

AN UNMANNED AERIAL VEHICLE ROUTING PROBLEM IN OFFSET
SURVEILLANCE MISSIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AZİZE DEMİR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

SEPTEMBER 2025

Approval of the thesis:

**AN UNMANNED AERIAL VEHICLE ROUTING PROBLEM IN OFFSET
SURVEILLANCE MISSIONS**

submitted by **AZİZE DEMİR** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Zeynep Pelin Bayındır
Head of Department, **Industrial Engineering**

Prof. Dr. Sinan Gürel
Supervisor, **Industrial Engineering, METU**

Examining Committee Members:

Assoc. Prof. Dr. Ayşegül Altın Kayhan
Industrial Engineering, TOBB ETU

Prof. Dr. Sinan Gürel
Industrial Engineering, METU

Assoc. Prof. Dr. Mehmet Hakan Akyüz
Industrial Engineering, METU

Assist. Prof. Dr. Gonca Yıldırım
Industrial Engineering, Gazi University

Assist. Prof. Dr. Nader Ghaffarinasab
Industrial Engineering, METU

Date: 01.09.2025



I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Azize Demir

Signature :

ABSTRACT

AN UNMANNED AERIAL VEHICLE ROUTING PROBLEM IN OFFSET SURVEILLANCE MISSIONS

Demir, Azize

M.S., Department of Industrial Engineering

Supervisor: Prof. Dr. Sinan Gürel

September 2025, 65 pages

This thesis investigates the routing problem of an unmanned aerial vehicle (UAV) with a particular focus on offset surveillance operations, where the mission is executed from a certain distance and side.

The primary objective is to formulate this real-life problem and provide a mathematical model for it. However, the computational results reveal that the exact solution approach becomes impractical for most of the problem sizes due to excessive solution times and memory requirements. Therefore, a metaheuristic approach based on a neighborhood search algorithm with simulated annealing is also designed to provide a solution approach for the management of such a daily operation.

The main contribution of this study comes from the introduction of a novel variant of the UAV routing problem that reflects offset surveillance requirements. To the best of our knowledge, an unmanned aerial vehicle routing in offset surveillance has not been studied in the literature before, thereby opening a new line of research in operations research.

In addition to that, it demonstrates the effectiveness of metaheuristic approaches for

addressing the computational intractability of the exact solution approach in such problems by a practical solution methodology that can be adapted to real-world UAV surveillance applications.

Keywords: UAV, Routing, Offset Surveillance, Mission Planning, Mathematical Modeling, Simulated Annealing



ÖZ

BİR İNSANSIZ HAVA ARACININ KAYDIRILMIŞ GÖZETLEME GÖREVLERİ İÇİN ROTALANMASI PROBLEMİ

Demir, Azize

Yüksek Lisans, Endüstri Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Sinan Gürel

Eylül 2025 , 65 sayfa

Bu tez, gözetleme görevlerinde kullanılan insansız hava aracı (İHA) rotalama problemini, özellikle görevin belirli bir mesafe ve yönden icra edildiği kaydırılmış gözetleme operasyonlarına odaklanarak incelemektedir.

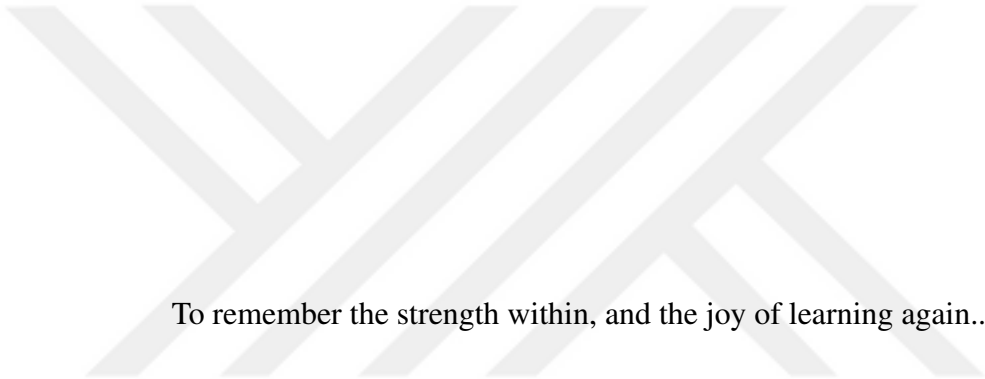
Çalışmanın temel amacı, bu gerçek yaşam problemini formüle etmek ve matematiksel model geliştirmektir. Ancak yapılan analizler sonucunda, artan problem boyutlarında kesin çözüm yöntemlerinin yüksek bellek gereksinimleri ve uzun çözüm süreleri nedeniyle pratikte uygulanabilir olmadığı gözlemlenmiştir. Bu nedenle, söz konusu operasyonların etkin bir şekilde yönetilebilmesi için benzetilmiş tavlama temelli bir komşu arama algoritmasına dayalı meta-sezgisel bir yaklaşım tasarlanmış ve alternatif bir çözüm metodolojisi olarak sunulmuştur.

Bu çalışmanın temel katkısı, kaydırılmış gözetleme gereksinimlerini yansıtan yeni bir insansız hava aracı (İHA) rotalama problemi versiyonunun literatüre kazandırılmasıdır. Bildiğimiz kadarıyla, literatürde belirli bir mesafe ve yön gereksinimi olan gözetleme operasyonları için İHA rotalaması problemine yönelik bir çalışma bulun-

mamaktadır. Bu yönüyle çalışma, yöneylem araştırması ve İHA görev planlaması literatüründe yeni bir araştırma alanı açmaktadır. Ayrıca, bu tür problemlerin hesaplamalı zorluklarını ele almak için meta-sezgisel yaklaşımların etkinliğini ortaya koyarak, gerçek dünyadaki İHA gözetleme operasyonlarına uyarlanabilecek pratik bir çözüm metodolojisi sunmaktadır.

Anahtar Kelimeler: İHA, Rotalama, Kaydırmalı Gözetleme, Görev Planlama, Matematiksel Modelleme, Tavlama Benzetimi





To remember the strength within, and the joy of learning again..

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Dr. Sinan Gürel, for his invaluable guidance, positive attitude, and continuous support throughout the entire process. The completion of this thesis has been made possible through the encouragement of the advisor.

I also want to sincerely thank the members of my thesis jury, Assoc. Prof. Dr. Ayşegül Altın Kayhan, Assoc. Prof. Dr. Mehmet Hakan Akyüz, Assist. Prof. Dr. Gonca Yıldırım, Assist. Prof. Dr. Nader Ghaffarinasab, for generously dedicating their time and expertise to my thesis evaluation.

I am grateful to my colleagues with whom I had the opportunity to work during my academic journey. Their encouragement, collaboration, and moral support have always been greatly appreciated.

Last but not least, I want to express my heartfelt gratitude to my family and friends, who have always accompanied me with their endless support, patience, and encouragement. The “Golden Girls” team, Hacer, Azime, Nil, and our newcomer Doğa, witnessed every single day of this journey, reminding me that true companionship transcends distance. Also, I want to express my appreciation to my lovely friend Yasemin Sipahi who has always been there to increase my motivation to complete my thesis by making plans with me for a holiday. I want to present my gratitude to my dear friend Ebru Eryiğit for her unlimited support and friendship during this challenging period. I owe special thanks to Çiya Aydoğan for his generous support and the calm refuge of his office, complete with tea, throughout this thesis. To Taner Ateş, for always being more than a friend, like a brother, and for being just a phone call away. I would like to thank my lifelong friend, Mustafa Onur Özcan, for standing by me through both the good and the difficult days. The ongoing presence of these individuals, with those whose names are not mentioned here, has been a source of strength, inspiration, and a priceless reminder that I was never alone.

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
ACKNOWLEDGMENTS	x
TABLE OF CONTENTS	xi
LIST OF TABLES	xiii
LIST OF FIGURES	xiv
LIST OF ABBREVIATIONS	xvi
CHAPTERS	
1 INTRODUCTION	1
2 LITERATURE REVIEW	9
2.1 Review on Travelling Salesman Problem	9
2.2 Review on Unmanned Aerial Vehicle Operations Area	12
3 PROBLEM DEFINITION	17
3.1 Problem Characteristics	18
3.2 Assumptions	20
3.3 Modeling Technique	21
3.4 Mathematical Model	22
3.5 Complexity	25

4	A METAHEURISTIC APPROACH FOR THE PROBLEM	27
4.1	Nearest Neighbor-based Initial Solution Algorithm	29
4.2	A Simulated Annealing Based Neighborhood Search Algorithm . . .	31
5	COMPUTATIONAL RESULTS	37
5.1	Experimental Settings	38
5.2	Parameter Tuning for Simulated Annealing Algorithm	41
5.3	Performance Evaluation and Discussion	43
5.3.1	Performance of the Exact Method	45
5.3.2	Comparison of the Performance of the Metaheuristic Approach with Exact Method	50
5.3.3	Consistency of the Metaheuristic Approach	53
6	CONCLUSION	55
	REFERENCES	59
	APPENDICES	61
A	EXPERIMENTAL STUDY DETAILS FOR PARAMETER TUNING OF THE METAHEURISTIC APPROACH	63

LIST OF TABLES

TABLES

Table 5.1	Levels of the Factors in Scenarios	41
Table 5.2	Exact Approach vs. Metaheuristic Performance Summary	44
Table 5.3	Performance of the Exact Method Based on Solution Status	47
Table 5.4	Performance of the Exact Method Based on CPU Time	48
Table 5.5	Performance of the Exact Method Based on Optimality Gap	49
Table 5.6	Comparison of Objective Values between the Metaheuristic Approach and the Exact Method	51
Table 5.7	Performance Comparison between the Metaheuristic Approach and the Exact Method	52
Table 5.8	Assessment of the Consistency of the Metaheuristic Approach	54

LIST OF FIGURES

FIGURES

Figure 1.1	Illustration of a Surveillance and Reconnaissance Mission	2
Figure 1.2	A Two-sided View Illustrating the Problem	3
Figure 1.3	A Two-sided View Illustrating Both the Target and the UAV's Observable Field	4
Figure 1.4	Illustration of Problem: Targets, Surveillance Points and the Op- timal Tour	5
Figure 3.1	Illustration of Offset Surveillance Mission	18
Figure 3.2	Controlled Airspace Classification	19
Figure 4.1	General Flow of the Metaheuristic Approach	28
Figure 4.2	Illustration of the 1-Opt and 2-Opt Neighborhood Movements . .	33
Figure 5.1	Problem Area	39
Figure 5.2	Current Solution at the Iteration	42
Figure A.1	Current Solution at the Iteration for Sample Problem - 1	63
Figure A.2	Current Solution at the Iteration for Sample Problem - 2	64
Figure A.3	Current Solution at the Iteration for Sample Problem - 3	64
Figure A.4	Current Solution at the Iteration for Sample Problem - 4	65



LIST OF ABBREVIATIONS

DGCA	Directorate General of Civil Aviation
GCS	Ground Control Station
GTSP	Generalized Travelling Salesman Problem
ICAO	International Civil Aviation Organization
ILP	Integer Linear Programming
MILP	Mixed-Integer Linear Programming
NOTAM	Notice to Airmen
TSP	Travelling Salesman Problem
SPT	Shortest Processing Time
UAV	Unmanned Aerial Vehicle

CHAPTER 1

INTRODUCTION

Unmanned aerial vehicles (UAVs) are undeniably among the most epoch-making technological advancements in human history. Owing to their ability to eliminate human risk in hazardous tasks, UAVs are widely employed in military operations. Beyond military applications, their use is expanding rapidly in civilian domains such as search and rescue, as well as damage assessment during natural disasters like earthquakes, floods, and wildfires.

With increasing variety and rapidly advancing payload capabilities, UAVs exhibit significant potential in surveillance, reconnaissance, and combat missions. According to Precedence Research (Research, 2025), the global UAV market size is estimated to be approximately USD 37 billion in 2024. Growing at a compound annual growth rate (CAGR) of 15.50% from 2025 to 2034, the market is expected to reach USD 157 billion by 2034. This indicates that studies concerning unmanned aerial vehicles (UAVs) are expected to gain increasing prominence in the future.

UAVs are engineered with diverse features and capabilities tailored to specific operational demands. For instance, Medium-Altitude Long-Endurance (MALE) UAVs operate within an altitude band of 10,000 to 30,000 feet and are capable of maintaining flight durations ranging from 24 to 48 hours. These characteristics enable a UAV to perform multiple surveillance or reconnaissance missions within a single flight operation, thereby introducing the necessity for route optimization.

UAVs are heavily used for surveillance and reconnaissance. In this thesis, we consider a problem that can be seen as a route optimization in which a UAV is tasked with executing a series of such operations. The route optimization problem for UAVs can

be modeled as a tour generation problem among predefined target coordinates. In this framework, a UAV departs from a runway, visits each designated target location from a specific altitude, and then returns to its point of origin. The objective is to determine a tour that minimizes metrics such as total travel distance or fuel consumption, subject to operational constraints. To better illustrate the operational context, Figure 1.1 presents a visual representation of the problem environment.



Figure 1.1: Illustration of a Surveillance and Reconnaissance Mission

In such problems, the set of target locations to be visited is predetermined by the relevant organization or client and delivered to the UAV operator in advance. In addition to that, fuel consumption parameters are inherently tied to the specific UAV model selected. The determination and procurement of a UAV is a strategic-level decision, occurring far less frequently than the operational-level task of planning daily missions. Therefore, the UAV model can be considered a fixed input to the problem and is not subject to decision-making within the scope of this study.

Additionally, external factors such as weather conditions and flight altitude can affect

the operational cost. However, these factors are excluded from consideration in this model, as they are beyond the control of the operator and are typically addressed through separate studies and mitigation strategies. Overall, given the fixed nature of the UAV and mission parameters, scheduling emerges as the most critical component for optimizing objectives such as fuel consumption.

During an operation, the flight altitude of a UAV is regulated by the appropriate aviation authorities to ensure safe integration with air traffic. As a result, a UAV initially flies at the assigned altitude and subsequently descends to a lower altitude to carry out its mission tasks. Since altitude levels are predefined and not subject to operator control, the problem can be modeled as occurring within a two-sided operational environment. To clarify the structure of the problem, a bird's-eye view is illustrated in Figure 1.2.

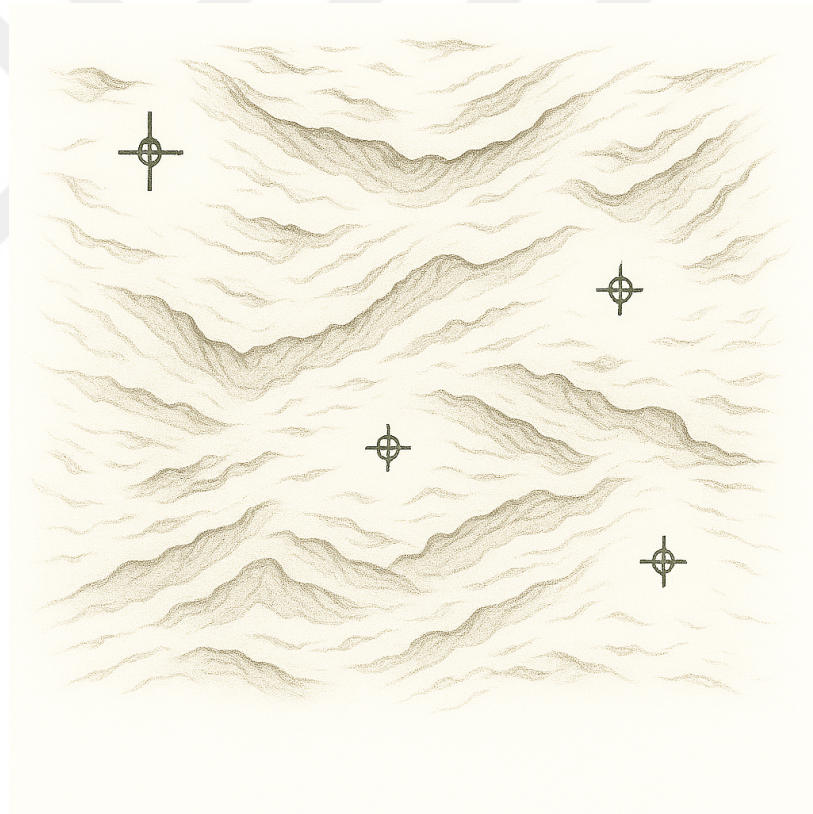


Figure 1.2: A Two-sided View Illustrating the Problem

Surveillance and reconnaissance missions involve remote collection of visual or sensory data at target locations. These tasks are typically carried out using onboard

high-resolution electro-optical, infrared cameras, or radar-based sensors on a UAV. Since these technological devices allow us to meet needs from a distance, the vehicle does not need to visit the exact coordinates of the given target. In other words, the target can be seen as a group of coordinates or a circle under an assigned altitude. Figure 1.3 illustrates the potential observable area for each target in the absence of any external constraints.

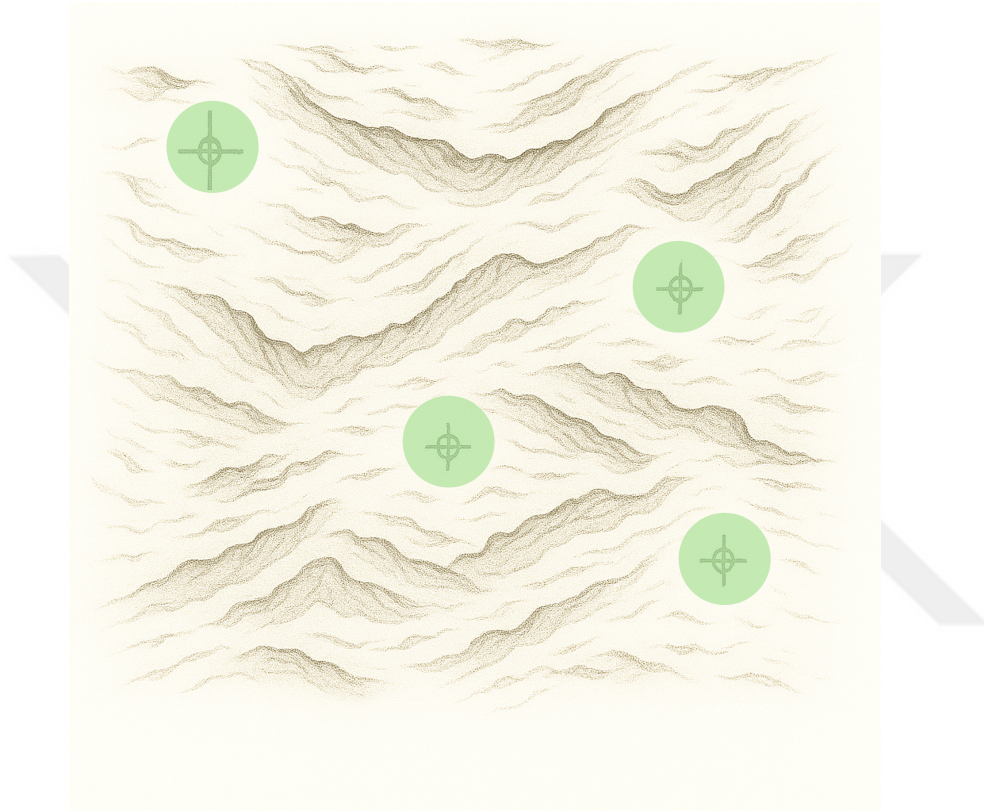


Figure 1.3: A Two-sided View Illustrating Both the Target and the UAV's Observable Field

On the other hand, every real-life UAV mission is inherently subject to various operational constraints. One of the most critical among them is the presence of physical obstacles. For instance, certain geographical features may obstruct line-of-sight visibility from specific directions, transforming the theoretical circular visibility area around a target into a partial arc. In this study, the candidate positions along such arcs are referred to as “surveillance points”.

Wind conditions also play a significant role in the selection of appropriate surveillance

points. In military missions, for example, the UAV's engine noise may be detected by the target if the vehicle approaches from the windward side. Therefore, it is often necessary to approach from the opposite direction to prevent such an undesired case. A similar constraint arises in fire damage assessment missions, where smoke carried by the wind can obstruct visibility and prevent effective monitoring from certain directions. These environmental and situational factors add a layer of complexity to the problem, and that makes the specialty of the problem in this study.

Specifically, the objective of the problem is to jointly determine the visiting sequence of the targets and the optimal surveillance point on the corresponding arc. To make the problem tractable, the enumeration of the surveillance arc is employed, thereby discretizing the continuous set of possible observation points. Figure 1.4 illustrates the targets, the enumerated surveillance points for each, and an example of an optimal tour. Once a UAV visits one of the designated surveillance points for a target, the mission for that target is considered completed. Thus, the goal is to construct a route among the surveillance points that minimizes the overall mission cost.

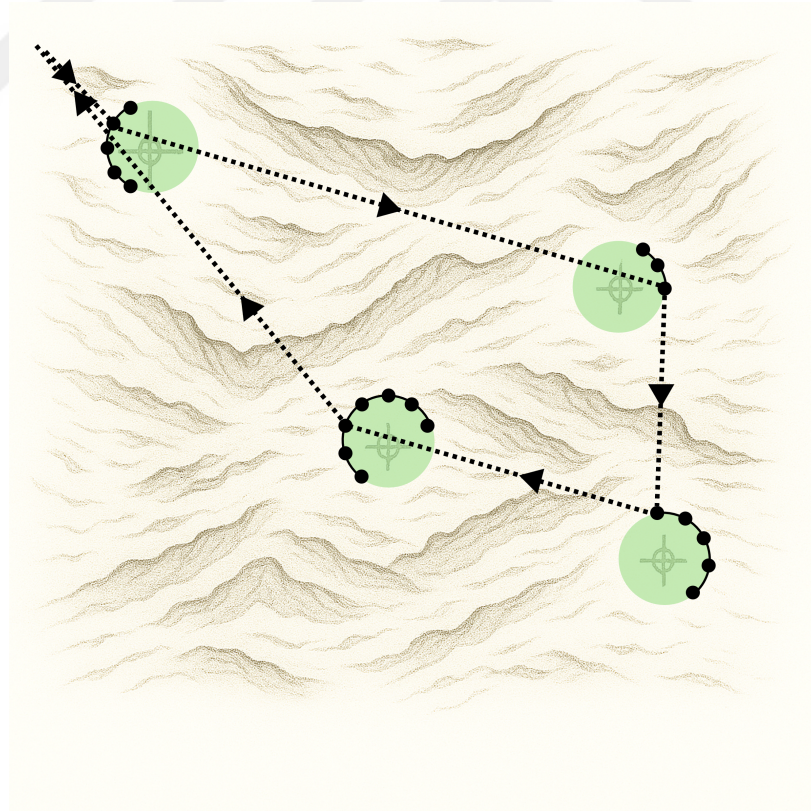


Figure 1.4: Illustration of Problem: Targets, Surveillance Points and the Optimal Tour

The primary focus of this study is tour construction, which aligns the problem with the well-known Travelling Salesman Problem (TSP) in the operations research literature. However, unlike the classical TSP, where a single point represents each target, the targets in this study are associated with a set of candidate surveillance points, increasing the problem's complexity. Since the TSP is classified as an NP-hard problem, the problem addressed in this study is also NP-hard by extension.

To the best of our knowledge, the problem considered in this thesis has similarities in the existing generalized travel salesman problem (GTSP) literature. Accordingly, the principal contribution of this study is the introduction of a new problem definition for real life issues, which broadens the scope of research in this field.

A mathematical model is formulated to deal with the problem. However, as expected, the model struggles to deliver a solution within acceptable time limits when the problem size increases. Given that tour generation is an operational-level decision, solution time becomes a critical factor; delays are often unacceptable in real-world scenarios. Therefore, a metaheuristic approach is also proposed.

The metaheuristic method begins with the construction of an initial solution using a greedy approach, which is then refined using a simulated annealing algorithm. The performance of both the exact and metaheuristic approaches is assessed through comprehensive experimental analysis, and the results are discussed in detail in the corresponding sections of this report.

In summary, the exact approach is only feasible for small-sized problems because of the time required to provide a solution. As the problem size grows, the quality of the solution deteriorates, and the time required increases significantly. In contrast, the metaheuristic approach yields feasible solutions across all tested scenarios within a reasonable time frame. Notably, in small instances, the metaheuristic performs comparably to the exact method, while in larger instances, it delivers superior solutions within seconds.

To conclude, with ongoing technological advancements, unmanned aerial vehicles are achieving greater endurance and are increasingly equipped with advanced long-range cameras. These developments will inevitably lead to more sophisticated and complex

optimization problems, emphasizing the importance of scalable and adaptive solution approaches such as the one presented in this study.

Outline of the rest of the thesis: Chapter 2 reviews the related literature. Chapter 3 introduces the problem definition together with its mathematical formulation. The proposed metaheuristic solution approach is presented in Chapter 4. Chapter 5 reports the computational results and their analysis. Finally, Chapter 6 concludes the thesis with closing remarks and potential directions for future research.





CHAPTER 2

LITERATURE REVIEW

The special case of the problem addressed in this thesis can be regarded as the Traveling Salesman Problem (TSP). Unlike the classical waypoint assignment, the extension constitutes a specific case of the UAV Routing Problem, where the objective is not only to construct a tour, as in the TSP, but also to determine the surveillance coordinates of each target from an arc defined as the offset surveillance set. This structure distinguishes the problem from traditional TSP models to generalized or set TSP and positions it as a novel problem within the area of unmanned aerial vehicle operations. As a result, an extensive review of the existing literature on TSP, as well as studies on the area of operation of unmanned aerial vehicles, is conducted in a thorough way.

2.1 Review on Travelling Salesman Problem

The TSP is a well-known combinatorial optimization problem. The problem owner wants to visit a set of nodes. Using the set of nodes and the distances between each pair of nodes, the goal is to find the shortest possible route while visiting each node only once and finishing the tour at the starting point. The TSP is classified as an NP-hard problem. This class of problems is at least as hard as the hardest problems in NP.

The Traveling Salesman Problem (TSP) has been recorded since 1930. The first mathematical formulation of the problem is provided by Merrill M. Flood (Flood, 1948). The work by Dantzig et al., 1954 helped the issue reach prominence. Their paper with Fulkerson and Johnson introduced the cutting-plane method, an optimization technique used to solve integer linear programming (ILP) and mixed-integer linear

programming (MILP) problems. This work laid a foundation for the development of exact algorithms. With existing computational power, Dantzig achieved solving problems with up to 49 nodes by leveraging linear programming techniques. The increase in computational power has enabled researchers to solve larger instances by covering more nodes. For even larger instances, limiting factors lead researchers to develop heuristic approaches for practical near-optimal solutions.

Studies showed that exact methods guarantee optimal solutions. The trade-off is that they are computationally expensive and exhaustive for large instances. Using dynamic programming for TSP with a time complexity of $O(n^2 2^n)$ with a number of nodes not exceeding 30 is suggested by the Held-Karp algorithm published by Held and Karp, 1962. Later, pruning branches is suggested in developing algorithms in order to decrease the size of the solution space. The methods are widely known as branch-and-bound and branch-and-cut methods, described by Applegate et al., 2006. Solver called Concorde, accepting the branch-and-cut algorithm as the main algorithm, emerged, developed by Applegate, Bixby, Chvátal, and Cook. The solver remains a gold standard for exact TSP solutions, solving instances with thousands of cities using sophisticated cutting-plane techniques.

Exact methods for large-scale TSP problems are computationally infeasible; the computational infeasibility of exact methods led researchers to heuristic and metaheuristic algorithms. A simple greedy heuristic example is the nearest neighbor algorithm. The algorithm constructs a tour by always visiting the closest unvisited city, offering sub-optimal solutions that are quickly obtained (Rosenkrantz et al., 1977). Simulated Annealing is a probabilistic optimization algorithm inspired by the annealing process in metallurgy (Kirkpatrick et al., 1983b). Genetic Algorithms are inspired by evolutionary principles; they imitate the evolutionary process (Holland, 1975). The foraging behavior of ants inspires Ant Colony Optimization (Dorigo and Gambardella, 1997). Mimicking natural processes to explore the solution space is rooted in the general principle of the empirical fine-tuning point of view in engineering, often finding near-optimal solutions for large instances in much quicker timespans compared to exact methods.

TSP studies conducted recently focus on hybrid approaches and machine learning

integration. There have been cases where reinforcement learning has been applied to TSP. The trial-and-error models can be seen in an overview at Bello et al., 2016. The paper uses neural networks to learn policies for constructing tours. Graph Neural Networks (GNNs) can be used to create the environment where an agent makes decisions and tries to learn via feedback from the interaction with this environment. There are studies that have shown promise in capturing TSP's graph-based structure, as demonstrated in the paper by Joshi et al., 2019. Recently, quantum computing approaches have been suggested. Classical computing uses bits that represent either 0 or 1; inspired by quantum mechanics, quantum computers use quantum bits or qubits. Qubits can exist in a superposition of 0 and 1 simultaneously at differing proportions. This enables parallel computation, which potentially solves certain problems much faster, such as those explored by Farhi et al., 2014 with quantum approximate optimization algorithms (QAOA). Currently, the modeling of real phenomena in quantum bits is not developed, and the problems that can be solved this way are limited.

The TSP literature can be divided into multiple major lines of research, ranging from formulation methods to various extensions and solution methodologies. The set TSP, or generalized TSP, is one of them, and the problem addressed in this study is of that scope. The classical TSP problem is extended by dividing the vertices into a given number of clusters, and the GTSP searches for the shortest tour visiting exactly one node from each cluster. This structure enables us to model many real-world applications, such as the UAV routing problem with offset surveillance missions. By Pop et al., 2024, a review of GTSP literature is provided.

Since the TSP can be viewed as a special case of GTSP, where each cluster consists of a single node, the GTSP also exhibits NP-hard complexity. Therefore, exact solution approaches could not provide a reasonable solution even for small or medium-sized instances, which is criticized at Jafarzadeh et al., 2017. Given its computational complexity, heuristic solution approaches play a critical role for such a problem. This study also provides an enhanced genetic algorithm that combines the nearest-neighbor search and heuristic mutation operators, analyzing their performance. Similarly, J. Yang et al., 2008 provides the ant colony optimization (ACO) based metaheuristic solution approach to GTSP. By W. Yang and Wang, 2012, an enhanced simulated annealing algorithms that incorporate memory functions and dynamic programming to

improve both the speed of convergence and the quality of solutions has been introduced. The study, conducted by Schmidt and Irnich, 2022, contributes to the literature by proposing new neighborhood structures and an iterated local search framework that jointly optimize cluster sequencing and representative selection, setting a new benchmark for GTSP solution quality. Pop et al., 2024 demonstrates and highlights that early studies effectively adapted well-known TSP heuristics for GTSP, whereas later studies introduced hybrid and population-based metaheuristics.

2.2 Review on Unmanned Aerial Vehicle Operations Area

Unmanned aerial vehicles (UAVs), owing to their numerous advantages, are widely used in both civil and military operations, including surveillance, reconnaissance, attack, and logistics missions. Since UAVs have different capabilities for different mission requirements, this directly affects the environment and scope of the studies done. This broad scope has diversified the related literature into several distinct streams. For small UAVs, drones, logistics, and delivery tasks have emerged as particularly popular application areas. Due to their limited flight endurance, most studies have focused on operational efficiency for broader systems, such as integrating UAVs with ground vehicles like trucks. For operative UAVs, mission scheduling, routing, and path planning seem the critical issues. In addition, obstacle avoidance has emerged as a popular and recurring theme, reflecting the importance of safe navigation. Within this landscape, the most relevant strand of literature for the present study is that of routing and path planning, which directly addresses the optimization challenges inherent in UAV operations.

An important aspect of Unmanned Aerial Vehicle (UAV) applications is navigation, both in single-vehicle operations and in swarm configurations. The navigation of a single UAV is called waypoint assignment, and the UAV follows these points in order to achieve the targets. TSP is adapted for waypoint assignment, where a UAV must visit predefined waypoints generated by the TSP models (e.g., for surveillance, delivery, or data collection) while minimizing travel distance or time. UAV waypoint assignment extends beyond TSP by addressing trajectory optimization, obstacle avoidance, and environmental constraints like battery life, wind, and no-fly zones.

This review explores TSP applications for UAV waypoint assignment, general UAV planning approaches, and their integration, highlighting classical methods, recent advancements, and challenges.

The application of TSP to UAV routing emerged as UAVs transitioned from military to civilian roles, such as logistics and surveillance. Early work by Ryan et al., 1998 modeled UAV routing as a TSP variant, using exact methods like branch-and-bound for small-scale missions. Applegate et al., 2006 adapted the Concorde solver to solve TSP instances with dozens of waypoints using cutting-plane techniques. Exponentially increasing problem size limited both methods; brute force algorithms failed to solve larger instances. High computational complexity limited scalability for real-time applications.

Due to TSP's NP-hard nature, exact methods are neither scalable nor fast. The routing operations for UAV surveillance missions require near real-time solutions, which is why heuristic and metaheuristic methods dominate. The Nearest Neighbor algorithm provides fast, suboptimal TSP solutions by selecting the closest unvisited node, route, or part Rosenkrantz et al., 1977. Metaheuristics like Genetic Algorithms (GAs) Holland, 1975, Ant Colony Optimization (ACO) Dorigo and Gambardella, 1997, and Particle Swarm Optimization (PSO) are widely applied. Chen et al., 2017 proposed an Improved Genetic Algorithm (IGA) and a PSO-ACO hybrid for UAV TSP, enhancing convergence speed and solution quality.

There are some studies emphasizing the adaptation of traditional Vehicle Routing Problem (VRP) and Traveling Salesman Problem (TSP) frameworks for UAV surveillance routing. Banaszak and Smith, 2020 conducted a review, analyzing 396, identifying VRP and TSP variants as dominant approaches. They noted that 54 recent papers applied these frameworks to surveillance tasks, with a focus on wireless sensor networks (WSNs) for monitoring large regions. However, simplifications in UAV kinematics and flight dynamics often limit real-world applicability.

UAV TSP variants in commercial planning incorporate constraints like battery life and payload. The Flying Sidekick TSP (FSTSP), introduced by Murray and Chu, 2015, FSTSP models a UAV collaborating with a ground vehicle, with the UAV launching and returning to the vehicle for deliveries. The Multiple Flying Sidekicks TSP with

Variable Drone Speeds (mFSTSP-VDS) considers UAV speed as a decision variable, optimizing energy consumption based on payload and flight range Raj et al., 2020. The TSP with Drone Resupply, proposed by Pina-Pardo et al., 2022, allows UAVs to receive new orders mid-route via depot resupply, using mixed-integer linear programming for larger instances. The TSP-CPP problem, combining TSP with Coverage Path Planning (CPP), optimizes UAV routes to cover multiple regions, as explained in detail by Xie et al., 2019.

Recent studies leverage machine learning, algorithms learn patterns and make predictions or decisions from data, to enhance UAV TSP solutions. Bello et al., 2016 applied reinforcement learning to optimize UAV waypoint sequences, an agent learns to make decisions by interacting with the environment, training neural networks to approximate optimal tours. Graph Neural Networks (GNNs), as explored by Ma et al., 2020, model waypoint graphs to account for spatial and temporal constraints, improving scalability for large missions. Additionally, hybrid approaches combining metaheuristics with machine learning, such as those by Zhang et al., 2023, integrate real-time data (e.g., weather updates) to adapt UAV routes dynamically.

In some cases, multiple UAVs need to be deployed for distribution or surveillance. These cases extend TSP to the Multiple Traveling Salesman Problem (MTSP), where multiple UAVs coordinate to visit multiple surveillance points. Kitjacharoenchai et al., 2019 proposed a mathematical model for MTSP with drones, using K-means and tabu search to optimize routes. Although different from the problem investigated in this paper, cooperative path planning, as explored by Maza and Ollero, 2007, addresses multi-UAV missions, ensuring coverage of convex regions with energy constraints. This approach differs by assigning a sequence of waypoints to optimize the given route, rather than finding the best route. Luo et al., 2022 developed a hybrid multi-objective optimization for truck-drone systems, incorporating flexible time windows and customer priorities.

To the best of our knowledge, the main structure of the problem that assigning an optimal sequence of visits for a UAV considering surveillance points is similar with generalized TSP literature. In this thesis, different than the existing TSP, generalized TSP and UAV studies in the literature, we state a novel problem setting in which the

mission is executed from offset surveillance arcs rather than the exact target locations. We handled the problem with a two-sub-decision, which is not observed in the literature. We develop a assignment based mathematical model for the problem and propose a simulated annealing-based solution approach.





CHAPTER 3

PROBLEM DEFINITION

Unmanned Aerial Vehicles (UAVs) are increasingly utilized in surveillance and reconnaissance operations thanks to their ability to operate in hazardous mission environments. In practice, these missions mostly require observing specific targets not directly from their coordinates, but from offset surveillance positions determined because of operational constraints such as safety, line-of-sight, or environmental conditions. That makes a special routing problem, since the issue is not only constructing an efficient tour, but also deciding from which point in a predefined surveillance area each target should be observed.

In this problem, a single unmanned aerial vehicle (UAV) is assigned to perform a surveillance mission on n targets. For each target, a set of candidate surveillance points is defined along a circular arc that represents feasible observation positions. Figure 3.1 illustrates the operational environment of a UAV in an offset surveillance mission. The green circles represent the theoretical observable areas of each target, while the black lines indicate the feasible surveillance coordinates that are determined based on operational constraints.

The aim is to select exactly one surveillance point for each target and construct a closed tour that begins and ends at the runway while visiting the chosen surveillance points in order. So, the problem can be defined as two intertwined decisions: (i) selecting surveillance coordinates from the arcs associated with each target and (ii) determining the optimal visiting order of these surveillance points. The objective is to minimize the total tour length, thereby improving operational efficiency and reducing mission costs.

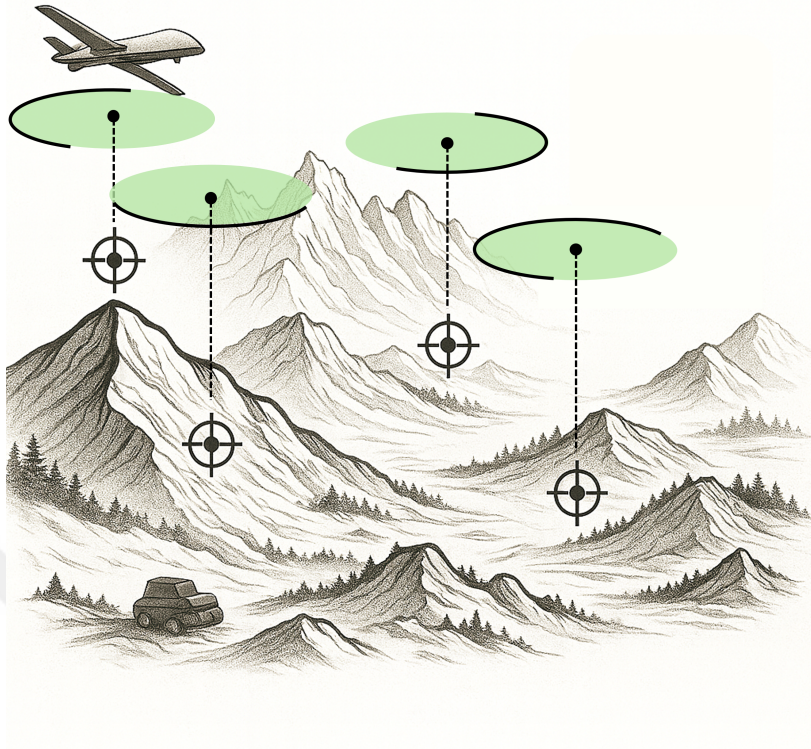


Figure 3.1: Illustration of Offset Surveillance Mission

The problem that is the subject of this thesis is an example application of the Generalized Travelling Salesman Problem (GTSP) within the operations research literature. Unlike the classic Traveling Salesman Problem (TSP), where each node has a fixed location, this variant incorporates both routing and surveillance coordinate selection, significantly increasing the problem dimensionality and computational complexity. This structure differentiates the problem from standard routing problems and underlines the need for specialized mathematical formulations and efficient solution approaches.

3.1 Problem Characteristics

In an unmanned aerial vehicle (UAV) surveillance operation, targets are visited to accomplish missions such as detection, image acquisition, or reconnaissance. Owing to the advanced resolution and capabilities of modern UAV payloads, it is not necessary

to reach the exact location of each target. Instead, visiting one of the feasible observation points—referred to as surveillance coordinates—within the visible region of the target is desired to execute the mission. This transforms the problem into a special extension of the classical Travelling Salesman Problem (TSP) in 3-dimensional space.

Although the actual operation is carried out in a 3-dimensional space, the problem in this study is modeled in a two-dimensional plane due to altitude restrictions imposed by the authorities. In various publications, airspace is recognized and defined as a common resource, highlighting the critical role of air traffic management. To control and ensure air traffic safety, given the increasing usage of unmanned aerial vehicles (UAVs), the determination of flight altitude is strictly regulated by civil aviation authorities. According to the International Civil Aviation Organization (ICAO), airspace is divided into classes to manage traffic effectively. In Figure 3.2 Havacılık, 2022, the controlled airspace classes are shared. In addition to the controlled airspace classes, there are also two uncontrolled classes for small-sized UAVs, or drones.



Figure 3.2: Controlled Airspace Classification

In Turkey, the Directorate General of Civil Aviation (DGCA) is the primary authority responsible for air traffic management. In accordance with international civil aviation regulations, the DGCA assigns the corresponding airspace class and enforces the ap-

plicable rules. The UAV class, which can be used in such a surveillance operation, needs to be flown in class E or class A. While the UAV is executing the mission, it may need to lower its attitude with the approval of all related parties. Since the relevant air traffic control units determine the flight altitude of the operation before it begins, the operator does not have control over the vertical dimension. Such an operation environment both underlines the importance of planning and also enables us to handle the problem in a 2-dimensional space.

At the given altitude, the theoretical observable area of each target can be determined based on the payload capabilities, such as the resolution of the camera. However, due to safety constraints, the farthest feasible surveillance points are generally preferred for mission execution. In addition, environmental factors may further restrict the surveillance area. For instance, geographical obstacles can limit the line of sight, while wind direction may cause the UAV's engine noise to propagate toward the target, which is undesirable. Therefore, offsetting may not be feasible in every direction. As a result, the surveillance set is represented by an arc around the projection of each target at the given altitude with a specified offset distance.

All in all, as stated above, the problem can be identified as a UAV routing problem in offset surveillance missions. The objective is to select one surveillance point on the corresponding arc for each target and to construct a tour visiting these points. Since the selection of surveillance coordinates directly affects both the tour length and the sequencing of the tour, this extension represents a significant and meaningful contribution.

3.2 Assumptions

The problem definition and mathematical formulation are developed under the following assumptions.

- The UAV capabilities are determined by the selected vehicle and payloads, which cannot be changed during operation. So, we have static and defined parameters for problem construction.

- The time required to complete the mission at each surveillance point is assumed to be negligible or a fixed time.
- There is no time window restriction for executing a mission. A mission can be performed at any time of the tour.
- Operation duration or fuel is not explicitly constrained, allowing the UAV to complete the tour without time-bound limitations.
- Fuel consumption or distance covered during the UAV's ascent and descent is not taken into account since these phases are not directly controlled in the context of the mission.
- It is assumed that the needed approvals for class passes during the operation can be obtained.
- The intersection of possible surveillance arcs for different targets may lead to a restricted feasible set. In this study, it is assumed that the surveillance sets are predefined in this manner.
- It is assumed that the mission can be executed in compliance with flight dynamics.

3.3 Modeling Technique

The problem addressed in this study can be classified as Generalized Travelling Salesman Problem (GTSP) in the literature. The Travelling Salesman Problem (TSP) is classically defined as starting from a base, visiting each node exactly once, and returning to the base. Structurally, the problem can be interpreted as a single-machine scheduling problem with sequence-dependent processing times.

Scheduling problems need to be formulated as integer or binary optimization models, which makes them computationally difficult to solve. Therefore, constructing a stronger formulation is often required to achieve efficiency. To identify an effective modeling technique, this study benefits from the work of Keha et al., 2009. The primary contribution of that research is a computational performance comparison

of four different formulations for single-machine scheduling problems with varying complexity, classified according to the type of decision variables employed: (1) completion time variables, (2) time-indexed variables, (3) linear ordering variables, and (4) assignment and positional date variables. Accordingly, this study adopts a hybrid mathematical formulation. To simplify the calculation of the objective function, an assignment-based decision variable (assigning each job to another job) is defined. In addition, a positional assignment variable is introduced, which eliminates the need for explicit subtour elimination constraints. The complete mathematical model is presented in Section 3.3.

3.4 Mathematical Model

This section introduces the mathematical formulation of the problem, presenting its decision variables, objective function, and constraints.

As stated earlier, the aim is to construct a tour that starts from the runway, visits the selected surveillance coordinate of each target, and returns to the runway while minimizing the total tour length. Choosing a surveillance point on a continuous arc for each target and creating a tour with these points causes a significant challenge in the mathematical formulation of the problem. To overcome this, each surveillance arc is discretized into equal parts, and a finite set of candidate surveillance points is generated and used in the formulation of the problem.

There are n target points to visit, and ns possible surveillance points for each target. The arc is divided into $ns - 1$ parts to construct the surveillance point set. In addition to these, the set t is used to record the place in the constructed tour.

Two types of decisions must be addressed in the model; the first is to determine the order in which the targets will be visited. The second is to select a surveillance point for each target.

In order to facilitate the computational process, the distances between surveillance points, as well as the distances between the runway and the surveillance points, are pre-calculated and defined as parameters in the model. The objective function, which

aims to minimize the total length of the tour, is calculated with these parameters.

To ensure that each target is assigned to a position in the tour, an assignment constraint is included in the model. Additionally, another constraint is introduced to guarantee that each position in the tour is occupied by only one target. With Constraints (3.1) and (3.2), the model ensures the construction of a feasible tour among the targets. The relationship between each target and its selected surveillance point is established through Constraint (3.3). Specifically, if a target is assigned to a position in the tour, it must be visited from exactly one surveillance point. Constraint (3.4) defines the transition between surveillance points corresponding to consecutive targets in the tour. Lastly, decision variables are introduced in Constraint (3.5) to complete the formulation. Consequently, the complete problem is established as an integer programming model.

Sets:

I : Targets to be observed (1,2,...,n)

K_i : Surveillance points of target i (1,2,...,ns)

T : Visit time/Order (1,2,...,n)

Parameters:

D_{ijkl} : Distance between the surveillance point k of target i and the surveillance point l of target j

P_{ik} : Distance between the surveillance point k of target i and the runway

Decision Variables:

x_{ijkl} : 1 if the surveillance point k of target i is visited just after the surveillance point l of target j , 0 otherwise.

y_{ikt} : 1 if the surveillance point k of target i is visited in order t

z_{it} : 1 if the target i is visited in order t

$$\min \sum_{i \in I} \sum_{k \in K_i} P_{ik} \cdot y_{ik1} + \sum_{\substack{i,j \in I \\ i \neq j}} \sum_{k \in K_i} \sum_{l \in K_j} D_{ijkl} \cdot x_{ijkl} + \sum_{i \in I} \sum_{k \in K_i} P_{ik} \cdot y_{ikn}$$

Subject to:

$$\sum_{i \in I} z_{it} = 1 \quad \forall t \in T \quad (3.1)$$

$$\sum_{t \in T} z_{it} = 1 \quad \forall i \in I \quad (3.2)$$

$$\sum_{k \in K} y_{ikt} \geq z_{it} \quad \forall i \in I, \forall t \in T \quad (3.3)$$

$$x_{ijkl} \geq y_{ikt} + y_{jlt+1} - 1 \quad \forall i, j \in I, i \neq j, \quad (3.4)$$

$$\forall k \in K_i, \forall l \in K_j, \forall t \in T, t \neq ns$$

$$x_{ijkl} \in \{0, 1\}, y_{ikt} \in \{0, 1\}, z_{it} \in \{0, 1\} \quad (3.5)$$

3.5 Complexity

The Traveling Salesman Problem (TSP) is a well-known optimization problem in operations research, famous for its computational intractability as the problem size increases. The main objective is to determine the shortest possible route that visits each node exactly once and returns to the origin. Despite its simple definition, the TSP is classified as an NP-hard problem. As the number of nodes increases, the number of possible routes increases significantly. Then, the problem could not be solved optimally in polynomial time, even for moderately sized instances.

In the context of UAV operation planning, the routing problem often shares structural similarities with the TSP, as the UAV must visit a set of target locations while minimizing total travel distance. In the problem that is the subject of this study, the mission for each target can be executed by a set of surveillance coordinates, which increases the complexity. The problem is a more complex variant of the TSP, still belonging to the NP-hard class.

This complexity necessitates the use of heuristic and metaheuristic approaches when the problem size is larger.



CHAPTER 4

A METAHEURISTIC APPROACH FOR THE PROBLEM

The complexity inherent in the problem leads to prolonged solution times and substantial computational effort. Considering that the unmanned aerial vehicle (UAV) routing problem is a daily operational task for an operator, it can be classified as an operational-level decision-making problem. Consequently, addressing this problem through a metaheuristic approach becomes inevitable.

The metaheuristic approach starts with finding an initial feasible solution for the problem. Then, the algorithm implements a neighborhood procedure iteratively to obtain improved solutions. Figure 4.1 presents the general flow of the proposed algorithm.

The process begins with the construction of an initial feasible tour among target coordinates by using the nearest neighbor-based algorithm and the assignment of a surveillance point to each target for the execution of the offset surveillance mission. With those two decisions, the initial solution will be generated. Then, with the simulated annealing-based neighborhood search algorithm, the neighbor tours are generated by using both 1-opt and 2-opt procedures, and surveillance point selection using the same procedure for the updated tour is executed. During the evaluation of the neighbor solution, the simulated annealing algorithm is employed to obtain a balance between exploration and exploitation. Overall, the proposed heuristic integrates both greedy and simulated annealing methodologies.

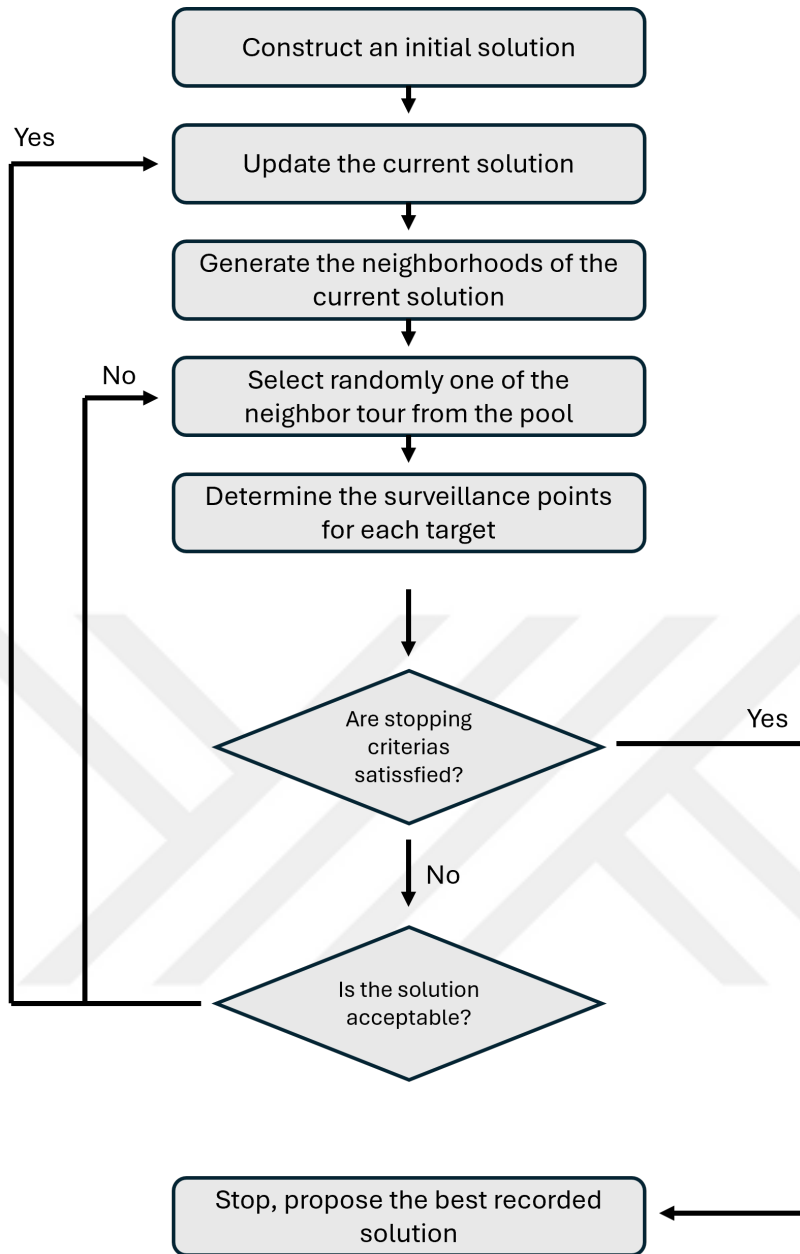


Figure 4.1: General Flow of the Metaheuristic Approach

The proposed metaheuristic solution approach has two main components: (i) initial tour construction and selection of surveillance points for the offset surveillance mission, and (ii) improvement of the current solution using a simulated annealing algorithm. The detailed algorithmic structures of them are provided in Section 4.1 and Section 4.2.

4.1 Nearest Neighbor-based Initial Solution Algorithm

As stated earlier, the primary objective of this study is to construct a tour that starts from the runway, visits each target through an appropriate surveillance coordinate, and returns to the runway. To achieve this, an initial solution is first generated. The initial solution procedure is divided into two phases. In the first phase, the visiting order of the targets is determined using a nearest neighbor-based approach, as described in Algorithm 1: Initial Tour Construction with Nearest Neighbor Algorithm. In the second phase, surveillance points corresponding to the predetermined target order are selected through a dynamic programming procedure for SPT, the details of which are presented in Algorithm 2: Selection of Surveillance Point.

Algorithm 1 Initial Tour Construction with Nearest Neighbor Algorithm

Input : Number of Target(n), Target Set, Target Distances with Runway, Distance Matrix Between Target Coordinates

Output: Tour

- 1 Mark all targets as unvisited
 - 2 Select the target(u) nearest to the runway,
assign that target to the first tour position, $Tour[1] \leftarrow u$,
 - 3 and update the unvisited target set by removing u
 - 4 **for** order $t \in [2, n]$ **do**
 - 5 *Select the nearest unvisited target(u) to the previously scheduled one, $Tour[t - 1]$,
assign that target to the t^{th} place, $Tour[t] \leftarrow u$,
and update the unvisited target set by removing u*
 - 6 **return** $Tour$
-

The algorithm constructs a target visiting order based on the given target set and the associated distance matrices, which are provided as input. At the beginning, all targets are marked as unvisited, and the algorithm starts with the target that has the minimum distance to the runway, placing it in the first position of the tour. Then, at each subsequent step, the algorithm identifies the unvisited target that is nearest to the last visited target in the partial tour and appends it to the sequence. After a target is added to the tour, it is removed from the set of unvisited targets to ensure that each

target is visited exactly once. This process continues until all targets are included in the tour, resulting in an initial feasible tour that serves as the starting point for further optimization.

Another critical aspect of the unmanned aerial vehicle routing problem in offset surveillance missions is the selection of surveillance coordinates for each target. In addition to determining the visiting order of the targets, this component of the problem constitute a shortest-processing-time (SPT) scheduling problem can be solved with an exact method. To maintain computational efficiency and also the solution quality, a dynamic programming approach is adopted for assigning surveillance points. The method developed by Dijkstra, 1959 is used in our solution procedure. The details of this procedure are presented in Algorithm 2: Surveillance Point Selection with Dynamic Programming Approach.

Algorithm 2 Surveillance Point Selection with Dynamic Programming Approach

Input: Tour, Distance Matrix Between Surveillance Sets' Coordinates

Output: Surveillance Point for Each Target, Objective Function Value

1 Step 1: Initialization

A directed graph is introduced with given tour

The stage represents the visiting order of the targets, whereas the state represents the surveillance point for the corresponding target at that stage

The shortest distance from runway to every stage will keep, the visiting order

2 Step 2: Calculation of Shortest Distance

For every stage node, calculate the distances by adding the distance between the corresponding surveillance points to the previous stage level.

Assign the minimum of that as a shortest distance to node

3 Step 3: Construct the Shortest Path Tree

In the last stage, the tour is completed by turning the runway

Select the shortest distance, and form the shortest path tree

4 return *Surveillance – Points, Objective – Value*

Given the predetermined tour and the distance matrix between the surveillance sets, the dynamic programming approach determines the surveillance coordinates of each

target. Algorithm iteratively solve the subset problems and assigns a surveillance point for each target. With stepwise manner, the stages are constructed by establishing the relation between the previous one. In the last stage, an optimal surveillance coordinate selection is obtained for given tour. After determining the surveillance coordinates for a given tour, a complete solution is obtained, enabling the calculation of the objective function value, namely the total tour length. It is calculated by summing the distances between consecutive surveillance points and the runway.

However, this sequential approach may lead to missing better solutions. Therefore, a simulated annealing-based algorithm is incorporated to further refine the initial solution by exploring alternative tour configurations and improving solution quality in the following part.

4.2 A Simulated Annealing Based Neighborhood Search Algorithm

Metaheuristic neighborhood search algorithms enable us to explore the solution space in addition to the exploitation. Since the special case of the problem in this thesis is the well-known TSP in literature, is regarded as NP-Hard, the solution space is also more complex. Therefore, exploration with a metaheuristic approach is a critical issue to handle the local optima problem. Thus, a simulated annealing-based neighborhood search algorithm is implemented to improve the current solution obtained from greedy approaches.

As proposed by Kirkpatrick et al., 1983a, simulated annealing is a stochastic optimization technique inspired by the physical annealing process. In this process, a material is heated to a high temperature and then slowly cooled, allowing its atoms to rearrange into a stable crystalline structure. The fundamental idea of the simulated annealing algorithm is to imitate the annealing process in physics by allowing occasional acceptance of worse solutions with a decreasing probability. This means that instead of constantly moving to better neighborhood solutions, the algorithm occasionally accepts worse solutions with a probability that decreases over time. That enables the search to escape local optima at the beginning and gradually converge to high-quality solutions as the search progresses.

In the algorithmic analogy, in each iteration, a neighboring solution is generated and evaluated. If the new solution improves the objective value, it is always accepted; otherwise, it may still be accepted with a probability that depends on the deterioration in the objective value and the level of toleration limit. In the algorithm, the acceptance of worse solutions is controlled by the Metropolis criterion. If the objective function value increases by Δf , the new solution is accepted with a probability defined as:

$$P = \exp \left(-\frac{\Delta f + \lambda}{\tau} \right)$$

where τ denotes the current temperature parameter, the current temperature is updated through the search with a cooling rate to decrease the acceptance probability. Therefore, this probability, often referred to as the *acceptance level*, is relatively high at the beginning of the search when τ is large, and it gradually decreases as the temperature is reduced. As the temperature is gradually reduced according to a cooling schedule, the algorithm transitions from exploration to exploitation, converging towards high-quality solutions.

In the simulated annealing algorithms, the parameter set determines the search dynamics. The *temperature* parameter determines the likelihood of accepting worse solutions to escape local optima. This value is initialized as τ_0 and is gradually reduced according to a *cooling rate* (denoted by α) over iterations. A higher α results in a slower cooling process, allowing more exploration, whereas a lower α accelerates convergence. Additionally, a *stopping criterion parameter* (such as λ) can be used to limit the number of iterations at a given temperature or to terminate the search when no improvement is observed. The careful tuning of these parameters is essential, as they collectively influence the balance between exploration of the solution space and exploitation of high-quality solutions.

Algorithms 3 and 4 present the main procedure of the simulated annealing-based neighborhood search algorithm designed in this study. The algorithm starts by setting the initial parameters, including the starting temperature, cooling rate, acceptance level, and the maximum iteration number. Once these parameters are initialized, a neighborhood set is generated by constructing all possible order pairs between targets, ensuring that each pair corresponds to a potential swap move in the solution. The subsequent steps of the main loop involve the generation of a neighborhood solution and the systematic evaluation of it iteratively.

At each iteration, a random pair (i, j) is selected from the neighborhood set, and two candidate solutions are constructed by applying 1-Opt and 2-Opt operations. In the 1-Opt move, the i -th target is reinserted into the j -th position, whereas in the 2-Opt move, the positions of the i -th and j -th targets are exchanged. Figure 4.2 is provided to illustrate the neighborhood movement.

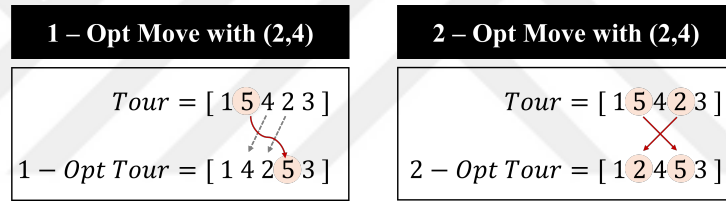


Figure 4.2: Illustration of the 1-Opt and 2-Opt Neighborhood Movements

For both candidate tours, the objective function values are recalculated using the surveillance point selection algorithm, and the better of the two is selected as the candidate solution. If this solution improves upon the current best solution, it is directly accepted and updates the record of the best tour. If this candidate solution improves upon the current solution, it is also directly accepted. Otherwise, acceptance is determined by the Metropolis criterion, where worse solutions may still be accepted with a probability that depends on the deterioration level, the acceptance parameter, and the current temperature. Subsequently, the temperature and iteration count are updated, and the algorithm proceeds iteratively until the maximum number of iterations is reached or no neighbor remains available.

Algorithm 3 Simulated Annealing Main Algorithm

Input : Number of target (n), Current Solution, Objective Value of the Current Solution (opt_t), Distance Matrix Between Surveillance Sets' Coordinates, Initial Temperature (τ_0), α , λ , Iteration Number

Output: $T_{initial}$, $opt_{initial}$, T_{best} , opt_{best}

1 Set the Simulated Annealing Parameters:

Initial Temperature(τ_0)=1000

Cooling Rate(α)=0.95

Acceptance Level (λ)=1

Iteration Number=10000

Iteration=1

2 Initialize Neighbors Set: Generate all possible order pairs, i not equal to j

Algorithm 4 Simulated Annealing Main Algorithm (Part 2)

Main Loop:

```
while  $Iteration < Iteration\_Number$  do
    Select one of the pair  $(i, j)$  randomly from  $Neighbors$ 

    Construct new candidate solutions by using 1-Opt and 2-Opt algorithms with pair  $(i, j)$ 

    1-Opt: Update the current tour by putting the  $i$ th ordered target to  $j$ th order and making appropriate arrangements,  $T\_Temp\_1$ 

    Calculate the objective function by using the selection of surveillance point algorithm for the new tour,  $opt\_temp\_1$ ;

    2-Opt: Update the current tour by changing the place of the  $i$ th ordered target and  $j$ th ordered target,  $T\_Temp\_2$ 

    Calculate the objective function value by using the selection of surveillance point algorithm for the new tour,  $opt\_temp\_2$ ;

    Select better of  $opt\_temp\_1$  and  $opt\_temp\_2$  as  $opt\_temp$  with tour  $T\_Temp$ 

    if  $opt\_temp < opt_{best}$  then
         $T_{best} \leftarrow T\_Temp$ ,  $opt_{best} \leftarrow opt\_temp$ 

    if  $opt\_temp < opt\_t$  then
        Accept  $T\_Temp$  as new  $T$   $opt\_t \leftarrow opt\_temp$ , and initialize the neighborhood

    else
         $Gap(\Delta) \leftarrow opt\_temp - opt\_t$   $RHS \leftarrow \exp(-(\Delta + \lambda)/\tau)$ 
        Generate random number  $r \in [0, 1]$ 
        if  $r < RHS$  then
            Accept  $T\_Temp$  as new  $T$   $opt\_t \leftarrow opt\_temp$ , , and initialize the neighborhood set
        else
            Remove  $(i, j)$  from  $Neighborhood$  if  $|Neighborhood| > 1$ 

     $\tau \leftarrow \tau \cdot \alpha$ 
     $Ite \leftarrow Ite + 1$ 

return  $T_{best}$ ,  $opt_{best}$ 
```



CHAPTER 5

COMPUTATIONAL RESULTS

In this study, we solve the problem under different parameter settings to evaluate the performance of both the exact and metaheuristic solution approaches in various scenarios and ensure that a proper solution method is offered. For that purpose, this section provides a detailed evaluation and critical discussion of the computational results obtained in the context of the unmanned aerial vehicle routing problem under offset surveillance operations.

First of all, the problem environment is introduced briefly, and the experimental factors are shared in Section 5.1.

Section 5.2 presents the parameter selection process for the metaheuristic approach.

Section 5.3 discusses the performance of exact and metaheuristic approaches and provides a comparative analysis. In that part, the performance of the mathematical model is provided for different scenarios, and the challenge of the exact approach will be underlined. The comparison between the exact and metaheuristic approaches is shared to demonstrate the usage of the offered metaheuristic solution approach in such a problem. In addition to that, the consistency of the metaheuristic approach is criticized to assess the reliability of its performance.

In the computational experiments, all problem instances were solved using Gurobi 12.0 on a computer equipped with an Intel(R) Core(TM) i7-4770S processor, 16 GB of RAM, and a Windows operating system. The metaheuristic algorithms were implemented in Python and executed via the Spyder IDE.

5.1 Experimental Settings

In an unmanned aerial vehicle routing problem with offset surveillance missions, the sequence of targets and the selection of the surveillance points are the decisions that need to be made. The unmanned aerial vehicle (UAV) is assumed to take off and land from a runway, and it must generate a closed tour that begins and ends at this location. It is assumed that there are no obstacles or NOTAM areas that prevent the direct travel between the target coordinates in the assigned attitude. Also, the UAV is assumed to travel at a constant speed throughout the operation, disregarding variations due to environmental factors such as wind. The objective is, therefore, to determine the shortest possible tour, where the tour length is computed as the sum of the direct (Euclidean) distances between the consecutive coordinates.

In such a system, the operation area, which is the space that the UAV's responsible region, determines the problem dynamics. The operator which provide a service like tracking and surveillance via unmanned aerial vehicle, first divides the region into problem spaces to manage properly. In our experiment, the problem area is modeled as a rectangular region with an area approximately equivalent to that of the Southeastern Anatolia Region of Türkiye. It is assumed that the runway, the takeoff point, and an integrated ground control station are positioned in accordance with the geographical coordinates of Ankara, the capital city of Türkiye. Figure 5.1 shows a visual representation of the problem area. The area and the runway location are classified as the environmental factors of the problem in this study.

The number of targets, the enumeration of the surveillance arc, and the offset distance can be seen as the input of our problem structure, and are described as experimental factors below.

Number of Targets:

Within the experimental design, the number of targets to be visited is selected as one of the primary experimental factors. In the Traveling Salesman Problem (TSP), the complexity is strongly dependent on the number of nodes to be visited. Each addi-

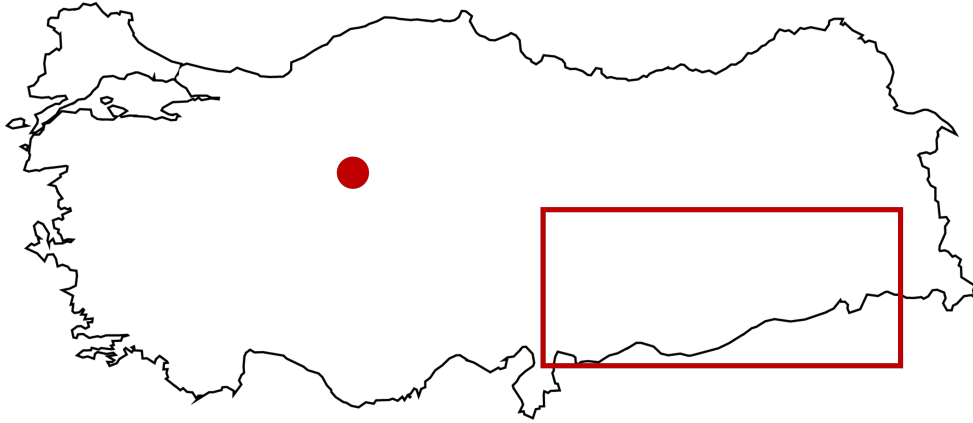


Figure 5.1: Problem Area

tional node increases the number of possible tours exponentially, making the problem computationally hard. A similar effect is observed in the unmanned aerial vehicle (UAV) routing problem considered in this study, where the number of targets plays a role equivalent to the node size in TSP. That will highlight the inherent NP-hardness of the problem and justify the necessity of employing heuristic and metaheuristic approaches when dealing with large-scale instances.

The number of targets is considered at five levels—5, 10, 20, 40, and 80—representing progressively increasing levels of problem complexity. These levels are designed to explore the scalability limits of the proposed approaches in experimental design.

Surveillance Point Set Size:

The second major issue of the problem is the determination of surveillance coordinates along an arc around each target. This aspect introduces additional complexity into the modeling process, which is addressed in this study through an enumeration procedure. The arc is discretized into a finite number of candidate points, then one of them is selected for each of the targets. The granularity of this discretization plays a critical role: a higher number of subdivisions increases the solution accuracy by providing a more precise representation of the feasible surveillance positions, but at the same time, it significantly enlarges the problem size and computational burden. Conversely, a smaller number of subdivisions reduces the problem complexity but

may compromise solution quality due to limited spatial resolution. Therefore, the Surveillance Point Set Size is treated as the second experimental factor -with levels 5, 10- to examine the trade-off between solution accuracy and computational efficiency.

Offset Distance:

The third experimental factor examined in this study is the offset distance, which defines the spatial separation between a target and its associated surveillance points. This parameter directly influences both the geometry of the feasible surveillance region and the resulting total tour length, thereby making it a critical factor in the experimental design. The concept of offset distance stems from operational and safety considerations in UAV surveillance missions, where direct overflight of targets may not always be operationally feasible or desirable. Moreover, the offset distance is inherently linked to the payload and sensor capabilities of UAVs. With the advancements in camera technologies, detection can currently be achieved from distances of up to 30 km, and this range is expected to increase further with future technological improvements. For this reason, the offset distance is incorporated as an experimental factor in order to assess its impact on the performance of the proposed approaches and to anticipate its implications in practical applications.

Finally, the levels of all experimental factors are summarized in Table 5.1: Levels of the Factors in Scenarios. Based on these factor settings, a total of 20 distinct scenarios are generated. For each scenario, five randomly created instance problems are solved, resulting in 100 sample problem instances in total, which are used to evaluate the performance of the solution approaches.

It should also be noted that, due to the stochastic nature of the metaheuristic approach, each problem instance is solved five times using the metaheuristic algorithm. This repetition is performed to evaluate the consistency and robustness of the results obtained from the heuristic method.

Table 5.1: Levels of the Factors in Scenarios

Parameter Name	Levels
Number of Targets:	5, 10, 20, 40, 80
Surveillance Point Set Size:	5, 10
Offset Distance:	30, 50

5.2 Parameter Tuning for Simulated Annealing Algorithm

The performance of the Simulated Annealing algorithm is highly sensitive to its parameter settings, including the initial temperature (τ_0), cooling rate (α), the λ coefficient in the acceptance probability equation, and the maximum number of iterations (*IterationNumber*). These parameters directly influence both the quality of the final solution and the computational time required by the algorithm.

To determine appropriate parameter values, an experimental study was conducted in which multiple configurations of the simulated annealing algorithm were tested and compared. Under different parameter settings, the objective value of the current solution was tracked across iterations, and the progression of the search was visualized through graphical representations. Such a graph provides valuable insights into how different parameter settings influenced the algorithm's behavior, particularly for convergence speed and quality of the final solution. For an appropriate configuration of the parameter set, it is essential to achieve a balance between exploration and exploitation. Specifically, exploration should dominate in the initial stages of the search, where the algorithm deliberately accepts worse solutions to diversify the search space. Then, exploitation takes on greater significance, which means the algorithm gradually shifts its focus toward refining the solution and converging to higher-quality neighborhoods. The results from the experiments validate that parameter settings maintain this balance.

Based on the comparative results, the parameter set $\tau_0 = 1,000$, $\alpha = 0.95$, $\lambda = 1$, and *IterationNumber* = 10,000 was selected as the most effective configuration, offering a balance between solution quality and computational efficiency. In order

to ensure that a convergence is also obtained before the termination, the maximum number of iterations is determined appropriately.

The outcome of these experiments was plotted against the iteration count, as illustrated in Figure 4.2: Current Solution at the Iteration. The figure presents the evolution of the current solution over the iterations with the selected parameter set for five different problem samples in different colors.

As illustrated in Figure 5.2, most of the problem instances reached a steady state before 2,000 iterations. Nevertheless, since extending the maximum iteration number does not significantly increase the computational burden, the limit was set to 10,000 to guarantee that the exploration phase was fully completed and convergence was achieved. At the early stages of the search, the algorithm prefers to accept worse solutions, which enables us to escape from a local optima, and subsequently progresses toward better solutions as the iterations continue. Accordingly, setting the initial temperature (τ_0) to 1,000 and the cooling rate to 0.95 proved to be appropriate, as these parameters successfully balanced exploration and exploitation throughout the search process.

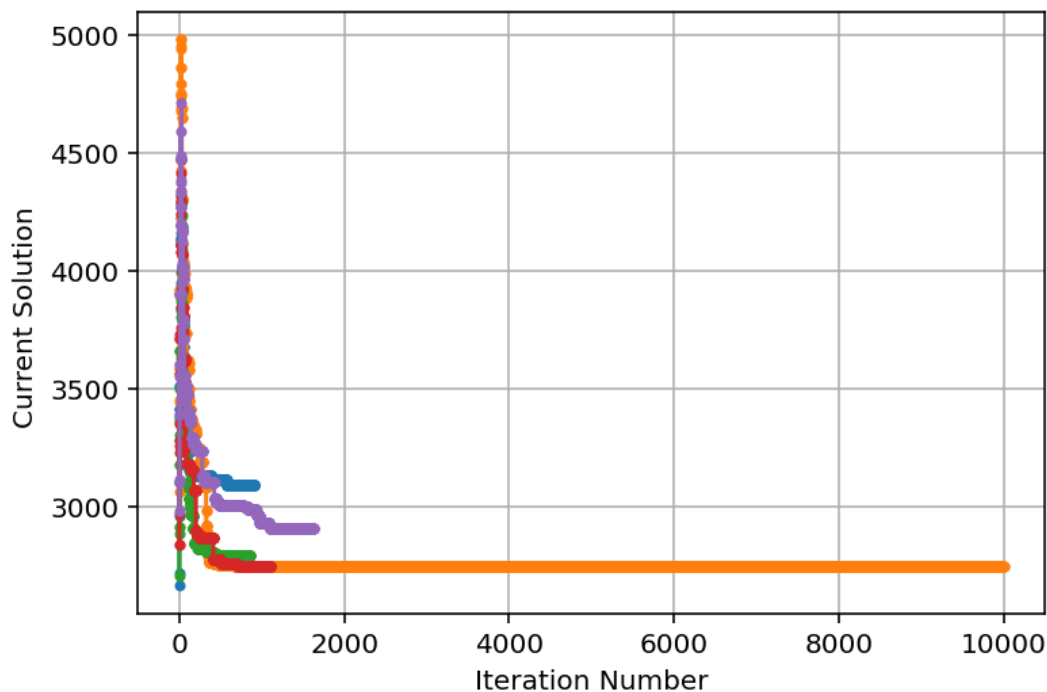


Figure 5.2: Current Solution at the Iteration

5.3 Performance Evaluation and Discussion

The unmanned aerial vehicle routing problem in surveillance missions is solved by using both exact and metaheuristic solution methods in this study. An experiment is designed, and the results are shared to ensure that a systematic solution methodology is proposed for the defined problem in this section.

Before the detailed discussion, the general summary of the results is shared. Table 5.2 provides the number of best solutions obtained by the exact and metaheuristic approaches, as well as their corresponding average CPU times, under different experimental settings. The findings reveal a noticeable difference in the effectiveness of the exact and metaheuristic approaches based on the size of the problem. As can be seen from the table, for small instances, when the number of targets is 5, the exact method consistently finds the best solutions in a reasonable time. However, the CPU times rise sharply to around 3,600 seconds after that level. By considering both the quality and the computational efforts, the exact method can be used only for small cases. On the other hand, the metaheuristic approach is suggested to deal with such a problem.

In the following part of this section, the discussion will be detailed with the exact method performance evaluation in Section 5.3.1, which serves as a reference point for the description of solution quality. The performance of the exact method is stated in terms of both the quality and combinatorial effort in the section. Next, a comparison is made between the heuristic approach and the exact method in Section 5.3.2. Then, the assessment of the metaheuristic approach for consistency is discussed in Section 5.3.3. Unlike deterministic methods, metaheuristics integrate stochastic elements into the search process.

Table 5.2: Exact Approach vs. Metaheuristic Performance Summary

Scenario				Number of Best Time			Average CPU Time	
Number of Targets	Surveillance	Point Set Size	Offset Distance	Exact	Metaheuristic	Total	Exact	Metaheuristic
5	5	30	5	0	5	5	0.69	0.01
5	10	30	5	0	5	5	8.37	0.02
5	5	50	5	0	5	5	1.03	0.02
5	10	50	5	0	5	5	7.86	0.02
10	5	30	4	1	4	5	TL	0.03
10	10	30	4	1	4	5	TL	0.03
10	5	50	5	0	5	5	TL	0.03
10	10	50	5	0	5	5	TL	0.03
20	5	30	5	0	5	5	TL	0.34
20	10	30	0	5	0	5	TL	0.26
20	5	50	5	0	5	5	TL	0.23
20	10	50	1	4	1	5	TL	0.27
40	5	30	0	5	0	5	TL	2.04
40	10	30	0	5	0	5	-	2.13
40	5	50	0	5	0	5	TL	2.25
40	10	50	0	5	0	5	-	2.05
80	5	30	0	5	0	5	-	7.47
80	10	30	0	5	0	5	-	6.87
80	5	50	0	5	0	5	-	7.05
80	10	50	0	5	0	5	-	7.64

*TL: Time Limit

5.3.1 Performance of the Exact Method

To provide a comprehensive evaluation of the performance of the exact solution approach, the corresponding performance measures of the mathematical model are presented in Tables 5.3, 5.4, and 5.5.

Table 5.3 illustrates the solution status of the exact approach under different experimental scenarios. The results indicate that, as the number of targets increases, obtaining an optimal solution becomes progressively more challenging. Beyond a certain problem size, even a feasible solution could not be identified within acceptable computational limits. Similarly, when the size of the surveillance point set is doubled, the solution space expands combinatorially, rendering exhaustive search impractical and leading to infeasibility within reasonable time limits. This behavior parallels the pattern observed with the increasing number of targets. Overall, across the set of scenarios, approximately one-third of all runs resulted in the absence of a feasible solution, which underlines the computational hardness of the problem.

In Table 5.4, the performance of the exact method based on the computation time is provided. As can be seen in the table, the CPU time required to solve the problem increases as the problem's size grows. While small instances are solved in less than a second, the computational effort escalates dramatically for medium- and large-scale cases. Therefore, no more optimal solution could be obtained due to the time limit set by the planning timeline; only an hour is given to the mathematical model.

To reinforce the performance assessment of the exact method, optimality gap(%) values are shared in Table 5.5: Performance of the Exact Method Based on Optimality Gap. For small-scale scenarios, the gaps remain at 0%, which means exact optimality. However, as the number of targets increases, the optimality gaps expand significantly, often reaching levels above 50% for medium to large instances. This indicates that even when the solver concludes with a feasible solution, the disparity between the best-discovered and demonstrably optimal solutions continues to be excessively large. Together, these results show that while the exact method ensures solution accuracy for small problems, it cannot effectively cope with the computational complexity of larger scenarios.

Beyond these results, it can be said that the method often fails to solve within acceptable time limits, rendering it impractical for real-time or operational decision-making contexts.



Table 5.3: Performance of the Exact Method Based on Solution Status

Scenario				Solution Status			
Number of Targets	Surveillance	Point Set Size	Offset Distance	Optimal	Best Feasible	No Solution	Total
5	5		30	5	0	0	5
5	10		30	5	0	0	5
5	5		50	5	0	0	5
5	10		50	5	0	0	5
10	5		30	0	5	0	5
10	10		30	0	5	0	5
10	5		50	1	4	0	5
10	10		50	0	5	0	5
20	5		30	0	5	0	5
20	10		30	0	5	0	5
20	5		50	0	5	0	5
20	10		50	0	5	0	5
40	5		30	0	2	3	5
40	10		30	0	0	5	5
40	5		50	0	3	2	5
40	10		50	0	0	5	5
80	5		30	0	0	5	5
80	10		30	0	0	5	5
80	5		50	0	0	5	5
80	10		50	0	0	5	5

Table 5.4: Performance of the Exact Method Based on CPU Time

Scenario				CPU Time			Number of Solution
Number of Targets	Surveillance	Point Set Size	Offset Distance	Min	Mean	Max	
5	5	30		0.6	0.686	0.75	5
5	10	30		6.2	8.366	10.55	5
5	5	50		0.5	1.026	2.37	5
5	10	50		5.3	7.858	11.3	5
10	5	30		TL	TL	TL	5
10	10	30		TL	TL	TL	5
10	5	50		TL	TL	TL	5
10	10	50		TL	TL	TL	5
20	5	30		TL	TL	TL	5
20	10	30		TL	TL	TL	5
20	5	50		TL	TL	TL	5
20	10	50		TL	TL	TL	5
40	5	30		TL	TL	TL	2
40	10	30		-	-	-	0
40	5	50		TL	TL	TL	3
40	10	50		-	-	-	0
80	5	30		-	-	-	0
80	10	30		-	-	-	0
80	5	50		-	-	-	0
80	10	50		-	-	-	0

*TL: Time Limit

Table 5.5: Performance of the Exact Method Based on Optimality Gap

Scenario				Optimality Gap			Number of Solution
Number of Targets	Surveillance	Point Set Size	Offset Distance	Min	Mean	Max	
5	5	30	0%	0%	0%	0%	5
5	10	30	0%	0%	0%	0%	5
5	5	50	0%	0%	0%	0%	5
5	10	50	0%	0%	0%	0%	5
10	5	30	6%	13%	20%	20%	5
10	10	30	19%	28%	40%	40%	5
10	5	50	0%	14%	24%	24%	5
10	10	50	20%	28%	35%	35%	5
20	5	30	37%	43%	45%	45%	5
20	10	30	55%	57%	61%	61%	5
20	5	50	38%	43%	47%	47%	5
20	10	50	52%	58%	67%	67%	5
40	5	30	76%	77%	77%	77%	2
40	10	30	-	-	-	-	0
40	5	50	75%	79%	84%	84%	3
40	10	50	-	-	-	-	0
80	5	30	-	-	-	-	0
80	10	30	-	-	-	-	0
80	5	50	-	-	-	-	0
80	10	50	-	-	-	-	0

5.3.2 Comparison of the Performance of the Metaheuristic Approach with Exact Method

As mentioned in the previous part, an alternative metaheuristic solution approach is also offered in this study. It is critical to address its performance against a reference point. Therefore, in Table 5.6 and Table 5.7, the comparisons of the metaheuristic and exact methods' performances are provided. The comparison is conducted with respect to both solution quality and computational effort. Solution quality is evaluated based on the difference between the tour length obtained by the metaheuristic and that of the exact method. The optimality gap is computed by dividing this difference by the objective function value of the exact solution. The results in Table 5.6 reveal that the proposed metaheuristic approach produces objective values that are close to those of the exact solver in small cases. In medium- and large-scale scenarios where the exact solver struggles with optimality gaps, the heuristic even delivers solutions that outperform the best feasible solutions identified by the exact method within the time limits. When the objective value gaps are evaluated within the scenario, it seems that similar differences are obtained.

Table 5.7 is especially shared to illustrate the trade-off between solution quality and computational effort. While the exact method requires extensive computational resources, reaching thousands of seconds in many instances, the metaheuristic achieves better results in only a fraction of that time, often within seconds. Notably, the average objective value gap between the metaheuristic and the exact method remains relatively small (generally under 3% for most scenarios). This indicates that the heuristic achieves near-optimal solutions while maintaining substantial efficiency gains. From a practical perspective, this balance between quality and time makes the metaheuristic solution approach a more suitable alternative for our problem, where timely decision-making is essential in the daily operation planning of an unmanned aerial vehicle.

Table 5.6: Comparison of Objective Values between the Metaheuristic Approach and the Exact Method

Scenario				Objective Value Gap Between the Exact and Metaheuristic Approach			
Number of Targets	Surveillance Point Set Size	Offset Distance		Min	Mean	Max	Number of Comparison
5	5	30		1%	2%	5%	5
5	10	30		0%	2%	5%	5
5	5	50		1%	2%	5%	5
5	10	50		1%	2%	5%	5
10	5	30		0%	3%	8%	5
10	10	30		0%	3%	7%	5
10	5	50		1%	2%	4%	5
10	10	50		1%	2%	3%	5
20	5	30		2%	8%	13%	5
20	10	30		-22%	-16%	-10%	5
20	5	50		2%	6%	11%	5
20	10	50		-34%	-18%	1%	5
40	5	30		-51%	-51%	-50%	2
40	10	30		-	-	-	0
40	5	50		-62%	-53%	-48%	3
40	10	50		-	-	-	0
80	5	30		-	-	-	0
80	10	30		-	-	-	0
80	5	50		-	-	-	0
80	10	50		-	-	-	0

Table 5.7: Performance Comparison between the Metaheuristic Approach and the Exact Method

Scenario				Average Objective Value Gap		Average CPU Time	
Number of Targets	Surveillance Point Set Size	Offset Distance			Exact	Metaheuristic	
5	5	30	2%		0.69	0.01	
5	10	30	2%		8.37	0.02	
5	5	50	2%		1.03	0.02	
5	10	50	2%		7.86	0.02	
10	5	30	3%		TL	0.03	
10	10	30	3%		TL	0.03	
10	5	50	2%		TL	0.03	
10	10	50	2%		TL	0.03	
20	5	30	8%		TL	0.34	
20	10	30	-16%		TL	0.26	
20	5	50	6%		TL	0.23	
20	10	50	-18%		TL	0.27	
40	5	30	-51%		TL	2.04	
40	10	30	-		-	2.13	
40	5	50	-53%		TL	2.25	
40	10	50	-		-	2.05	
80	5	30	-		-	7.47	
80	10	30	-		-	6.87	
80	5	50	-		-	7.05	
80	10	50	-		-	7.64	

*TL: Time Limit

5.3.3 Consistency of the Metaheuristic Approach

Lastly, the consistency of the metaheuristic approach is criticized. As mentioned before, five different sample problems are solved for each scenario, and 5 replications are obtained for each of the problem samples. These replications are taken to evaluate the robustness and the accuracy of the algorithm. Table 5.8 summarizes the calculated coefficient of variation measures. The findings show that the fluctuation in objective values is minimal, with coefficients of variation typically below 7% even for the more complex instances. This low level of variability shows that the randomness inherent in the algorithm does not result in volatility in the outcome.

Given that the metaheuristic includes stochastic elements, consistency needs to be controlled before evaluation. Such reliability is essential for practical implementation, as decision-makers need trustworthy results that do not vary significantly between solutions. The results indicate that the metaheuristic not only generates competitive solutions in terms of both quality and efficiency but also provides stability. Thereby enhancing its reputation as a strong alternative to exact optimization in large-scale and time-sensitive scenarios.

To conclude, these complementary analyses provide a comprehensive validation of the methodological framework adopted in this research. By critically analyzing the strengths and weaknesses of the exact and metaheuristic methods, this section adds to a comprehensive understanding of the solution landscape. The insights presented here not only affirm the methodological approaches adopted in the research but also illustrate their practical significance for UAV surveillance operations.

Table 5.8: Assessment of the Consistency of the Metaheuristic Approach

Scenario				Coefficient of Variation (%)		
Number of Targets	Surveillance	Point Set Size	Offset Distance	Min	Mean	Max
5	5	30		0%	0%	0%
5	10	30		0%	0%	0%
5	5	50		0%	0%	0%
5	10	50		0%	0%	0%
10	5	30		0%	0%	1%
10	10	30		0%	0%	1%
10	5	50		0%	0%	1%
10	10	50		0%	0%	1%
20	5	30		0%	2%	4%
20	10	30		0%	2%	5%
20	5	50		0%	4%	7%
20	10	50		0%	3%	5%
40	5	30		3%	5%	7%
40	10	30		3%	5%	8%
40	5	50		4%	6%	8%
40	10	50		5%	8%	10%
80	5	30		4%	6%	9%
80	10	30		1%	6%	11%
80	5	50		3%	6%	8%
80	10	50		4%	7%	9%

CHAPTER 6

CONCLUSION

This thesis has examined the routing problem for unmanned aerial vehicles (UAVs) performing offset surveillance operations, where the objective is to construct a tour that starts from a runway, visits each designated target for a surveillance mission, and returns to the runway. The distinctive aspect of the problem is that the surveillance task is not conducted at the exact coordinate of the target. Due to security and environmental concerns, it is more desirable and feasible to perform the mission from an offset distance. This requirement enlarges the problem by introducing an additional decision layer: the selection of a surveillance coordinate from a predefined set associated with each target.

UAV operators commonly face this problem in their everyday tasks. Given that UAVs are increasingly deployed in both military and civilian surveillance activities due to their operational advantages, the significance and potential of this research are extremely pertinent.

To represent a realistic context, the target area was modeled in accordance with the Southeastern Anatolia region of Turkey, while the runway was located in Ankara, the capital. Since UAV flight altitude is subject to regulatory authority, the study focused exclusively on two-dimensional coordinates. It aimed to identify a minimum-length tour among the feasible surveillance point sets.

The routing problem for an unmanned aerial vehicle (UAV) with offset surveillance operations is modeled to find an exact solution. However, as this problem can be regarded as Generalized Traveling Salesman Problem (GTSP) in the operations research literature, its computational complexity is considerable. As expected for NP-

hard problems, the exact solution method quickly became infeasible for medium- and large-scale instances. This challenge highlighted the need for alternative solution strategies in practical scenarios.

To address this challenge, a metaheuristic solution framework was created by combining a greedy tour initialization with a simulated annealing-based enhancement process. Computational experiments demonstrated that the approach was capable of producing acceptable solutions within reasonable computational times. Furthermore, the robustness of the proposed method across a variety of scenarios confirmed its potential as a decision-support tool for daily UAV operations in offset surveillance missions.

The primary contribution of this thesis is the introduction and definition of a real-life operational problem, unmanned aerial vehicle routing with offset surveillance missions. While UAV path planning has been studied extensively in the literature, the specific case of offset surveillance—where missions are executed from a feasible surveillance point rather than the exact target coordinates—has not previously been addressed.

To the best of our knowledge, this study contributes to the literature by providing both a mathematical formulation and a metaheuristic solution approach, thereby establishing a new research direction and offering a practical solution method for widely faced decision-making problems.

In this study, the effect of wind on unmanned aerial vehicle speed has been disregarded. Nevertheless, the direction of the wind plays a crucial role in the duration of travel. The ground speed is calculated by taking into account the real speed of the vehicle and the vertical component of the wind speed, which changes the effort of the travel between two surveillance coordinates. Therefore, it directly affects both the routing decisions and the overall mission duration. Therefore, incorporating wind dynamics into the model constitutes a promising direction for the future extension of this thesis.

Another future research direction involves expanding the geographical scope of the problem. Evaluating the space effect of the solution approach on results would be

valuable both for the literature and the practical operational aspect.

Additionally, the current study addresses a single-UAV routing problem. Extending the model to consider multiple homogeneous or heterogeneous UAV operations would constitute a promising direction for future research.

To conclude, this study demonstrates that while exact optimization method provides valuable theoretical insights for small-scale instances, metaheuristic approaches offer practical and scalable solutions for the unmanned aerial vehicle routing problem in offset surveillance missions.





REFERENCES

- Applegate, D. L., Bixby, R. E., Chvátal, V., & Cook, W. J. (2006). *The traveling salesman problem: A computational study*. Princeton University Press.
- Banaszak, J., & Smith, K. (2020). A survey of vehicle routing problem applications in uav surveillance. *Journal of Unmanned Aerial Systems*, 12(3), 45–67.
- Bello, I., Pham, H., Le, Q. V., Norouzi, M., & Bengio, S. (2016). Neural combinatorial optimization with reinforcement learning.
- Chen, J., Ye, F., & Li, Y. (2017). Travelling salesman problem for uav path planning with two parallel optimization algorithms. *Progress in Electromagnetics Research Symposium*, 832–837.
- Dantzig, G., Fulkerson, R., & Johnson, S. (1954). Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America*, 2(4), 393–410.
- Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269–271.
- Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1), 53–66.
- Farhi, E., Goldstone, J., & Gutmann, S. (2014). A quantum approximate optimization algorithm.
- Havacılık, H. İ. (2022). *Air spaces class nedir?* [Accessed: 23 August 2025]. <https://herkesicinhavacilik.com/2022/01/28/air-spaces-class-nedir/>
- Held, M., & Karp, R. M. (1962). A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics*, 10(1), 196–210.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press.

- Jafarzadeh, H., Moradinasab, N., & Elyasi, M. (2017). An enhanced genetic algorithm for the generalized traveling salesman problem. *Engineering, Technology & Applied Science Research*, 7(6), 2260–2265. <https://doi.org/10.48084/etasr.1570>
- Joshi, C. K., Laurent, T., & Bresson, X. (2019). An efficient graph convolutional network technique for the traveling salesman problem.
- Keha, A. B., Khowala, K., & Fowler, J. W. (2009). Mixed integer programming formulations for single machine scheduling problems. *Computers & Industrial Engineering*, 56(1), 357–367. <https://doi.org/10.1016/j.cie.2008.06.008>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983a). Optimization by simulated annealing. *Science*, 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983b). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
- Kitjacharoenchai, P., Ventresca, M., Moshref-Javadi, M., Lee, S., Tanchoco, J. M. A., & Brunese, P. A. (2019). Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Computers & Industrial Engineering*, 129, 14–30.
- Luo, Q., Wu, G., Ji, B., Wang, L., & Suganthan, P. N. (2022). Hybrid multi-objective optimization approach with pareto local search for collaborative truck-drone routing problems considering flexible time windows. *IEEE Transactions on Intelligent Transportation Systems*, 23(8), 13011–13025. <https://doi.org/10.1109/TITS.2021.3119080>
- Ma, Y., Hu, M., & Yan, X. (2020). Graph neural networks for uav route optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 56(4), 3210–3223.
- Maza, I., & Ollero, A. (2007). Multiple uav cooperative path planning for coverage missions. *Journal of Field Robotics*, 24(5), 385–404.
- Murray, C. C., & Chu, A. G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54, 86–109.
- Pina-Pardo, J. C., et al. (2022). The traveling with drone resupply. *OR Spectrum*, 44(4), 1045–1086.

- Pop, P. C., Cosma, O., Sabo, C., & Pop Sitar, C. (2024). A comprehensive survey on the generalized traveling salesman problem. *European Journal of Operational Research*, 314, 819–835. <https://doi.org/10.1016/j.ejor.2023.07.022>
- Raj, R., Murray, C., & Chu, A. (2020). The multiple flying sidekicks traveling salesman problem with variable drone speeds. *Transportation Research Part C: Emerging Technologies*, 110, 368–398.
- Research, P. (2025). Unmanned aerial vehicle market size and forecast 2025 to 2034 [Accessed June 2025].
- Rosenkrantz, D. J., Stearns, R. E., & Lewis, P. M. (1977). An analysis of several heuristics for the traveling salesman problem. *SIAM Journal on Computing*, 6(3), 563–581.
- Ryan, J. L., Bailey, T. G., & Moore, J. T. (1998). Reactive tabu search in unmanned aerial reconnaissance simulations. *Proceedings of the Winter Simulation Conference*, 873–879.
- Schmidt, J., & Irnich, S. (2022). New neighborhoods and an iterated local search algorithm for the generalized traveling salesman problem. *EURO Journal on Computational Optimization*, 10, 100029. <https://doi.org/10.1016/j.ejco.2022.100029>
- Xie, J., Garcia Carrillo, L. R., & Jin, L. (2019). An integrated traveling salesman and coverage path planning problem for unmanned aircraft systems. *IEEE Control Systems Letters*, 3(1), 67–72.
- Yang, J., Shi, X., Marchese, M., & Liang, Y. (2008). An ant colony optimization method for generalized tsp problem. *Progress in Natural Science*, 18, 1417–1422. <https://doi.org/10.1016/j.pnsc.2008.03.028>
- Yang, W., & Wang, Y. (2012). Improved simulated annealing algorithm for the generalized traveling salesman problem. *Proceedings of ECC'01*, 1202–1205.
- Zhang, Q., Li, H., & Wang, Z. (2023). Hybrid metaheuristics with machine learning for dynamic uav routing. *Robotics and Autonomous Systems*, 159, 104297.



Appendix A

EXPERIMENTAL STUDY DETAILS FOR PARAMETER TUNING OF THE METAHEURISTIC APPROACH

This appendix section provides visual representations of the experimental study for each sample problem, which are summarized in a compact form in the main report.

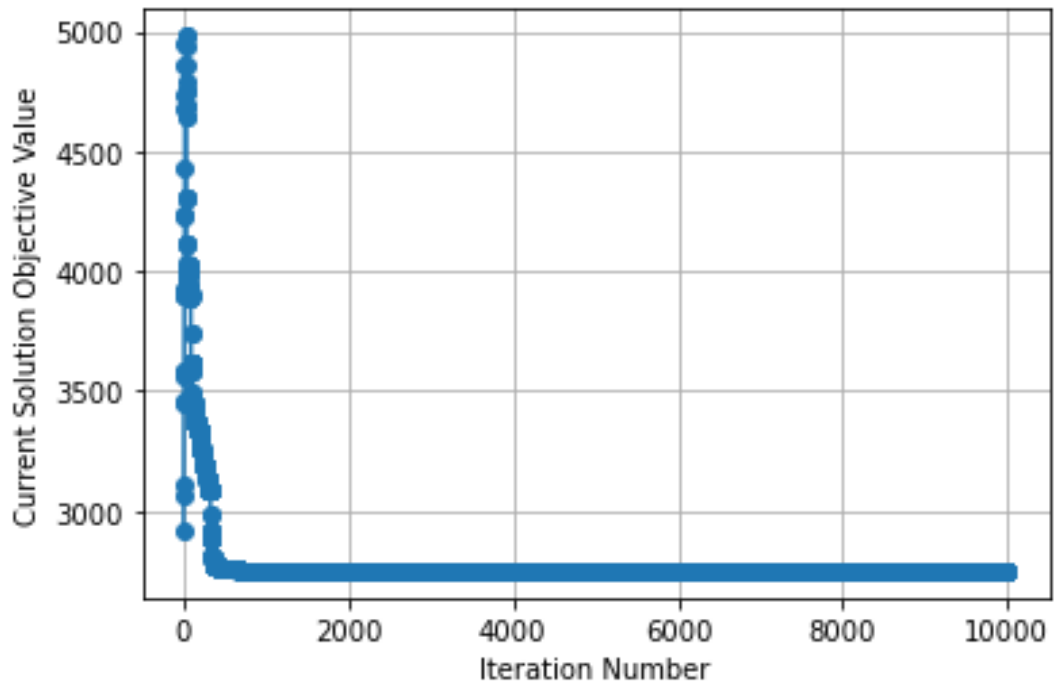


Figure A.1: Current Solution at the Iteration for Sample Problem - 1

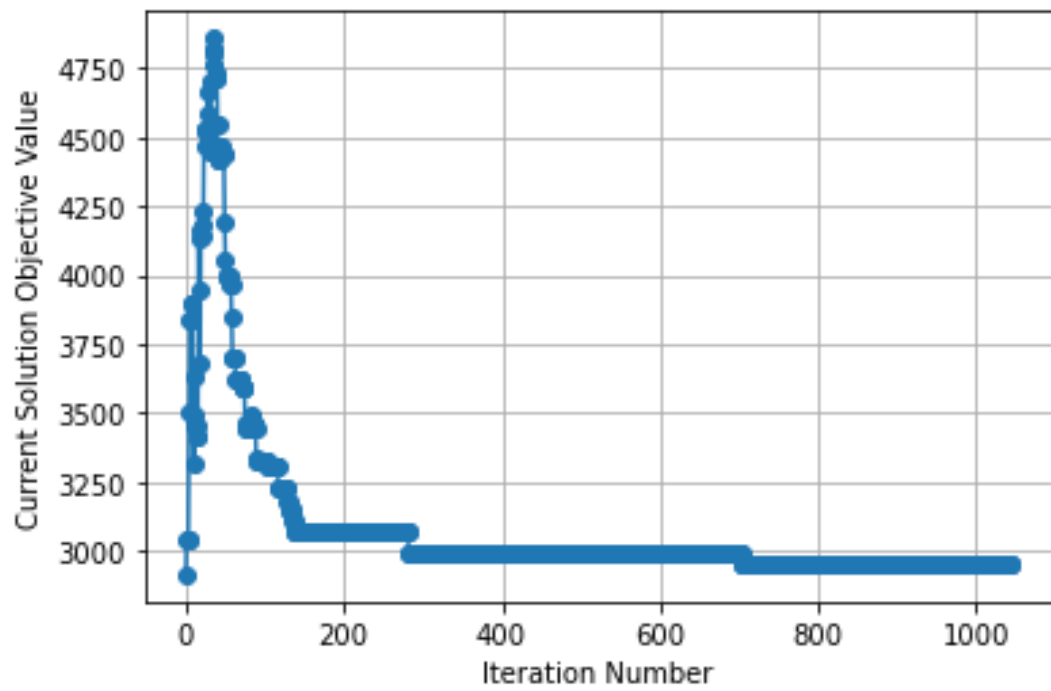


Figure A.2: Current Solution at the Iteration for Sample Problem - 2

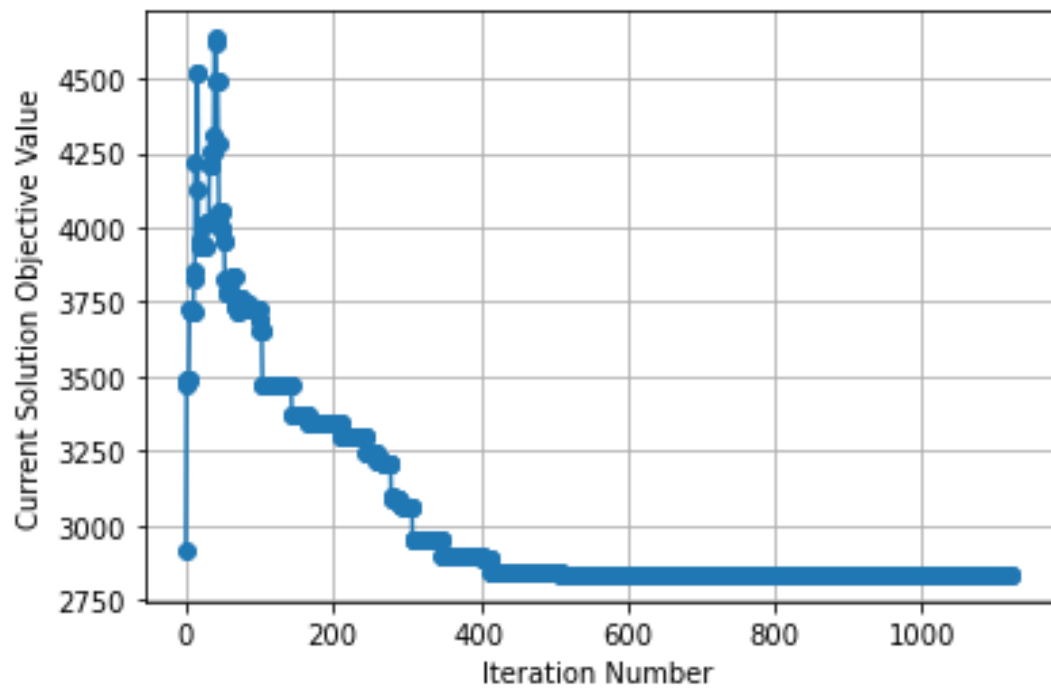


Figure A.3: Current Solution at the Iteration for Sample Problem - 3

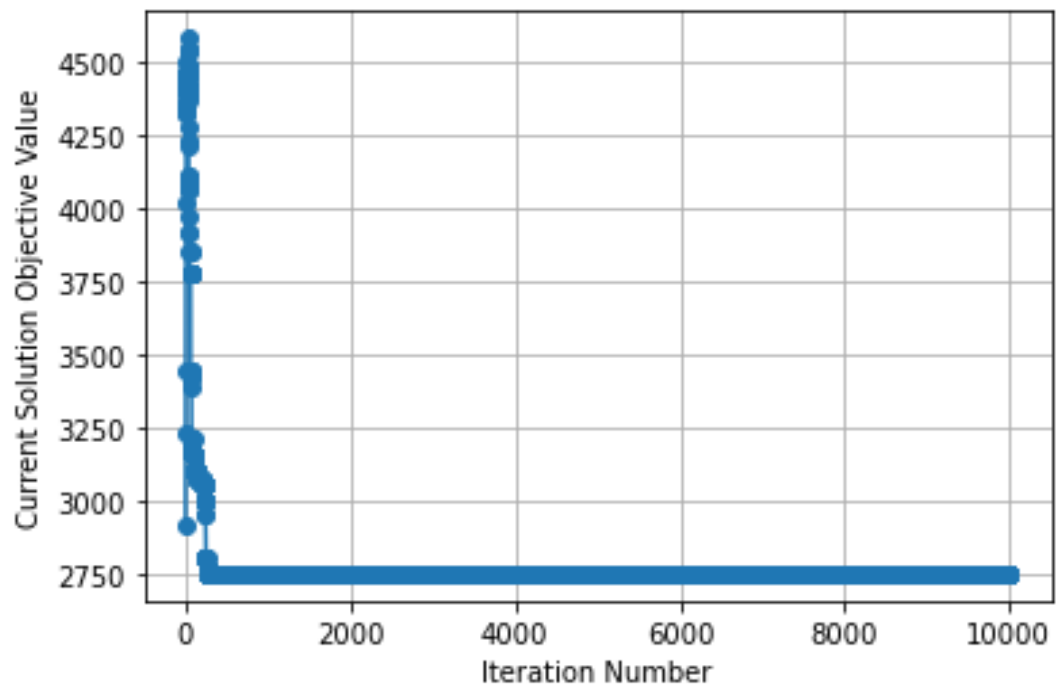


Figure A.4: Current Solution at the Iteration for Sample Problem - 4

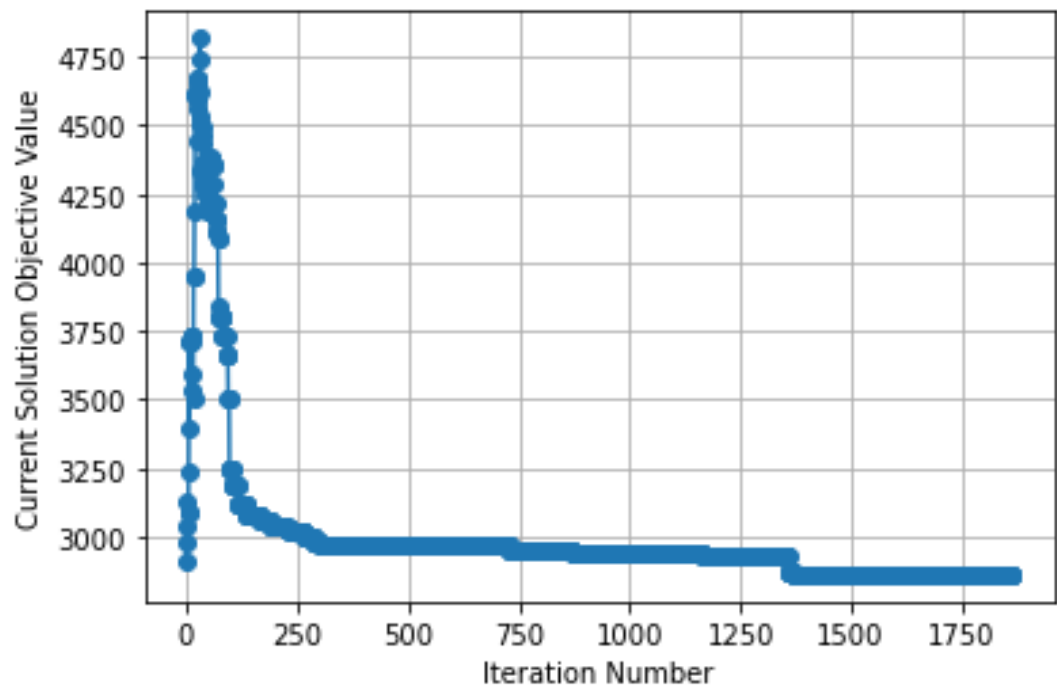


Figure A.5: Current Solution at the Iteration for Sample Problem - 5