DEVELOPMENT OF A COLLABORATIVE DELIVERY SYSTEM WITH UNMANNED AERIAL VEHICLES AND DELIVERY TRUCKS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

OF

ATILIM UNIVERSITY

BY

Salah Aboharba

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF

DOCTOR OF PHILOSOPHY

IN

MODELING AND DESIGN OF ENGINEERING SYSTEMS (MODES) PhD

PROGRAM

(Main Field of Study: Mechatronics Engineering)

MAY 2018

Approval of the Graduate School of Natural and Applied Sciences, Atılım University.

Prof. Dr. Ali Kara
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of **Doctor of Philosophy in Modeling and Design of Engineering Systems Department, Atılım University.**

Assoc. Prof. Dr. Ender Keskinkılıc
Head of Department

This is to certify that we have read the thesis DEVELOPMENT OF A COLLABORATIVE DELIVERY SYSTEM WITH UNMANNED AERIAL VEHICLES AND DELIVERY TRUCKS submitted by **Salah Aboharba** and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy

Assoc. Prof. Dr. Mehmet Turan
Co-Supervisor Assoc.

Asst. Prof. Dr. Kutluk Bilge Arıkan
Supervisor

Examining Committee Members:

Asst. Prof. Dr. Buğra Koku
Mechanical Engineering Department, METU

Assoc. Prof. Dr. Funda Kurtuluş
Aerospace Engineering Department, METU

Asst. Prof. Dr. Hakan Tora
Electronics Engineering Department, Atılım University

Asst. Prof. Dr. Ahmet Güneş
Mechatronics Engineering Department, Atılım University

Asst. Prof. Dr. Kutluk Bilge Arıkan
Mechanical Engineering Department, TED University

Date: May 03, 2018

I declare and guarantee that all data, knowledge and information in this document has been obtained, processed and presented in accordance with academic rules and ethical conduct. Based on these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Salah Aboharba

Signature: ……………….

ABSTRACT

DEVELOPMENT OF A COLLABORATIVE DELIVERY SYSTEM WITH UNMANNED AERIAL VEHICLES AND DELIVERY TRUCKS

Aboharba, Salah

PhD in Modeling and Design of Engineering Systems (MODES)

Supervisor: Asst. Prof. Dr. Kutluk Bilge Arıkan

Co-Supervisor: Assoc. Prof. Dr. Mehmet Turan

May 2018, 91 pages

This thesis studies the new application for an unmanned aerial vehicle in the delivery system. Considering a problem of the limited flight time of UAV due to the small battery package that challenges the distribution of the goods directly from the main warehouse difficult, therefore, a collaborative delivery system with UAVs and delivery trucks is proposed. This research focuses on the optimization of the routing problems where a delivery truck is utilized as the base for the UAV when it performs a delivery task. First, the mathematical formulation is developed, with two stages, namely the UAV power consumption model and integer linear programming model, followed by the problem being solved with the K-means algorithm (to partition customers into groups and find the best location for the delivery truck) and with an ant colony optimization algorithm and nearest neighbor algorithm to tackle the routing problem for the UAV for each group. All the algorithms are implemented in MATLAB to find the location of the delivery truck, to minimize the distance traveled and minimize delivery time taking into account the power consumption of UAVs. Finally, comparisons between this system and truck usage is presented. The results show that the delivery time in the collaborative delivery system is reduced compared with truck only usage. Moreover, the issue of limited flight time is solved by applying this system. In addition, a method is developed to weight between the highest demand and shortest distance for the UAV to select a path at minimum power consumption when the demand of the customers is not equal. This method is enforced in nearest neighbor algorithm and ant colony optimization algorithm and the results show that nearest neighbor algorithm is more efficient then ant colony optimization algorithm.

Keywords: UAV; delivery optimization routing problem; integer linear programming; K-means cluster algorthm; nearest neighbor algorithm; ant colony optimization algorithm**.**

ÖZ

İNSANSIZ HAVA ARACI VE FIRLATMA KAMYONUNUNDAKİ FIRLATMA SİSTEMİNİN BİRLİKTE ÇALIŞMASININ GELİŞTİRİLMESİ

Aboharba, Salah
Mühendislik Sistemlerinin Modellenmesi ve Tasarımında Doktora (MODES)
Tez Yoneticisi: Yar. Doç. Dr. Kutluk Bilge Arıkan
Ortak Tez Yoneticisi: Doç. Dr. Mehmet Turan

Mayıs 2018, 91 sayfa

Bu tez, insansız hava aracının (İHA) paket dağıtma amacıyla farklı bir şekilde kullanımını incelemektedir. İHA'nın paketleri doğrudan ana depodan alarak dağıtmasını zorlaştıran kısıtlı batarya kapasitesi ve sınırlı uçuş süresi problemleri göz önünde bulundurularak, İHA ve dağıtım kamyonu işbirliği yapan bir sistem önerilmektedir. Bu çalışma, paketleri de taşıyan bir kamyon ve bu kamyonu üs olarak kullanan İHA'nın dağıtım güzergâhı optimizasyonuna odaklanmıştır. Öncelikle, güç tüketimi ve tam sayılı doğrusal programlama modelleri geliştirilmiş ve ardından paketlerin teslim edileceği müşterileri gruplara ayırmak ve kamyon için en uygun bekleme konumunu hesaplamak üzere K-ortalama algoritmasından faydalanılmıştır. Her bir müşteri kümesi içerisinde İHA'nın izleyeceği rota ise karınca kolonisi optimizasyon algoritması ve en yakın komşuluk algoritması ile hesaplanmıştır. Teslimat kamyonu için en uygun konumun hesaplanması, kat edilen mesafenin minimize edilmesi, güç sarfiyatını dikkate alarak teslimat süresinin minimize edilmesi için kullanılan tüm algoritmalar MATLAB ortamında uygulanmıştır. İHA-kamyon iş birlikteliğine sahip bu sistem ile tek başına kamyonun dağıtım amaçlı kullanımı karşılaştırılmıştır. Sonuçlar, önerilen sistem ile dağıtımın daha kısa sürede tamamlandığını sunmaktadır. Aynı zamanda İHA'nın sınırlı uçuş süresi sorununun işbirlikçi sistem kullanılarak giderildiği de gösterilmektedir. Bunlara ek olarak, müşteri taleplerinin eşit olmadığı durumlarda, İHA'nın güç tüketimini en aza indirecek rotayı hesaplamak adına, en yüksek talep

ve en kısa mesafe isteklerini oranlayarak kullanan bir yöntem de geliştirilmiştir. Bu yöntemde de karınca kolonisi optimizasyonu ve en yakın komşuluk algoritması uygulanmış olup sonuçlar en yakın komşuluk algoritmasının daha etkili olduğunu göstermektedir.

Anahtar Kelimeler: İHA, teslimat güzergâhı optimizasyonu, tamsayı doğrusal programlama, K-ortalama gruplandırma algoritması, en yakın komşu algoritması, karınca kolonisi optimizasyonu

To my Father and my Mother. To my wife and kids. To all of my family and my friends

ACKNOWLEDGMENTS

In the name of Allah, the Most Gracious and the Most Merciful and Beneficent, all praises to Allah for the strengths and his blessing in completing this thesis.

I would like to extend my thanks and gratitude to my supervisor Asst. Prof. Dr. Kutluk Bilge Arıkan and my Co-supervisor Assoc. Prof. Dr. Mehmet Turan for their encouragement, advice, and guidance during my research study inspire me to work hard, continual, and confident.

I would like to thank my committee members for oaring me their comments and suggestions to accomplish my work appropriate.

Finally, I would like to thank my family, friends, and everyone, who helped me during the study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACOA	Ant Colony Optimization Algorithm
ERAST	Environmental Research Aircraft and Sensor Technology
DC	Direct current
DRP	Drone Routing Problem
GPS	Global Positioning System
GV	Ground Vehicle
ILPM	Integer Linear Programming Model
IPCACOA	Improved Power Consumption for Ant Colony Optimization Algorithm
IPCNNA	Improved Power Consumption for Nearest Neighbor Algorithm
MILP	Mixed Integer Linear Programming
NASA	National Aeronautics and Space Administration
NNA	Nearest Neighbor Algorithm
PC	power consumption
TSP	Traveling Salesman Problem
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
USA	United States of America

VRP	Vehicle Routing Problem

LIST OF SYMBOLS

V	vertices set
E	arc set
m	Mass $[Kg]$
v	speed of drone $[\frac{m}{s}]$
P	Power [watt]
T	Thrust
v_h	induced velocity in the air $[\frac{m}{s}]$
ρ	air density $[\frac{Kg}{m^3}]$
A_p	propeller disk area $[m^2]$
m_D	weight of the drone frame (including battery and propeller) [kg]
m_L	payload weight [kg]
g	gravity $[\frac{m}{s^2}]$
n_r	number of rotors
μ	power consumed rate per unit weight [watt / kg]
γ	power required to keep the body drone in the air [watt]
V_c	customers set
N_0	customers subset
d_i	customer demand
Q_T	truck capacity
D_{ij}	distance matrix between customers and trucks location
d_{ij}	distance matrix for customers subset

λ_j	time required for landing, taking off and delivering the package
t_{ij}	time spent to service the customers
Q_D	drone capacity

CHAPTER 1

INTRODUCTION

1.1 Introduction

Small Unmanned Aerial Vehicles (UAVs), also called drones, are one of the research areas that have received considerable attention in recent years. This autonomous aerial vehicle can fly at low altitudes and evade obstacles at low altitudes more easily. Small UAVs have already been utilized in many civilian applications such as weather monitoring, search and rescue, pollutant estimation, traffic surveillance and disaster monitoring and management [1-5].

In this chapter, a brief history of UAVs and their classification are given first, followed by usage of UAVs for logistic purposes. Then, the motivations and objectives of this study is clarified. Next, the statement of the problem will be discussed, followed finally by the outline of the thesis.

1.2 Unmanned Aerial Vehicles (UAVs)

A UAV (or 'flying robot') may be defined as an aircraft that flies without an on-board pilot. In fact, UAVs can be controlled remotely from the ground by a pilot, or they can be controlled autonomously by an on-board computer programmed to perform a specific task [6, 7]. UAVs were first created by Lawrence and Sperry (USA) in 1916. They were called the 'Aviation Torpedo,' as shown in Figure (1.1), and they utilized a gyroscope to stabilize the body. The 'Aviation Torpedo' was able to fly a distance of over 30 miles [7, 8]. At the end of the 1950s, UAVs began to gain advantage during the Cold War with full-scale research and expansion continuing well into the 1970s, after which time researchers started developing cheaper and smaller UAVs powered with small engines such as those found in motorcycles and which were able to carry a camera to transmit images to a base. It was in this period that the prototype of the UAV was born. In 1991, the USA utilized UAVs for practical use in the Gulf War, after which UAVs in military applications progressed quickly.

Figure 1.1: Aviation Torpedo Ref. [7]

The most famous UAV for military use is the Predator, shown in Figure (1.2). During this period, NASA was started to focus on UAV research for civil use. The most typical example from this time was the ERAST (Environmental Research Aircraft and Sensor Technology) project. It started in the 1990s and was a full-scale research attempt for a UAV that contained the development of the technology required to fly at high altitudes of up to 30,000 m along with extended flight technology, sensors, engines, and so on [7- 9].

Figure 1.2 : Predator Military UAV Ref. [8]

1.2.1 Classification of Unmanned Aerial Vehicles (UAVs)

There are different ways to classify UAVs, according to their range of action, aerodynamic configuration, size and payload, as shown in Figure (1.3).

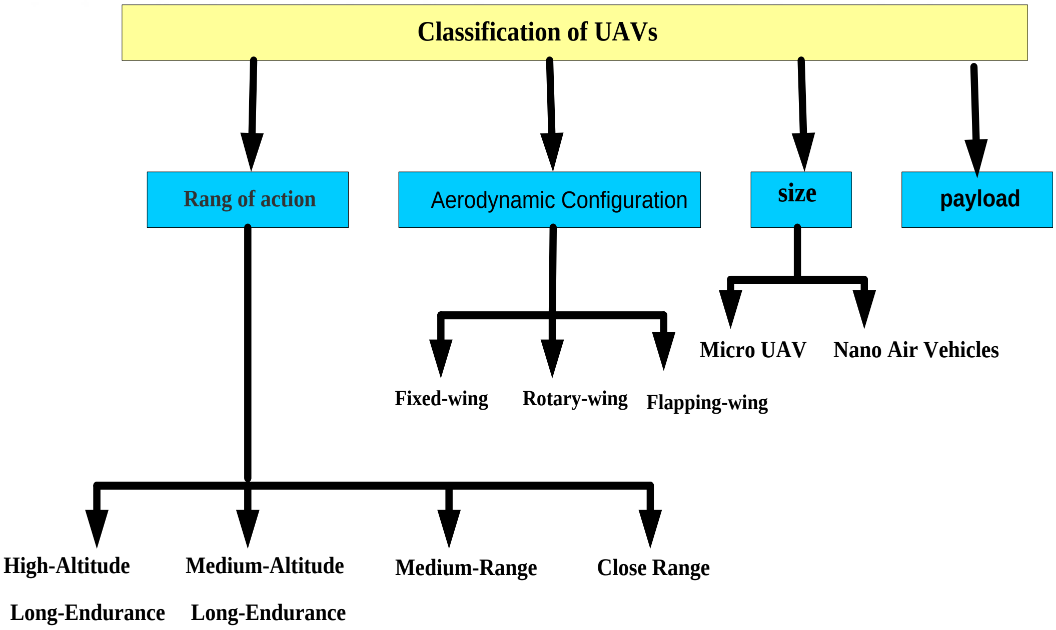

Figure 1.3: Unmanned aerial vehicles classification

There are two common categories of UAV based on the aerodynamic configuration used in many applications: fixed-wing and rotary-wing. Fixed-wing UAVs Figure (1.4 (a)) are unmanned aircraft that utilize forward propulsion over a fixed wing to gain lift. A high forward velocity is needed to generate this lift; consequently, this type is not appropriate for use in restricted environments, in which UAVs require maneuverability to perform tasks.

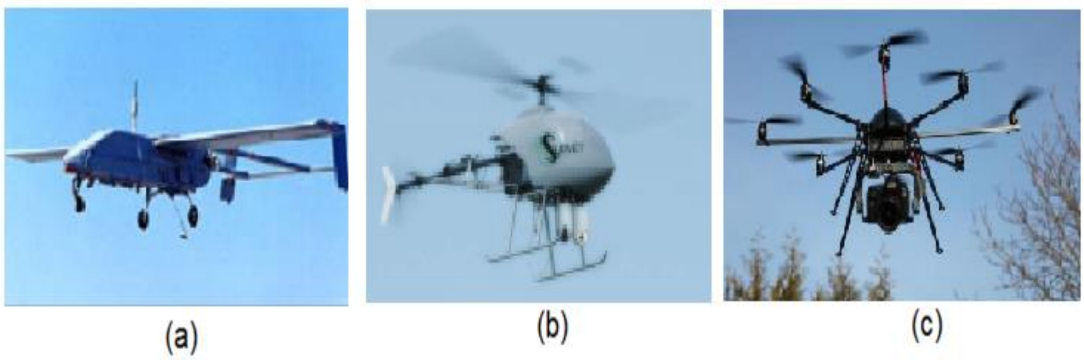

Figure 1.4: (a) Fixed-wing Ref. [10] (b) Helicopter Ref. [7] (c) Multi rotor Ref. [11]

On the other hand, there are four types of rotary-wing aircraft based on the number of rotors. Those with one rotor are called helicopters Figure (1.4-b) and utilize a single large rotor to produce lift but requiring a tail rotor for directional control and stability. The second type is the multi rotors which utilize multiple rotors to produce and control the motion of vehicles Figure (1.4-c). The third type is the coaxial, which has two rotors constructed on the same shaft and rotating in opposite directions, as shown in Figure (1.5-a). Finally, the Quad-rotor has four rotors fitted in a cross link configuration, as shown in Figure (1.5-b). The multi-copter is a type of rotary-wing UAV with 4, 6 or more rotors powered by DC motors. It has several features, including its small size, high maneuverability, vertical takeoff and landing and its ability to carry loads of 50% to 100% of its body weight. However, it is powered by a lithium polymer (Li-Po) battery, which makes the flight time limited.

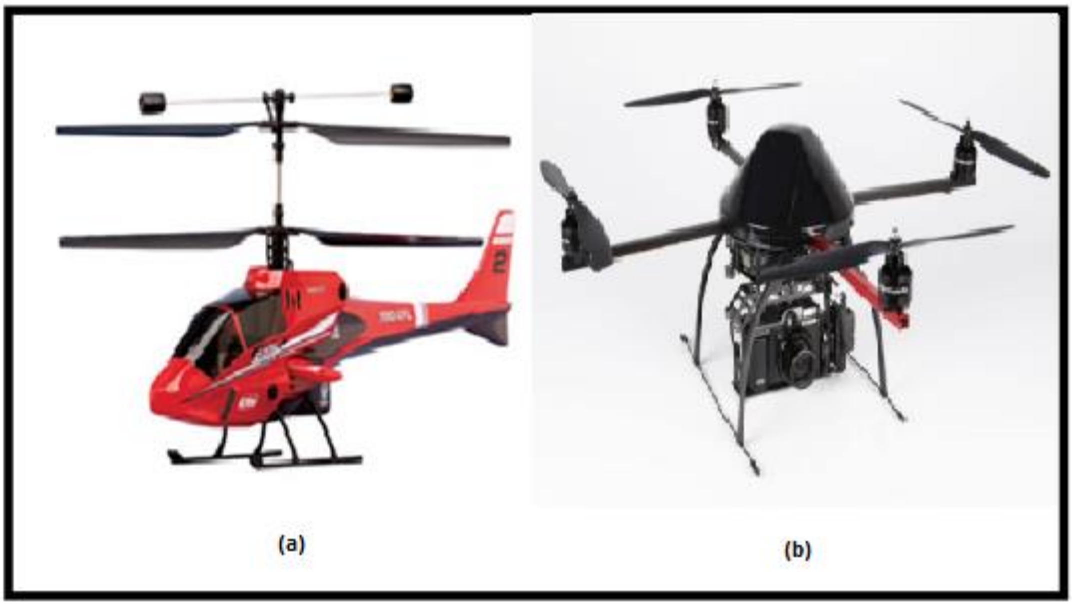

Figure 1.5: (a) Coaxial Ref. [7] (b) Quadrotor Ref. [12]

The most popular rotary craft is the quadrotor, which has numerous advantages over classical helicopters. One of the advantages is that the rotors are small and can be enclosed, making them safer for indoor flights. Moreover, they have higher payload capacity and better maneuverability in comparison to the classical helicopter. It is also possible to accomplish more stationary hovering with four thrust effects at a distance from the center of gravity than with one thrust force effect through the center of gravity, as is the case with classical helicopters. The main disadvantage of the quadrotor is its high energy consumption due to its use of four motors [13]. In addition, advantages will be enhanced by adding more rotors (multirotor) to the aircraft, but this will increase energy consumption.

1.2.2 UAVs in Logistic.

In recent years, unmanned aerial vehicles (UAVs, also known as drones) have been becoming utilized in many commercial and military applications providing limitless advantages such as reliably and effectively saving time and effort of doing work. The multi-copter is a rotary craft type of UAV that has good features, including plain mechanical structure, high maneuverability and a perfect relationship between the payload capacity and total weight. This feature makes it an ideal candidate to transport goods in 3D space. On the other hand, efforts have been made over many years to reduce emissions in logistics systems by introducing greener fuel sources [14]. One of these efforts includes a truck powered by electricity using an in-vehicle routing problem [15]. In addition, high-density traffic on roads, especially during peak times, produces situations in which traditional delivery trucks cannot easily reach a location. Therefore, this necessitates the introduction of UAVs for delivery tasks to make more feasible and improve the delivery process.

There are many logistic companies working to utilize UAVs in delivery systems to make delivery processes more efficient at lower cost and in less time. Regular delivery trucks cannot easily reach some locations, especially in the rural areas; therefore, a drone would be an appropriate solution in such cases. DHL first tested UAVs to deliver blood samples across a river, followed by delivery of medication to a small island called Juist. In 2016, DHL tested a third-generation delivery drone called the "parcelcopter" shown in Figure (1.6-a), which can fly at speeds of up to 70 km/h to a distance of 8.3 km with a payload of up to 2.2 kg. Moreover, it can make deliveries in 8 minutes. A standard delivery vehicle would take approximately 30 minutes for the same distance [16]. Amazon has tested a delivery drone (as shown in Figure (1.6-b) that can fly at altitudes of up to 122 m with a payload of up to 2.3 kg at a top speed of 88 km/h. The drone is fully autonomous and utilizes GPS when making a delivery; however, the location should be within a 24-kilometer range [17].

A drone delivery system, made by Australian drone company Flirtey, Figure (1.7), has been tested successfully to transport 4.5 kg of medical supplies to a countryside health clinic in UA. The test was approved by the Federal Aviation Authority in participation with NASA. The test demonstrated that drones could be useful for the delivery of packages, especially in the countryside and in remote areas that are difficult to reach with ground vehicles. However, the flight time of drones is limited [18]. The national postal service in Singapore has started testing drones for delivery. Drone have been flown 2.3 km to deliver letters to a small island called Pulau Ubin, located northeast of the main island of Singapore, with a trip taking approximately 5 minutes [19].

Figure 1.6: Delivery drone, (a) DHL parcelcopter for fast delivery Ref. [16]
(b) Amazon delivery drone Ref. [17]

In the same manner, the authors in [20] make some calculations to compare the cost of using drones in delivery systems. They state that the operating costs of using drones are on the order of 10 cents per 2 kg of weight in a 10-kilometer, in contrast to 60 cents per item used over a decade ago in their traditional delivery systems. Therefore, it is economically feasible to deliver small packages using drones. The consequence of this is that the delivery of goods becomes faster, more effective and efficient at lower cost. UAVs are an ideal candidate, owing to their vertical takeoff and landing, good maneuverability and the possibility of carrying cargo at 50% to 100% of the UAV body weight [21].

Figure 1.7: Flirtey drone delivery Ref. [18]

In contrast, the traditional delivery truck can carry many delivery packages and has a long operation time, but it is slow and heavy, as shown in Table (1.1). In addition, in some cases, the truck cannot reach customers easily due to high traffic congestion or bumpy roads in rural areas.

Table 1.1: Comparing between delivery truck and UAV

	weight	speed	capacity	Operation time
Truck	heavy	low	high	long
UAV	light	high	low	short

1.3 Motivations and Objectives

In the last few years, the use of Unmanned Aerial Vehicles (UAV), also known as drones, has been greatly increased in many military and civilian applications due to numerous features, which has them perform human work at high levels of reliability and with greater effectiveness. The multi-copter is rotary-wing type with numerous characteristics, such as small size, takeoff and landing in a small area, good relationship between total weight and payload, simple mechanical structure and high maneuverability. These characteristics make it an ideal candidate for the distribution of goods in 3D space.

This work started in 2015 when many logistics companies were working to introduce UAVs into their delivery systems to improve logistics tasks. This inspired new challenges needed to investigate how to make route planning more efficient. These challenges came from several limitations of UAVs. For instance, the flight time is limited due to small battery power, which makes UAVs unable to complete missions as well as possibility of crashing to the ground if batteries are not recharged or replaced during a journey. Due to UAVs being a new application in delivery systems, only a small amount of study in the literature exists that investigates this problem in the routing side. Hence, handling the problem of limited flight time will make it possible to use UAVs to distribute goods.

This thesis focuses on the routing problem for multiple vehicles, namely delivery trucks and drones working together to deliver packages. The aims of the thesis are:

- To develop a collaborative delivery system with a UAV and delivery truck on the routing side to solve the limited flight time issue.
- To present a drone routing problem that minimizes delivery time and power consumption by considering payload weight and distance travelled.
- To improve the power consumption of a drone by proposing a method of weighting between shortest distances and highest weights when a drone selects its customers.

1.4 Problem Statement

One of the difficulties encountered in the use of drones in a delivery system is the limited flight time of drones due to small battery power, which makes it difficult to carry out every delivery directly from a warehouse, as shown in Figure (1.8-a). There are *n* customers needing service by drone from the warehouse; nevertheless, a number of them could not be served because of the limited flight range, thereby making the drone incapable of serving these customers. This problem can be overcome by constructing new warehouses close to customers, but this will increase the cost of delivery and would be unsuitable in many areas, especially rural areas. The appropriate solution to this problem is to develop a collaborative system between a drone and a delivery truck, where a delivery truck is loaded with packages and carrying a multi-copter to a point closer to customers, which suggests that the delivery truck would function as a sub-warehouse and a base for drones, as shown in Figure (1.8-b).

In this scenario, the delivery truck is loaded with packages in the main warehouse and at the same time carries the drones. The delivery truck then proceeds to the point close to customers. The drone will take off, fly to the given customer, deliver the package and return to the base (delivery truck). When the drone is at the base, the battery will swiped with recharged one and a new package is loaded for the next mission. The drone may serve one or two customers per trip depending on the payload and flight time. This problem is studied as a routing optimization problem for both the drone and delivery truck. First, the problem is modeled to find the best stop locations for the delivery truck by partitioning customers into groups and finding the center for each group. Second, for each group the problem is modeled as a drone delivery optimization problem to minimize the distance traveled by the drone, the delivery time and the power consumption so as to serve every customer taking into account the payload weight and limited flight time of the drone.

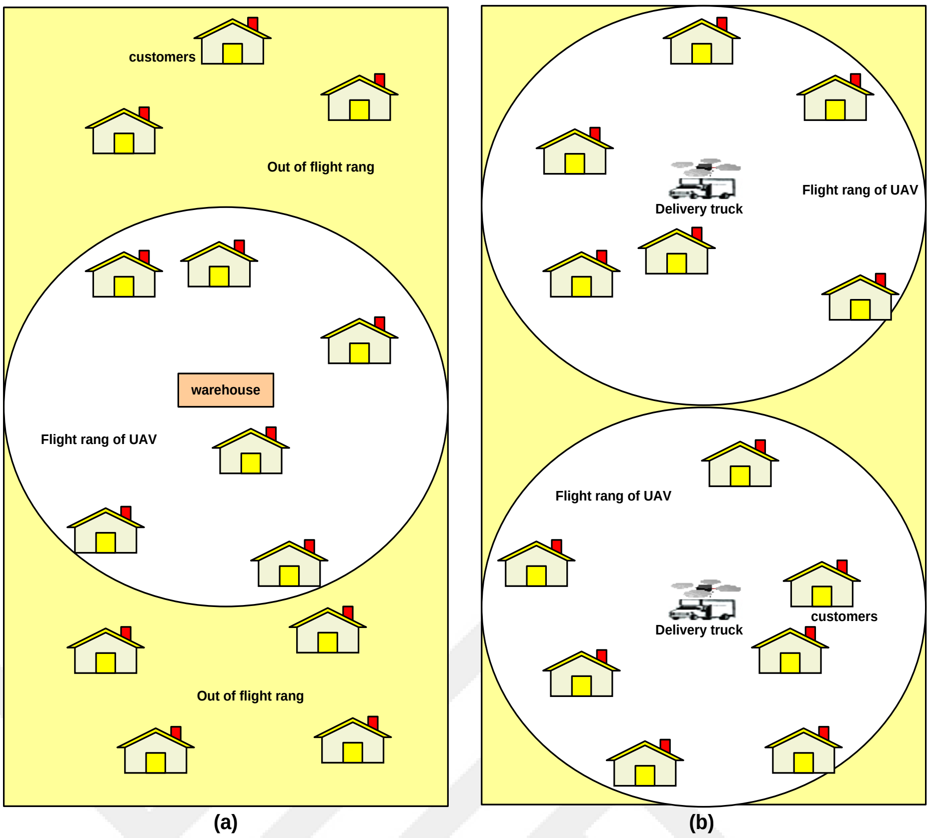

Figure 1.8: Delivery directly from warehouse, numbers of customers cannot be reached due to limited flight time. (b) All customers can be reached by patriating the customers into groups and using a collaborative system between UAV and a delivery truck.

1.5 Contribution

The contribution of this thesis may be summarized as follows:

1 – The issue of the limited flight time of the UAV is addressed by collaboration between a drone and a delivery truck when customers are divided into groups and served by a drone, while a truck is used as a base for the drone and assigned to the location close to customers (middle of the group).

2 – We develop an integer linear programming model (ILPM) for the drone routing problem (DRP) with two objective functions, the first one is to minimize the time of delivery with taking into account the distance traveled and the speed of the drone. The second objective function is to minimize the power consumption by taking into account the payload weight of the drone and the distance traveled.

3 – We introduce a method for the drone to select between two criteria, namely highest demand and shortest distance when it moves to serve the customer, which will minimize the power consumed by the drone. This method can be followed to construct a new distance matrix which would be utilized to improve the NNA and ACOA to minimize the power consumption.

1.6 Thesis Outline

The thesis organizes in six chapters as following:
In the first chapter, an introduction is presented including a short history of UAVs, their classification, logistical uses, followed by the motivations and objectives of the presented work. Then the problem statement is described, followed finally by the contribution and structure of the thesis being presented.

Chapter Two commences with some background needed to understand and solve the routing problem, such as the optimization problem, NP-hard, integer linear programming, and graph theory. This is followed by a presentation of the literature survey with confirmation of published work in relation to the present work, namely the problem of limited flight time, collaboration with two vehicles and the drone routing problem, all of which are discussed.

Chapter Three presents the mathematical formulations began by the power consumption model for UAVs, followed by integer linear programming for the collaborative system.

Chapter Four presents the methodology to find solutions. Three algorithms are proposed, and discussed, namely K-means clustering to divide customers into groups and find the best location for a delivery truck, nearest neighbors and ant colony optimization to find and solve the drone routing problem.

Chapter Five presents the results found from simulating drone power consumption and integer linear programming models by implementing all the algorithms in MATLAB.

Chapter Six presents the discussion of results, conclusions and future work for this thesis.

CHAPTER 2

BACKGROUND INFORMATION AND LITERATURE SURVEY

2.1 INTRODUCTION

In many applications, the UAV needs to work together with another vehicle to complete its work successfully. This cooperation may be between two UAVs or it may be between a UAV and a ground vehicle (GV). This issue occurs from the application itself, such as in search and rescue in an extensive area, in dangerous places, or from the limitations of a UAV, such as limited flight time. In this chapter, some background about the routing problem is given first to realize the optimization problem, followed by a review of the previous study in the area of limited flight time due to the small battery power of the UAV, collaboration with two vehicles, and the drone routing problem (DRP) in the delivery system.

2.2 Combinatorial Optimization Problem

The routing problem studied in this thesis is an optimization problem. An optimization problem can be formulated as either maximizing or minimizing. Our problem in this thesis is only considered as a minimization problem which can be modeled in general form as:

$$\begin{aligned} &\min \;\; f(x) \\ &subject\ to\;\; x \in S \end{aligned} \tag{2.1}$$

where, f is called an objective function and the variables x_i, with $i = 1,2, \ldots\ldots, n$ are called decision variables. S is the solution space in which there is a set of feasible solutions in the solution space that can be determined by a number of constraints.

The feasible solution set is usually implicitly described by a set of equalities and inequalities. The optimization problem can be modeled in a number of different ways with respect to the decision variables, objective function and depiction of the solution space. This will require different algorithms to solve the problem and subsequently, various calculation effects.

The combinatorial optimization problem is an optimization problem in which a set of feasible solutions is limited. Nevertheless, it generally becomes exponentially large in relation to the problem data. The feasible solutions for the combinatorial problem is usually discrete or can be reduced to discrete. For small sized problems, efficient algorithms can be used to find an optimal solution, while for large problems, only a suboptimal solution can be found in a reasonable time using heuristic or metaheuristic algorithms [22].

2.3 NP-hard Problems

The NP is a problem with decisions (problems with ‘yes’ or ‘no’ answers) which the answer can be verified in a polynomial time. The routing problems are NP-hard problems and they are not solved optimally in reality because the solution grows exponentially with the number of customers. If we suppose that there are n customer services for one vehicle, then the total number of feasible solutions is $(n - 1)!$, which grows exponentially [23]. For $n = 5$, the feasible solution is 24, while for $n = 8$, the feasible solutions is 5040. There are two main approaches to solve the routing problem, namely the exact and approximate methods [24]:

2.3.1 Exact Method

The exact method guarantees that an optimal solution is found in a limited amount of time; however, it is suitable only for small-sized problems. Time grows exponentially when the size of a problem increases and the search for a solution no longer becomes feasible. The algorithms normally used to provide exact solutions include branch and bound or branch and cut.

2.3.2 Approximate Methods

Approximate methods contain heuristic and meta-heuristic approaches to find near optimal solutions. A heuristic method is a procedure that is likely to discover a very good feasible solution, but not necessarily an optimal solution. The main advantage of heuristic methods is their speed and ability to handle large problems that cannot be solved by exact methods. The most used algorithm to implement a heuristic approach in the routing problem is the nearest neighbor algorithm.

On the other hand, the meta-heuristic is a more generic method than the heuristic. Moreover, it is a non-exact solution method; however, it can be utilized to solve a problem in an acceptable time to find a near optimal solution for a large or very large problem. The main difference it has with the heuristic method is that the meta-heuristic is an independent method that can be applied to any wide range problems containing multiple local optima, while the heuristic method is created to solve a particular problem. It also becomes trapped in the first local optima. There are many algorithms used to implement meta-heuristic methods, such as ant colony optimization (ACO), particle swarm optimization (PSO), simulated annealing (SA) and genetic algorithms (GA) [25].

2.4 Integer Linear Programming

Integer linear programming is used to solve optimization problems with discrete decisions and it has an application in many fields, such as electrical power systems, control engineering and operation research. In the model (2.1), if the decision variables are allowed to be fractional, the model is referred to as integer linear programming (ILP) and when some, but not all, of them are restricted to be integers, the model is called mixed integer linear programming (MILP). A special type of integer linear programming is known as a zero-one programming problem in which the decision variables can take a value of 0 or 1 [26]. For instance, in the drone routing problem, if the drone moves from customer i to customer j, the decision variable $x_{ij} = 1$; otherwise, it equals 0.

2.5 Graph Theory

To demonstrate the routing problem and design an effective algorithm, the formulate graph theory should be introduced which used as the main tools to solve the routing problem, therefore, some key points and notations need to be proposed. The graph $G(V, E)$ is a mathematical structure governing a set of vertices or nodes in set V and arc set E or paths linking the vertices. The arc $e = \{i, j\} \in E$ represents the vertices i and j, which are adjacent to each other.

For the undirected graph, there is no singularity between the two nodes related to an edge, and so edges $\{i, j\}$ and $\{j, i\}$ are coinciding, where, as in a directed graph, edges have direction. A directed edge is also called an arc. Arc $a = (i, j)$ has a direction that points from its initial vertex i, also referred to as the arc tail, to its end vertex j, also referred to as the arc head. The degree of a vertex is the number of edges that connect to that vertex; in a directed graph, a vertex has in-degrees and out-degrees that are the number of arcs pointing to that vertex and away from it, respectively. A path in a graph G is a consecutive sequence of edges $[\{v_1, v_2\}, \ldots\ldots, \{v_{n-1}, v_n\}]$ from edge set E that connects a distinct set of vertices $\{v_1, \ldots, v_n\} \in V$ at their end points. Vertex b is reachable from a if there exists a path in the graph that starts from a and ends at b and the two vertices a and b are connected if they are both reachable from one another. Clearly, connectivity is implied by reachability in an undirected graph. G is described as a connected graph if every pair of vertices in V are connected and it becomes a complete graph if every vertex in V is adjacent to every other vertex in the graph.

A cycle is a path that starts and ends at the same vertex. A graph G is a tree if it is connected and does not have any cycles. A spanning tree of a connected undirected graph G, is a tree that contains every vertex of G and all of whose edges are in E. [27, 28].

2.6 Literature Survey

Many researchers are working to improve the energy efficiency of drones, which will increase their endurance. This issue was addressed by considering the drone itself in terms of mechanical design, reducing the payload or by considering the surroundings of the drone to perform missions. In this section, we summarize the previous study in the field of the limited flight time of the drone, followed by collaboration with drones and other vehicles in a number of applications. Finally, the drone routing problem is investigated in the field of delivery systems.

2.6.1 Limited Flight Time

There are many studies that present how to improve energy efficiency or reduce the energy consumption of UAVs. Some of them are proposed approaches to increase the efficiency of energy consumption; nevertheless, these are appropriate for a team of UAVs, while others studied the power systems of drones and the improvement of the mechanical design was studied to solve this problem. In Ref. [29] a new rotor configuration, called a “Triangular Quadrotor,” is proposed to improve power efficiency. This configuration utilizes one large rotor for lifting and three small rotors for control. The design, however, is still in a prototype stage. The authors in Ref. [30] introduce an algorithm that is used to increases energy efficiency for a swarm of flying robots utilized for surveillance. The swarm is incrementally deployed by launching one robot at a time. The robots are launched individually, separated by adjustable times between them, which reduces the energy consumption of the swarm by up to 30.8%. However, this increases the search time. This system works for teams of UAVs utilized for indoor exploration. Ref. [31] proposes a model to estimate the endurance of UAVs used for aerial exploration within indoor environments, and a ceiling attachment is introduced as a means of preserving energy while maintaining a bird’s eye view. Other studies in Ref. [32, 33] and [34] proposed an automated battery management platform system used to extend the operational time of battery powered UAVs. This system is used to quickly swap a consumed battery with a replenished battery while the other batteries are simultaneously recharged. The change/recharge system can be maintained for long duration missions; however, the UAVs must be landed on the platform more than once during their missions. In Ref. [35], the authors propose a technique that extends the endurance of UAVs by dumping consumed batteries out of the UAV while in flight. It was found that endurance can be extended by 17% compared with fixed weight models, but the battery dumping system would add extra weight. Nevertheless, the system could be simplified by dividing the battery into even packs, which would extend flight time. Ref. [36] introduces a method that extends the endurance of a rotorcraft by decreasing the payload; this is achieved by sub-dividing the battery into multiple smaller capacity batteries which are sequentially discharged and released. However, this is limited by the additional weight of the switching circuitry and release mechanism. Additionally, the battery efficiency will decrease and the size reduce. In the above studies, most work to improve the flight time by a percentage unless in Ref. [32, 33], and [34] they introduce an automated battery platform system that can solve the problem by replacing or charging the battery many times during a mission. Nevertheless, this system is inappropriate in the delivery task where drones are needed for freight with new packages in addition to replacing the battery.

2.6.2 Collaborations with UAVs and Another Vehicles

In the past few years, many researchers have studied the advantages of cooperation between flying robots and ground vehicles. The collaborative system between UAVs and ground robots is rapidly spreading as an innovative tool suitable for use in many applications, such as search and rescue operations, safety problems and civil protection as in Ref. [37, 38]. Ref. [39] presents a cooperative system between UGVs and small air vehicles utilized in an observation missions area when the UGV cannot observe it. In Ref. [40], the researcher group at Cincinnati University and AMP electric vehicles developed a delivery truck-drone system utilized for delivery to rural areas Figure (2.1-a). The drone collaborates with the delivery truck; the drone takes off, delivers the package to the customer and returns to land on the top of electric trucks to pick up another package, while the driver serves other customers on the main route.

(a) (b)

Figure 2.1 (a) Delivery drone collaborative with AMP electric vehicle Ref. [40]

(b) Delivery drone launches from UPS truck Ref. [41]

In Ref. [41], the UPS and Workhorse groups successfully tested a UAV that releases from the top of a delivery truck Figure (2.1-b) and can autonomously serve a customer, after which it then returns to the vehicle while the truck driver continues serving other customers on the road. In Ref. [42] and [43], new mixed integer linear programming formulations called "Flying Sidekick Traveling Salesman Problem" (FSTSP) and "Traveling Salesman Problem with Drone" (TSP-D) were proposed, respectively, where the delivery truck is working collaboratively with the drone. In Ref. [42], two issues are studied to minimize the time required to complete all deliveries by two vehicles. In the first case, the drone is working in coordination with the delivery truck and customers are served by either a truck or a drone. Some customers are served by truck due to exceeding the drone's payload capacity, the fact that delivering packages requires signatures, customers' being out of flight time. In the second case, parallel drone scheduling TSP is considered when customers are located within a drone's flight range from the depot. Therefore, the truck serves customers along a TSP route while the drone serves other customers directly from the depot. The two problems are solved by the Clarke-Wright saving heuristic and the nearest neighbor heuristic. The results shows that delivery times, when using the cooperative system, are reduced in contrast to using only a delivery truck. Ref. [43] is similar to the FSTSP in collaboration between drone and truck, with the difference being a TSP tour constructed first using Kruskal's minimum spanning tree algorithm and second the tour being split into drone tour and truck tour as a sub-tour being solved by a fast greedy heuristic and exact partitioning algorithm. Ref. [42] and [43], are quite similar to the present study in the collaboration between drone and truck while the main difference is the problems in Ref. [42] and [43] considered as a TSP by adding a drone to serve customers by both truck and drone. On the other hand, this research focuses on the problem of DRP by adding a truck as the sub warehouse. In addition, the formulation differs because the problems in Ref. [42, 43] involve missions and routing solutions where delivery is made by both vehicles. Mission resolving is required to determine which vehicle, truck or drone, is to serve which customers. Apart from this issue, routing resolving is required to distinguish the order of customers each vehicle serves. Hence, our problem is modeled as a routing issue. First, customers are divided into groups and then for each group, the drone routing problem is studied to serve customers with minimum distance, service time and power consumption while a delivery truck plays the role of a sub-depot. In addition, the K-means clustering algorithm and the ant colony optimization algorithm are used to solve this problem and compared with the nearest neighbor heuristic algorithm.

2.6.3 Drone Routing Problem (DRP)

The drone routing problem is a new issue related to utilizing the drone in the logistics of distributing goods to customers Ref. [44], and it is somewhat similar to the vehicle routing problem (VRP). The VRP was first applied in the field of logistics distribution by Dantzig and Ramser in the early 1960s Ref. [45]. The purpose of the VRP is to determine the minimum cost such that the distance travelled or time of delivery of the routes for several vehicles which leave from the depot with a certain capacity to serve a number of customers and return to the depot. The DRP is different from the VRP such that the DRP uses a small autonomous aerial vehicle powered by a small battery, which adds some limitations to the problem, such as limited flight times and limited carrying capacity, which means that in general it can only carry a small number of packages per route. This also adds more constraints when we model the problem, such as reuse of the drone, maximum flight time, and the number of customers on the route. This allows the drone to perform multi-trips with a fixed number of customers, possibly one, two or three customers.

There are many studies in the literature with respect to traveling salesman problems and vehicle routing problems. Nevertheless, all these studies consider the truck as the vehicle in the problem except for one study using the drone, which is somewhat related to our problem. The VRP and TSP have been important problems in the field of logistics and distribution of goods to customers and they are closely related to each other. The travelling Salesman Problem (TSP) is the most widely utilized of routing problems. The aim of the problem is to minimize the cost (distance travelled) when a salesman has to travel to a number of cities and return to the starting node Ref. [46]. An expanded version of this problem is the multiple Travelling Salesman Problem (mTSP), in which m salesmen have to travel to n cities such that each city is visited by exactly one salesman. All salesmen begin at the same node, travel to specific cities and return to the starting point. The aim is to minimize the sum of the cost (distances travelled) by every salesman. In fact, the VRP is reduced to several TSPs. For a general overview about formulations, types and solution procedures of the VRP there are a number of books [47, 48], and papers [49, 50] dealing with this matter. There are numerous researchers who expand the VRP to consider minimum fuel consumption Ref. [51, 52] and safe environment impact. Ref. [53] and [54] introduce a problem called the Green Vehicle Routing Problem (G-VRP) and the Green Capacitated Vehicle Routing Problem with Fuel Consumption Optimization Model, which use a truck powered by greener fuel sources, such as electricity, hydrogen or natural gas to reduce pollutants and emissions. Another study considers a number of parameters with VRP, such as time windows Ref. [55], pick-up and delivery Ref. [56], and truck and trailer routing problems Ref. [57, 58]. However, all of these studies consider the delivery truck as the vehicle in the problem and, as mentioned previously, there is a difference somewhat between VRP and DRP.

Another related study named “Vehicle Routing Problems for Drone Delivery” concentrated on of drone routing is conducted by Ref. [59]. The main difference between two studies is the cost function and solution approach, as Ref. [59] suggests a function that minimizes the budget cost by considering the UAV reuse and energy consumption and solved by simulated annealing algorithm to find a suboptimal solutions. Our study focuses on the minimization of power consumption and delivery time by considering payload weight and distance travelled using the nearest neighbor and ant colony optimization algorithms, and improving these algorithms by using methods with weighting between the shortest distance and highest demand weight of the customer to minimize power consumption.

CHAPTER 3

MATHEMATICAL MODEL

3.1 INTRODUCTION

In this chapter a mathematical formulation for the collaborative system is derived with two parts; the drone power consumption model and integer linear programming model. In the first part the theories of rotor aerodynamics of rotor wing helicopter [60] is utilized to obtain the model, which will used to calculate the power consumed in the second part. For the optimization problem, the integer linear programming is used to derive the model with multiple objective functions and numerus of constraints.

3.2 **Drone Power Consumption Model**.

The drone is powered by a lithium-ion polymer (LiPo) small battery which strongly limits the class of missions that a drone can successfully carry out. The power consumed by the drone is effected by distance traveled, payload carried, environment condition and the speed of the drone. The data obtain from the power consumption model is used in the optimization part for calculation. Refer to Ref. [60] "the main purpose of the rotor in the hover are to provide a vertical lifting force in opposition to the weight of the helicopter". This force need a power required induced in the air called an ideal power and given by:

$$P = Tv_h \tag{3.1}$$

Where, T is the thrust generated by the rotor to endure the vehicle in hover and it is equal the weight of the vehicle, v_h is the induced velocity in the air at hover condition, and by using momentum theory Ref. [60] [61] it is given by:

$$v_h = \sqrt{\frac{T}{2\,\rho\,A_p}} \tag{3.2}$$

By substituting equation (3.2) in (3.1)

$$P = T\sqrt{\frac{T}{2\,\rho A_p}} = \frac{T^{3/2}}{\sqrt{2\,\rho\,A_p}} \tag{3.3}$$

Where, P is the hover power, T is the rotor thrust, ρ is the air density and A_p is the propeller disk area. The thrust, T for a single rotor is equal to the drone weight when the drone in hovering phase Ref. [60], that is:

$$T = (m_D + m_L)g \tag{3.4}$$

Where, m_D and m_L are the weight of the drone frame (including battery and propeller) and the weight of payload respectively and g is the gravity in $\frac{m}{s^2}$ equation becomes:

$$P = (m_D + m_L)^{3/2}\sqrt{\frac{g^3}{2\,\rho\,A_p}} \tag{3.5}$$

Equation (3.5) gives the power consumed by one rotor, on the other hand, the total weight of the drone frame and payload is distributed evenly on n_r rotors, so we can calculate the power consumption in hover for n_r rotors as following:

$$P = Pn = (m_D + m_L)^{3/2}\sqrt{\frac{g^3}{2\,\rho\,A_p\,n_r}} \tag{3.6}$$

We considered the DJI Phantom 3 quadrotor Ref. [62] with specification agree to drone that used in delivery system, the physical parameters of DJI Phantom 3 are listed in table (3.1) and for missing ones, we reckoned on the values reported in Ref. [63], for similar drone. Equation (3.6) can approximate to a linear equation [59] as following:

$$P(m_L) = \mu\, m_L + \alpha \tag{3.7}$$

where, μ is the power consumed rate per unit weight for payload, and α is the power required to keep the body drone in the air. The power required for hovering is considered as an upper limited power consumed by the drone Ref. [60]. At the forward speed an event called translational lift will occur, when the air passes horizontally through the rotor system due to forward speed. This will improves the rotor efficiency and the required power at the rotor is considerably lower than in the hover case Ref. [60].

Table 3.1: The physical parameters values of the drone

Parameter	Description	Value	Units
m_D	mass	1.3	kg
v	speed	16	$\frac{m}{s}$
	Max flight time	23	min.
g	Gravity	9.81	$\frac{m}{s^2}$
ρ	Air density	1.225	$\frac{kg}{m^3}$
A_p	propeller disk area	0.2	m^2
n_r	Number of rotor	4	

3.3 Integer Linear Programming Model.

The main goal of this section is to drive a minimization power consumption model for the collaborative system with a drone and a delivery truck. The integer linear programming model (ILPM) is needed, where we will first model the system of dividing the customers into groups based on distance between customers, number of customers and payload of the delivery truck. Next, the drone routing problem (DRP) will be modeled with two objective functions. The DRPs are related to vehicle routing problems (VRPs) where it is the main problems in logistics, distribution and transportation system. The VRP is a combinatorial optimization problem and is modeled as an integer linear programming problem aimed to minimize the total cost of the logistic system such as, delivery time, distance when the customers are served with the fleet of vehicles to get the optimal routes with minimum index. The collaborative delivery problem is considered under the following assumption:

- The number of groups equal to the number of delivery truck.
- Each customer assigned to only one delivery truck.
- The demand is stable and as a result equal for all customers.
- The total demand in the group does not exceed the capacity of a delivery truck.
- Each customer must be served only once

To model the problem a complete directed graph is developed *G = (V, E)*: where *V* is a vertex set (nodes), and E is the arc set (paths between nodes). There are two types of vertexes, $V_c = 1, \dots, n$ is the customers set, where n is the number of customers, and $V_T = 1, \dots, m$ is the candidate location for delivery truck (sub-depot), where m is the number of the delivery truck. The D_{ij} is the distance matrix between customers $i \in V_c$ and the trucks location $\in V_T$. The customers are partitioned into sub-sets $\subseteq V_c$, each sub set has a center location represent the vertex for the delivery truck denotes by 0, and $n_1, n_2, \dots, n_n$ representing the number of customers in the set. Each location $i \in N_0$, where the set $N_0 = N \setminus 0$, each customer has a demand d_i, and each truck has a capacity Q_T. There is also another distance matrix d_{ij} representing the nonnegative cost of traveling between vertices in the subset N_0 which associated with the arc between customers and truck location. The cost matrix is symmetric, that is $d_{ij} = d_{ji},\ for\ all\ \forall(i,j) \in N_0$. The following decision variable is required to complete the model:

$$g_{ij} = \begin{cases} 1 & ,if\ a\ customer\ i\ th\ allocates\ to\ the\ truck\ location\ \ j\ th \\ 0, & otherwise \end{cases} \quad i \in V_c\ , j \in V_T$$

$$h_j = \begin{cases} 1 & ,if\ the\ truck\ is\ located\ at\ place\ j \in\ V_T \\ 0 & ,otherwise \end{cases}$$

$$x_{ij} = \begin{cases} 1 & ,if\ the\ drone\ moves\ from\ customer\ i\ to\ j \\ 0\ , & otherwise \end{cases} \quad \forall(i,j) \in N_0$$

First, the model is created to divide the customers into subsets or groups. The objective function (3.8) is to minimize the distance between the customers and the corresponding truck location:

$$\min \quad \sum_{i \in V_c}^{n} \sum_{j \in V_T}^{m} D_{ij} \; g_{ij} \tag{3.8}$$

Subject to:

$$\sum_{j=1}^{m} g_{ij} = 1 \quad , \quad \forall i = 1, \dots, n \tag{3.9}$$

$$\sum d_i \, g_{ij} \; \leq \; Q_T \, h_j \tag{3.10}$$

$$\sum_{j=1}^{m} h_j = N_j \quad , \quad \forall j \in V_T \tag{3.11}$$

$$g_{ij} \; \in \; \{0, 1\}, \quad h_j \in \; \{0,1\}, \quad \forall i \in V_c \, , \quad \forall j \in V_T \tag{3.12}$$

Constraint (3.9) insures that each customer is allocated to exactly one truck. Constraint (3.10) insures that the total demand assigned to each truck does not exceed the capacity of the truck. Constraint (3.11) insures that only N_j groups will be selected, and the integer condition is provided by constraint (3.12).

Second, the DRP is modeled to minimize the delivery time and power consumption. The first, objective function is to minimize the time spent to travels from a delivery truck (sub-depot) serving customers and arrival to delivery truck (base) eq. (3.13). The second objective function is to minimize the power consumption (PC) by the drone when travels from base to customers or from customer *i* to customer *j* considering the distance traveled, the payload weight and the speed of the drone (3.14).

$$\min \sum_{\substack{i \in N_0 \\ i \neq j}} \sum_{j \in N_o} t_{ij}\, x_{ij} \;=\; min \sum_{\substack{i \in N_0 \\ i \neq j}} \sum_{j \in N_o} \left(\frac{d_{ij}}{v} + \lambda_j\right) x_{ij} \qquad (3.13)$$

Where t_{ij} represent the time spent to travel from delivery truck, service customers and arrival to delivery truck, d_{ij} represents the distance matrix, v represents the speed of the drone, and λ_j represents the time required for landing, taking off and submitting the package for customer j.

$$\min \sum_{\substack{i \in N_0 \\ i \neq j}} \sum_{j \in N_o} PC_{ij} \;=\; \min \sum_{\substack{i \in N_0 \\ i \neq j}} \sum_{j \in N_o} P_{m_{ij}} \left(\frac{d_{ij}}{v} + \lambda_j\right) x_{ij} \qquad (3.14)$$

Where PC_{ij} represents the power consumption when traveling between customers or between customers and truck, and $P_{(m_{ij})}$ represents the power consumption in watts as a function of payload so it can be calculated by using equation (3.7)

Subject to:

$$\sum_{\substack{i=1 \\ i \neq j}}^{n} x_{ij} + x_{0j} = 1 \;, \quad \forall j = 1, \dots, n \qquad (3.15)$$

$$\sum_{\substack{j=1 \\ i \neq j}}^{n} x_{ij} + x_{i0} = 1 \;, \quad \forall i = 1, \dots, n \qquad (3.16)$$

$$\sum_{j=1}^{n} x_{0j} = 1 \qquad (3.17)$$

$$\sum_{i=1}^{n} x_{i0} = 1 \qquad (3.18)$$

$$u_i - u_j + (n+1) x_{ij} \leq n \quad , \forall (i,j) \in N_0 \qquad (3.19)$$

Constraint (3.15) ensures that each customer is reached either from the delivery truck or from another customer. Constraint (3.16) ensures that from each customer we depart to another customer or to the delivery truck. Constraints (3.15) and (3.16) ensure that the drone visits each customer exactly once. Constraint (3.17) and (3.18) ensures that the drone departed from a delivery truck must return to the delivery truck. Constraint (3.19) implies that the number of customers visited at each route never exceeds the customers *n* allowed in that route.

The energy constraints are:

$$y_j \le y_i - p(m_{ij})\left(\frac{d_{ij}}{v} + \lambda_i\right) + B(1 - x_{ij}) \qquad ,\forall (i,j) \in N_0 \qquad (3.20)$$

$$y_i \ge \min\left(p(m_{ij})\left(\frac{d_{ij}}{v} + \lambda_i\right) + p(m_{j0})\left(\frac{d_{j0}}{v}\right)\right) \qquad ,\forall (i,j) \in N_0 \qquad (3.21)$$

The above constraints track the level of energy, constraint (3.20) forced the current energy available y_j is equal to the total power consumed along the route to reach the customer *j* where B is the maximum battery capacity, while constraint (3.21) ensures that there is enough power to return to the sub-depot (delivery truck). Both constraints ensure that at any customer the available power is never negative.

The carrying capacity constraints are:

$$f_j \le f_i - d_i x_{ij} + Q_D(1 - x_{ij}) \qquad ,\forall (i,j) \in N \qquad ,i \ne j \qquad (3.22)$$

$$0 \le f_i \le Q_D \quad , \quad \forall i = N \qquad (3.23)$$

The f_j is a capacity variation illustrating the weighing capacity of the drone when traveling along the route up to customer *j*. Therefore, constraint (3.22) ensures that the load of the drone at customer *j* depends on the previous load at customer *i* and the demand d_i , while constraint (3.23) restricts the load of the drone at f_i never to exceed the maximum capacity Q_D of the drone as well as ensures a positive value.

The time constraint is:

$$t_j \geq t_i + \lambda_i + \frac{d_{ij}}{v} - T(1 - x_{ij}) \tag{3.24}$$

$$\sum_{i=0}^{n} \left(\frac{d_{ij}}{v} + \lambda_j\right) x_{ij} + \frac{d_{j0}}{v} \leq T \quad , \forall j = 1, \dots, n \ , i \neq j \tag{3.25}$$

The t_j is the time variation illustrating the visited time by the drone when traveling along the route up to customer *j*. Therefore, the constraint (3.24) ensures that the time at customer *j* depends on the previous time at customer *i*, plus the time spend to move from customer i to *j* which is expected on the speed of the drone v and distance between customer *i* and *j* $d_{i\,j}$ and the time λ_i spent at customer *i* on landing, package delivery and taking off. While constraint (3.25) ensures that the time spent to reach all customers in the route when the drone departures from the sub-depot and return back never to exceed the maximum flight time *T*. This time includes travelling time between customers $\frac{d_{ij}}{v}$, the service time λ_j and the departure and arrival time to the delivery truck.

Due to limited carrying capacity the drone should make multi-trip when it do the delivery task, therefore, the drone will use more than one after it returns to the sub-depot to replace the battery and load with new package, so, the reuse constraints are:

$$\sum_{j=1}^{n} c_{ij} \leq x_{i0} \ , \qquad \forall i = 1, \dots, n \tag{3.26}$$

$$\sum_{j=0}^{n} c_{ji} \leq x_{0i} \ , \qquad \forall i = 0, \dots, n \tag{3.27}$$

The c_{ij} is the reuse decision variable where, $c_{ij} = 1$ if the drone depart from customer i to the sub-depot, replace the battery and loaded with package, then moves to customer ; and otherwise $c_{ij} = 0$. Constraint (3.26) ensures that if the drone returns to the sub-depot from customer , it is ready for use again to fly to another customer. Constraint (3.27) ensures that if the reuse drone depart from the sub-depot to customer, it is previously, arrived from another customer.

3.3.1 Collaboration with two UAVs to Carries Heavy Load

In some applications in logistics the UAV cannot carry some heavy load due to overloaded, so it needs to cooperation with another flying vehicle to carry the load. We consider the routing problem of M drones to carry load from location i to location j. Refer to equation (3.14) which represent the cost function for one drone to minimize the power consumption when it travels from customer i to customer j. This equation becomes as following for M drones:

$$\min \sum_{i=0}^{n}\sum_{j=1}^{n}(M(P_{(m_{ij})}))\left(\frac{d_{ij}}{v}+\lambda_j\right)x_{ij} \tag{3.28}$$

As we mentioned previously in section (3.2), the $P_{(m_{ij})}$ represent the power consumption of UAV when moves from i to j as a function of carrying load and can be calculate by equation (3.7), the second part of the equation (3.28) represent the time as a function of distance traveled and speed of UAV. This is the time taken by the drone to moves from customer i to j . The cost function (3.28) is subjected to constraints (3.15), (3.16), (3.17) and (3.18), also there are two additional constraints:

$$y_j \geq \min\left(\left(M\left(P_{(m_{j0})}\right)\right)\left(\frac{d_{j0}}{v}\right)\right) \tag{3.29}$$

$$\sum_{i=0}^{n}\sum_{j=1}^{n} q_{ij} \leq MQ_D \tag{3.30}$$

Constraint (3.29) guarantees that there is enough power for the M drones to return back to the base after complete the mission, while constraint (3.30) guarantees that the load carrying from location i to location j does not exceed the carrying capacity for M drones.

CHAPTER 4

SOLUTION APPROACHES

4.1 INTRODUCTION

The collaborative delivery system is consisting of two vehicles, delivery truck, and drone. The system is solving by use two phases as shown in Figure (4.1):

1 – Grouping phase: Which is the process used to partition the customers into groups based on similarity in some ways. After the clustering is completed the center of the group performed as a sup depot which is the location of the delivery truck. The delivery trucks are loaded with packages, caries the drones, and moves to the center of the groups.

2 – Routing phase: Which is the process used to find the optimal route for the drone to serve the customers in each group with minimum time, distance and power consumption. The drone is loaded with package on the delivery truck, take off, flying on the optimal route and when it complete the task return to landing on the delivery truck.

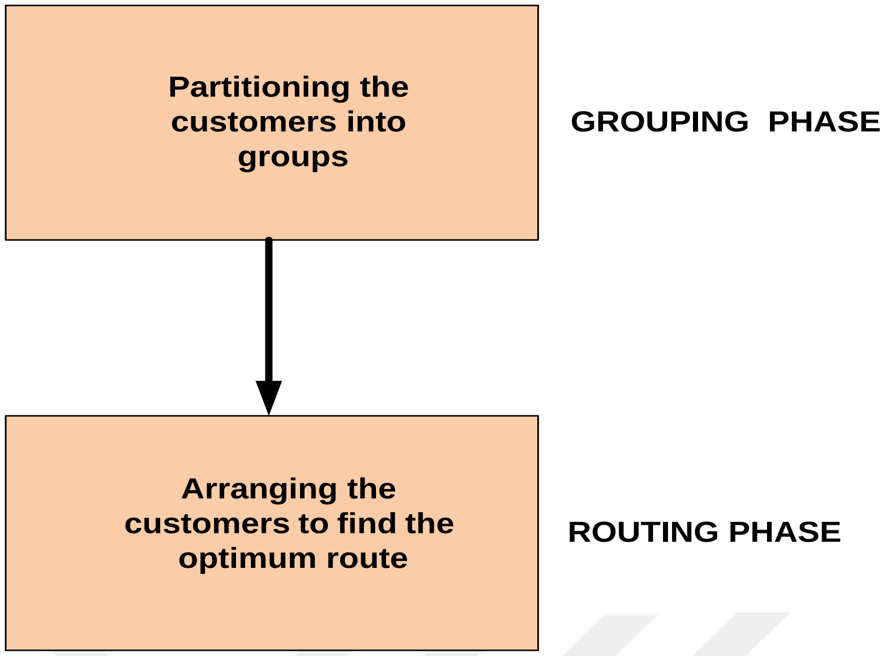

Figure 4.1 The collaborative delivery phases

4.2 Partition the Customers into Groups

The customers are divided into groups based on the distance between customers, number of customers or payload of the delivery truck. Each group has a middle point in a close proximity to all customers in the group. The number of groups is equal to the number of the delivery trucks. The delivery truck is loaded with packages in the main warehouse, at the same time it carries the drone, and moves to the middle point of the group. Once the truck stops, the drone can start the delivery assignment. The *K* means-clustering algorithm used to divide customers into groups Ref. [64]

4.2.1 K-Means Clustering

The **K**-means clustering is an algorithm applied to arrange the objects into a **K** number of groups based on their similarities Ref. [65], where **K** is an integer positive number representing the center of a group and is called a "centroid". The centroids are positioned far away from each other and contain data set needed for a cluster. Next step is that the objects are assigned to the nearest centroid. Then the **K** new centroids need to be re-calculated as by center of the clusters resulting from the previous step. After that **K** new centroids appear to assign a new task has to be done between the objects and the nearest new centroid. Following this, a loop is generated. As a result of this loop, it can be noticed that the **K** centroids gradually change their location until no more changes are done. The procedure of the algorithm is shown in Figure (4.2):

Step (1) calculates the number of groups used equation (4.1).
Step (2) places K points into the space represented by the customers that are clustered. These points represent the initial group centroid.
Step (3) calculates the distance matrix by using Euclidean distance formula equation (4.2).
Step (4) assigns each customer to the group that has the closest centroid.
Step (5) recalculates the positions of the K centroids using equation (4.3) after all customers have been assigned.
Step (6) repeats steps 3, 4, and 5 until the centroids no longer move.

Figure 4.2 The K-means clustering algorithm.

The number of clusters (groups) P_j is selected based on the customer's demand (requirement) and the capacity of delivery truck [66] by using Eq. (4.1)

$$P_N = \sum_{i=1}^{n} \frac{d_i}{Q_T} \tag{4.1}$$

Where P_N is the number of groups (clusters), n is the number of customers, d_i is the demand of each customer and Q_T is the capacity of the delivery truck. The number of groups is generally equal to the number of delivery trucks, which means that equation (4.2) optimizes the number of delivery trucks.

The distance between the customers and the centroids is calculated by the Euclidean distance equation (4.2), and the result is a distance matrix with rows represent the customers and columns represent the centroids.

$$D_{ij} = \sqrt{\left(x_i - \boldsymbol{x}_j\right)^2 + \left(y_i - \boldsymbol{y}_j\right)^2} \qquad \forall i \in V_c\,, \quad \forall j \in V_T \tag{4.2}$$

Where (x_i ,y_i) represents the customer's location and ($\boldsymbol{x}_j$,$\boldsymbol{y}_j$) represents the cluster's centroid, the D_{ij} is calculated for all customers i to every cluster . Therefore, all the customers are assigned to the nearest centroid. Then the centroids are positioned by an iterative procedure and in each step, the centroid of each cluster is calculated until there is no change in the centroid location. The new centroid ($\boldsymbol{x}_j$,$\boldsymbol{y}_j$) of each group can compute in each iteration based on the individual of the cluster as following:

$$\boldsymbol{x_j} = \frac{1}{n_j}\sum_{m=1}^{n_j} x_m\,, \qquad \boldsymbol{y_j} = \frac{1}{n_j}\sum_{m=1}^{n_j} y_m \tag{4.3}$$

Where n_j represents the number of the customers in the cluster j, and x_m , y_m represents the coordinates of the customers in the j^{th} cluster. The K-means clustering algorithm aims to minimize the objective function called Mean Squared Error (MSE).

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i^{(j)} - C_j\right\|^2 \tag{4.4}$$

In the above objective function, $x_i^{(j)}$ shows the location of customer i which is allocated by center C_j to the cluster j . The function is minimize the sum of squared distance between each customer and the centroid of cluster that customer belong to it.

4.3 Routing Problem for the Drone

To find the optimal path with minimum distance, time, and power consumed for the drone when it serve the customers in each group we use two algorithms, nearest neighbor heuristic and the ant colony optimization to compare the solution between them. The first one is a heuristic algorithm while the second is the meta-heuristic algorithm. The concept and procedure are discussed in the following sections:

4.3.1 Nearest Neighbor Heuristic for DRP

The nearest neighbor heuristic, is a simple approach for solving the RP and find the optimal tours for the drone. It was one of the first algorithm used to determine a solution to the RP. The algorithm starts at node 0 which represent the location of the delivery truck and repeatedly visits the nearest customers until all have been visited. All the constraints are checked, if it satisfied the drone will return to the node 0. It quickly yields a short tours, but usually not an optimal [23]. The input to the algorithm is the customer coordinates and the coordinates of the delivery truck. Therefore, the distance matrix d_{ij} is calculated based on those coordinates by using the Euclidean distance formula (4.5). The procedure of the nearest neighbor heuristic algorithm with the constraints is shown in Figure (4.3).

Step (1) start from the location of delivery truck as a starting node.
Step (2) choose the next unvisited customer which nearest to the starting node.
Step (3) check the number of customers can be served in each tour, if it satisfied return to the starting node (location of delivery truck).
Step (4) check the level of the battery drone, if there is enough power to serve next customer and return to the home node go there, if not return to the starting node.
Step (5) check the carrying capacity of the drone, if it satisfied return to the starting node.
Step (6) check if all customers in the group are visited, if not return to the starting node.
Step (7) stop.

Figure 4.3: Procedure of the nearest neighbor heuristic algorithm

The Euclidean distance formula is:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{4.5}$$

4.3.2 Ant Colony Optimization Algorithm.

The ant colony optimization algorithm (ACOA) is an artificial intelligence algorithm initially offered by Macro Dorigo in his Ph.D. research named "Optimization, learning and Natural Algorithms" in 1991 Ref. [67]. The actual behavior of ants in nature was studied while the ants were moving from their nests to search for food. The ants can find the shortest paths while moving from a colony to the food sources and back by providing indirect communication by the means of pheromone trail. While moving the ants lay a constant amount of chemical substances called pheromone on the ground so that the other ants can follow this pheromone trail in probability to obtain the shortest path.

To clarify the real behavior of ants in nature and how they can find the shortest path, an example shown in the Figure (4.4) explains how to reconnect a broken line after an unexpected obstacle on the initial path Ref. [68].

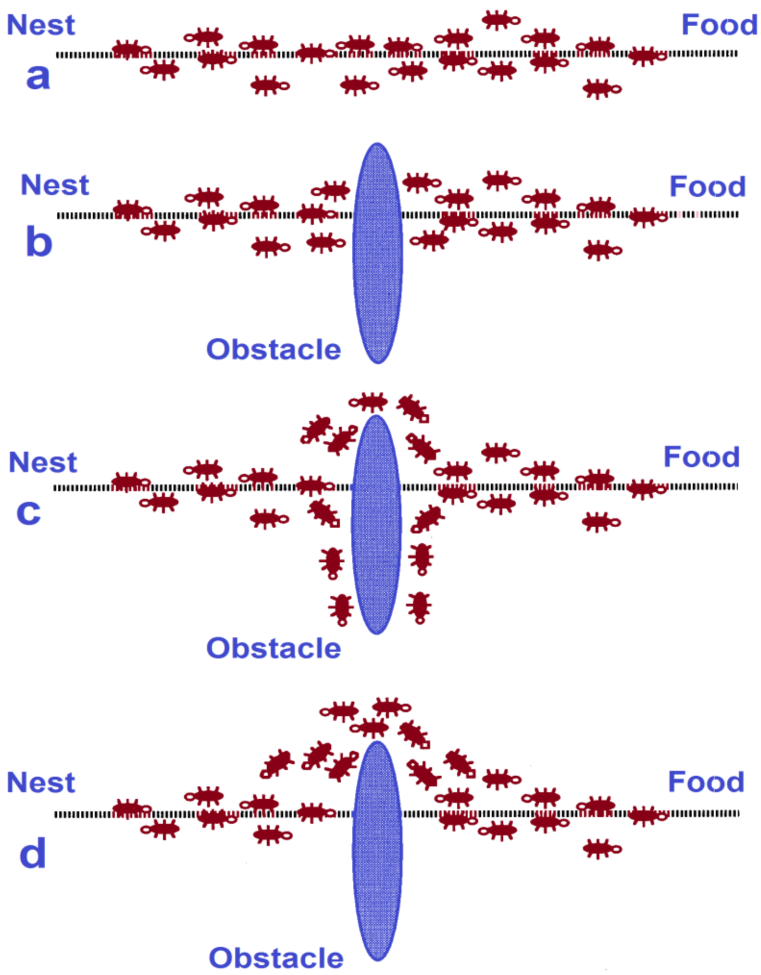

Figure 4.4: Actual behavior of ants in nature.

Ants are moving from their nest in a straight line to link the food source Figure (4.4-a). Suddenly an obstacle occurs and interrupts the previous path, therefore, the ants appearing just in front of the obstacle could not persist to follow the pheromone trail Figure (4.4-b) and consequently, they modify their direction to choose between turning right or left. In this case, half of the ants are expected to choose to turn right and the other half to turn left Figure (4.4-c). The ants which randomly select the shortest path around the obstacle will faster reconnect the interrupted pheromone trail compared to those who choose the longer path. Thus, the shorter path will get more amount of pheromone trail per time unit and a larger number of ants will probably prefer the shortest path Figure (4.4-d). The ants are able to find the shortest path from the nest to the food source around their nest without using any spatial information, but only communicating by using pheromone. The algorithm is described as follows:

- A few of ants search randomly for food around the nest.
- One of them finds the food source
- The ant laying down pheromone trail when it goes back to the nest after discovering the food.
- When other ants travel randomly and suddenly find a pheromone trail, they are probably not to keep moving at random but instead follow the pheromone trail.
- By following the pheromone trail the ants finally find the food and when they go back to the nest they will reinforce the trail with more pheromones.
- Due to their stochastic behavior, some ants do not follow the pheromone trail, and thus, uncover more possible paths.
- However, at some point, the pheromone trail starts to evaporate which leads to its attractive strength to reduce
- The shortest path is obtained.

4.3.2.1 Ant Colony Optimization Algorithm for DRP.

The vehicle routing problem includes finding out the shortest planned route which has minimum total routing distance or time, or minimum combined total routing time and service time. The Ant Colony Optimization (ACO) is a meta-heuristic method that models the intelligent behavior of ant colonies. It is applied to solve the hard combinatorial optimization problems such as VRP Ref. [69, 70]. In ACO for DRP, there are k ants, each ant k placed on a starting node (delivery truck). Each ant generates one solution, which means that each iteration contains k solutions and one of them is saved as the best solution, where k is the number of ants. In ACO algorithm the drone is simulated by an individual artificial ant and at each iteration, each ant k constructs its route by incrementally selecting customers until all customers have been visited. The ant k starts and finishes at the same location (sub-depot), at each step the ant moves from current customer *i* to next customer *j* based on the state probabilistic transition rule using one of two methods Ref. [71] [72]:

- Exploitation method: It leads the ant k to select the next customer which path has highest value of pheromone by using equation (4.6) as following:

$$j = \begin{cases} \arg\max \left\{ \left(\tau_{ij}\right)\left(\eta_{ij}\right)^{\beta} \right\} & for\ j \notin M_k\ ,\quad if\ q \leq q_0 \\ P_{ij} & otherwise \end{cases} \tag{4.6}$$

Where τ_{ij} is the amount of pheromone on the route between the current location *i* and possible location j, which represent the number of possible unvisited customers. η_{ij} is the inverse distance between customer locations $\left(\frac{1}{d_{ij}}\right)$, and β establishes the importance of distance in comparison to the pheromone quantity in the selection algorithm ($\beta > 0$). M_k is the ants working memory keeping track of customers already visited.

- Exploration method: It leads the ant k to select the next customer randomly according to the probability distribution formula given by equation (4.7), which support the selection based on short paths and high value of pheromone:

$$P_{ij}^{k} = \begin{cases} \dfrac{\left[\tau_{ij}\right]^{\varepsilon}\left[\eta_{ij}\right]^{\beta}}{\sum_{j \notin M_k}\left[\tau_{ij}\right]^{\varepsilon}\left[\eta_{ij}\right]^{\beta}} & if\ j \notin M_k \\ 0 & otherwise \end{cases} \tag{4.7}$$

Where P_{ij}^{k} is the probability that ant *k* moves from customer *i* to customer *j*, and ε is a parameter controlling the influence of τ_{ij} .

To weighting between exploitation and exploration a parameter q_0 is used in equation (4.6). This parameter is a predetermined to find the relative importance of the exploitation versus exploration and it values between 0 and 1. The exploitation is selected when $q < q_0$ which governs to use equation (4.6), otherwise the exploration is selected which governs to use equation (4.7). The q value is a random uniform variable distributed in [0,1].

If the number of customers, energy consumption, vehicle capacity and flight time constraints are satisfying, the ant k will return to the starting location (sub-depot) before moving to j customer. This selection process continues until an ant visits all customers.

To improve the solution the pheromone trail must be updated to track the colony's movements. This update is done locally after individual solutions have been constructed and globally for the best solution found after a predefined number of solutions m has been constructed. The local update is made by reducing the amount of pheromone deposited on each edge (i,j) visited by an ant when moving from customer i to customer j. Therefore, last ants will select the arc in a current cycle which simulates the natural evaporation of pheromone. It is given by the following local trail updating formula:

$$\tau_{ij} = (1 - \gamma)\tau_{ij} + \gamma\, \tau_0 \tag{4.8}$$

where $(0 \leq \gamma \leq 1)$ is the rate of evaporation of the pheromone trail and τ_0 is the initial pheromone value for all edges. The global update is done by increasing the amount of pheromone deposited on all edges of complete routes obtained as the best solutions after all ants have completed a route. This update is given by the following global formula:

$$\tau_{ij} = (1 - \gamma)\tau_{ij} + \gamma L^{-1} \tag{4.9}$$

where L is the value of the best solution. This update guides the other ants to follow this best route with more probability in the sequent iteration to find the optimal solution with the repetition process continuing until a terminating condition is met. The procedure of the algorithm is show in Figure (4.5).

Step (1) start the ACO.
Step (2) Input the DRP instance to be solved.
Step (3) Initialize all parameters.
Step (4) check the maximum iteration, if it exceed go to step (18).
Step (5) Move ants to the start point.
Step (6) Start construct the routs from sub-depot.
Step (7) Compute ant's probability of going to unvisited nodes using equation (4.6) or equation (4.7).
Step (8) Select a node according to the probability.
Step (9) Check the constrains condition, if they satisfied go to step (5).
Step (10) Go to the selected node.
Step (11) Have all customers been visited? If not, go to step (7).
Step (12) Save the best solution.
Step (13) Have all ants constructed their routs? If not, go to step (6).
Step (14) Update the pheromones trails by using equation (4.9).
Step (15) Evaporate the pheromone trails by using equation (4.8).
Step (16) Store the best cost.
Step (17) Go to step (4)
Step (18) Stop.

Figure 4.5: Procedure of ACOA for DRP.

CHAPTER 5

EXPERIMENT AND RESULTS

5.1 Introduction

In this chapter the collaborative delivery system model is simulated in MATLAB by implement the optimization algorithms. First, the drone power consumption model is simulated by implement the equation (3.6) in MATLAB/SIMULINK to obtain the relation between the payload weight and the power consumed, also the power required to keep the drone frame in the air and the power consumed rate per Kg are obtained from the relation. Next, the customers are patriating into groups and the location of the delivery truck is selected by implement the K-means clustering algorithm. Then, for each group the ACOA and NNA are implemented to solve the drone routing problem. To tests the algorithms we generated randomly Euclidean instances with different size and distance square area. This random generation methods are commonly used in Ref. [59, 73]. Firstly, two scenarios are studying in detail, one with 10 customers, and the other with 32 customers. Then, different size instances with different square area are generated to compere between algorithms in the solution and run time.

5.2 Power Consumption and Payload Weight Relationship.

To find the relation between payload weight and power consumed by the drone the equation (3.6) is implement in SIMULINK which represent the model of power consumption of the drone as shown in figure (5.1). The data in table (3.1) is used in the SIMULINK model, and the payload weight is variant from 0 to 4 Kg. The result obtain from this model is used to drive the relation between the payload weight and the power consumption and shown in figures (5.2) and (5.3).

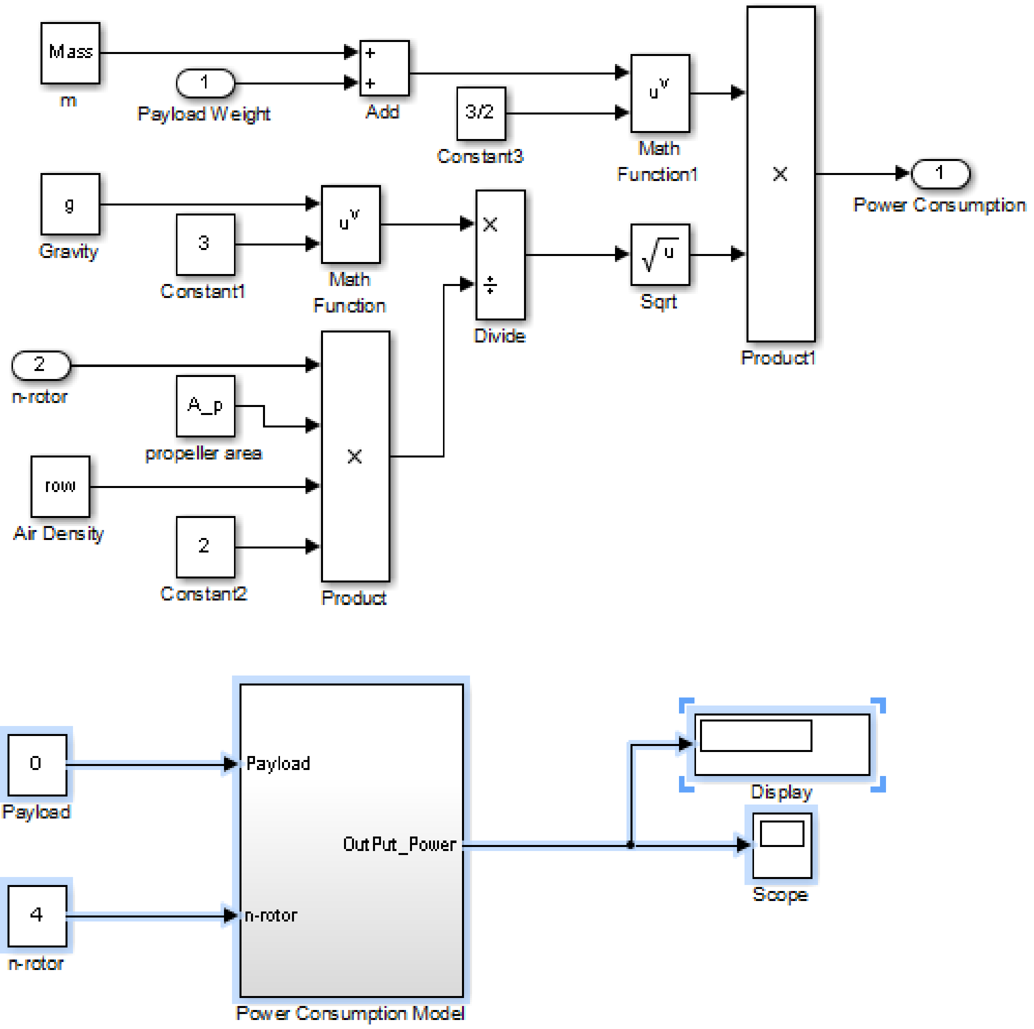

Figure 5.1: The power consumption model in SIMULINK.

The result is shown in figures (5.2), (5.3), and (5.4) for three types of UAVs, quadrotor with 4 rotors, hexacopter with 6 rotors and UAV with 8 rotors. The figures present that the linear approximate model dashed thin lines equation (3.7) is closely fitted to the exactly model continuous thick line equation (3.6). For drone with 4 rotor (quadrotor), the error percentage is variation from 0% to 4.57% as the payload variant from 0.5 kg to 4 kg with difference variant in power from 0 to 11.6 watts. The result in figure (5.2) show that the power required to keep the drone frame in the air $\alpha = 21.44\, watt$, which obtain when the payload equal to zero, and the power consumed rate per kg is, $\mu = 69.4\, watt/kg$.

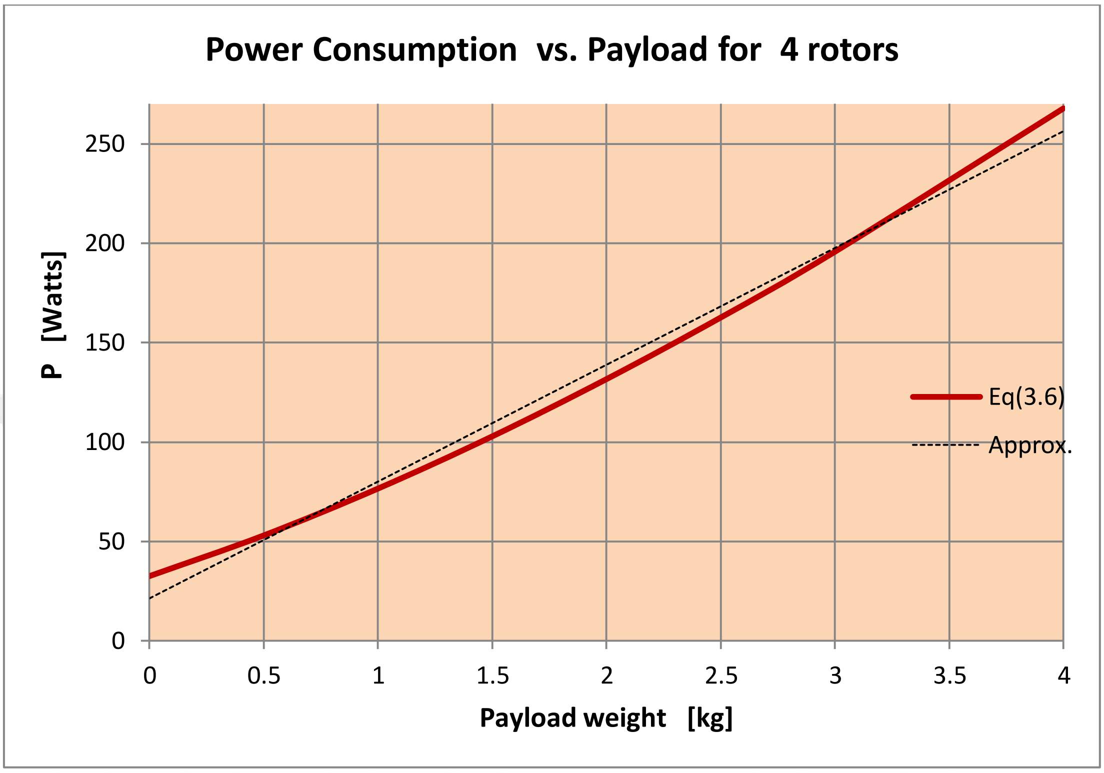

Figure 5.2: The relation between payload and power consumption for 4 rotors, obtain from Eq. (3.6) and Eq. (3.7). The continuous thick line represents Eq. (3.6) and dashed thin line represents the approximate.

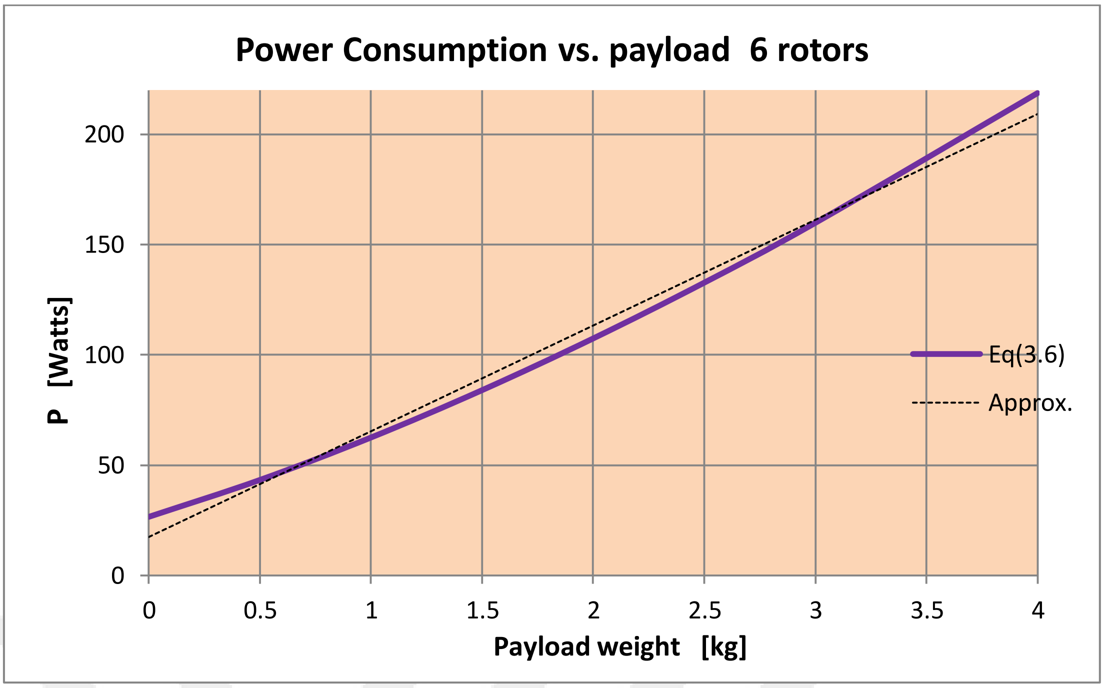

Figure 5.3: The relation between payload and power consumption for 6 rotors, obtained from Eq. (3.6) and Eq. (3.7). The continuous thick line represents Eq. (3.6) and dashed thin line represents the approximate.

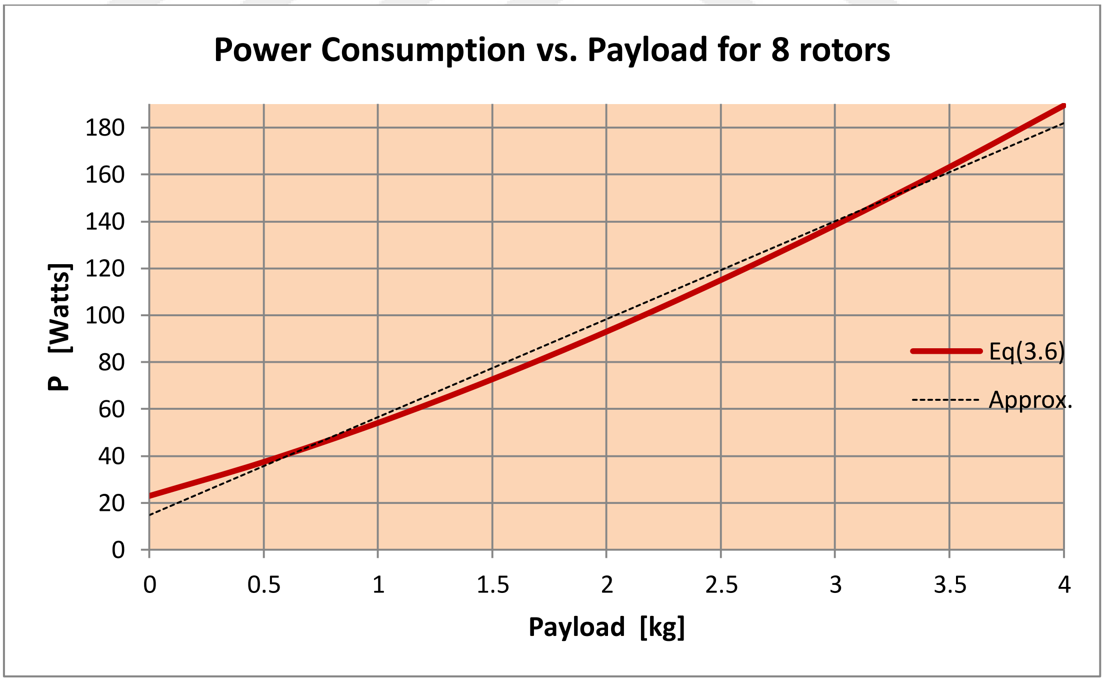

Figure 5.4 The relation between payload and power consumption for 8 rotors, obtained from Eq. (3.6) and Eq. (3.7). The continuous thick line represents Eq. (3.6) and dashed thin line represents the approximate.

5.3 Assumptions

The collaborative delivery system is considered under the following conditions:

- All the customers will service by drones exactly once, and the delivery truck works as a base for the drone (sub-depot).
- The drone flies from customers i to j at constant speed without any obstacles.
- The drone could service two customers with the same demand and the combined demand of two customers must be less than the drone's carrying capacity.

For the first scenario 10 customers are considered as one group, and serve with one drone the location coordinates for the customers are lasted in Table (5.1).

Table 5.1: Location coordinates for n=10

Customers No.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
***x*-coordinate /km**	15	5	12	14	7	13	16	8	3	3
***y*-coordinate /km**	10	2	3	5	5	7	9	8	9	12

Next we run the K-means clustering algorithm to find the center of the group which, represent the coordinate location of delivery truck at (9.6,7) the result is show in Figure (5.5).

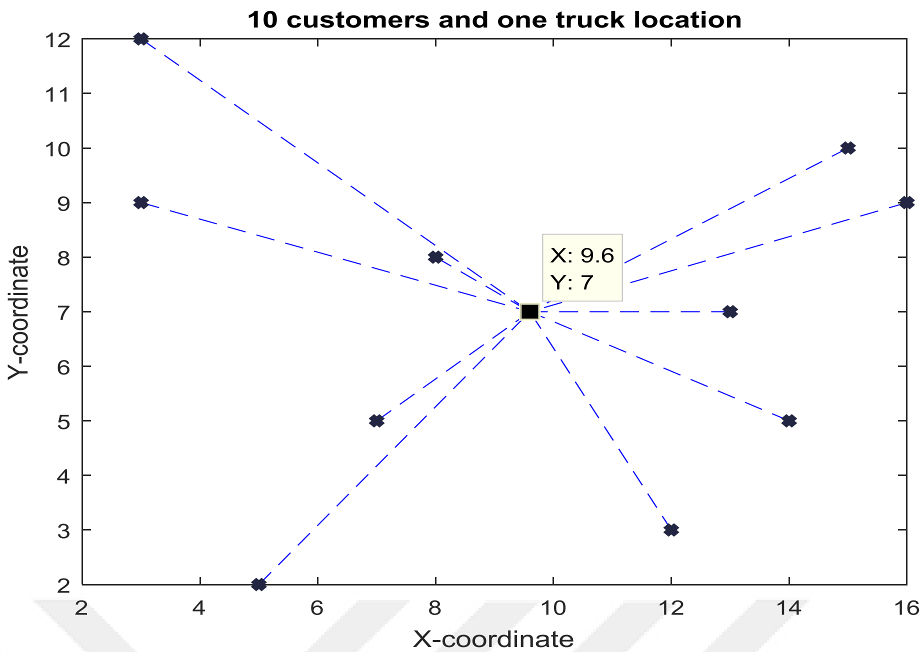

Figure 5.5: The location of delivery truck at coordinate x=9.6km and y=7km.

After introduce the location of the delivery truck (the center of the group) which is taken as the start location to serve all customers, the problem is solved by the drone to minimize the time of delivery and the power consumed, the drone will serve two customers at each tour, the speed is assumed constant when travel from customer i to j refer to Table (3.1), the speed $v = 16\ \frac{m}{sec} = 0.016\ \frac{Km}{sec} =\ 0.960\ \frac{Km}{min}$. We assume the time at each customer govern the landing, deliver packages and take off is $\lambda = 1min,$ and the recovery service time for the drone to be ready for the next mission when it is at base (sub-depot) is $t_r = 2min$. Therefore the NNA and ACOA are run to find the delivery time and the optimal path with minimum power and the result is shown in Figures (5.6) and (5.7) and Tables (5.2), (5.3). The result obtain for the drone will compare with a regular delivery truck, the speed of truck is assume and equal $40\ \frac{Km}{h}$, the tours and the delivery time is shown in Figures (5.8) and (5.9) and Table (5.4).

Table 5.2: The tours, distance traveled, delivery time for each customer using a nearest neighbor algorithm.

Tour No	Customers	Distance (Km)	Delivery time by drone for each tour (min.)	Delivery time (min.)
1	0→8→5→0	8.3293	2.9654→7.2595→10.6764	2.9654→7.2595→10.6764
2	0→6→4→0	10.4693	4.5417→ 7.8709→12.9055	17.2180→20.5473→25.5819
3	0→3→2→0	18.5300	5.8591→14.2248→21.3020	33.4410→41.8067→48.8839
4	0→1→7→0	14.2968	7.4348→9.9079→16.8925	58.3187→60.7918→67.7764
5	0→9→10→0	18.1765	8.1837→12.3087→20.9338	77.9601→82.0851→90.7102
Average Delivery time			8.0557	40.2394

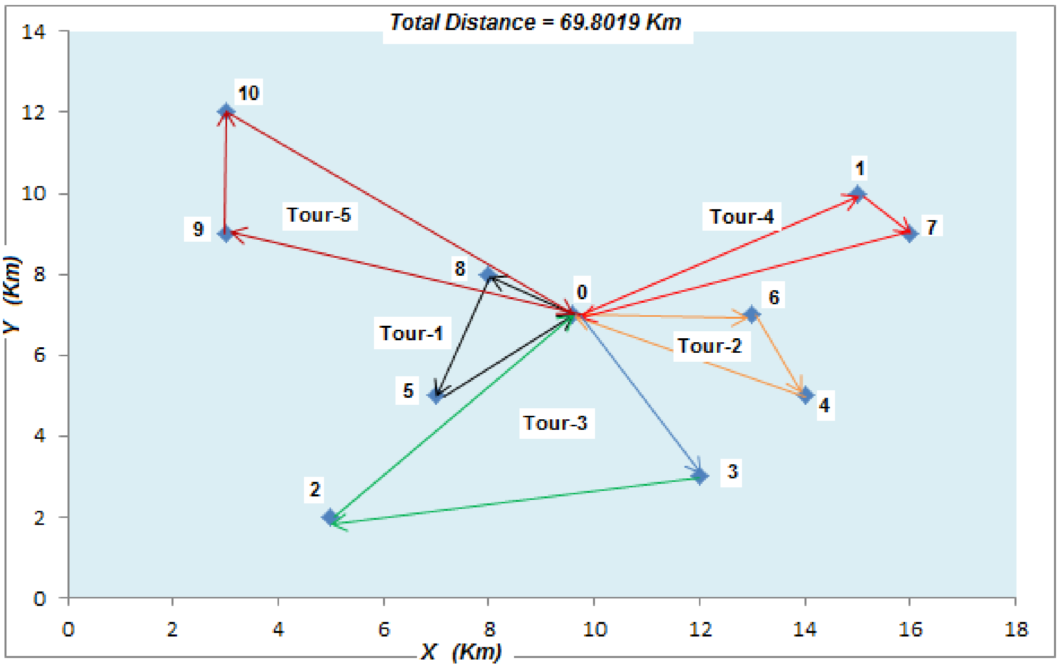

Figure 5.6: The tours obtains by NNA for the drone with 10 customers, location 0 represent the delivery truck.

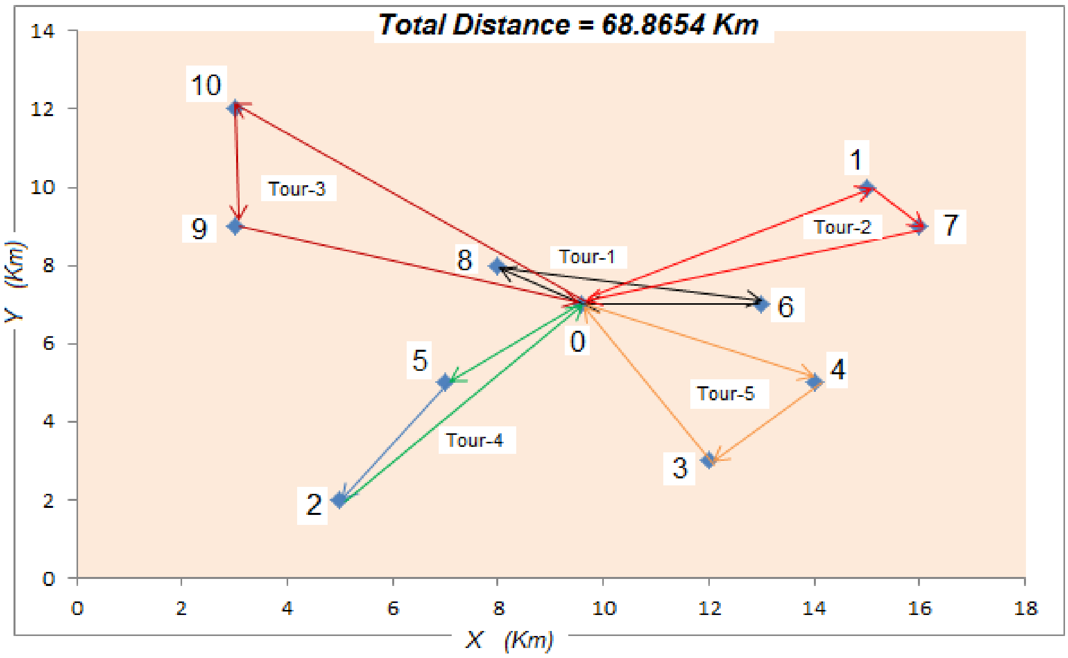

Figure 5.7: The tours obtains by ACOA for the drone with 10 customers, location 0 represent the delivery truck.

Table 5.3: The tours, distance traveled, delivery time for each customer using ant colony optimization algorithm by using one drone and a drone for each tour.

Tour No	Customers	Distance (Km)	Delivery time by drone for each tour (min.)	Delivery time by one drone (min.)
1	0→8→6→0	10.3858	2.9654→9.2768→12.8184	2.9654→9.2768→12.8184
2	0→1→7→0	14.2968	7.4347→9.9078→16.8923	22.2531→24.7262→31.7107
3	0→10→9→0	18.1765	9.6251→13.7501→20.9338	43.3335→47.4585→54.6422
4	0→5→2→0	13.6799	4.4168→9.1726→16.2497	61.0590→65.8148→72.8919
5	0→4→3→0	12.3264	6.0344→9.9806→14.8397	80.9264→84.8726→89.7317
Average Delivery time			8.2564	44.2686

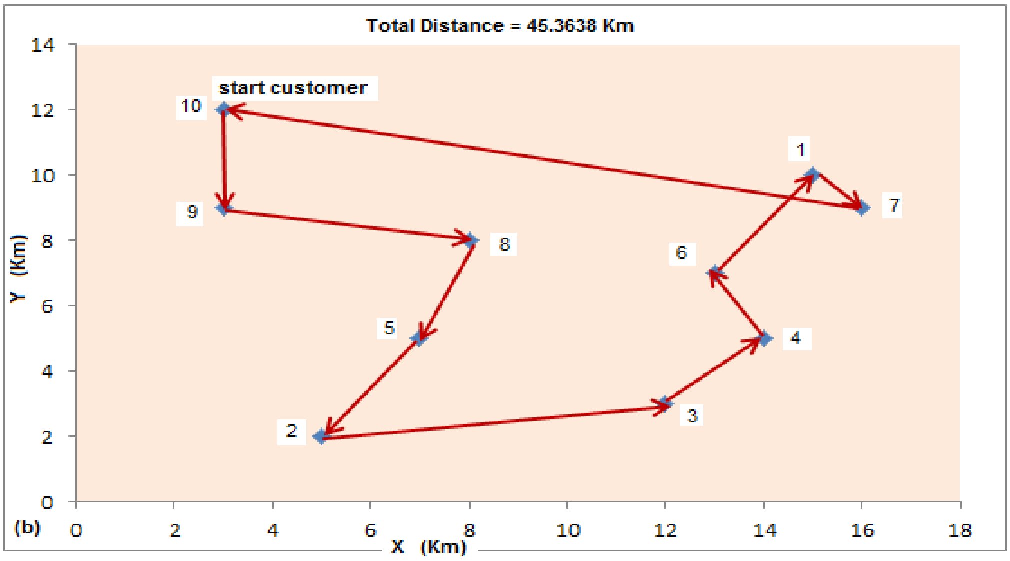

Figure 5.8: Tours and distance traveled by the truck for n=10 obtained by (a) NNA

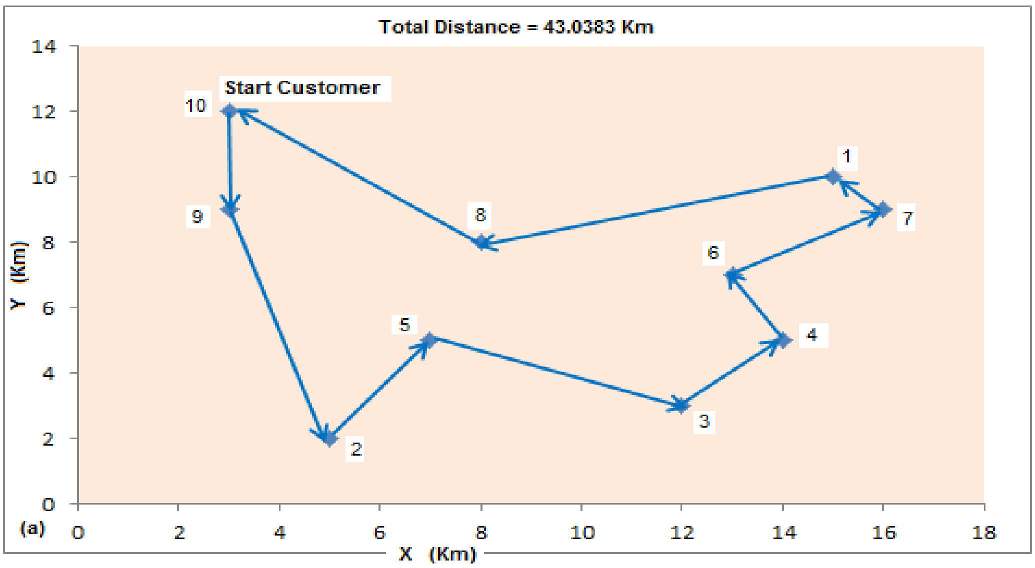

Figure 5.9: Tours and distance traveled by the truck for n=10 obtained by ACOA.

Table 5.4: Tours, distance traveled and delivery time using a truck only for n=10 obtain by ACOA and NNA.

Algorithm	ACOA	NNA
Tours	10→9→2→5→3→4→6→7→1→8→10	10→9→8→5→2→3→4→6→1→7→10
Distance traveled for each customer (Km)	3→ 10.2801→ 13.8857→ 19.2708→ 22.0993→ 24.3353→ 27.9409→ 29.3551→ 36.6352→ 43.0383	3→ 8.0990→ 11.2613→ 14.8668→ 21.9379→ 24.7663→ 27.0024→ 30.6080→ 32.0222→ 45.3638
Delivery time for each customer (min)	6.4978→ 19.4125→ 26.8181→ 36.8918→ 43.1323→ 48.4847→ 55.8904→ 60.0106→ 72.9253→ 84.5252	6.4978→ 16.1425→ 22.8835→ 30.2891→ 42.8904→ 49.1309→ 54.4834→ 61.8890→ 66.0093→ 86.0118
Average Delivery time	**45.4589**	**43.6228**

We consider the energy consumption at each customer by using a equation (3.7) to calculate the power consumed by the drone when travel between customers and delivery truck.

$$P(m_L) = \mu\, m_L + \alpha$$

We assume that the maximum payload is $m_L = 1.5kg$, and from figure (5.2) the power consumed per kilogram $\mu = 69.4\ W/kg$, and the power required to keep the drone frame in the air $\alpha = 21.44\ W$. As we mention previously the drone serves two customers with the same demands at each tour, therefore, the drone moves from base 0 to customer j with maximum payload which will make the power consumption to the maximum level. In the same way, when the drone moves to the second customer after visited the first customer the payload reduced to 50 % from the maximum which will reduce the power also. Finally, when the drone return to the base (delivery truck) the payload is zero, wherefore, from equation (3.7) the power equal to the required power to keep the drone in the air α .

The energy consumed by the drone is changed depending on the payload and the flight time and can calculated by:

$$E = P\, t_f \tag{5.1}$$

Where, E is the energy consumed in joules (J), P is the power in watts [W] which calculated by equation (3.7), t_f is the flight time in seconds [sec], the result for ten customers is shown in table (5.5).

Table 5.5: The tours, distance traveled, and power consumption in each customer location for n=10, served by drone using a nearest neighbor algorithm.

Tour No	Customers	Distance [km]	Power consumption [W.sec]
1	0→8→5→0	8.3293	372.2779→687.8469 →779.9111
2	0→6→4→0	10.4693	570.1608→ 814.8265→ 886.2053
3	0→3→2→0	18.5300	735.5548→ 1350.3→ 1529.7
4	0→1→7→0	14.2968	933.3609→ 1115.1→ 1168.1
5	0→9→10→0	18.1765	1027.4→1330.5→ 1419

For the second scenario 32 customers are consider with coordinate locations show in Table (5.6) to find the tours, delivery time and power consumption . First, the problem is solved by K-means cluster algorithm to patriating the customers in to groups based on equation (4.1) and find the location of delivery truck. The problem is solve two times, first, the customers are consider as one group and in the second time they consider as two groups, the result is show in Figures (5.10) and (5.11).

Table 5.6: The coordinate locations for 32 customers.

Customers No.	x-coordinate /km	y-coordinate /km
1	10	16
2	7	18
3	12	7
4	18	3
5	0	10
6	18	4
7	6	16
8	5	10
9	15	19
10	4	2
11	10	3
12	10	5
13	17	5
14	10	7

15	10	9
16	8	7
17	13	8
18	9	12
19	12	15
20	1	12
21	9	13
22	10	12
23	4	6
24	10	18
25	5	16
26	17	18
27	7	16
28	2	18
29	13	10
30	15	2
31	12	2
32	16	9

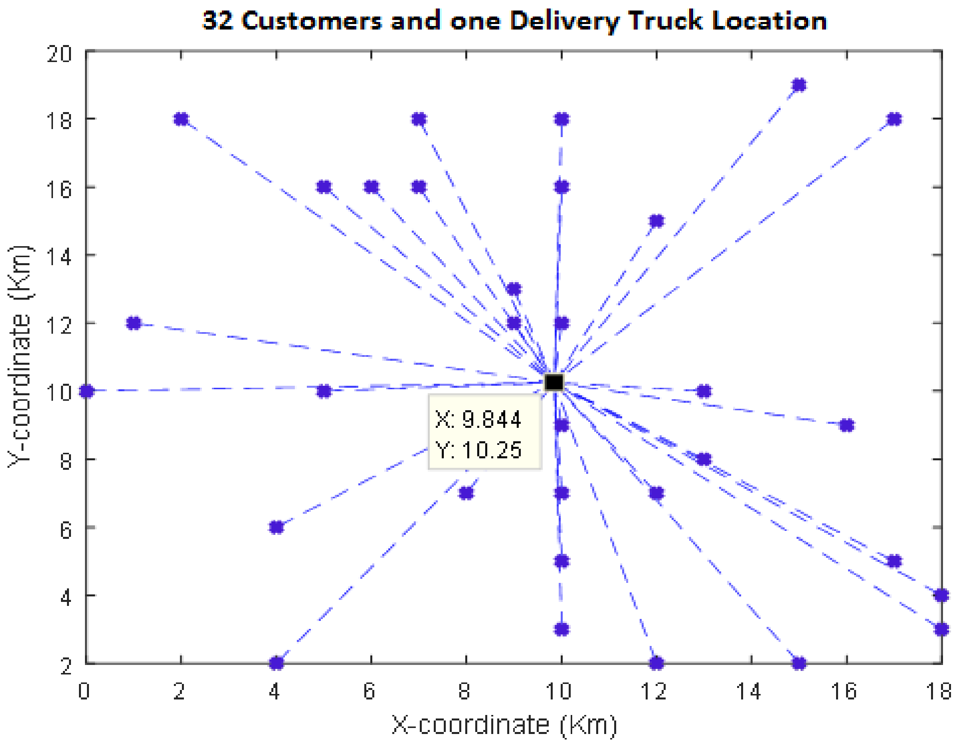

Figure 5.10: The location of delivery truck at coordinate x=9.844km and y=10.25km.

Figure 5.11: The location of 32 customers and two delivery truck, one at coordinate x=11.88km and y=5.563km, and the other at coordinate x=7.813km and y=14.94km.

The solution of 32 customers and two delivery trucks are represent in Table (5.7), where there are two delivery trucks and 32 customers, each customer allocated to one delivery truck.

Table 5.7: Solution for 32 customers and two delivery truck.

Customers No.	Delivery Truck
1	2
2	2
3	1
4	1
5	2
6	1
7	2
8	2
9	2
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1
18	2
19	2
20	2
21	2
22	2
23	1
24	2
25	2
26	2
27	2

28	2
29	1
30	1
31	1
32	1

We take the location of the delivery truck as a start node for each group and run the algorithms to obtain the tours and average delivery time to serve the customers by one drone for each group or use more than one drone for each group, the results are shown in Figures (5.12) and (5.13).

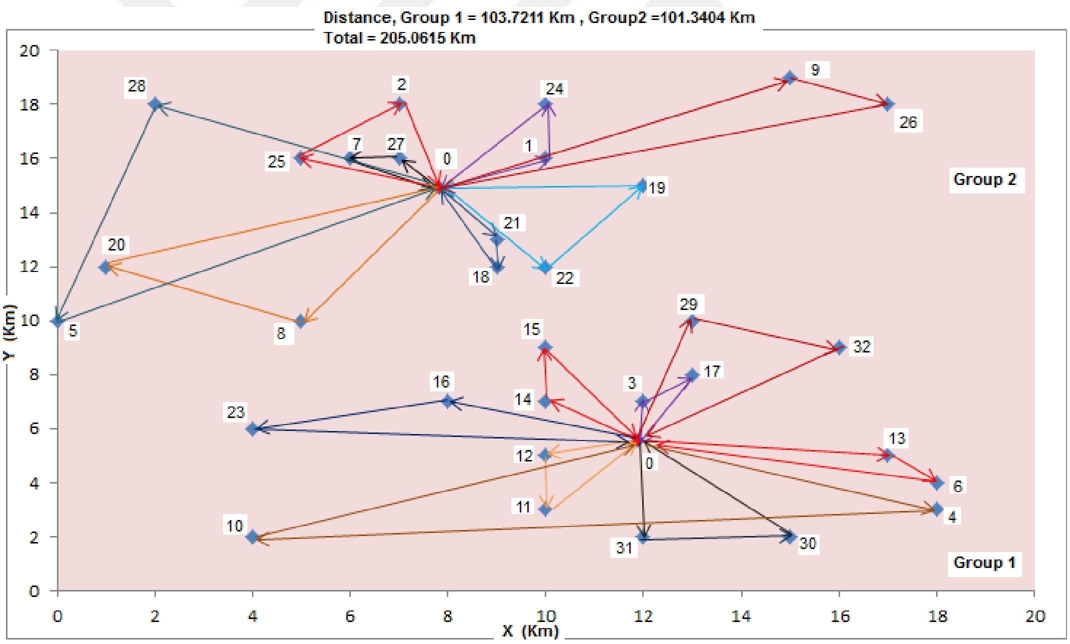

Figure 5.12: The tours and total distances for two groups served by drone NNA.

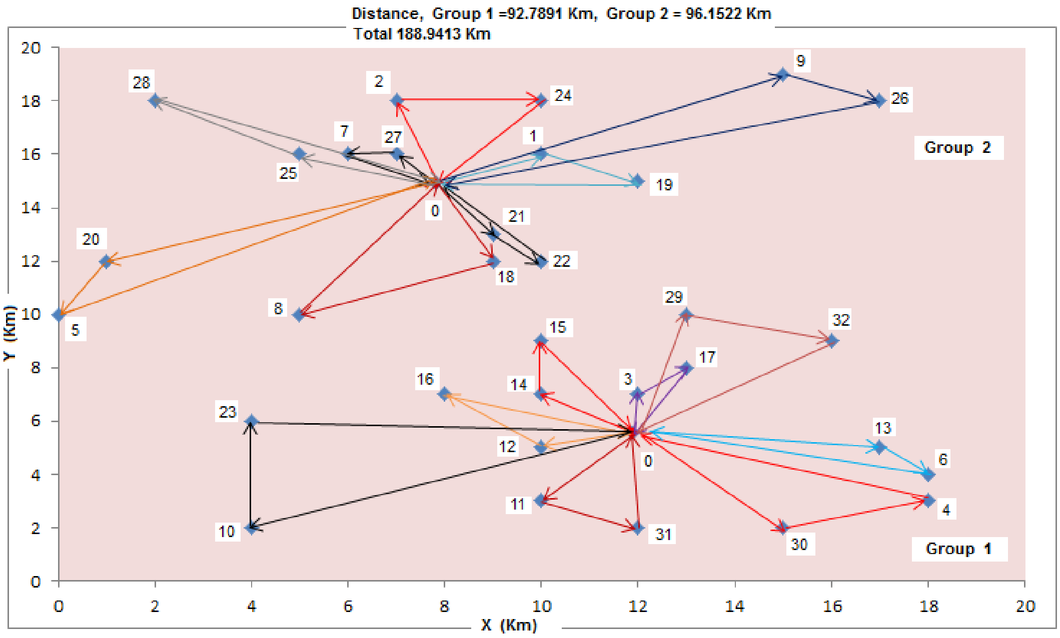

Figure 5.13: Tours and total distances traveled for two groups served by drone obtained by ACOA.

Tables (5.8) - (5.12) show the average delivery time for two groups serve by one drone for each tour, one drone for two tours, one drone for 4 tours and all tours in the group serve by one drone.

Table 5.8: The tours, distance traveled, and delivery time in each customer location for n=32, served by drone using a nearest neighbor algorithm.

Tour No	Customers	Distance [km]	Delivery time [min]	Average Delivery time [min]
		Group one		
1	0→3→17→0	5.5383	2.5021→4.9752→7.7690	7.4128
2	0→12→11→0	7.1411	3.0443→6.1276→9.4386	
3	0→14→15→0	8.2839	3.4649→6.5482→10.6290	
4	0→31→30→0	11.3010	4.7136→8.8386→13.7719	
5	0→16→23→0	16.1528	5.3100→10.6049→18.8258	
6	0→29→32→0	13.1038	5.7668→10.0609→15.6498	
7	0→13→6→0	12.8815	6.3655→8.8386→15.4182	
8	0→4→10→0	29.3188	7.9115→23.5320→32.5404	
		Group two		
1	0→27→7→0	4.4360	2.3915→4.4332→6.6208	7.4783
2	0→21→18→0	6.4449	3.3691→5.4108→8.7134	
3	0→1→24→0	8.1915	3.5316→6.6149→10.5328	
4	0→25→2→0	9.0007	4.1313→8.0776→11.3757	
5	0→22→19→0	11.4572	4.8169→9.5727→13.9346	
6	0→8→20→0	17.5772	6.9216→12.5801→20.3096	
7	0→28→5→0	24.0592	7.8429→17.4327→27.0616	
8	0→9→26→0	20.1738	9.5984→12.9277→23.0143	

Table 5.9: The tours, distance traveled, and delivery time in each customer location for n=32, served by 8 drones using a ACOA.

Tour No	Customers	Distance [km]	Delivery time [min]	Average Delivery time [min]
		Group one		
1	0→11→31→0	8.9797	4.3110→ 7.6403→ 11.3538	**6.9908**
2	0→12→16→0	8.9285	3.0443→ 6.9905→ 11.3005	
3	0→14→15→0	8.2839	3.4649→ 6.5482→ 10.6290	
4	0→30→4→0	14.5333	5.9333→ 10.2273→ 17.1388	
5	0→3→17→0	5.5383	2.5021→ 4.9752→ 7.7690	
6	0→29→32→0	13.1038	5.7668→10.0609→15.6498	
7	0→13→6→0	12.8815	6.3655→8.8386→15.4182	
8	0→10→23→0	20.5402	10.0084→15.1751→23.3960	
		Group two		
1	0→27→7→0	4.4360	2.3915→4.4332→6.6208	**6.8591**
2	0→21→22→0	7.3528	3.3691→5.8422→9.6591	
3	0→25→28→0	13.1809	4.1313→ 8.8871→15.7301	
4	0→2→24→0	9.9274	4.2981→8.4231→12.3410	
5	0→1→19→0	8.8538	3.5316→6.8608→11.2228	
6	0→18→8→0	13.3275	4.3027→9.9612→15.8828	
7	0→9→26→0	20.1738	9.5984→12.9277→23.0143	
8	0→20→5→0	18.9001	8.7295→12.0587→21.6876	

Table 5.10: The tours, distance traveled, and delivery time in each customer location for n=32, served by 4 drones using a ACOA.

Tour No	Customers	Distance [km]	Delivery time [min]	Average Delivery time [min]
		Group one		
1	0→11→31→0	8.9797	4.3110→ 7.6403→ 11.3538	**13.6148**
2	0→12→16→0	8.9285	16.3981→ 20.3444→ 23.6543	
3	0→14→15→0	8.2839	3.4649→ 6.5482→ 10.6290	
4	0→30→4→0	14.5333	18.5623→ 20.8564→ 28.7678	
5	0→3→17→0	5.5383	2.5021→ 4.9752→ 7.7690	
6	0→29→32→0	13.1038	16.3580→20.6520→24.4189	
7	0→13→6→0	12.8815	6.3655→8.8386→15.4182	
8	0→10→23→0	20.5402	27.4267→32.5933→39.8143	
		Group two		
1	0→27→7→0	4.4360	2.3915→4.4332→6.6208	**14.9326**
2	0→21→22→0	7.3528	11.9899→14.4631→18.2800	
3	0→25→28→0	13.1809	4.1313→ 8.8871→15.7301	
4	0→2→24→0	9.9274	22.0281→26.1531→30.0710	
5	0→1→19→0	8.8538	3.5316→6.8608→11.2228	
6	0→18→8→0	13.3275	17.5254→23.1839→29.1055	
7	0→9→26→0	20.1738	9.5984→12.9277→23.0143	
8	0→20→5→0	18.9001	33.7438→37.0730→46.7019	

Table 5.11: The tours, distance traveled, and delivery time in each customer location for n=32, served by two drones using a ACOA.

Tour No	Customers	Distance [km]	Delivery time [min]	Average Delivery time [min]
		Group one		
1	0→11→31→0	8.9797	4.3110→ 7.6403→ 11.3538	**27.1553**
2	0→12→16→0	8.9285	16.3981→ 20.3444→ 24.6543	
3	0→14→15→0	8.2839	30.1192→ 33.2025→ 35.2833	
4	0→30→4→0	14.5333	45.2166→ 49.5107→ 53.4222	
5	0→3→17→0	5.5383	2.5021→ 4.9752→ 7.7690	
6	0→29→32→0	13.1038	15.5359→19.8299→25.4189	
7	0→13→6→0	12.8815	33.7843→36.2575→42.8371	
8	0→10→23→0	20.5402	54.8455→60.0122→64.2331	
		Group two		
1	0→27→7→0	4.4360	2.3915→4.4332→6.6208	**27.7790**
2	0→21→22→0	7.3528	11.9899→14.4631→18.2800	
3	0→25→28→0	13.1809	24.4113 → 29.1671 → 36.0100	
4	0→2→24→0	9.9274	42.3081 → 46.4331 → 50.3510	
5	0→1→19→0	8.8538	3.5316→6.8608→11.2228	
6	0→18→8→0	13.3275	17.5254→23.1839→29.1055	
7	0→9→26→0	20.1738	40.7040 → 44.0332 → 54.1199	
8	0→20→5→0	18.9001	64.8493 → 68.1786 → 74.8075	

Table 5.12: The tours, distance traveled, and delivery time in each customer location for n=32, served by one drone using a ACOA.

Tour No	Customers	Distance [km]	Delivery time [min]	Average Delivery time [min]
		Group one		
1	0→11→31→0	8.9797	4.3110→ 7.6403→ 11.3538	**56.3664**
2	0→12→16→0	8.9285	16.3981→ 20.3444→ 24.6543	
3	0→14→15→0	8.2839	30.1192→ 33.2025→ 37.2833	
4	0→30→4→0	14.5333	45.2166→ 49.5107→ 56.4222	
5	0→3→17→0	5.5383	60.9242→ 63.3974→ 66.1912	
6	0→29→32→0	13.1038	73.9580→78.2521→83.8410	
7	0→13→6→0	12.8815	92.2065→ 94.6796→ 101.2592	
8	0→10→23→0	20.5402	113.2677→ 118.4343→ 126.6553	
		Group two		
1	0→27→7→0	4.4360	2.3915→4.4332→6.6208	**53.9545**
2	0→21→22→0	7.3528	11.9899→14.4631→18.2800	
3	0→25→28→0	13.1809	24.4113 → 29.1671 → 36.0100	
4	0→2→24→0	9.9274	42.3081→ 46.4331→ 50.3510	
5	0→1→19→0	8.8538	55.8826→59.2119→63.5738	
6	0→18→8→0	13.3275	69.8765→75.5349→81.4566	
7	0→9→26→0	20.1738	93.0550→ 96.3842→ 106.4709	
8	0→20→5→0	18.9001	117.2004→ 120.5296→ 130.1585	

The power consumption for each tour in groups is show in Tables (5.13) and (5.14)

Table 5.13: The tours, distance traveled, and power consumption in each customer location for n=32, served by drone using a nearest neighbor algorithm.

Tour No	Customers	Distance [km]	Power consumption for each tour [kJ]	Total Power consumption [kJ]
			Group one	
1	0→3→17→0	5.5383	18.847→10.905→3.594	33.3460
2	0→12→11→0	7.1411	22.931→13.596→4.259	40.7860
3	0→14→15→0	8.2839	26.099→13.596→5.250	44.9450
4	0→31→30→0	11.3010	35.504→18.189→6.346	60.0390
5	0→16→23→0	16.1528	39.997→23.347→10.575	73.9190
6	0→29→32→0	13.1038	43.438→18.934→7.190	69.5620
7	0→13→6→0	12.8815	47.947→10.905→8.464	67.3160
8	0→4→10→0	29.3188	59.592→68.877→1.159	129.6280
			Group two	
1	0→27→7→0	4.4360	18.014→ 9.003→2.814	29.8310
2	0→21→18→0	6.4449	25.377→9.003→4.249	38.6290
3	0→1→24→0	8.1915	26.601→13.596→5.040	45.2370
4	0→25→2→0	9.0007	31.119→17.401→4.243	52.7630
5	0→22→19→0	11.4572	36.283→20.970→5.611	62.8640
6	0→8→20→0	17.5772	52.136→24.950→9.943	87.0290
7	0→28→5→0	24.0592	59.076→42.285→12.387	113.7480
8	0→9→26→0	20.1738	72.299→14.680→12.976	99.9550

Table 5.14: The tours, distance traveled, and power consumption in each customer location for n=32, served by drone using a ACOA.

Tour No	Customers	Distance [km]	Power consumption for each tour [kJ]	Total Power consumption [kW.sec]
			Group one	
1	0→11→31→0	8.9797	32.472→14.680→4.777	51.9290
2	0→12→16→0	8.9285	22.931→17.401→5.544	45.8760
3	0→14→15→0	8.2839	26.099→13.596→5.250	44.9450
4	0→30→4→0	14.5333	44.692→18.934→8.891	72.5170
5	0→3→17→0	5.5383	18.847→10.905→3.594	33.3460
6	0→29→32→0	13.1038	43.438→18.934→7.170	69.5420
7	0→13→6→0	12.8815	47.947→10.905→8.464	67.3160
8	0→10→23→0	20.5402	75.387→22.782→10.575	108.7440
			Group two 494.2150	
1	0→27→7→0	4.4360	18.014→9.003→2.814	29.8310
2	0→21→22→0	7.3528	25.377→10.905→4.910	41.1920
3	0→25→28→0	13.1809	31.119→20.970→8.803	60.8920
4	0→2→24→0	9.9274	32.375→18.189→5.040	55.6040
5	0→1→19→0	8.8538	26.601→14.680→5.611	46.8920
6	0→18→8→0	13.3275	32.410→24.950→7.617	64.9770
7	0→9→26→0	20.1738	72.299→14.680→12.976	99.9550
8	0→20→5→0	18.9001	65.754→14.680→12.386	92.8200

For the truck we solve the problem with 32 customers by NNA and ACOA to compare it with the delivery system, the tours are show in Figures (5.14) and (5.15) and the distance traveled, average time are show in Table (5.15).

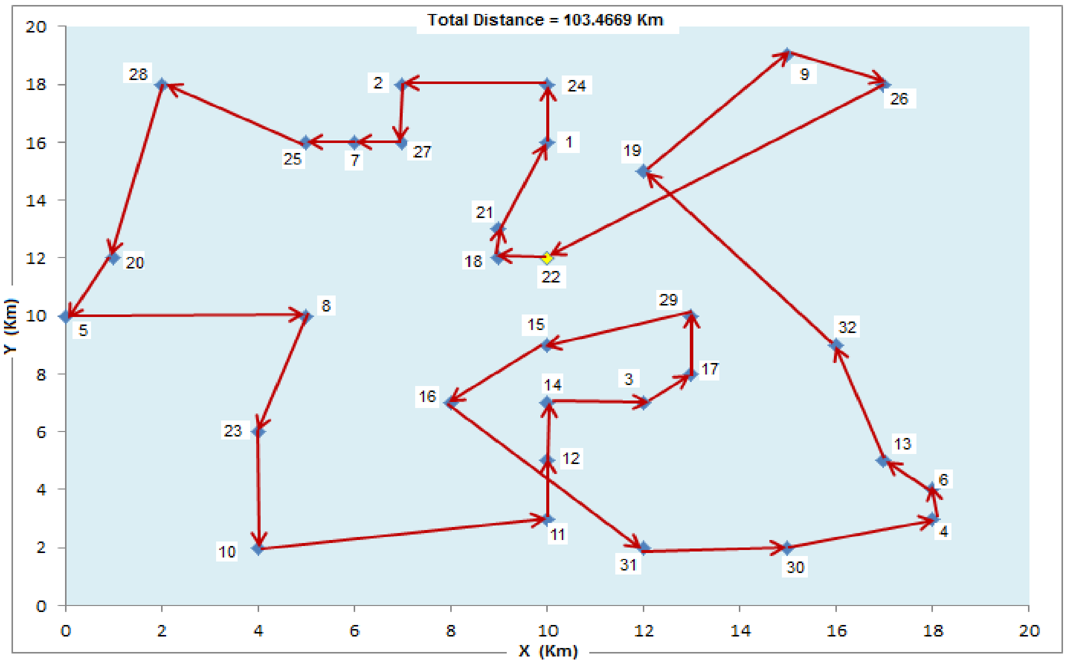

Figure 5.14: The tour and the distance traveled by truck Nearest Neighbors algorithm.

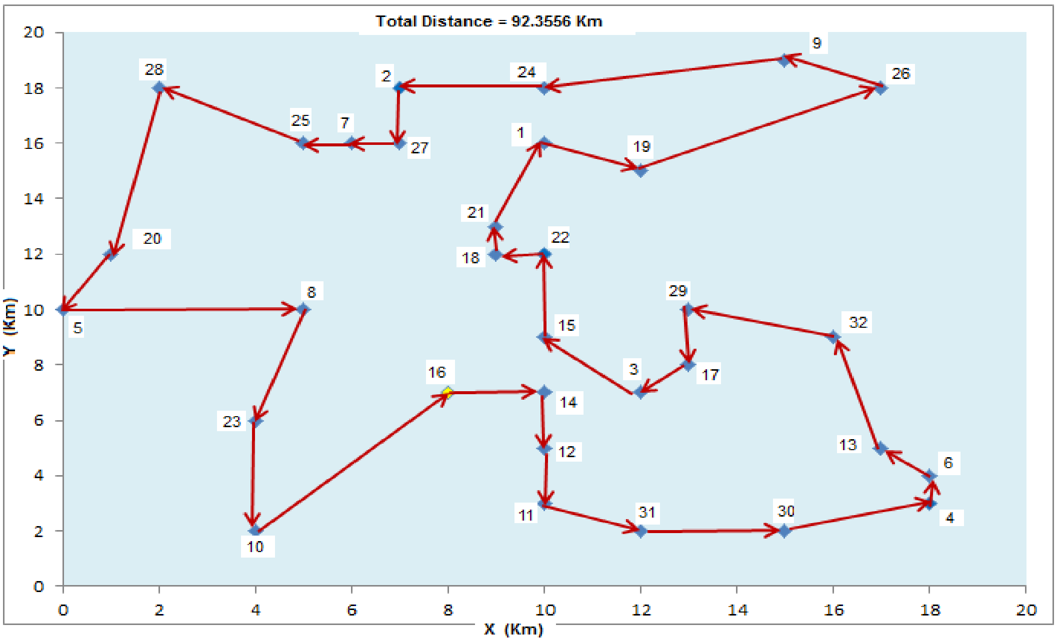

Figure 5.15: The tour and the distance traveled by truck ACOA.

Table 5.15: Tours, distance traveled and delivery time using a truck only for n=32 obtained by ACOA and NNA.

Algorithm	ACOA	NNA
Tours	16→14→12→11→31→30→4→6→13→32→29→17→3→15→22→18→21→1→19→26→9→24→2→27→7→25→28→20→5→8→23→10→16	22→18→21→1→24→2→27→7→25→28→20→5→8→23→10→11→12→14→3→17→29→15→16→31→30→4→6→13→32→19→9→26→22
Distance traveled for each customer [km]	2→ 4→ 6→ 8.2361→ 11.2361→ 14.3983→ 15.3983→ 16.8126→ 20.9357→24.0979→26.0979→27.5122→30.3406→33.3406→34.3406→35.3406→38.5029→40.7389→46.5699→48.8059→53.9050→56.9050→58.9050→59.9050→60.9050→64.5105→70.5933→72.8293→77.8293→81.9525→85.9525→92.3556	1→ 2→5.1623 →7.1623→10.1623 →12.1623 →13.1623 →14.1623 →17.7678 →23.8506→26.0867→31.0867→35.2098→39.2098→45.2925→47.2925→49.2925→51.2925→52.7067→54.7067→57.8690→60.6974→67.1006→70.1006→73.2628→74.2628→75.6771→79.8002→87.0113→92.0113→94.2473→103.4669
Delivery time for each customer [min]	4.9985→ 9.9970→ 14.9955→20.3479→ 26.8457→ 33.5867→ 37.0860→ 41.2062→49.3878→56.1288→61.1274→65.2476→71.4881→77.9859→81.4851→84.9844→91.7254→97.0779→107.8199→113.1723→122.8170→129.3148→134.3133→137.8125→141.3118→148.7174→159.8370→165.1894→174.6857→182.8672→190.8643→202.4641	3.4993→ 6.9985→13.7395→18.7380→25.2358→30.2343→33.7335→37.2328→44.6384→55.7580→61.1104→70.6067→78.7883→86.7853→97.9048→102.9033→107.9018→112.9003→117.0206→122.0191→128.7601→135.0007→146.6006→153.0983→159.8394→163.3386→167.4589→175.6404→188.4517→197.9479→203.3004→217.1228
Average Delivery time [min]	**94.9028**	**102.0096**

In the case of carrying a heavy payload with two drones, we consider M= 2 ,to carry load with 5 kg from location i to location j , the speed of two drones is constant and equal to $v = 16\ \frac{m}{sec}$, we use two drone with 4 rotors (quadrotor) the power consumed per rate weight for each drone is $\mu = 69.4\ W/kg$, and the power required to keep the drone frame in the air $\alpha\ = 21.44\ W$. The power consumption is calculated for variant distances traveled as shown in Table (5.16).

Table 5.16 Distance traveled and power consumption for two drones (Quadrotor)

Distance [km]	2.5	5	10	20
Power consumption [kJ]	115.140	230.275	460.550	921.100

On the other hand, refer to Figure (5.4) which represent Equation (3.6) and Equation (3.7) with 8 rotors. From the relation we can find that the power required to keep the drone frame in the air $\alpha\ = 14.91\ W$, and the power consumed rate per kilogram $\mu = 49.56\ W/kg$. The speed of the drone with 8 rotors is considered as the two drones with $= 16\ \frac{m}{sec}$. The power consumption versus the distances traveled is shown in table (5.17)

Table 5.17 Distance traveled and power consumption for one drone with 8 rotors.

Distance [km]	2.5	5	10	20
Power consumption [kJ]	41.048	82.097	164.19	328.39

5.4 Improving Power Consumption for the Drone

In the previous case, we consider the demand to be the same for all customers; this is known as regular delivery or daily delivery. In this section, power consumption is considered when demand is different for the customer; therefore, the nearest neighbor algorithm and the ACOA are improved to decrease power consumption. For instance, if the drone is at the base and there are two customers that need to be served with the same distance, it makes no difference to go to either of them if the demand is identical. Nevertheless, if the demand differs, there is less power consumption when the drone first serves the customer with greater demand. This idea comes from the fact that the drone should disburden itself of the heavier payload as soon as possible to carry fewer payloads throughout the tour. Equations (3.7) and (3.14) show that power consumption is affected by the payload and the distance traveled; therefore, a method should be followed to determine the minimum power consumption when the drone selects the next customer by comparing between two criteria. These two criteria are the heaviest demand and the nearest distance, and we can call this is the priority to select the next customer.

To illustrate this problem with an example, we assume the drone is at the base and there are three customers with variant demands requiring service. The demand for each customer and the distances between them and the base are shown in Table 5.16. If the drone selects the customer based on the smallest distance, Customer 3 will be selected. Similarly, if the drone selects the customer based on the heaviest demand, Customer 1 will be selected.

Table 5.18: Demand and distances between customers and base (sub-depot)

Customers	**1**	**2**	**3**
Distance	8	16	4
Demand	6	1	2

Now we will put Customers 1 and 3 in the same decision by dividing every customer's distance by the minimum distance (Customer 3), and by dividing the heaviest demand (Customer 1) by every customer's demands with each customer. We do this by applying Equation (5.2) to Table 5.18, the results of which are shown in Table 5.19. The results show that Customer 3 and Customer 1 are in the same decision and could be taken as being realized to the heaviest criteria and closest criteria.

$$d'_{ij} = \frac{d_{ij}}{\min d_{ij}} \quad , \qquad d'_j = \frac{\max d_j}{d_j} \tag{5.2}$$

where, d_{ij} is the distance between node i and node j, and d_j is the demand of customer j.

Table 5.19: Result after applying Equation (5.2).

Customers	1	2	3
Distance	2	4	1
Demand	1	6	3

For weighting between distance and demand, Equation (5.2) can be written as follows:

$$(1-\sigma)d'_{ij} + \sigma\, d'_j \qquad (5.3)$$

where, σ is a parameter of weighting between the distance (first part of the equation) and demand (second part of the equation). To minimize the objective function Equation (5.3), a suitable σ should be investigated which would lead to a least value for the customer for the drone selected.

We apply Equation (5.3) to Table 5.19 and vary σ from 0 to 1 to find the minimum value of this equation. This will give the priority of the customer who will be served first so as to minimize power consumption. The classical nearest neighbor algorithm can be updated by using Equation (5.3) to improve power consumption, as shown in Figure 5.16.

Step (1) start from the delivery truck location as a starting node.
Step (2) choose the next unvisited customer j which having the least value of Equation (5.3).
Step (3) visit the selected customer j.
Step (4) check the constraints condition, if they satisfied, return to the starting node.
Step (5) go to step (2) until all customers are completed.
Step (6) go back to the started node.

Figure 5.16: The minimize Power consumption for nearest neighbor algorithm (MPCNNA).

This algorithm is applied to the problem with $n = 32$ to obtain the distance traveled, tours and power consumption, and compared with the classical NNA when the demand is varying. The coordinates and demands of each customer are shown in Tables 5.20 and 5.21.

Table 5.20: The coordinate location and demand of group one.

Customers No.	x-coordinate /km	y-coordinate /km	Demand [kg]
0	11.88	5.563	-
3	12	7	0.1
4	18	3	0.5
6	18	4	1.75
10	4	2	2
11	10	3	0.2
12	10	5	1.8
13	17	5	0.15
14	10	7	2
15	10	9	0.75
16	8	7	0.25
17	13	8	1.9
23	4	6	0.2
29	13	10	2
30	15	2	0.75
31	12	2	2
32	16	9	1.5

Table 5.21: The coordinate location and demand of group two.

Customers No.	x-coordinate /km	y-coordinate /km	Demand [kg]
0	7.813	14.94	-
1	10	16	0.1
2	7	18	0.5
5	0	10	1.75
7	6	16	2

8	5	10	0.15
9	15	19	2
18	9	12	0.15
19	12	15	2
20	1	12	0.75
21	9	13	0.25
22	10	12	1.9
24	10	18	0.2
25	5	16	0.75
26	17	18	0.75
27	7	16	2
28	2	18	1.5

The results in Tables 5.22 and 5.23 show the tours, distances traveled and power consumed when the demand variants are obtained through MPCNNA.

Table 5.22: Tours, distances traveled, and power consumption in each customer location for n=32, served by drone using a nearest neighbor algorithm with variant demands.

Tour No	Customers	Distance [km]	Power consumption for each tour [kW.sec]	Total Power consumption [kW.sec]
		Group one		
1	0→3→17→0	5.5383	24.056→ 22.748→3.594	50.3980
2	0→12→11→0	7.1411	29.269→6.5342→4.259	40.0622
3	0→14→15→0	8.2839	44.134→13.596→5.250	62.9800
4	0→31→30→0	11.3010	60.039→18.189→6.346	84.5740
5	0→16→23→0	16.1528	16.781→11.221→10.575	38.5770
6	0→29→32→0	13.1038	91.465→32.344→7.190	130.9990
7	0→13→6→0	12.8815	58.550→21.203→8.464	88.2170
8	0→4→10→0	29.3188	92.536→150.18→11.588	254.3040
		103.7212	Group two	750.1112
1	0→27→7→0	4.4360	42.910→ 19.629→2.814	65.3530
2	0→21→18→0	6.4449	9.9456→3.9016→4.249	18.0962
3	0→1→24→0	8.1915	8.9547→6.5342→5.040	20.5289
4	0→25→2→0	9.0007	48.322→13.293→4.243	65.8580
5	0→22→19→0	11.4572	84.421→45.724→5.611	135.7560
6	0→8→20→0	17.5772	36.285→24.950→9.943	71.1780
7	0→28→5→0	24.0592	116.23→82.217→12.387	210.8340
8	0→9→26→0	20.1738	114.27→14.680→12.976	141.9260

Table 5.23: The power consumption obtain by ACOA for 32 customers with variant demands and without improved.

Tour No	Customers	Distance [km]	Power consumption for each tour [kW.sec]	Total Power consumption [kW.sec]
			Group one	
1	0→11→31→0	8.9797	45.038→32.009→4.777	81.8240
2	0→12→16→0	8.9285	52.086→ 8.3630→5.544	65.9930
3	0→14→15→0	8.2839	44.134→13.596→5.250	62.9800
4	0→30→4→0	14.5333	38.515→14.464→8.891	61.8700
5	0→3→17→0	5.5383	24.056→22.748→3.594	50.3980
6	0→29→32→0	13.1038	91.465→32.344→7.170	130.9790
7	0→13→6→0	12.8815	58.550→21.203→8.464	88.2170
8	0→10→23→0	20.5402	104.56→10.949→10.575	126.0840
			Group two	
1	0→27→7→0	4.4360	42.910→19.629→2.814	65.3530
2	0→21→22→0	7.3528	34.496→ 22.748→4.910	62.1540
3	0→25→28→0	13.1809	44.021→35.822→8.803	88.6460
4	0→2→24→0	9.9274	35.425→32.009→5.040	72.4740
5	0→1→19→0	8.8538	8.9547→7.0553→5.611	21.6210
6	0→18→8→0	13.3275	11.806→11.991→7.617	31.4140
7	0→9→26→0	20.1738	114.27→14.680→12.976	141.9260
8	0→20→5→0	18.9001	102.10→28.543→12.386	143.0290

Table 5.24: The tours, distance traveled, and power consumption in each customer location for n=32, served by drone using a IPCNNA with variant demands.

Tour No	Customers	Distance [km]	Power consumption for each tour [kW.sec]	Total Power consumption [kW.sec]
			Group one	
1	0→12→14→0	6.3288	52.086→29.644→3.171	84.9010
2	0→17→3→0	5.5383	36.475→4.2113→1.932	42.6183
3	0→31→30→0	11.3010	60.039→18.189→6.346	84.5740
4	0→29→15→0	11.6560	73.455→18.934→5.250	97.6390
5	0→32→13→0	14.6394	53.746→10.119→6.902	70.7670
6	0→6→4→0	13.9514	80.764→6.8771→8.891	96.5321
7	0→10→16→0	19.1888	106.64→17.851→5.544	130.0350
8	0→11→23→0	17.7789	12.726→16.928→10.575	40.2290
		100.3826	Group two	647.2954
1	0→27→2→0	6.5020	27.972→10.386→4.243	42.6010
2	0→7→25→0	6.1062	40.602→9.0025→4.028	53.6325
3	0→22→18→0	7.8348	47.315→3.9016→4.249	55.4656
4	0→19→3→0	8.8538	53.784→5.6690→3.257	62.7100
5	0→28→8→0	20.7980	65.608→20.980→7.618	94.2060
6	0→9→26→0	20.1738	114.27→ 14.680→12.976	141.9260
7	0→5→20→0	18.9001	124.32→14.680→10.136	149.1360
8	0→21→24→0	11.1345	10.647→13.375→5.040	29.0620

Figure (5.17) shows the tours and total distance traveld for each group when applying IPCNNA. The ACOA could improve by applying Equation (5.3) to construct the new distance matrix d_{ij} for the two groups and applying the algorithm based on this distance matrix with a variant of the value of σ. The resulting power consumption, distance traveled and tours are shown in Table 5.24, while Table 5.25 shows the tours, distances traveled and the power consumption for 32 customers without improvement of ACOA.

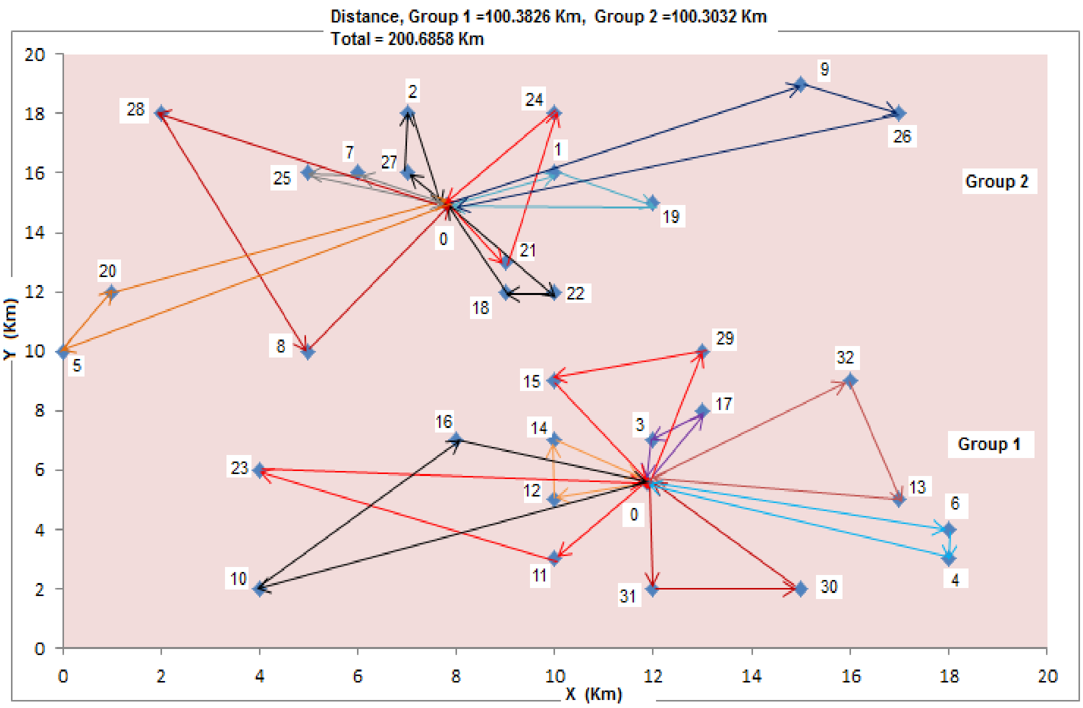

Figure 5.17: Tours and distances traveled obtaining by IPCNNA.

Table 5.25 : Tours, distances traveled, and power consumption in each customer location for n=32, served by drone using imporved IPCACOA to minimize the power consumption.

Tour No	Customers	Distance [km]	Power consumption for each tour [kW.sec]	Total Power consumption [kW.sec]
		Group one		
1	0→6→4→0	5.8083	34.756→5.2503→1.932	41.9383
2	0→32→29→0	11.3010	60.039→18.189→6.346	84.5740

3	0→14→17→0	14.6394	53.746→10.119→6.902	70.7670
4	0→16→15→0	11.9186	38.845→14.689→5.544	59.0780
5	0→31→30→0	13.9514	80.764→6.8771→8.891	96.5321
6	0→10→23→0	22.3171	60.247→23.861→10.575	94.6830
7	0→12→11→0	7.1411	29.269→6.5342→4.259	40.0622
8	0→13→3→0	21.7852	64.716→ 101.95→11.588	178.2540
		108.8621	Group two	665.8886
1	0→27→7→0	7.5024	37.284→11.214→4.243	52.7410
2	0→21→22→0	7.2158	25.483→11.069→3.048	39.6000
3	0→25→28→0	7.8348	47.315→3.9016→4.249	55.4656
4	0→2→24→0	12.1524	21.657→14.007→5.040	40.7040
5	0→1→19→0	18.9001	124.32→14.680→9.943	148.9430
6	0→18→8→0	20.1738	114.27→14.680→12.976	141.9260
7	0→9→26→0	20.7980	65.608→20.980→7.618	94.2060
8	0→20→5→0	8.8538	53.784→5.6690→3.257	62.7100

To get the minimum value of power consumed by the drone, the parameter σ in Equation (5.3) should be variating between 0 and 1. The results after running the two algorithms are show in Tables (5.26) and (5.27).

Table 5.26: The variant of σ with total power consumption and distance for two groups using IPCNNA.

Value of σ	**Group1**		**Group 2**	
	Power consumption [kW.sec]	Distance [km]	Power consumption [kW.sec]	Distance [km]
0	676.68	106.2832	743.20	117.2083
0.1	697.70	104.5896	747.36	117.0445
0.2	657.26	97.4771	805.64	117.7637
0.3	659.17	100.3825	722.42	108.3132
0.4	647.30	100.3825	682.91	113.8367
0.5	647.30	100.3825	655.76	100.3033
0.6	647.30	100.3825	655.76	100.3033
0.7	647.30	100.3825	655.76	100.3033
0.8	647.30	100.3825	655.76	100.3033
0.9	781.09	114.8030	756.27	125.6118
1	761.18	122.0873	802.58	126.0750

Table 5.27: The variant of σ with total power consumption and distance for two groups using IPCACOA.

Value of σ	**Group1**		**Group 2**	
	Power consumption [kW.sec]	Distance [km]	Power consumption [kW.sec]	Distance [km]
0	696.38	93.2712	706.69	98.5737
0.1	639.43	93.2712	729.97	98.5737
0.2	634.13	102.3609	702.14	98.2658
0.3	665.89	108.8621	652.39	98.2658
0.4	667.12	108.8621	657.79	103.4310
0.5	665.89	108.8621	688.15	103.4310
0.6	667.12	108.8621	657.79	103.4310
0.7	701.33	108.8621	657.79	103.4310
0.8	667.12	108.8621	726.75	119.3864
0.9	652.55	115.9163	716.04	130.6265
1	652.55	115.9163	715.01	130.6265

CHAPTER 6

DISCUSSION and CONCLUSION

6.1 Discussion

In this section, the result obtained from the previous section is analyzed and discussed. The results obtained in Tables (5.12) and (5.15) for the problem with 32 customers show that the average delivery time using the collaborative system is lass by 41.87% than using truck only. This value is obtained without taken into account the traffic status of roads when the truck moves to serve the customers, which will increased the average delivery time for the truck. In the same way, the Tables (5.9 - 5.12) show that if the number of drones is increased the average delivery time is decreased as an inverse exponential as shown in Figure (6.1). This gives that the collaborative system is faster than the delivery truck and more suitable to utilize in delivery packages. In addition, the problem of limited flight time is solved by using this system as show in Figure (5.14) the distance between some customer is faraway and cannot serve by the drone directly. As follows, the distance between customer 10 and 26 is 20.616 km, and this distance will take 42.95 min which cannot be completed by the drone in one journey due to out of flight time, in comparison if the distance is divide in two segments as show in Figure (5.13) the drone can reach each customer without any difficulty.

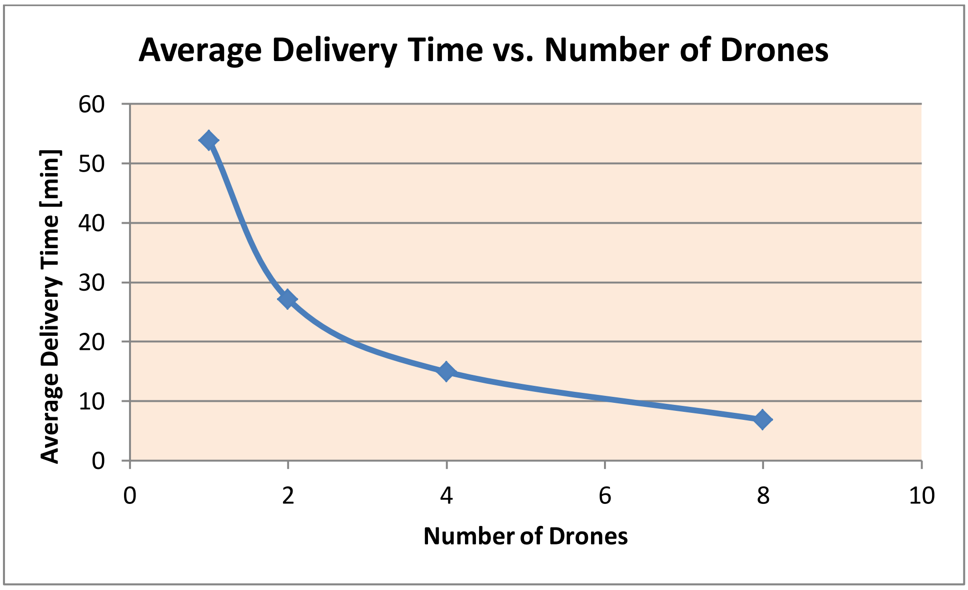

Figure 6.1: The relation between average delivery time and number of drones

Figure (6.2) show the relations between the power consumption and the time of delivery for the drone with variant payloads (1, 1.5 and 2 kg) of one tour with total distance 8.9797 km as show in Table (5.14). The Figure clarifies that the power consumption is at high level when moves from the base to the first customers and then, it reduced when moves to second customer depending on decreasing of the payload which reduced by half. Finally, the power is at low level when the drone returns to the base, for three different payloads, 1, 1.5 and 2 kg the power consumed when the drone moves from the second customer to the base are the same value (4.777 kJ). Because there is no any payload to carry, and this power represent only the power required to keep the drone in the air. This gives that serve the customer with high demand first will reduce the power consumption, but a comparison must be take into account between short distance and high demand to find the priority for served.

The three algorithms are tested for many instance generated randomly in a certain area, first, the K means algorithm is tested for difference instances and difference areas and the result is shown in Table (6.1). The result shows that the run time is acceptable pending for big instance and wide area.

Table 6.1 :The result obtain by runing the K-means algorathm for difference instances.

Scenario		**No. of groups**	**Run time (sec)**
Area (Km^2)	No. of customers		
5	50	2	9.65
10	50	2	23.81
5	100	3	51.61
10	100	3	73.61
10	200	4	151.25
20	200	4	126.35
20	500	10	423
40	500	10	339.64

The NNA and ACOA are tested for different size of customer locations as shown in Table (6.2). The result shows that ACOA is better than NNA with accepted run time and gap between them is 11.78% at customer size equal to 25 and 0.78% at customer size equal to 100. That is means the gap is decreases as the size is increase. On the other hand, the ACOA is a stochastic algorithm and need to run may times and take the average for the result.

Table 6.2: The result of run time and cost obtain by NNA and ACOA for different size of customers.

No. of customers	**ACOA**		**NNA**
	Run time (sec)	Cost	Cost
25	483.73	51.53	57.6
50	227.75	103.6	107.25
75	853.5	161.63	166.42
100	1026.62	203.82	205.41

In addition, the ACOA is tested to solve variant size of problem as shown in Table (6.3) and the result shows the effect of ants on the runtime and the solution, as we increase the number of ants the runtime is increasing and a small change occurs in the solution. In small size we can select a few ants to solve the problem with short runtime, on the other hand, we can select the number of ants equal to the number of customers for large size to get an acceptable runtime.

Table 6.3: The effect of ants on the runtime and the solution

NO. of customers	NO. of ants	Iteration	Runtime (min)	Cost
10	15	100	0.5003	212.3215
	10		0.3632	212.3215
	5		0.2156	212.3215
30	45	100	3.9561	494.345
	30		2.7015	491.8693
	15		1.394	496.0931
60	90	100	17.5233	900.647
	60		11.1277	904.3654
	30		5.5659	905.627

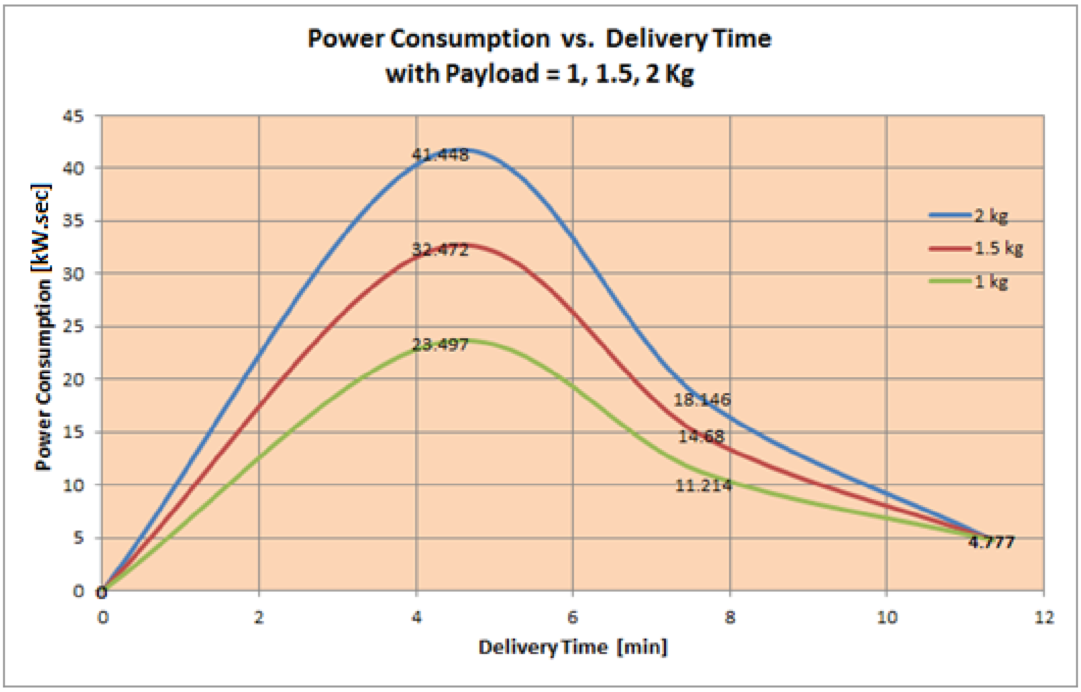

Figure 6.2: Power consumption vs. delivery time for drone with variant payload (1, 1.5 and 2 kg).

6.1.1 The Affected on Power Consumption of Carrying Payload with two Drones

The power consumption facing variant distances for two cases Tables (5.16) and (5.17), two drones with 4 rotors and one drone with 8 rotors are shown in Figure (6.3). The result clarify that the power consumption is lass when we use one drone with 8 rotors than use two drones. This come from increasing the number of rotors in Equation (3.6), and this will increases the effective of disc area.

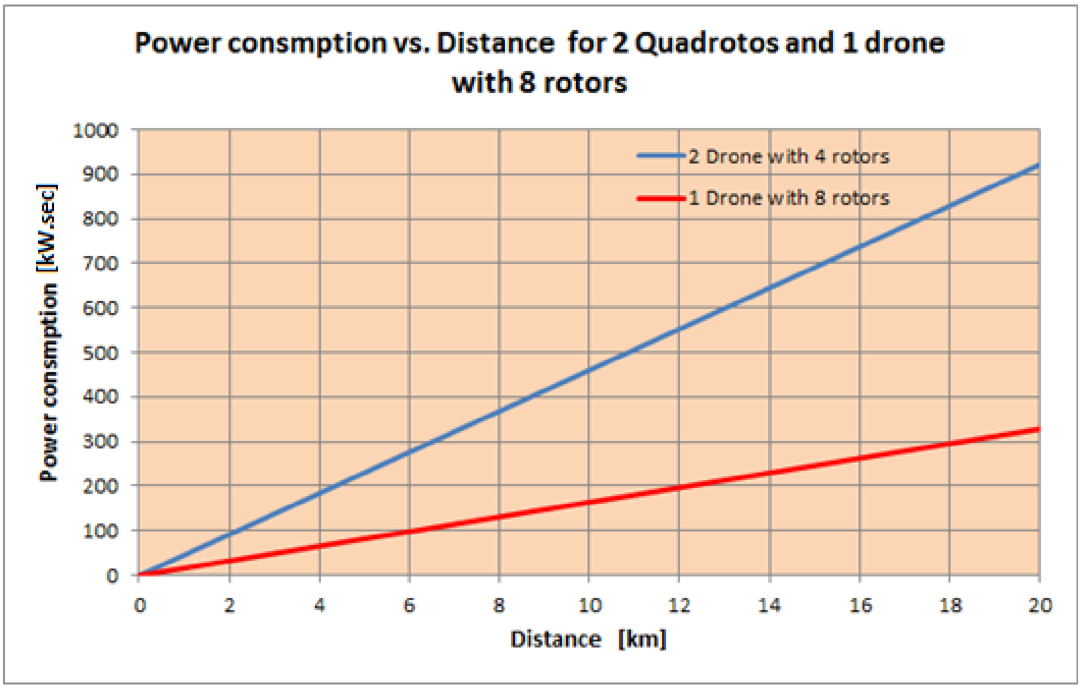

Figure 6.3 Power consumption vs. distance for 1 drone with 8 rotors, red line and 2 Quadrotors blue line.

6.1.2 The Impact of Variant Demand on Power Consumption

The payload of the drone affects the power consumed, as shown in Equation (3.7). Customer demands differ from one another in irregular delivery, which impacts power consumption. Therefore, a comparison should be made between distance and demand for each customer to minimize the power to be consumed when the drone moves between the sub-depot and customers. As shown in Table 5.22, the total power consumed to serve 32 customers with varying demands is 1479.6413 kJ. This value was acquired by the NNA. Similarly, the total power consumed to serve the same number of customers acquired by IPCNNA is 1303.06 kJ, as shown in Table 5.24. This means that the power consumed is reduced by 11.93% when using minimized power consumption for the nearest neighbor algorithm (IPCNNA). The total distance traveled is 200.6858 km when using IPCNNA compared to 205.0617 km using NNA, which was reduced by 2.134%. It might be concluded from this that the IPCNNA is more efficient and more effective when we minimize power consumption. On the other hand, an investigation into parameter σ in Equation (5.3) should be performed to determine the proper value of the power consumption minimization, as shown in Figure 6.4. The minimum power may be obtained at σ equaling 0.4 to 0.8 for Group 1 (blue line), and equaling 0.5 to 0.8 for Group 2 (red line).

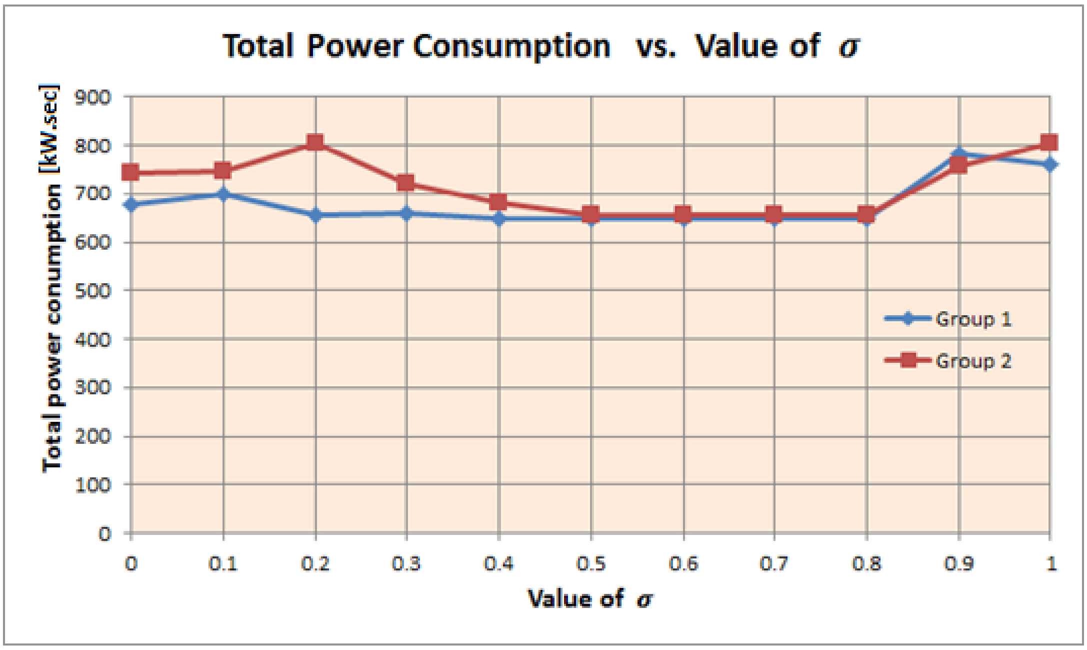

Figure 6.4: The total power consumption vs. the value of σ (blue line for group 1, and red line for group 2) obtained by IPCNNA.

In addition, the distance is affected by the value of parameter σ, thus, the minimum distance is acquire at σ equaling 0.3 to 0.8 for Group 1, as shown in Figure 6.5 with the blue line, and equaling 0.5 to 0.8 for Group 2 with the red line.

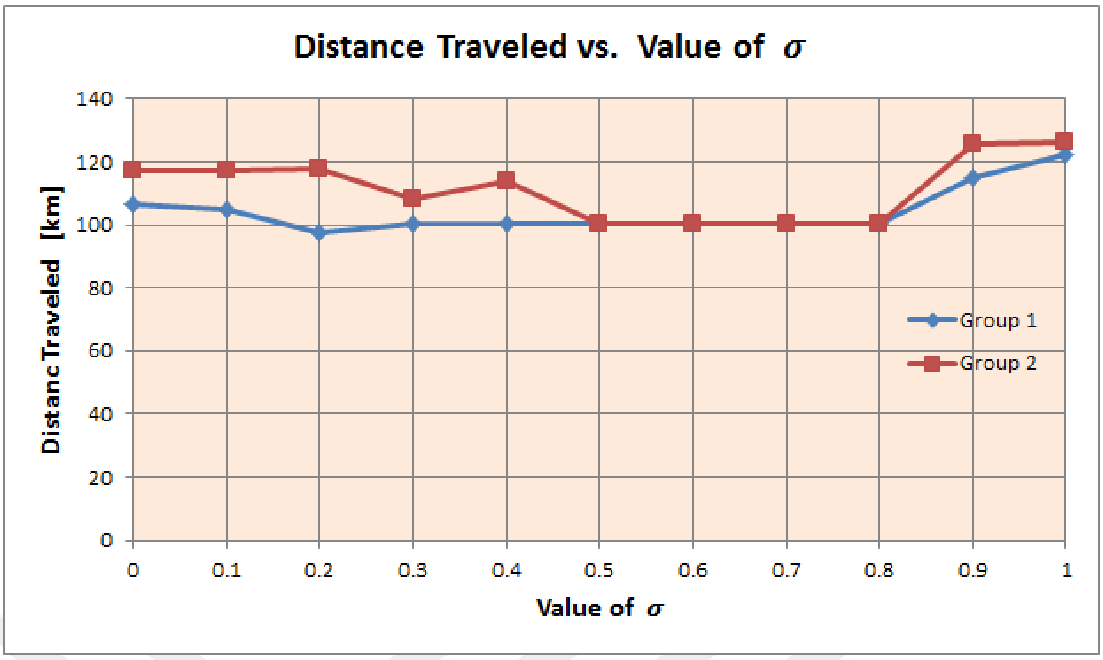

Figure 6.5: The distance traveled vs. the value of σ (blue line for group 1, and red line for group 2) obtained by IPCNNA.

Results from Figures 6.4 and 6.5 show that the power consumption and distance traveled are minimize at some value of parameter σ and this is difference from case to case depends on the distance between customers and weight of customer demand, as shown for Group 1 and Group 2.

Another improvement of power consumption occurs in the ACOA. The results in Table 5.23 show that the power consumption for Group 1 with different demand weights is 634.13 kJ and for Group 2 it is 644.95 kJ, compared with 667.92 kJ for Group 1 and 651.98 kJ obtained with ACOA without improvement, as shown in Table 5.25. This clarifies that the power decreases by 5.1% for Group 1, and for Group 2, it decreases by 1.08%. This occurs due to the ACO being a stochastic algorithm, which implies that the ACO could not improve the power consumption all the time; however, it depended on the difference in the demand weight and the distance between customers. Moreover, the distance traveled increased by 10.32% for Group 1 and 7.57% for Group 2 when using IPCACOA compared with using ACOA without improvement. The variant of parameter σ in Equation (5.3) for the power consumption and distance traveled are shown in Figures 6.6 and 6.7.

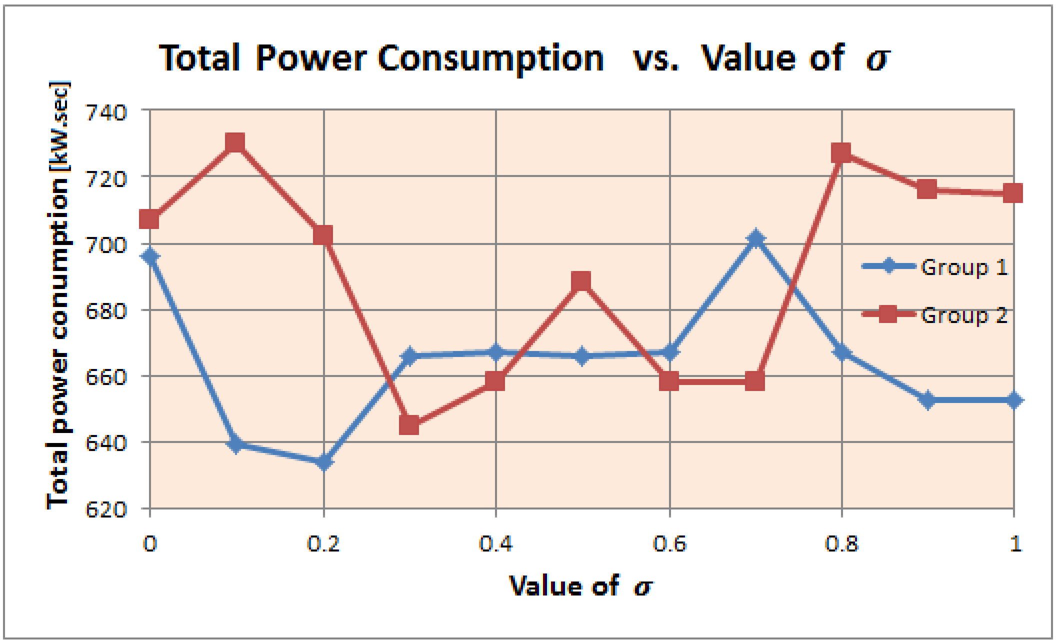

Figure 6.6: The power consumption vs. the value of σ (blue line for group 1, and red line for group 2) obtaining by IPCACOA.

As we have mentioned above, the ACOA is a stochastic algorithm that needs several runs for each testing instance to evaluate the stability of the solution. Figure 6.6 shows that the power consumption changes up and down for the two groups as the value of parameter σ increase from 0 to 1. This comes from the order of each tour that may change in each iteration, which will influence the power consumption. The figure also shows that the minimum power consumption obtained at σ equaling 0.2 for Group 1 and σ equaling 0.3 for Group 2. On the other hand, Figure 6.7 shows the relation between the distance traveled and the value of parameter σ as it increases. The distance will have a small increase and in some value of σ it remains stable, as shown in the figure. When σ is between 0.3 and 0.7, the distance is 108.8621 km for Group 1 and 103.4310 km for Group 2.

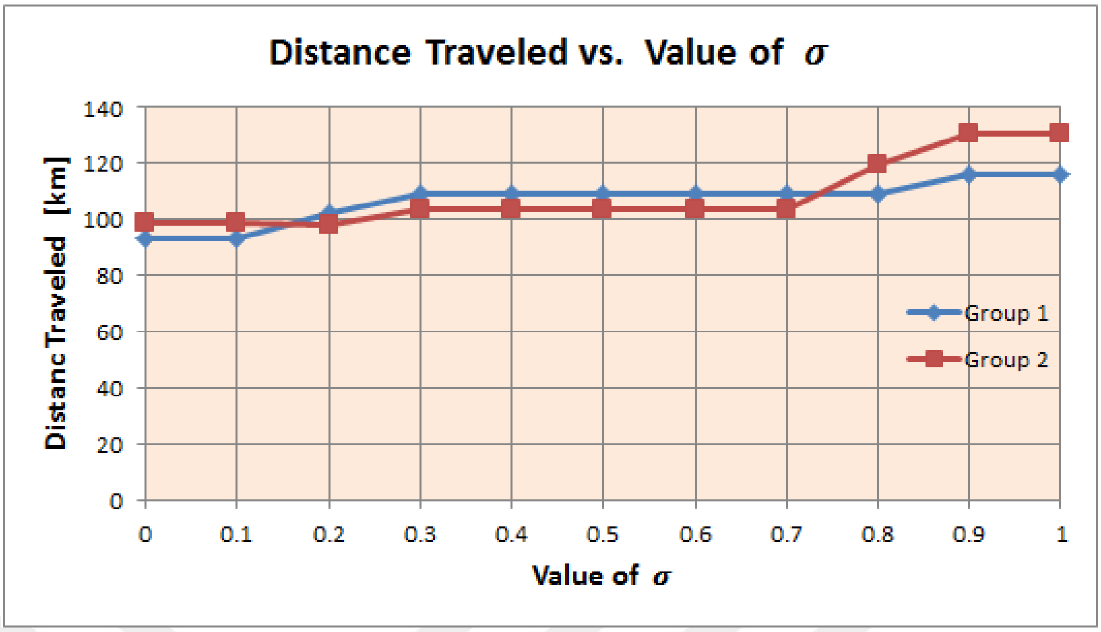

Figure 6.7: The distance traveled vs. the value of σ (blue line for group 1, and red line for group 2) obtaining by IPCACOA.

6.2 Conclusion

In this thesis, a collaborative delivery system with a UAV and a delivery truck is proposed to solve the problem of limited flight times when UAVs are used to deliver packages. A mathematical formulation in two stages is presented. In the first stage, the drone power consumption model was driven by using the theories of aerodynamics of the rotor-wing aircraft. This model is used to obtain the relation between the payload weight and the power consumed by the drone. From this relation, we found that the power consumption rate per kilogram was $= 69.4\, watts/kg$, and the power required to keep the drone frame in the air was $\alpha = 21.44\, watts$. Second, the integer linear programming model is driven for the routing problem with an objective function to find the best location for the delivery truck, and two objective functions to minimize the delivery time, distance traveled and power consumption for the drone. Then, the K-means algorithm is used to divide the customers into groups and find the best location for the delivery trucks. Moreover, the routing problem for the UAV is solved for each group with an ant colony optimization algorithm and nearest neighbor algorithm. The model was simulated in MATLAB and the results show that the average delivery time was reduced by 58% when using the combined system instead of the truck only for a problem with 32 customers. On the other hand, increasing the number of UAVs to serve one group will decrease the average delivery time as an inverse exponential. The results also show that the problem of limited flight time is solved by using this system when the customers partitioned into groups and each group has a middle location for the delivery truck.

The results also show that the power consumption model derived in the first part of the mathematical formulation is more realistic when taking into account the payload of the drone. This directly impacts the power consumption and therefore the flight time of the drone. Moreover, the power consumption can be minimized by serving high-demand customers first and taking into account a comparison that must be made between the short distances and high demands to determine priority when demands are not equal.

This priority is made by using the criterion equation (Equation (5.3)) with weighting between the heaviest demand and the shortest distance, which applies in the NNA and ACOA to calculate a new distance matrix to improve the power consumed by the drone. The results show that using IPCNNA will reduce the power by 11.93% relative to the results obtained from NNA. Moreover, the distance traveled is reduced by 2.134% when using the same algorithm for the same instance of the problem with variant demands. This shows that the IPCNNA is more efficient if we minimize the power consumption and distance traveled when demands differ. On the other hand, using this criterion to improve ACOA will reduce the power by 3.1%, but the distance traveled will increase by 8.945%, which occurs due to the ACOA being a stochastics.

6.3 Future Research

It has been observed that the collaborative delivery system is a new study with the potential to solve the issues of limited flight time. Therefore, several types of research can be carried out in the future:

- using multiple delivery trucks combined with UAVs to allow UAVs to launch from one truck and return to another truck, which may increase the system's working time.

- Considering another parameter that effect the power consumption such as, weather condition.
- Adding time windows to the customers could be ensure that the packages are delivered within a specific time.

References

1. Waharte, S. and N. Trigoni. *Supporting search and rescue operations with UAVs*. in *Emerging Security Technologies (EST), 2010 International Conference on*. 2010. IEEE.

2. Naidoo, Y., R. Stopforth, and G. Bright. *Development of an UAV for search & rescue applications*. in *AFRICON, 2011*. 2011. IEEE.

3. Villa, T.F., et al., *An overview of small unmanned aerial vehicles for air quality measurements: Present applications and future prospectives.* Sensors, 2016. **16**(7): p. 1072.

4. Liu, X.-f., et al., *An optimization model of UAV route planning for road segment surveillance.* Journal of Central South University, 2014. **21**(6): p. 2501-2510.

5. Adams, S.M. and C.J. Friedland. *A survey of unmanned aerial vehicle (UAV) usage for imagery collection in disaster research and management*. in *9th International Workshop on Remote Sensing for Disaster Response*. 2011.

6. Bendea, H., et al., *Low cost UAV for post-disaster assessment.* The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2008. **37**(B8): p. 1373-1379.

7. Habib, M.K., W.G.A. Abdelaal, and M.S. Saad, *Dynamic modeling and control of a Quadrotor using linear and nonlinear approaches*. 2014.

8. Nonami, K., *Prospect and recent research & development for civil use autonomous unmanned aircraft as UAV and MAV.* Journal of system Design and Dynamics, 2007. **1**(2): p. 120-128.

9. Gupta, S.G., M.M. Ghonge, and P. Jawandhiya, *Review of unmanned aircraft system (UAS).* International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 2013. **2**(4): p. pp: 1646-1658.

10. Sonmezocak, E. and S. Kurt, *Optimum route planning and scheduling for unmanned aerial vehicles*. 2008, NAVAL POSTGRADUATE SCHOOL MONTEREY CA.

11. Dean, D.B., et al., *Applications of UAV-capable systems for post-disaster damage assessment of transportation infrastructure.* 2013.

12. Imam, A. and R. Bicker, *State of the Art in Rotorcraft UAVs Research.* IJESIT, 2014. **3**: p. 221-233.

13. Azzam, A. and X. Wang. *Quad rotor arial robot dynamic modeling and configuration stabilization*. in *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*. 2010. IEEE.

14. Lin, C., et al., *Survey of green vehicle routing problem: past and future trends.* Expert Systems with Applications, 2014. **41**(4): p. 1118-1138.

15. Lin, J., W. Zhou, and O. Wolfson, *Electric vehicle routing problem.* Transportation Research Procedia, 2016. **12**: p. 508-521.

16. GmbH, D.I. *Successful Trial Integration of DHL Parcelcopter into Logistics Chain*. 2016, September, 05; Available from: http://www.dhl.com/en/press/releases/releases_2016/all/parcel_ecommerce/successful_trial_integration_dhl_parcelcopter_logistics_chain.html.

17. Inc., A.c. *Amazon Prime Air*. 2016, December, 07; Available from: www.amazon.com/primeair.

18. Guardian, T. *First successful drone delivery made in the US*. 2015; Available from: https://www.theguardian.com/technology/2015/jul/20/first-successful-drone-delivery-us.

19. Mashable. *Singapore Post has started testing drone deliveries*. 2015; Available from: https://mashable.com/2015/10/08/singapore-post-drone/#HfCzaHxhusqO.

20. D'Andrea, R., *Guest Editorial Can Drones Deliver?* IEEE Transactions on Automation Science and Engineering, 2014. **11**(3): p. 647-648.

21. Faust, A., et al., *Automated aerial suspended cargo delivery through reinforcement learning.* Artificial Intelligence, 2014.

22. Larsen, J., *Parallelization of the vehicle routing problem with time windows*. 1999: Institute of Mathematical Modelling, Technical University of Denmark Lyngby, Denmark.

23. Taiwo, O.S., et al., *IMPLEMENTATION OF HEURISTICS FOR SOLVING TRAVELLING SALESMAN PROBLEM USING NEAREST NEIGHBOUR ANDNEAREST INSERTIONAPPROACHES.* 2013.

24. Geetha, S., P. Vanathi, and G. Poonthalir, *Metaheuristic approach for the multi-depot vehicle routing problem.* Applied Artificial Intelligence, 2012. **26**(9): p. 878-901.

25. Bianchi, L., et al., *A survey on metaheuristics for stochastic combinatorial optimization.* Natural Computing, 2009. **8**(2): p. 239-287.

26. Arora, R.K., *Optimization: algorithms and applications*. 2015: CRC Press.

27. Wilson, R.J., *Introduction to graph theory*. 1979: Pearson Education India.

28. Dehghan Nasiri, S., *Vehicle routing on real road networks*. 2014, Lancaster University.

29. Driessens, S. and P. Pounds, *The triangular quadrotor: a more efficient quadrotor configuration.* IEEE Transactions on Robotics, 2015. **31**(6): p. 1517-1526.

30. Stirling, T., S. Wischmann, and D. Floreano, *Energy-efficient indoor search by swarms of simulated flying robots without global information.* Swarm Intelligence, 2010. **4**(2): p. 117-143.

31. Roberts, J.F., J.-C. Zufferey, and D. Floreano. *Energy management for indoor hovering robots*. in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. 2008. IEEE.

32. Ure, N.K., et al., *An automated battery management system to enable persistent missions with multiple aerial vehicles.* IEEE/ASME Transactions on Mechatronics, 2015. **20**(1): p. 275-286.

33. Lee, D., J. Zhou, and W.T. Lin. *Autonomous battery swapping system for quadcopter*. in *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*. 2015. IEEE.

34. Michini, B., et al., *Automated battery swap and recharge to enable persistent UAV missions*, in *Infotech@ Aerospace 2011*. 2011. p. 1405.

35. Chang, T. and H. Yu, *Improving electric powered UAVs' endurance by incorporating battery dumping concept.* Procedia Engineering, 2015. **99**: p. 168-179.

36. Abdilla, A., A. Richards, and S. Burrow. *Endurance optimisation of battery-powered rotorcraft*. in *Conference Towards Autonomous Robotic Systems*. 2015. Springer.

37. L. Cantelli, M.M., C.D. Melita, G. Muscato, *UAV/UGV cooperation for surveying operations in humanitarian demining*. IEEE, 2013.

38. L. Cantelli, M.L.P., M. Mangiameli, C.D. Melita, G. Muscato, *Autonomous Cooperation Between UAV and UGV to Improve Navigation and Environmental Monitoring in Rough Environments.*

39. Melin, J., et al., *Cooperative Sensing and Path Planning in a Multi-vehicle Environment*⁎. IFAC-PapersOnLine, 2015. **48**(9): p. 198-203.

40. WIRED.COM. *The Next Big Thing You Missed: Amazon's Delivery Drones Could Work—They Just Need Trucks.* October, 2014; Available from: https://www.wired.com/2014/06/the-next-big-thing-you-missed-delivery-drones-launched-from-trucks-are-the-future-of-shipping/.

41. ATLAS, N. *UPS tests truck-launchable delivery drones*. February 21, 2017 Available from: http://newatlas.com/ups-truck-launch-delivery-drone-test/48046/.

42. Niels Agatz, P.B., Marie Schmidt, *Optimization Approaches for the Traveling Salesman Problem with Drone.* August 1, 2015.

43. Murray, C.C. and A.G. Chu, *The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery.* Transportation Research Part C: Emerging Technologies, 2015. **54**: p. 86-109.

44. Gatteschi, V., et al., *New Frontiers of Delivery Services Using Drones: A Prototype System Exploiting a Quadcopter for Autonomous Drug Shipments.* 2015: p. 920-927.

45. Dantzig, G.B. and J.H. Ramser, *The truck dispatching problem.* Management science, 1959. **6**(1): p. 80-91.

46. Anbuudayasankar, S., K. Ganesh, and S. Mohapatra, *Models for practical routing problems in logistics*. 2016: Springer.

47. Paolo Toth, D.V., *THE VEHICLE ROUTING PROBLEM*. 2002: Society for Industrial and Applied Mathematics, 3600 University City Science Center, Philadelphia, PA 19104-2688.

48. Bruce Golden, S.R., Edward Wasil, *THE VEHICLE ROUTING PROBLEM: LATEST ADVANCES AND NEW CHALLENGES*, ed. S.V. Ramesh Sharda. 2008, Spring Street, New York, NY 10013, USA: Springer Science+Business Media, LLC.

49. LIONG CHOONG YEUN, W.R.I., KHAIRUDDIN OMAR & MOURAD ZIROUR, *VEHICLE ROUTING PROBLEM: MODELS AND SOLUTIONS.* Journal of Quality Measurement and Analysis, 2008. **4**(1).

50. Bektas, T., *The multiple traveling salesman problem: an overview of formulations and solution procedures.* Omega, 2006. **34**(3): p. 209-219.

51. Xiao, Y., et al., *Development of a fuel consumption optimization model for the capacitated vehicle routing problem.* Computers & Operations Research, 2012. **39**(7): p. 1419-1431.

52. Rao, W. and C. Jin. *A model of vehicle routing problem minimizing energy consumption in urban environment*. in *2012 Asian conference of management science and Applications. Proceedings*. 2012.

53. Erdoğan, S. and E. Miller-Hooks, *A green vehicle routing problem.* Transportation Research Part E: Logistics and Transportation Review, 2012. **48**(1): p. 100-114.

54. Kucukoglu, I., et al., *Green capacitated vehicle routing problem fuel consumption optimization model.* Computational Engineering Research, 2013. **3**: p. 16-23.

55. Kallehauge, B., et al., *Vehicle routing problem with time windows*, in *Column generation*. 2005, Springer. p. 67-98.

56. Desaulniers, G. and G. décisions, *The VRP with pickup and delivery*. 2000: Montréal: Groupe d'études et de recherche en analyse des décisions.

57. Villegas, J.G., et al., *A matheuristic for the truck and trailer routing problem.* European Journal of Operational Research, 2013. **230**(2): p. 231-244.

58. Derigs, U., M. Pullmann, and U. Vogel, *Truck and trailer routing—problems, heuristics and computational experience.* Computers & Operations Research, 2013. **40**(2): p. 536-546.

59. Dorling, K., et al., *Vehicle Routing Problems for Drone Delivery.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017. **47**(1): p. 70-85.

60. Gordon, L.J., *Principles of helicopter aerodynamics.* Cambridge Aerodynamic Series, 2000.

61. Hoffmann, G., et al. *Quadrotor helicopter flight dynamics and control: Theory and experiment*. in *AIAA Guidance, Navigation and Control Conference and Exhibit*. 2007.

62. DJI. *Phantom 3*. Available from: https://www.dji.com/phantom-3-pro/info#specs.

63. Pounds, P., R. Mahony, and P. Corke, *Modelling and control of a large quadrotor robot.* Control Engineering Practice, 2010. **18**(7): p. 691-699.

64. Tan, P.-N., M. Steinbach, and V. Kumar, *Cluster analysis: basic concepts and algorithms.* Introduction to data mining, 2006. **8**: p. 487-568.

65. Sahraeian, R. and P. Kaveh. *Solving Capacitated P-Median Problem by Hybrid K-Means Clustering and Fixed Neighborhood Search algorithm*. in *Proceedings of the 2010 International Conference on Industrial Engineering and Operation Management*. 2010.

66. Geetha, S., G. Poonthalir, and P. Vanathi, *Improved k-means algorithm for capacitated clustering problem.* INFOCOMP, 2009. **8**(4): p. 52-59.

67. KomalChadha, N. *Review Of Ant Colony Optimization Algorithms OnVehicle Routing Problems And Introduction ToEstimation-Based ACO*. in *Proceedings of International Conference on Environment Science and Engineering (ICESE 2011)*. 2011.

68. Maniezzo, A. *Distributed optimization by ant colonies*. in *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*. 1992. Mit Press.

69. Mazzeo, S. and I. Loiseau, *An Ant Colony Algorithm for the Capacitated Vehicle Routing.* Electronic Notes in Discrete Mathematics, 2004. **18**: p. 181-186.

70. NajmeZehraNaqvi, H., KomalChadha, *Review Of Ant Colony Optimization Algorithms OnVehicle Routing Problems And Introduction ToEstimation-Based ACO*, in *International Conference on Environment Science and Engineering IPCBEE vol.8 (2011)*. 2011: Singapore.

71. Bell, J.E. and S.E. Griffis, *Swarm Intelligence: Application of the Ant Colony Optimization Algorithm to Logistics-Oriented Vehicle Routing Problems.* Journal of Business Logistics, 2010. **31**(2): p. 157-175.

72. Chen, R.-M., F.-R. Hsieh, and D.-S. Wu. *Heuristics based ant colony optimization for vehicle routing problem*. in *Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on*. 2012. IEEE.

73. Ha, Q.M., et al., *Heuristic methods for the Traveling Salesman Problem with Drone.* Comput. Sci, 2015.