optimization. Journal of Aircraft 54:1860–1876
68. Arias-Montano A, Coello CAC, Mezura-Montes E (2012) Multiobjective Evolutionary Algorithms in Aeronautical and Aerospace Engineering. IEEE Trans Evol Computat 16:662–694
69. Li H, Gu J, Wang M, Zhao D, Li Z, Qiao A, Zhu B (2016) Multi-objective optimization of coronary stent using Kriging surrogate model. BioMed Eng OnLine 15:148
70. Chakraborty S, Mali K (2018) Application of Multiobjective Optimization Techniques in Biomedical Image Segmentation—A Study. In: Mandal JK, Mukhopadhyay S, Dutta P (eds) Multi-Objective Optimization. Springer Singapore, Singapore, pp 181–194
71. Pereira JLJ, Oliver GA, Francisco MB, Cunha SS, Gomes GF (2021) A Review of Multi-objective Optimization: Methods and Algorithms in Mechanical Engineering Problems. Arch Computat Methods Eng. https://doi.org/10.1007/s11831-021-09663-x
72. Cerda-Flores SC, Rojas-Punzo AA, Nápoles-Rivera F (2022) Applications of Multi-Objective Optimization to Industrial Processes: A Literature Review. Processes 10:133
73. Fei Z, Li B, Yang S, Xing C, Chen H, Hanzo L (2017) A Survey of Multi-Objective Optimization in Wireless Sensor Networks: Metrics, Algorithms, and Open Problems. IEEE Commun Surv Tutorials 19:550–586
74. Elmusrati M, El-Sallabi H, Koivo H (2008) Applications of Multi-Objective Optimization Techniques in Radio Resource Scheduling of Cellular Communication Systems. IEEE Trans Wireless Commun 7:343–353
75. Tolmidis ATh, Petrou L (2013) Multi-objective optimization for dynamic task allocation in a multi-robot system. Engineering Applications of Artificial Intelligence 26:1458–1468

76. Kouritem SA, Abouheaf MI, Nahas N, Hassan M (2022) A multi-objective optimization design of industrial robot arms. Alexandria Engineering Journal 61:12847–12867
77. Bish DR, Sherali HD, Hobeika AG (2014) Optimal evacuation planning using staging and routing. J Oper Res Soc 65:124–140
78. Baou E, Koutras VP, Zeimpekis V, Minis I (2018) Emergency evacuation planning in natural disasters under diverse population and fleet characteristics. J Humanit Logist Sup 8:447–476
79. Saadatseresht M, Mansourian A, Taleai M (2009) Evacuation planning using multiobjective evolutionary optimization approach. European Journal of Operational Research 198:305–314
80. Stepanov A, Smith JM (2009) Multi-objective evacuation routing in transportation networks. European Journal of Operational Research 198:435–446
81. Coutinho-Rodrigues J, Tralhao L, Alcada-Almeida L (2012) Solving a location-routing problem with a multiobjective approach: the design of urban evacuation plans. Journal of Transport Geography 22:206–218
82. Zhao X, Xu W, Ma Y, Hu F (2015) Scenario-Based Multi-Objective Optimum Allocation Model for Earthquake Emergency Shelters Using a Modified Particle Swarm Optimization Algorithm: A Case Study in Chaoyang District, Beijing, China. Wen-Bo Du, Beihang University, China. https://doi.org/10.1371/journal.pone.0144455
83. Ikeda Y, Inoue M (2016) An Evacuation Route Planning for Safety Route Guidance System after Natural Disaster Using Multi-Objective Genetic Algorithm. Procedia Comput Sci 96:1323–1331
84. Gai WM, Du Y, Deng YF (2018) Evacuation risk assessment of regional evacuation for major accidents and its application in emergency planning: A case study. Safety Sci 106:203–218
85. Ghasemi P, Khalili-Damghani K, Hafezalkotob A, Raissi S (2019) Uncertain multi-objective multi-commodity multi-period multi-vehicle location-allocation model for earthquake evacuation planning. Applied Mathematics and Computation 350:105–132
86. Niyomubyeyi O, Pilesjo P, Mansourian A (2019) Evacuation Planning Optimization Based on a Multi-Objective Artificial Bee Colony Algorithm. Isprs Int Geo-Inf. https://doi.org/ARTN 110 10.3390/ijgi8030110
87. Deb K, Sindhya K, Okabe T (2007) Self-adaptive simulated binary crossover for real-parameter optimization. In: Proceedings of the 9th annual conference on Genetic and evolutionary computation. ACM, London England, pp 1187–1194
88. Sahebgharani A (2016) Multi-Objective Land Use Optimization Through Parallel Particle Swarm Algorithm: Case Study Baboldasht District Of Isfahan, Iran. J Urb and Environ Engng 10:42–49
89. Liu M, Xi J (2021) Multi-Objective Optimization of the Spatial Structure and Layout of the Protected Area Based on Ecosystem Services: A Case Study of the Yellow River's Headwaters Region in the Three-River-Source National Park. Chn J Urb EnvironStud 09:2150016

90. Medrano-García JD, Ruiz-Femenia R, Caballero JA (2017) Multi-objective optimization of combined synthesis gas reforming technologies. Journal of CO2 Utilization 22:355–373
91. Zhao X, Tan K, Xie P, Chen B, Pu J (2021) Multiobjective Land-Use Optimization Allocation in Eucalyptus-Introduced Regions Based on the GMDP–ACO Model. J Urban Plann Dev 147:05021004
92. Caparros-Midwood D, Barr S, Dawson R (2015) Optimised spatial planning to meet long term urban sustainability objectives. Computers, Environment and Urban Systems 54:154–164
93. Shifa M, Jianhua H, Feng L LAND-USE SPATIAL OPTIMIZATION MODEL BASED ON PARTICLE SWARM OPTIMIZATION.
94. Rahman MdM, Szabó G (2021) Multi-objective urban land use optimization using spatial data: A systematic review. Sustainable Cities and Society 74:103214
95. Holloway R, Ho D, Delotavo C, Xie WY, Rahimi I, Nikoo MR, Gandomi AH (2023) Optimal location selection for a distributed hybrid renewable energy system in rural Western Australia: A data mining approach. Energy Strategy Reviews 50:101205
96. AlSaba MT, Hakami NA, AlJebreen KS, Abido MA (2023) Multi-objective distributionally robust approach for optimal location of renewable energy sources. Alexandria Engineering Journal 77:75–94
97. Gital Durmaz Y, Bilgen B (2020) Multi-objective optimization of sustainable biomass supply chain network design. Applied Energy 272:115259
98. Hai T, El-Shafay AS, Alizadeh A, Dhahad HA, Chauhan BS, Almojil SF, Almohana AI, Alali AF (2023) Comparison analysis of hydrogen addition into both anode and afterburner of fuel cell incorporated with hybrid renewable energy driven SOFC: An application of techno-environmental horizon and multi-objective optimization. International Journal of Hydrogen Energy S0360319923007206
99. Amjith L, Bavanish B (2022) A review on biomass and wind as renewable energy for sustainable environment. Chemosphere 293:133579
100. Delgarm N, Sajadi B, Kowsary F, Delgarm S (2016) Multi-objective optimization of the building energy performance: A simulation-based approach by means of particle swarm optimization (PSO). Applied Energy 170:293–303
101. Konis K, Gamas A, Kensek K (2016) Passive performance and building form: An optimization framework for early-stage design support. Solar Energy 125:161–179
102. Attia S, Gratia E, De Herde A, Hensen JLM (2012) Simulation-based decision support tool for early stages of zero-energy building design. Energy and Buildings 49:2–15
103. Deveci K, Güler Ö (2020) A CMOPSO based multi-objective optimization of renewable energy planning: Case of Turkey. Renewable Energy 155:578–590
104. Colak HE, Memisoglu T, Gercek Y (2020) Optimal site selection for solar photovoltaic (PV) power plants using GIS and AHP: A case study of Malatya Province, Turkey. Renewable Energy 149:565–576
105. Uyan M (2017) Optimal site selection for solar power plants using multi-criteria evaluation: A case study from the Ayranci region in Karaman, Turkey. Clean Techn Environ Policy 19:2231–2244

106. De Rainville F-M, Fortin F-A, Gardner M-A, Parizeau M, Gagné C (2012) DEAP: a python framework for evolutionary algorithms. In: Proceedings of the 14th annual conference companion on Genetic and evolutionary computation. ACM, Philadelphia Pennsylvania USA, pp 85–92
107. Blank J, Deb K (2020) Pymoo: Multi-Objective Optimization in Python. IEEE Access 8:89497–89509
108. Guerreiro AP, Fonseca CM, Paquete L (2022) The Hypervolume Indicator: Problems and Algorithms. ACM Comput Surv 54:1–42
109. Blonder B, Morrow CB, Brown S, Butruille G, Chen D, Laini A, Harris DJ (2023) High Dimensional Geometry, Set Operations, Projection, and Inference Using Kernel Density Estimation, Support Vector Machines, and Convex Hulls.
110. Izzo D PYGMO AND PYKEP: OPEN SOURCE TOOLS FOR MASSIVELY PARALLEL OPTIMIZATION IN ASTRODYNAMICS (THE CASE OF INTERPLANETARY TRAJECTORY OPTIMIZATION).
111. Blank J, Deb K (2020) A Running Performance Metric and Termination Criterion for Evaluating Evolutionary Multi- and Many-objective Optimization Algorithms. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, Glasgow, United Kingdom, pp 1–8
112. Rudeck J, Vogl S, Banneke S, Schönfelder G, Lewojohann L (2020) Repeatability analysis improves the reliability of behavioral data. Giuseppe Biagini, University of Modena and Reggio Emilia, ITALY. https://doi.org/10.1371/journal.pone.0230900
113. Bravi A, Longtin A, Seely AJ (2011) Review and classification of variability analysis techniques with clinical applications. BioMed Eng OnLine 10:90
114. Noto M, Sato H (2000) A method for the shortest path search by extended Dijkstra algorithm. In: SMC 2000 Conference Proceedings. 2000 IEEE International Conference on Systems, Man and Cybernetics. "Cybernetics Evolving to Systems, Humans, Organizations, and their Complex Interactions" (Cat. No.00CH37166). IEEE, Nashville, TN, USA, pp 2316–2320
115. Sphere Association (2018) The Sphere Handbook: Humanitarian Charter and Minimum Standards in Humanitarian Response., fourth edition. Geneva, Switzerland
116. OpenStreetMap contributors (2023) Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org
117. Sicuaio T, Niyomubyeyi O, Shyndyapin A, Pilesjö P, Mansourian A (2022) Multi-Objective Optimization Using Evolutionary Cuckoo Search Algorithm for Evacuation Planning. Geomatics 2:53–75
118. Niyomubyeyi O, Sicuaio TE, Díaz González JI, Pilesjö P, Mansourian A (2020) A Comparative Study of Four Metaheuristic Algorithms, AMOSA, MOABC, MSPSO, and NSGA-II for Evacuation Planning. Algorithms 13:16
119. Minister of Environment, Science and Technology (2011) ZONING GUIDELINES AND PLANNING STANDARDS.
120. United Nations Human Settlements Programme (2015) International Guidelines on Urban and Territorial Planning.

121. Niyomubyeyi O, Veysipanah M, Sarwat S, Pilesjö P, Mansourian A (2022) An improved non-dominated sorting biogeography-based optimization algorithm for multi-objective land-use allocation: a case study in Kigali-Rwanda. Geo-spatial Information Science 1–15

122. Xiao N, Bennett DA, Armstrong MP (2007) Interactive evolutionary approaches to multiobjective spatial decision making: A synthetic review. Computers, Environment and Urban Systems 31:232–252

123. Yang X-S, Deb S (2013) Multiobjective cuckoo search for design optimization. Computers & Operations Research 40:1616–1624

124. Qiang ZR, Miao XJ, Wu MP, Sawhney R (2019) Optimization of abrasive waterjet machining using multi-objective cuckoo search algorithm (vol 99, pg 1257, 2018). Int J Adv Manuf Tech 105:2747–2747

125. Srivastav A, Agrawal S (2018) Multi-Objective Optimization of Slow Moving Inventory System Using Cuckoo Search. Intell Autom Soft Co 24:343–349

126. Wang RP, Chen YR, Chen YC, Lei TC, Wang HP (2018) Neural network-based skyline evacuation route planning algorithm for different amounts of rainfall. Ieee Int C Electr Ta

127. Balasubbareddy M, Sivanagaraju S, Suresh CV (2015) Multi-objective optimization in the presence of practical constraints using non-dominated sorting hybrid cuckoo search algorithm. Eng Sci Technol 18:603–615

128. Wang H, Du L, Ma S (2014) Multi-objective open location-routing model with split delivery for optimized relief distribution in post-earthquake. Transportation Research Part E: Logistics and Transportation Review 69:160–179

129. Madni SHH, Abd Latiff MS, Ali J, Abdulhamid SM (2019) Multi-objective-Oriented Cuckoo Search Optimization-Based Resource Scheduling Algorithm for Clouds. Arab J Sci Eng 44:3585–3602

130. Yang X-S (2014) Nature-inspired optimization algorithms, First edition. Elsevier, Amsterdam ; Boston

131. Akbari R, Hedayatzadeh R, Ziarati K, Hassanizadeh B (2012) A multi-objective artificial bee colony algorithm. Swarm and Evolutionary Computation 2:39–52

132. Booch G, Maksimchuk RA, Engle MW, Young BJ, Connallen J, Houston KA (2008) Object-oriented analysis and design with applications, third edition. SIGSOFT Softw Eng Notes 33:29–29

133. Nerson J-M (1992) Applying object-oriented analysis and design. Commun ACM 35:63–74

134. Ramnath S, Dathan B (eds) (2011) Object-Oriented Analysis and Design. https://doi.org/10.1007/978-1-84996-522-4

135. Civelo F (1993) Roles for composite objects in object-oriented analysis and design.

136. Emmerich MTM, Deutz AH (2018) A tutorial on multiobjective optimization: fundamentals and evolutionary methods. Nat Comput 17:585–609

137. Black PE, Okun V, Yesha Y (2000) Mutation operators for specifications. In: Proceedings ASE 2000. Fifteenth IEEE International Conference on Automated Software Engineering. IEEE, Grenoble, France, pp 81–88

138. Pavai G, Geetha TV (2017) A Survey on Crossover Operators. ACM Comput Surv 49:1–43
139. Qiao H, Escobar LE, Saupe EE, Ji L, Soberón J (2017) A cautionary note on the use of hypervolume kernel density estimators in ecological niche modelling. Global Ecol Biogeogr 26:1066–1070
140. Seo M, Ryu N, Min S (2020) Sensitivity Analysis for Multi-Objective Optimization of the Benchmark TEAM Problem. IEEE Trans Magn 56:1–4
141. Yang L, Ho SL, Fu WN (2014) Design Optimizations of Electromagnetic Devices Using Sensitivity Analysis and Tabu Algorithm. IEEE Trans Magn 50:1–4
142. Osaba E, Villar-Rodriguez E, Del Ser J, Nebro AJ, Molina D, LaTorre A, Suganthan PN, Coello Coello CA, Herrera F (2021) A Tutorial On the design, experimentation and application of metaheuristic algorithms to real-World optimization problems. Swarm and Evolutionary Computation 64:100888
143. Bischl B, Kerschke P, Kotthoff L, et al (2016) ASlib: A benchmark library for algorithm selection. Artificial Intelligence 237:41–58
144. Zavala G, Nebro AJ, Luna F, Coello Coello CA (2016) Structural design using multi-objective metaheuristics. Comparative study and application to a real-world problem. Struct Multidisc Optim 53:545–566
145. Ceja-Cruz EM, Menchaca-Méndez A, Montero E, Zapotecas-Martínez S (2023) An Archive-Based Multi-Objective Simulated Annealing Algorithm for the Time/Weight-Balanced Cluster Problem in Delivery Logistics. In: 2023 IEEE Congress on Evolutionary Computation (CEC). IEEE, Chicago, IL, USA, pp 1–8
146. Li X, Gao B, Bai Z, Pan Y, Gao Y (2020) An Improved Parallelized Multi-Objective Optimization Method for Complex Geographical Spatial Sampling: AMOSA-II. IJGI 9:236
147. Liu L, Mu H, Yang J, Li X, Wu F (2014) A simulated annealing for multi-criteria optimization problem: DBMOSA. Swarm and Evolutionary Computation 14:48–65
148. Ma L, Cheng S, Wang X, Huang M, Shen H, He X, Shi Y (2017) Cooperative two-engine multi-objective bee foraging algorithm with reinforcement learning. Knowledge-Based Systems 133:278–293
149. Luo J, Liu Q, Yang Y, Li X, Chen M, Cao W (2017) An artificial bee colony algorithm for multi-objective optimisation. Applied Soft Computing 50:235–251
150. Abdi F, Farughi H, Sadeghi H, Arkat J (2024) Using common redundancy components for suppliers in a supply chain network design problem considering energy costs and environmental effects. Expert Systems with Applications 245:122989
151. Huo Y, Zhuang Y, Gu J, Ni S (2015) Elite-guided multi-objective artificial bee colony algorithm. Applied Soft Computing 32:199–210
152. Ibrahim I, Ahmad H, Ibrahim Z, Yusof Z (2020) An Improved Multi-State Particle Swarm Optimization for Discrete Combinatorial Optimization Problems. International Journal of Simulation Systems Science & Technology. https://doi.org/10.5013/IJSSST.a.16.06.14
153. Ab Wahab MN, Nefti-Meziani S, Atyabi A (2015) A Comprehensive Review of Swarm Optimization Algorithms. PLoS ONE 10:e0122827

154. Weise T, Chiong R, Lassig J, Tang K, Tsutsui S, Chen W, Michalewicz Z, Yao X (2014) Benchmarking Optimization Algorithms: An Open Source Framework for the Traveling Salesman Problem. IEEE Comput Intell Mag 9:40–52
155. Meng X, Chang J, Wang X, Wang Y (2019) Multi-objective hydropower station operation using an improved cuckoo search algorithm. Energy 168:425–439
156. Othman MS, Kumaran SR, Yusuf LM (2020) Gene Selection Using Hybrid Multi-Objective Cuckoo Search Algorithm With Evolutionary Operators for Cancer Microarray Data. IEEE Access 8:186348–186361
157. Moffaert KV, Drugan MM, Nowe' A (2013) Hypervolume-based Multi-Objective Reinforcement Learning. Computational Modeling Lab, Vrije Universiteit Brussel 7811:352–366
158. Liang M, Wang L, Wang L, Zhong Y (2024) A hypervolume-based cuckoo search algorithm with enhanced diversity and adaptive scaling factor. Applied Soft Computing 151:111073
159. Auger A, Bader J, Brockhoff D, Zitzler E (2012) Hypervolume-based multiobjective optimization: Theoretical foundations and pratical implications. Theoretical Computer Science 425:75–103
160. Ari N, Ustazhanov M (2014) Matplotlib in python. In: 2014 11th International Conference on Electronics, Computer and Computation (ICECCO). IEEE, Abuja, Nigeria, pp 1–6
161. Blonder B, Lamanna C, Violle C, Enquist BJ (2014) The *n* -dimensional hypervolume: The n-dimensional hypervolume. Global Ecology and Biogeography 23:595–609
162. Zitzler E, Brockhoff D, Thiele L (2007) The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 862–876
163. Sicuaio T, Zhao P, Shindyapin A, Mansourian A (2024) Sustainable and Resilient Land Use Planning: A Multi-Objective Optimization Approach.
164. Tan YS, Fraser NM (1998) The modified star graph and the petal diagram: two new visual aids for discrete alternative multicriteria decision making. J Multi-Crit Decis Anal 7:20–33
165. El-Shorbagy MA, Bouaouda A, Nabwey HA, Abualigah L, Hashim FA (2024) Advances in Henry Gas Solubility Optimization: A Physics-Inspired Metaheuristic Algorithm With Its Variants and Applications. https://doi.org/10.1109/ACCESS.2024.3365700.
166. Bi X, Wang C (2017) An improved NSGA-III algorithm based on objective space decomposition for many-objective optimization. Soft Comput 21:4269–4296