RESOURCE-AWARE COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

by

Ali Naci Zabun

B.S., Mechanical Engineering, Boğaziçi University, 2019

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Mechanical Engineering
Boğaziçi University
2022

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest appreciation to my advisor, Assist. Prof. Sinan ÖNCÜ, for his support and guidance especially during the difficult pandemic.

I would also like to thank the thesis committee members, Assoc. Prof. Evren SAMUR and Assoc. Prof. Volkan SEZER, for their participation. I am glad to get their brilliant suggestions and comments on my thesis.

I also wish to express my gratitude to my managers at TAAC. I would like to thank the General Manager, Enis ATA, Engineering Director, Göker AKINCI and Systems Engineering Manager, M. Caner AKAY for their belief in completing my M.Sc. degree, support and the opportunities they provided.

My warm and heartfelt thanks go to my parents Rümeysa and Serdar ZABUN, my sibling Betül, and my brother Emirhan for their tremendous support and hope they had given to me.

Lastly, I would like to thank my beloved wife Yağmur for her unconditional support and understanding. I am so grateful for everything she has done for me throughout pursuing the M.Sc. degree. I will always be in her debt.

ABSTRACT

RESOURCE-AWARE COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

In this study, a problem related to cooperative delivery mission planning of an unmanned aerial vehicle (UAV) and an autonomous ground vehicle (AGV) is considered. The problem is a new sub-variant of the Traveling Salesman Problem with Drone (TSP-D), known as a variant of the Traveling Salesman Problem (TSP) in the literature, and is called the Flying Sidekick Traveling Salesman Problem (FSTSP). Challenging part of the problem is that the UAV is able to carry only one package per flight and performs only one deliver mission as well. In the study, UAV is considered as able to take-off from warehouse or any delivery points on the route of AGV and land on the AGV on any delivery point on the route of AGV or warehouse. UAV takes the package for the next delivery point while staying on the AGV. Main constraints are taken as payload capacity (regarding weight of the package) and battery capacity of the UAV. Differently from original FSTSP model in the literature, a weight-aware energy consumption model is applied for the UAV and UAV has a chargeable battery with limited capacity. Main objective of the study is to develop a computation-time efficient algorithm by dynamic programming (DP) method to find the exact solution of cooperative delivery mission planning for UAV and AGV. The applicability of dynamic programming method instead of enumeration method was analyzed by considering external factors and constraints and it was proved that DP method has time efficiency compared to Enumeration method for increased number of delivery points. On the other hand, obtained results showed that determination of the value of battery capacity and charging rate of the UAV take lead role for the cooperative delivery mission planning for UAV and AGV.

ÖZET

İHA VE İKA İÇİN KAYNAK FARKINDALIĞINA SAHİP İŞBİRLİKLİ TESLİMAT GÖREVİ PLANLAMASI

Bu çalışmada insansız hava aracı ve otonom kara aracının işbirlikli teslimat görevi planlamasına ilişkin bir problem ele alınmıştır. Problem, literatürde Gezgin Satıcı Probleminin (TSP) bir çeşidi olarak bilinen İHA'lı Gezgin Satıcı Probleminin (TSP-D) yeni bir alt varyantıdır ve Uçan Yardımcılı Gezgin Satıcı Problemi (FSTSP) olarak adlandırılmaktadır. Problemin zorlu kısmı, İHA'nın uçuş başına sadece bir paket taşıyabilmesi ve aynı zamanda sadece bir teslimat görevi gerçekleştirebilmesidir. Çalışmada İHA'nın İKA güzergahındaki herhangi bir ambar veya teslim noktasından kalkış yapabileceği ve İKA veya antrepo güzergahındaki herhangi bir teslim noktasında İKA'ya inebileceği kabul edilmiştir. İHA, İKA'da kalırken bir sonraki teslimat noktası için paketi alır. Ana kısıtlar, İHA'nın faydalı yük kapasitesi (paket ağırlığına göre) ve batarya kapasitesi olarak alınmıştır. Literatürdeki orijinal FSTSP modelinden farklı olarak İHA için ağırlık duyarlı bir enerji tüketim modeli uygulanmakta ve İHA sınırlı kapasiteli şarj edilebilir bir bataryaya sahiptir. Çalışmanın temel amacı, İHA ve AGV için işbirlikli teslimat görev planlamasının kesin çözümünü bulmak için dinamik programlama (DP) yöntemi ile hesaplama zamanı verimli bir algoritma geliştirmektir. Sayım yöntemi yerine dinamik programlama yönteminin uygulanabilirliği dış faktörler ve kısıtlar dikkate alınarak analiz edilmiş ve artan teslimat noktaları için Sayım yöntemine kıyasla DP yönteminin zaman verimliliğine sahip olduğu kanıtlanmıştır. Öte yandan elde edilen sonuçlar, İHA ve İKA için ortak teslimat görevi planlamasında İHA'nın pil kapasitesi ve şarj hızı değerinin belirlenmesinin başrol oynadığını göstermiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$< i, j, k >$	A tuple to represent delivery operation of UAV from departure point to delivery point and delivery point to arrival point; $i \in P^d$, $j \in P^U$, $k \in P^a$, $i \neq k$, $j \neq k$, $i \neq k$, $\tau^U_{i,j} + \tau^U_{j,k} \leq \zeta$
B	Peak magnetic flux
D	Distance matrix for AGV consist of $d_{i,j}$ where $i \in P^d,\ j \in P^a$ and $i \neq j$, $D \in \mathbb{R}^{n+1 \times n+1}$
D^U	Distance matrix for UAV consist of $d_{j,k}$ where $j \in P^U,\ k \in P^T$ and $j \neq k$, $D \in \mathbb{R}^{n^U \times n+1}$
$d_{i,j}$	Distance between point i and j where $i \in P^d,\ j \in P^a$ and $i \neq j$, $d_{i,j} \in \mathbb{R}^{n+1}$
DR	Cooperative delivery route for UAV and AGV
DR^T	Route of only AGV in cooperative delivery route
E_{pro}	Energy consumption of vehicle
f	Frequency
I	Current
I_m	Current of back EMF
k	Steinmetz coefficient
M	Torque
n	Number of Cities
n^T	Number of Cities delivered by AGV
n^U	Number of Cities delivered by UAV
P	Set of delivery points, $P = \{1, 2, .., n\} \in \mathbb{N}^n$
P^a	Set of all arrival points, $P^a = \{1, 2, .., n, n+1\} \in \mathbb{N}^{n+1}$ where point $n+1$ represents warehouse
P^d	Set of all departure points, $P^d = \{0, 1, 2, .., n\} \in \mathbb{N}^{n+1}$ where point 0 represents warehouse
P^T	Set of deliverable by AGV
P^U	Set of deliverable by UAV

P^W	Set of all points on the route including delivery points and warehouse, $P^W = \{0, 1, 2, .., n, n+1\} \in \mathbb{N}^{n+2}$ where point 0 and $n+1$ represents warehouse
P_C	Copper loss
P_I	Iron loss
$p_{i,j}$	Auxiliary binary decision variable ($\in \{0, 1\}$) that equals to 1 if and only if $i \in P^d$ is visited by AGV before $j \in P$ when $i \neq j$
P_{in}	Propulsion system power transmission
P_m	Back EMF loss
P_M	Mechanical power of propellers
PR	Possible routes for UAV according to AGV route
$q_{i,j}$	Auxiliary binary decision variable ($\in \{0, 1\}$) that equals to 1 if and only if $i \in P^d$ is visited by UAV before $j \in P$ when $i \neq j$, $< i, k, l > \notin U$ and $< m, n, j > \notin U$
R_m	Resistance of propulsion system circuit
t_j^T	Decision variable for the time at AGV is ready to move, $j \in P$
t_j^U	Decision variable for the time at UAV is ready to departure, $j \in P$
U	Set of all possible tuples for UAV, $\in \mathbb{N}^{n+1\times n\times n+1}$
u_i	Auxiliary integer variable that specifies the position of point $i \in P^a$ in AGV route, $\in \mathbb{N}^{n+2}$
V_m	Voltage of back EMF
v^U	Velocity of UAV for horizontal travel in air in m/s
v_L^U	Velocity of UAV for landing in m/s
v_T^U	Velocity of UAV for take-off in m/s
v^T	Velocity of AGV in m/s
W^T	Set of weights of deliverables delivered by AGV, $\in \mathbb{R}^+$
W^U	Set of weights of deliverables delivered by UAV, $\in \mathbb{R}^+$
W_{El}	Weight of eliminated deliverable
W_i	Weight of deliverable in kg, $i \in P^U$
W_U	Weight of UAV in kg

$x_{i,j}$	Binary decision variable ($\in \{0,1\}$) that equals to 1 if and only if AGV departs from $i \in P^d$ and arrives to $j \in P^a$ when $i \neq j$
X_{El}	X coordinate of eliminated deliverable
X_i	X coordinate of deliverable, $i \in P$
X_w	X coordinate of warehouse
X^T	Set of X coordinate of deliverables delivered by AGV, $\in \mathbb{R}^+$
X^U	Set of X coordinate of deliverables delivered by UAV, $\in \mathbb{R}^+$
$y_{i,j,k}$	Binary decision variable ($\in \{0,1\}$) that equals to 1 if and only if UAV departs from $i \in P^d$, delivery to $j \in P^U$ and arrives to $k \in P^a$ when $< i,j,k > \in U$
Y_{El}	Y coordinate of eliminated deliverable
Y_i	Y coordinate of deliverable, $i \in P$
Y_w	Y coordinate of warehouse
Y^T	Set of Y coordinate of deliverables delivered by AGV, $\in \mathbb{R}^+$
Y^U	Set of Y coordinate of deliverables delivered by UAV, $\in \mathbb{R}^+$

β_i	Current battery status of UAV on point i in min, $i \in P^W$
β_f	Final battery status of UAV in min at reunion point of UAV and AGV after all UAV deliverables are delivered by UAV
χ	A data structure consist of $(X,\ Y)$ where X and Y are coordinates of warehouse and deliverables
χ^U	A data structure consist of $(X^U,\ Y^U)$ where X^U and Y^U are coordinates of deliverables delivered by UAV
χ^T	A data structure consist of $(X^T,\ Y^T)$ where X^T and Y^T are coordinates of deliverables delivered by AGV
χ_{El}	Eliminated data structure consisting of (X_{El}, Y_{El}) where X_{El} and Y_{El} are coordinates of eliminated deliverable
$\mathit{\Delta} L^T$	Delivery route length of AGV
$\mathit{\Delta} P$	Delivery time of the package from vehicle to customer on delivery point in s, $\in \mathbb{R}^+$
$\mathit{\Delta} T^T$	Delivery route time of AGV
ϵ	Big enough number
η^T	Specifications of AGV including $v^T,\ \mathit{\Delta} P$

η^U	Specifications of UAV including v^U, v_L^U, v_T^U, W_U, $\varDelta P$, κ, λ, ρ, $\varUpsilon$, ζ
κ	Constant charging rate of UAV per min
λ	Payload effect parameter for UAV
ω	Rotational Speed
$\varPsi$	A data structure consist of $(\chi(X, Y),\ W)$ where χ represents X and Y are coordinates of warehouse and deliverables, and W is weight value of deliverables
$\varPsi^T$	A data structure consisting of $(\chi^T(X^T, Y^T),\ W^T)$
$\varPsi^U$	A data structure consisting of $(\chi^U(X^U, Y^U),\ W^U)$
$\varPsi_{El}$	Eliminated data structure consisting of $(\chi^{El}(X^{El}, Y^{El}),\ W^{El})$
ρ	Payload limit of UAV in kg, $\in \mathbb{R}^+$
$\tau_{i,j}^T$	Travel time of AGV from point $i \in P^d$ to point $j \in P^a$, $\in \mathbb{R}^+$
$\tau_{i,j}^U$	Travel time of UAV from point $i \in P^U$ to point $j \in P^a$, $\in \mathbb{R}^+$
$\varTheta$	Prohibited Flight Scenario of UAV
$\varUpsilon$	Travelling height of UAV in 'm', $\in \mathbb{R}^+$
ζ	Battery capacity of UAV for maximum flight time in min, $\in \mathbb{R}^+$

LIST OF ACRONYMS/ABBREVIATIONS

/	‘and’ or ‘or’
A/C	Aircraft
AGV	Autonomous Ground Vehicle
CDMP	Cooperative Delivery Mission Planning
Cont.	Continuous
Coop.	Cooperative
CVRP	Capacitated Vehicle Routing Problem
Del.	Delivery/Deliveries
DIM	Dimension
ECDMS	Expected Cooperative Delivery Mission Scenario
EMF	Electromagnetic Field
Enum.	Enumeration
Epm	Energy Consumption Rate for Steady Level Flight
FSTSP	Flying Sidekick Traveling Salesman Problem
GV	Ground Vehicle
HTOL	Horizontal Takeoff and Landing
J	Joule
kg	Kilogram
LD	D’Andrea Technique for Energy Consumption Calculation
LPG	Liquefied Petroleum Gas
mAh	Milliampere Hour
max.	Maximum
min	Minute
min.	Minimum
PDSTSP	Parallel Drone Scheduling Traveling Salesman Problem
R2	Stolaroff Technique for Energy Consumption Calculation
R3	Kirschstein Technique for Energy Consumption Calculation
RH	Dorling Technique for Energy Consumption Calculation
s	Second

TSP	Travelling Salesman Problem
TSP-D	Travelling Salesman Problem with Drone
UAV	Unmanned Aerial Vehicle
V	Volt
VRD	Vehicle Routing with Drone
VRP	Vehicle Routing Problem
VRPD	Vehicle Routing Problem with Drone
VRPPD	Vehicle Routing Problem with Pick-Up and Deliver
VRPTW	Vehicle Routing Problem with Time Windows
VTOL	Vertical Takeoff and Landing
W	Watt

1. INTRODUCTION

Currently, the demand for faster, less energy consuming and autonomous delivery methods is increasing in the logistics industry within the scope of last-mile delivery. Even though conventional delivery methods such as delivery by a deliveryman with truck or a deliveryman with a bicycle in neighborhood dominated the industry, these methods have disadvantages about fuel consumption and delivery time. Hence, cooperative delivery mission planning for the UAV and the AGV was considered as an alternative solution for the last-mile delivery missions instead of conventional methods. Recent research and development activities generally focus on integration of unmanned aerial vehicles for last-mile delivery mission instead of conventional methods. Main motivation behind it is to use low energy consumption advantage and high-speed movement capability of UAVs to minimize delivery time and cost. Synchronization of UAVs with AGVs provides advantage to conduct delivery to hard-to-reach place. In addition, heterogeneous mobile platforms offer variability of strategies considering operational time, delivery time, fuel consumption etc. A brief comparison of UAV and AGV regarding their advantages and disadvantages are given in Table 1.1.

Table 1.1. Comparison of Advantages and Disadvantages of UAV and AGV for Cooperative Delivery Mission Planning.

VEHICLE	**ADVANTAGES**	**DISADVANTAGES**
UAV	High Travel Speed Independent from traffic, road signs, traffic lights etc. Low Power Consumption	Low Payload Capacity Single Delivery per Dispatch Limited Battery-Range
AGV	High Payload Capacity Multiple Deliveries per Dispatch High Battery-Range	Low Travel Speed Dependent to traffic, road signs, traffic lights etc. High Power Consumption

UAVs offer several advantages compared to conventional delivery methods. Since the UAVs cruise through the air, the delivery is independent from roads, traffic lights, speed limits, pedestrian crossings etc. Especially in residential areas, there are too many external factors influencing ground deliveries. Delivery distance of a UAV is considered as air distance in contradistinction to ground deliveries and it takes much less time than the ground route. On the other hand, main disadvantages of the UAVs are payload capacity and flight range/time. Increasing payload capacity of a UAV is possible but it costs more energy consumption for the UAV and the more energy consumption needs higher capacity of batteries which also causes more battery weight on drone. Moreover, increasing flight range capacity of a UAV with keeping payload capacity same is also possible but it also requires battery with more capacity which also causes more battery weight. Current battery technologies are not advanced enough to increase flight range or payload capacities of UAVs with a worthwhile opportunity cost. In summary, using UAV for long distance deliveries or with heavy payloads causes more consumption which equals heavier battery packs on UAV and it also causes more energy consumption. The situation goes into an endless loop.

Autonomous ground vehicles are also used for last-mile deliveries especially for short distances. There are two types of vehicles in general. First one is like ground version of a UAV, hereinafter called as Type I, which is able to carry only one package per delivery and need to turn back to a depot or warehouse to charge itself and take the next deliverable. This type of AGV is practical for small and uncrowded areas but they are not cost and time efficient. Vehicles have to follow legal instructions on the road and also sidewalks which causes losing lots of time. Energy consumption of this type of AGVs is not always less than a UAV which carry out similar delivery mission. It depends on road conditions, length difference between air and ground routes, elevation difference between start and finish points of delivery etc. Second type is like autonomous version of conventional delivery trucks, hereinafter called as Type II, which is able to carry all deliverables on its back during delivery mission. They generally follow a predetermined delivery path and drop each deliverable one-by-one. Since almost all delivery conditions are same compared to a conventional truck, energy consumption and delivery time savings are much easier to compare.

UAVs, Type I and Type II AGVs have different advantages and disadvantages compared to each other. Planning a cooperative delivery mission with more than one of them regarding advantage and disadvantage of each of them instead of only one, provides a cost-efficient delivery mission. A Type II AGV and a UAV were chosen for the cooperative delivery mission scenario. Main considerations are to eliminate ground limitations as much as possible and to consume less energy compared to only AGV delivery. Usage of the UAV provide elimination of ground limitations and faster deliveries compared to Type I AGV. On the other hand, usage of Type II AGV provides achieving delivery mission which exceed flight range of the UAV or payload capacity of the UAV. And the main advantage of using both AGV and UAV for delivery mission planning is that the AGV acts like a moving depot for the UAV which enable UAV to pick up next deliverable and to charge itself if necessary.

The most challenging part of planning delivery mission is to find an optimized algorithm for that. Main aim of the delivery mission is that to deliver each package by UAV or AGV to related customer for only once and to perform that by consuming the least energy within the shortest time. In fact, the problem is a simple permutational problem known as Travelling Salesman Problem in literature and has an exact solution which is described simply by Hahsler and Hornik [1]. By considering thousands or millions of possibilities and calculate the delivery time and energy consumption of vehicles in total, it is possible to find the best route for delivery mission. Furthermore, taking every constraint, external factors, legal and physical limitations into account causes complexity and increases computational time. Especially for large numbers of delivery points, computational time may reach infinite for the current technological potential. On the other hand, using of alternative optimization algorithms such as greedy algorithms or dynamic programming etc. might be decrease computational time dramatically. In this study, main motivation is to find proximate delivery mission planning solution to the exact solution in an acceptable computational time by using greedy algorithms and dynamic programming methods.

1.1. Literature Review

Literature is reviewed within the scope of UAVs, AGVs, delivery routing concepts and solution algorithms. Therefore, UAV and AGV are investigated in Section 1.1.1 and 1.1.2, respectively. In Section 1.1.3, delivery routing concepts are investigated under two major categories; Vehicle Routing Problems (VRP) and Travelling Salesman Problem (TSP). Finally, in Section 1.1.4, solution methods and algorithms related to VRP and TSP are handled.

1.1.1. Unmanned Aerial Vehicle

UAVs can be defined basically as flying vehicles which are remotely controlled and/or autonomous. These vehicles are classified based on their different specifications and areas of usage according to Singhal and Mathew [2].

1.1.1.1. Classification of UAVs. First classification method is based on aerodynamics of UAVs. Fixed-wing UAVs [3], multi-copter (also called multirotors or drones) [4], chopper (also called UAV helicopter) [5] and flapping-wing UAVs [6] are the major categories of UAVs within the aspect of aerodynamics. Major effect of aerodynamics on this categorization is about movement capability of the UAVs in air. Fixed-wing UAVs and flapping-wing UAVs are capable of high-speed gliding in opposite to multi-copters and choppers. On the other hand, multi-copters, choppers and VTOL UAVs are capable of hovering in opposite to fixed-wing UAVs [7]. In addition, maneuverability of UAVs is also directly-related to aerodynamics [8]. All these criteria are evaluated to choose the most convenient type of UAV for a determined operation.

Second classification criterion is landing style of UAVs. There are two major categories; horizontal takeoff and landing (called HTOL) [9] and vertical takeoff and landing (called VTOL) [10, 11]. HTOL UAVs are capable of horizontal takeoff and landing which provide high speed gliding and smooth horizontal landing. VTOL UAVs are capable of high maneuverability and vertical takeoff/landing [12]. Fixed wings and flapping wing UAVs are also categorized as HTOL and on the other hand, multi-copter and chopper UAVs are categorized as VTOL UAVs.

Table 1.2. Advantages and Disadvantages of Different Types of UAV.

Landing	**Aerodynamics**	**Concept**	**Advantage**	**Disadvantage**
HTOL	Fixed-Wing	Have Similar Concept to A/C	Long Range High Payload Capability	Low Maneuverability
	Flapping-Wing	Design Inspired by Insects	High Maneuverability	Low range High Energy Consumption
VTOL	Multi-copter	Quadcopter	High Maneuverability Hovering Capability	High energy consumption
	Chopper	Helicopter	High maneuverability	Hard Stabilization

Third classification criterion is application (usage) area of UAVs. According to Singhal [2], three main application area; civilian, environment and defense are identified in literature [13].

(i) *Civilian*

Since UAVs have multifarious design, they are able to satisfy different individual, commercial or social needs. UAVs provide different benefits for each different user. For instance, a multicopter UAV can perform a delivery mission for a place which is unable to reach physically or perform an observation mission of a field which is a dangerous and/or unsafe place for people to be there. On the other hand, a UAV can perform spraying a farmland mission with much less cost instead of a tractor for the farmer. Moreover, for aviation industry, turning an air vehicle into an unmanned version provides financial benefits and also design simplicity because regulations about human factor such as air supply, thermal control and cabin pressurization in cockpit cause additional cost and complicated design solutions.

According to Otto, main application areas for civilian purposes are listed and also shown in Figure 1.1 as physical infrastructure (including energy, gas, oil, road, railway, construction), agriculture, entertainment and media, transport, and security [14].

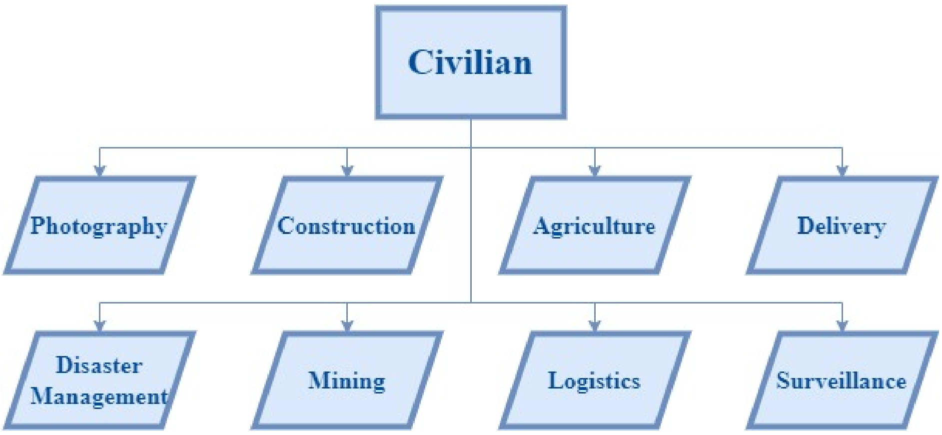


Figure 1.1. Civilian Application Areas of UAV.

(ii) *Environment*

UAVs provide cost and time beneficial service for environmental missions. Most environmental observation and monitoring duties necessitate stations or fleets to be conducted. Smith [15] researches and evaluates the usage of UAVs for environment management within the scope of international applications. Accordingly, UAVs are used in Africa for monitoring destruction of nature and in Brazil for monitoring deforestation of Amazon. Another popular application area of UAVs for environment is to monitor air quality. Several studies have conducted to evaluate effectivity of UAV regarding some challenges such as power consumption, weight constraints, choice of proper sensors, propeller effect to measurements etc. [16, 17]. Nevertheless, usage of UAVs to monitor air quality is becoming popular over the years.

Soil and crop monitoring are another application area of UAV usage. Since observing current status of soil and crop is crucial for agriculture industry, there are some researches in the literature by Kavoosi [18], Capolupo [19] and Corbane [20]. Kavoosi [18] focus on UAV usage for monitoring soil residue cover within the scope of crop residue management in agriculture. On the other hand, Capolupo [19] have studied the UAV use for detection of contaminated soil by copper in Italy region and Corbane [20] have studied the contribution of UAV usage to soil surface characteristics analyses in Mediterranean area. Since the study areas are similar to each other, these studies avail to understand positive effects of UAVs on environment applications. UAVs are also used for water and under water investigation and monitoring activities. These applications are also called hydrology applications. Water applications operates on the surface of water for several purposes. Most common purpose is to monitor water quality, coral reefs, fish farms etc. [21].

UAVs are also used to measure flow of water in physically challenging water environment during adverse hydrometeorological cases [22]. Environmental application areas of UAVs are listed in Figure 1.2 according to Singhal [2].

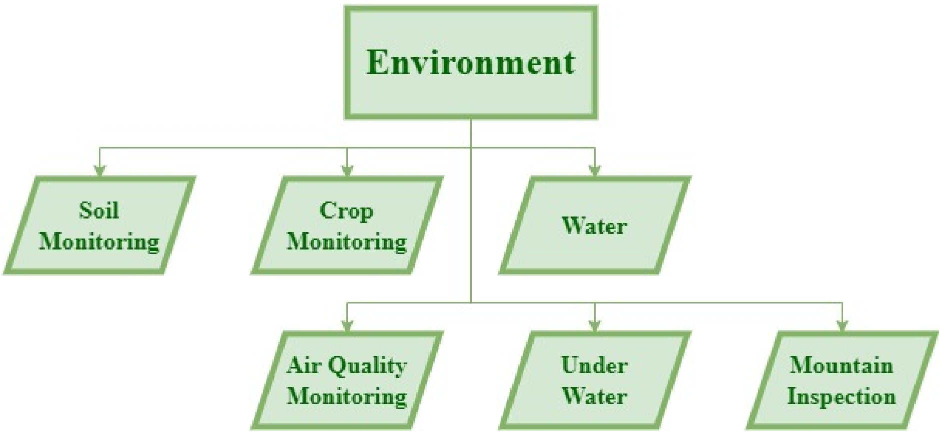


Figure 1.2. Environmental Application Areas of UAV.

(iii) *Defense*

There are several studies focused on military application of UAV usage in the literature. Main concept of the studies is to focus on acquisition of replacing human on the battlefield by UAVs to prevent death without any tactical loss. Sieg [23] have studied on offensive capabilities of UAVs which led to three operational concepts; suppressing enemy air defense, supporting counterinsurgency operations and finding and eliminating enemy targets. Even though UAVs have advantageous capabilities compared to human on the battlefield, when all the limitations, costs and other factors are evaluated, UAVs look like never replacing humans on the battlefield. Dunn [24] have also studied about usage of UAVs in military applications by underlining effects of UAV technology within the scope of conventional purport of safety and security. Application areas of UAVs in defense industry are listed in Figure 1.3 according to Singhal [2].

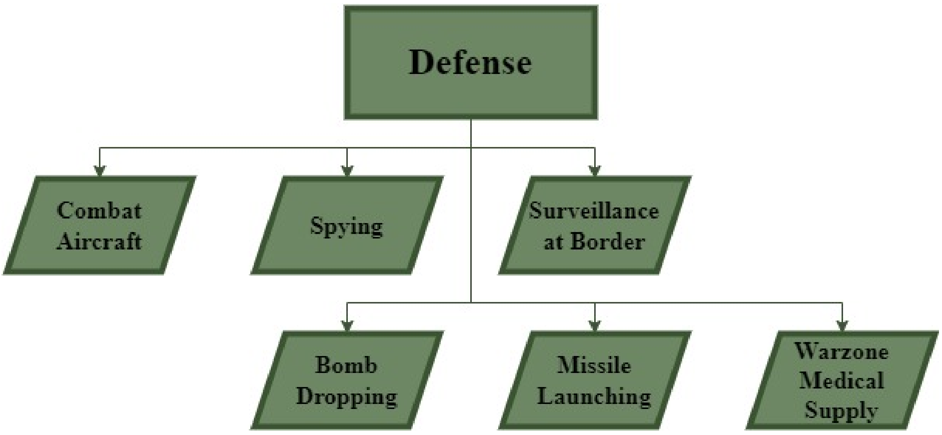


Figure 1.3. Application Areas of UAV for Defense.

1.1.1.2. Energy Consumption. Fuel/energy consumption is the most crucial objective of the unmanned autonomous vehicles which are operating last-mile delivery to customer considered in planning mission. Energy consumption creates vital importance for UAVs because out of charge/fuel while flying may cause loss of control and fail delivery mission even crashing on ground. Furthermore, fuel/energy consumption rate of a UAV may change with respect to motion of it such as taking-off, hovering and landing. All the components of a UAV affect the fuel/energy consumption rate. The rate depends on motor powers, number of motors, total weight of the vehicle, flying altitude, wind speed etc. Sum of objectives varies momentarily and the others stay constant. According to Chan and Kam, energy consumption rate of a battery-powered UAV can be modelled by calculating the propulsion system power transmission and converting it to energy consumption [25].

In this model, main power parameters are mechanical power of propellers (P_M), power loss of battery such as back EMF loss (P_m), iron loss (P_I) and copper loss (P_C). These are formulated as

$$\begin{aligned} P_M &= M * \omega && (1.1)\\ P_m &= V_m * I_m && (1.2)\\ P_I &= k * f * B^2 && (1.3)\\ P_C &= I^2 * R_m. && (1.4) \end{aligned}$$

According to this power consumption, propulsion system power transmission and energy consumption can be calculated as

$$\begin{aligned} P_{in} &= P_M + P_m + P_I + P_C && (1.5)\\ E_{pro} &= P_{in} * t. && (1.6) \end{aligned}$$

Since this energy consumption model do not consider external factors like disturbances, field tests result with the error rate between 15-25% [25]. Weight of the vehicle can be listed as the number one objective for energy consumption. As the weight increase, the vehicle consumes more energy and more energy requires more battery capacity and more battery capacity cause more weight on the vehicle. At this point, the equation is being deadlocked. According to another research belongs to Zhang, effect of weight of the vehicle has been investigated by the techniques published before such as D'Andrea, Dorling, Stolaroff, Kirschstein and Tseng [26]. With respect to each technique, energy consumption rate and range values are calculated and compared in graphs separately in the research. Energy consumption of the vehicle is increasing as the payload weight increases but the similar to other research, there is not a clear consumption assumption for the vehicle. Calculated values of all techniques differ from each other. It is very hard to predict the energy consumption of a UAV because of external disturbances and unexpected consumption effects so that while estimating energy consumption, taking average consumption data given by UAV manufacturers in the market help to construct control algorithm. Technical energy-related specifications of two samples of commercial UAVs in the market are listed below for comparison;

(i) DJI – Matrice 300 RTK [27];
  - Payload Capacity: 2.7 KG
  - Max. Take-Off Weight: 9 KG
  - Battery: 5000 mAh / 4920 mAh (One built-in + one external)
  - Voltage: 7.2 V / 7.6 V
  - Max Flight Time: 55 min (without payload)

(ii) DJI – Agras T20 [28];
  - Payload Capacity: Approximate 20 KG
  - Max. Take-Off Weight: 47.5 KG
  - Max Power Consumption: 8300 W
  - Hovering Power Consumption: 6200 W (with a takeoff weight of 47.5 kg)
  - Battery: 18000 mAh
  - Voltage: 51.8 V
  - Max Flight Time: 10 min (with payload)

1.1.2. Autonomous Ground Vehicle

AGVs are investigated within the scope of fuel/energy consumption only since UAV is the leading part of this study. Having enough carrying capacity and fuel/battery are the only parameters considered for the model.

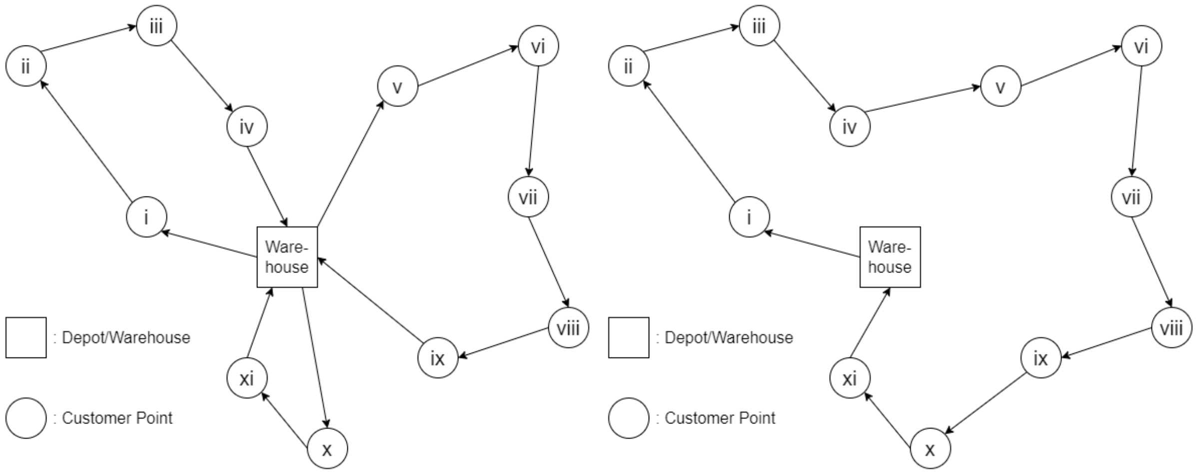
1.1.2.1. Fuel/Energy Consumption. Ground vehicles are more advantageous than aerial vehicles about fuel/energy consumption. Van industry has massive van manufacturers which are investing and developing new technologies for fuel economy on vans and they are also evaluating new fuel types such as LPG, electric, hydrogen and hybrid versions of vans. According to these developments in the industry, there are several types of van models which have fuel economy. Fully electric powered cargo vans contain huge batteries under the chassis for longer ranges. But in opposite hybrid model cargo vans which contain both electric powered engine and petrol engine can reach same ranges with smaller batteries because in the hybrid system, batteries store only the energy generated in petrol engine for electric-powered engine.

Furthermore, same condition about weight on the vehicle is valid for the ground vehicles like aerial vehicles. As the amount of weight increases, the fuel consumption increases too. In this situation, fully electric-powered cargo vans need much more battery capacity for stored delivery packages.

1.1.3. Delivery Routing Concepts

In the literature, delivery routing issue is addressed to and examined under two main concepts; Vehicle Routing Problem called as VRP and its variants, and Travelling Salesman Problem called as TSP and its variants. Accordingly, literature search of delivery routing problem is divided into two categories in this study as well. Main difference between VRP and TSP is the tour of vehicle dependant to number of the vehicles. TSP have only one vehicle to complete a single tour consist of all customer points. On the other hand, VRP is capable of to complete several tour consist of all customer points again but in different tours. Simple schematics of both VRP and TSP are given in Figure 1.4(a) and 1.4(b), respectively.

1.1.3.1. Vehicle Routing Problem (VRP). Vehicle routing problem can be defined as a combinatorial optimization problem consisting of a set of target points (customers), and a set of vehicles to find an optimal set of routes between target points (customers) for delivery. VRP is firstly proposed by Dantzig and Ramser in 1959 [29] within the scope of importance for distribution, transportation and logistics. Goods distribution, mail/package delivery, garbage collection, and snowplow are some of the major application area of VRP in daily activities. VRP cover a wide variety of subproblems



(a) VRP. (b) TSP.

Figure 1.4. Simple Vehicle Routing Problem and Traveling Salesperson Problem Schematics.

such as Vehicle Routing Problem with Drones (VRPD), Vehicle Routing with Drones (VRD) [30], Capacitated Vehicle Routing Problem (CVRP), Vehicle Routing Problem with Time Windows (VRPTW) and Vehicle Routing Problem with Pick-Up and Delivery (VRPPD).

(i) *Vehicle Routing Problem with Drones (VRPD)* is a vehicle routing problem which considers trucks (AGVs) and also drones (UAVs) - generally multicopters for delivery route planning. As a general concept, a set of AGVs equipped with single/several UAVs conducts delivery mission cooperatively. UAVs are able to take-off from and land on AGVs at depot or customer delivery point [31]. Main motivation behind it is to use low energy consumption advantage and high speed movement capability of UAVs to minimize delivery time and cost. Moreover, using UAVs synchronized with AGVs also provides advantage to conduct delivery to hard to reach place. Poikonen [32], studies on VRPD model of Wang, which is a case allows for a fleet of m homogeneous trucks, each equipped with k drones (all allowed for only single delivery per flight), to either deliver packages directly, or bring the drones within range to deliver packages [31], with some extensions: cost issues, limited battery life and other metrics.

(ii) *Vehicle Routing with Drones (VRD)* has a similar naming to vehicle routing problem with drones, but mechanics of delivery model is different. In VRPD, conditions are more flexible AGV and/or UAV may conduct delivery mission together or AGV may perform only mobile depot or charging station for UAV. And also UAV may conduct consecutive deliveries by carrying more than one piece at once. In VRD, both AGVs and UAVs are conducting delivery and UAV have to turn back to AGV after each parcel/item delivery. Problem is formulated as a bit more specified and all conditions/constraints are defined according to study of Daknama and Kraus [33].

(iii) *Capacitated Vehicle Routing Problem (CVRP)* can be defined as limited version of a VRP. In CVRP, vehicles (AGVs and/or UAVs) have carrying/payload limits which are quantitative such as weight or volume or any other physical or regulation-related limitations. Since the solution of VRP is increasing combinationally, it reaches infinite computational time to find an exact solution for around 20-25 delivery points. On the other hand, CVRP is able to be solved in much less computational time by courtesy of limitations/constraints. By constraints, some solution possibilities are eliminated and it decreases computational time very much. In the literature, there some cases which are able to reduce computational time and find an exact solution for exceeding 100 delivery points by means of determined constraints and chosen solving methods. Lysgaard [34] have modified the model of Augerat [35] and developed an brunch-and-cut algorithm to find an exact solution the modified model. According to report of Baldacci [36] algorithms developed by Augerat [35], Baldacci [36] and Lysgaard [34] were able to find an exact solution for 135 delivery points. That is the largest non-trivial CVRP delivery point set to be exactly solved to that date. In the report, it is declared that best proposed algorithm belongs to Fukasawa [37]. Algorithm of Fukasawa blends branch-and-cut and Set Partitioning (SP) [38] methods.

(iv) *Vehicle Routing Problem with Time Windows (VRPTW)* is a version of VRP which approaches to VRP from time window as its very name signifies. It can be defined simply as a combination of VRP and scheduling problem. Set of vehicles must conduct delivery mission under determined conditions and constraints and also satisfy defined time restrictions [39]. In the literature, there are both exact and heuristic approaches to VRPTW since the problem is convenient for different solution methods based on varying constraints and restrictions. Desrosiers [40] and Cordeau [41] have chosen exact solution technique and their models and approaches are detailed in the study of Larsen, and Cook and Rich. Nevertheless, heuristic solution are more common in the literature due to complex structure of VRPTW. In this scope, Campbell and Savelsbergh [42] has focused on insertion heuristics which is able to overcome time restrictions but also other constraints.

(v) *Vehicle Routing Problem with Pick-Up and Delivery (VRPPD)* is a vehicle routing problem which don't consider only delivery of a parcel from depot/warehouse to customer but also delivery of a parcel from customer to depot/warehouse. This vice versa situation causes complexity for delivery planning and turns into a challenge to find optimal route for the mission. Therefore, studies generally focus on simplifying problem firstly to optimize delivery routes. Bent and van Hentenryck [43], have developed a two-stage hybrid heuristic method to solve this problem; simulated annealing is applied in first stage to minimize number of routes and in second stage to minimize total route length. On the other hand, Ropke and Pisinger [44], have divided the problem into two subproblems in their study; first subproblem to consider only delivery mission and second subproblem to consider only pick up mission.

1.1.3.2. Travelling Salesman Problem (TSP). Travelling salesman problem can be defined as a specified vehicle routing problem. Main objective of TSP is to find the shortest route between the determined cities for travelling salesman by visiting each city only once and coming back to starting city as last stop [45]. Karl Menger is known as the first person to study and formulate TSP in 1930 [46]. Since the first day it was formulated, TSP is one of the most common problem in computational science. The reason of popularity of TSP is that its formulation is applicable for many different areas such as vehicle routing, transportation, logistics, manufacturing etc. In literature, TSP is classified as a NP-hard problem in combinatorial optimization which means that it is not possible to solve TSP in polynomial time [47]. Let's say salesman has to visit n cities, then there are $(n-1)!$ possible routes for the exact solution of shortest route for n cities. Since, large number of n reaches infinite polynomial time, exact solution methods are not preferred that much. Instead of exact methods, heuristic approaches are studied in common to obtain acceptable computational time. Zhang have studied comparison of varying exact and heuristic algorithms and heuristic algorithms conclude in hundred or thousand times less compared to exact algorithms [48].

(i) *Travelling Salesman Problem with Drone (TSP-D)* is a variant of TSP. Although it is derived from TSP, planning logic in the background is quite a change. Main difference is to consider at least one more vehicle for planning. And this additional vehicle is an aerial vehicle whose movement capabilities and parameters are completely different compared to a ground vehicle. This problem is defined by Agatz as to develop an optimal cooperative delivery planning by involving both assignment and routing decisions for a truck (AGV) and a drone (UAV) [49]. Since TSPD is a conceptual problem which is prone to generating variants, there are several similar problems such as Flying Sidekick Traveling Salesman Problem (FSTSP), Parallel Drone Scheduling Traveling Salesman Problem (PDSTSP), Travelling Salesman Problem with multiple Drone (TSPmD) etc. On the other hand, form of utilisation of UAV has strong effect on definition and mathematical model of the problem. Accordingly, all these problems (FSTSP, PDSTSP, TSPmD etc.) are addressed as different variants of TSPD in the literature instead of subproblems of TSPD.

(ii) *Flying Sidekick Traveling Salesman Problem (FSTSP)* is defined by Murray and Chu as a problem considering a set of customer/delivery points who/which have to be visited only once by a GV (driver-operated) or a UAV [50]. In the problem, warehouse is defined as a single point and able to visit only once(vehicles depart from and turn back when the delivery mission is completed.) and UAV is able to depart from and turn back to GV on delivery points. Murray and Chu also considers charging, flight range and payload capacity of UAV while planning, and apply heuristic algorithms to solve the problem for up to 10 customer/delivery points [13]. There are also additional studies based on Murray and Chu model, Minh Ha, Deville, Dung Pham and Hoang Ha have developed a hybrid genetic algorithm [51] with a heuristic approach and applied on both Murray and Chu model and also their model [52] for up to 100 customer/delivery points to prove the feasibility of their algorithm on FSTSP.

(iii) *Parallel Drone Scheduling Traveling Salesman Problem (PDSTSP)* is defined by Murray and Chu as a problem which assumes that GV and UAV perform their delivery missions separately. In other words, there is a single depot/warehouse, and UAV is only able to depart from and turn back to depart/warehouse for each delivery, on the other hand, GV is able to follow an optimal route to perform deliveries in order [50]. In this problem, only concern is to find an optimal path for GV, and additionally, physical and legal constraints of UAV are considered for planning. Only collaboration between GV and UAV is to perform delivery mission in same time period. Kim and Moon [53] have studied an extended version of Murray and Chu model. In the model of Murray and Chu, UAV is limited to serve customer/delivery points which are in the range of only depot/warehouse. Kim and Moon use additional points called drone stations, which store deliverables and also provide charging for UAVs, in their model. They have also used a decomposition approach similar to Murray and Chu for developing an algorithm with the difference of focusing on the conditions on the decomposition. Kim and Moon declares that their approach difference guarantees the optimal solution for delivery time for up to 80 customer/delivery points [13].

1.1.4. Solution Methods/Algorithms

Solution methods/algorithms to solve VRP/TSP and their variants can be divided into two categories; exact solutions and heuristic solutions. A detailed literature review has been carried out within the scope of FSTSP and its variants. It has been investigated, by inspiration of the literature review of Macrina [13], that who used which method on the models with which characteristics and the results are listed in Table A.1 accordingly.

1.1.4.1. Exact Solution Methods/Algorithms. These methods focus on to find exact solution for the problem. Solution gives the accurate result value, not the proximate value. All the exact solution methods/algorithms can be generalized as enumeration method since all of them focus on every single possibilities to find the best computational value.

(i) *Brute-force Algorithm* can be simply defined as combinatorial solution also known as Enumeration method. The method evaluates all the route possibilities as a total number of $(n-1)!$ for n customer/delivery points. Main advantage of using Brute-force method over heuristic methods is to find global maximum always. On the other hand, this methods is feasible for a small number of customer/delivery points. As number of customer/delivery points increases, route possibilities grow exponentially and computational time becomes too large for even current computers to achieve [47].

(ii) *Branch and Bound Algorithm* consists of two main parts; branching and bound, as the name suggests. Branching means that set of customer points are divided into two subsets, which are called branches, for every step of the process until there is no divisible subset left. Bound means the lower restriction to eliminate the branch with higher cost. Algorithm sets lower bound as minimum cost of first branch, and eliminates other branch with higher cost (delivery time). Algorithm continue to search for a better branch with lower cost and if it is found, lower cost is updated according to found one and older best option is eliminated. This search continues until there is no better branch is left. Although branch and bound algorithm decrease computational time by elimination of the branches with higher cost compared to brute-force algorithm, it is still not feasible for over 60 delivery points [54].

(iii) *Dynamic Programming Algorithm* solves the problem by dividing it into sub problems in a recursive manner and makes optimization about the given system while solving it. By storing the results of subproblems, the method reduces computational time. If the same subproblem occurs, it uses stored result for it, does not recompute it. This simple optimization technique provides reducing time complexities for the system from exponential to polynomial [55]. Dynamic programming methods evaluates $n * 2^n$ subsets(possibilities) for n cities. Implementation steps of dynamic programming on TSP are [48];

(a) Choose starting customer/delivery point as depot/warehouse and call it point s.
(b) Calculate distance value between s and other points. Call it base cost.
(c) If number of customer/delivery points is more than 2, then calculate the distance from the point to the closest customer/delivery point. Add this distance (cost) value to base cost and memorize the cost value of the subsets of customer/delivery points which consist of more than 2 customer/delivery points.
(d) Repeat step (c) until there is no customer/delivery point left which is not visited.
(e) Add the distance(cost) value between the last point visited and starting point (depot/warehouse) to obtain minimum distance (cost) value of TSP by completing the cycle.

For example, let's have a set of points as $P = s, 1, 2, .., n$ and call base cost C. Distance between $s \ and \ 1$, $s \ and \ 2$, $s \ and \ n$ are an element of base cost. If the cost of a subset such as $(s, \ 1 \ and \ 2)$ is evaluated, distance between $1 \ and \ 2$ is calculated only and the value is added to cost of $s \ and \ 1$ to obtain the subset cost. Algorithm memorize each subset cost and obtain new subset cost values recursively regarding memorized values. That is the main advantage of dynamic programming to reduce computational time.

1.1.4.2. Heuristic Solution Methods/Algorithms. Heuristic methods/algorithms are developed to solve the problem faster than exact solution methods/algorithms. They obtain faster but proximate results instead of exact results by compromising about precision, completeness and accuracy of results. These algorithms are generally used to solve NP-hard and NP-complete problems which are the categories of decision problems [56]. Since the main advantage of heuristic algorithms is quickness, they are generally preferred to be used when the computational time of exact solution method is too large.

(i) *Greedy Algorithm* solves the problem by adding shortest possible edge, which is the distance between two random customer/delivery points, in each iteration. Since it focuses on only local minimum, it is not always possible to obtain global optimum results. Time complexity of greedy algorithms are $n^2log_2(n)$ for n customer/delivery points [57]. Implementation steps of greedy algorithms are;

   (a) Subtour is initially 0 for n customer/delivery points.
   (b) Calculate each possible edge distances and sort in an array.
   (c) Choose the shortest edge value in the array and add it to subtour if it does not create a cycle with less than n edges or increase the degree of any customer/delivery points.
   (d) Repeat step(c) until there is no edge value left in the array and complete the tour.

   Agatz has two step solution for the presented TSP-D model in their study; first they use heuristic approach for TSP solution, secondly generate a greedy algorithm to partition the TSP tour into a subtour for the UAV and a subtour for the AGV [49].

(ii) *Genetic Algorithm* can be defined as a search technique inspired by Charles Darwin used to find proximate solution to combinatorial problems such as TSP. Algorithm based on natural selection where the best individuals are selected for reproduction of next generations. Accordingly, a set of solutions for a problem is selected as best ones to solve the problem. Genetic algorithm consists of five main phases which are initial population, fitness function, selection, crossover and mutation. According to Potvin, following terms; chromosome, decoded chromosome, population and fitness function, are equivalent to encoded solution, solution, set of solution and objective function, respectively [58]. Ferrandez use genetic algorithm to determine truck route between launch locations in their model. The model presented in their study focus on to determine optimal number of launch sites and locations for UAV [59].

(iii) *Nearest Neighbor Algorithm* is the simplest heuristic approach which considers only local optimum. Therefore, it is not capable to find global optimum for every case especially for large amount of customer/delivery points. The algorithm evaluates n^2 possibilities for n cities [60]. Implementation steps of nearest neighbor algorithm are;

  (a) Choose the starting point (depot/warehouse) and find the closest point to create an edge(route), then call closest point as ending point for next step.
  (b) Find the closest point to ending point and extend the route through ending point, then call closest point as ending point for next step.
  (c) Repeat step (b) until there is no customer/delivery point which is not visited.

(iv) *2-opt and 3-opt Algorithms* are similar algorithms to each other. 2-opt algorithm is a basic edge changing method. Two edges of the tour are removed and reconnects two paths via a new created path. When the two edge is removed, there is only one way to reconnect this edges in different combination, so the completion of tour is guaranteed during each change. This change is implemented if and only if created tour is shorter than existing one. Edge changing continues until there is no 2-opt possibility left. Then, the tour is called 2-optimal.

3-opt has similar principle to 2-opt, the only difference is the number of edges which are changing during new option tries. That means there are two possibility to reconnect the paths to satisfy a valid tour. Edge changing continues until there is no 3-opt possibility left. Then, the tour is called 3-optimal. If a tour is called 3-optimal, it is also 2-optimal [61].

(v) *Insertion Heuristic Methods* are generally simple and easy-to-apply methods and there are several variant of these methods in the literature such as nearest insertion [62], convex hull [57] etc. Nearest insertion evaluates only n^2 possibilities for n cities. Nearest method can be described as a method from cell to wall and in opposite convex hull can be described as from wall to cell. For nearest insertion [57];

  (a) Choose the closest customer/delivery point to starting point(depot/warehouse) and create a subtour of them

  (b) Find the shortest customer/delivery point to any point of subtour
  (c) Find the minimal edge insertion cost to obtain shortest extended subtour
  (d) Repeat step (b) and (c) until all the customer/delivery points are included in subtour.

  Convex Hull insertion evaluates $n^2 log_2(n)$ possibilities for n cities [57]. For convex hull;

  (a) Find the convex hull of the set of customer/delivery points (starting subtour)
  (b) Find the shortest customer/delivery point to any point of subtour
  (c) Find the minimal edge insertion cost to obtain shortest extended subtour
  (d) Repeat step (b) and (c) until all the customer/delivery points are included in subtour.

(vi) *Ant Colony Optimization Method* is developed as the imitation of movement of real ants which are finding shortest routes between food and the nest by Beckers, Deneubourg and Goss as a result of observation of nature [63]. Ants leave pheromones temperamentally behind them when they follow a new path/route. By means of left pheromones, other ants can reach to food through shortest path/route. Most important thing of this route following is the number of ants. Since the ants leave pheromone behind them, more ants mean more pheromone on the related path/route and more pheromone means more attractive path/route option for an ant. Therefore, the more ants follow a path/route, the closer you are to find global optimum [57]. Application of ant colony optimization is fictionalized as putting a set of ants on different cities randomly and then they are set to move to another random city. Ants are not able to visit a city again, they visit each city only once. When an ant follow a path/route, it leaves pheromone behind it, then another ant will be prone to follow the path/route which has more pheromone. As much as this process is repeated, a short enough path/route will be found. Dorigo [64] has explained mathematical model and the logic behind ant colony optimization in details in his study.

1.2. Research Objective & Contributions

Objectives of the study are;

- Development of a computationally efficient cooperative delivery mission planning algorithm for UAV and AGV by dynamic programming (DP) method while regarding external factors and constraints.
- Comparison of delivery time data with AGV-only delivery model and cooperative delivery model.
- Analysis of disponibility of dynamic programming method instead of enumeration method to find the exact solution of cooperative delivery route for large amount of delivery points.
- Observation effect of optimizing major parameters on computational time and the precision of the solution.

Regarding the literature review, contributions of the study are;

- A weight-aware energy consumption model is applied to original FSTSP model for UAV.
- Rechargeable battery with the limited capacity is used for UAV instead of changing battery for each delivery as in original FSTSP model.
- Exact algorithms are studied instead of heuristic methods to obtain exact time value for cooperative delivery mission planning for UAV and AGV.
- Dynamic programming method is used to decrease time complexity of system and obtain exact solutions for larger number of delivery points in similar computational times.
- Optimization parameters is defined for the solution algorithm to decrease computational times for the exact solution.

2. MODEL

In this section, mathematical model and model description is explained as variant of the FSTSP model presented model by Murray and Chu [50]. In section 2.1, the problem held in this study is defined, and considered assumptions for the model are held in Section 2.1.1. In section 2.2, expected solution for the problem are explained. Mathematical model, and further considerations on variables and constants are explained in Section 2.3 and 2.5, respectively. Differences from the model presented by Murray and Chu [50] are detailed in Section 2.4. Solution algorithm including all functions used for the solution is given in Section 2.6.

2.1. Problem Statement

Resource-aware cooperative delivery mission planning for UAV and AGV considers a system consisting of a single AGV and a single UAV. UAV and AGV serve cooperatively to n delivery points by starting from warehouse and ending at warehouse, together. The AGV, besides serving delivery requests itself, serves as a moving hub for the UAV, effectively extending the service range of the UAV. The UAV can pick up a package from the AGV, take off from any delivery point or warehouse to deliver it to a delivery point, and return to the AGV at any delivery point or warehouse. The AGV and UAV move simultaneously, but not independently. Their movements have to be synchronized to allow for the UAV to return to the AGV.

The problem is to find the routes of the AGV and the UAV that minimize the cooperative delivery time of UAV and AGV. A real-world delivery mission planning for UAV and AGV is given as an example in Figure 2.1. Green triangle represents warehouse and the AGV follow white-lined route for delivery. Black-white triangles represents the deliverables delivered by AGV and red circles represents the deliverables delivered by UAV. Green and red straight lines represent going and coming way of UAV to/from delivery point, respectively. Assumptions listed in Section 2.1.1 are considered for the mathematical model of problem.

Figure 2.1. Simultaneous Resource-aware Cooperative Delivery Mission of an AGV and a UAV.

2.1.1. Rules, Assumptions & Constraints

Since the problem depends on various specifications of vehicles, environmental aspects, physical constraints and legal regulations, it is not feasible to solve the problem regarding all conditions. Correspondingly, some assumptions, rules and constraints are defined for the problem. Rules are;

(i) Location and weight information of all deliverables are known before the start of delivery mission.

(ii) The UAV takes-off at any delivery point or depot/warehouse.

(iii) The UAV has to fly at a constant altitude, Υ *meter*. Since the UAV has to fly at Υ *meter*, UAV takes off vertically up to Υ *meter* with a constant ascend speed and then flies throughout the delivery point. Finally, UAV lands vertically on the delivery point with a constant descent speed. UAV implements same procedure for return flight to AGV.

(iv) The AGV continues to move while UAV is taking-off.

(v) When UAV is launched from the AGV, it may rendezvous with the AGV at any any delivery point or depot/warehouse.

(vi) When the UAV and the AGV are scheduled to meet at a node (any delivery point or depot/warehouse), the one arriving earlier waits for the other vehicle.
(vii) UAV departs from Warehouse only once and arrives to Warehouse also only once.
(viii) Each of deliverables can be delivered by one visit of either the AGV or the UAV.
(ix) UAV is recharged on the truck instantaneously with a constant charging rate.

Addition to Rules, some Assumptions are made for the formulation of cooperative delivery mission planning for UAV and AGV as;

(i) The AGV travels with an average speed.
(ii) Both UAV and AGV stop at each delivery point for $\mathit{\Delta P}$ *second* to perform delivery.

For the mathematical model of cooperative delivery mission planning, some Constraints are defined as;

(i) UAV can deliver only a single deliverable per dispatch.
(ii) UAV charge status can not exceed maximum battery capacity.
(iii) The flight range of the UAV is limited by maximum battery capacity.
(iv) UAV is only capable to perform deliveries which does not exceed its current battery status.
(v) UAV can not carry a deliverable which exceed the payload weight limit of UAV.

2.2. Description and Visualization of an Expected Delivery Mission Planning and Timeline

Expected Cooperative Delivery Mission Scenario (ECDMS) is fictionalized based on hypothetical delivery points and weight values to illustrate the concept of the model. Examples based on real data and computed results are given in Section 3.

ECDMS starts with getting location data of $Warehouse$ and 10 delivery points, and weight data of deliverables, $\mathit{\Psi}(\chi(X, Y), W)$, which are listed in Table 2.1, and continues with the only AGV delivery case and cooperative delivery case which are illustrated in Figure 2.2 and 2.3(a), respectively.

Table 2.1. $\Psi_n(\chi_n(X_n, Y_n), W_n)$ values for ECDMS.

Delivery Point	X_n	Y_n	W_n
Warehouse	1000	2100	-
Delivery Point 1	3500	2550	11
Delivery Point 2	1000	500	9
Delivery Point 3	4750	500	2
Delivery Point 4	2200	2900	5
Delivery Point 5	690	4100	7
Delivery Point 6	3840	120	6
Delivery Point 7	1700	4500	1
Delivery Point 8	4000	3900	8
Delivery Point 9	500	2580	3
Delivery Point 10	2500	3700	7
DIM	**[m]**	**[m]**	**[kg]**

First of all, if there is no UAV, only AGV case is conducted. In this case, all deliverables are delivered to customers by AGV. No UAV is considered. Reason of generating only AGV case is to observe the benefits of using UAV by comparing only AGV and cooperative delivery mission planning cases. Generated route for AGV delivery is illustrated in Figure 2.2.

Secondly, cooperative delivery case is conducted. In this case, when UAV and AGV started to delivery, the decision about which one will perform delivery must be made. Main criteria of decision making is the payload limit of the UAV, ρ. Because no matter how efficient UAV it is, if $W_j > \rho \quad \forall j \in P$, then UAV can not perform delivery mission. Accordingly, the model focus on which deliverable are able to be carried by UAV, $\in P^U$, and which one is not, $\notin P^U$. In Table 2.2, proper deliverables of UAV,

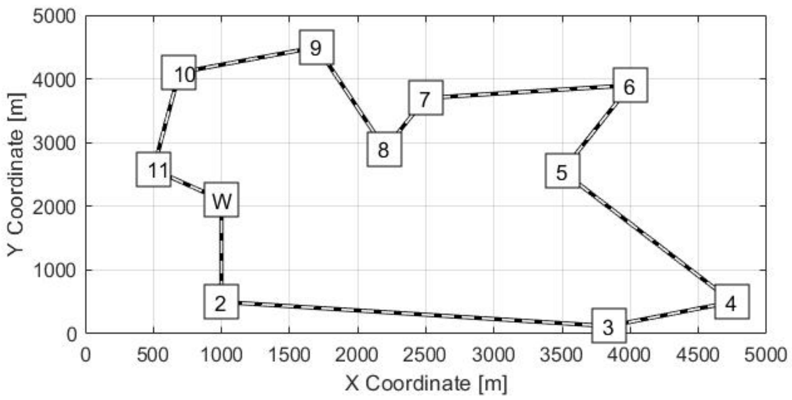


Figure 2.2. Expected Delivery Mission Planning only for AGV.

$W_j \leq \rho \quad \forall j \in P$, are given. $\rho = 4.5kg$ for ECDMS.

Table 2.2. Proper deliverables for UAV, $\Psi^U = \Psi_n(\chi_n(X_n, Y_n), W_n \leq \rho)$.

Delivery Point	X_n	Y_n	W_n	**Reordered Point # in Figure 2.3(a)**
Delivery Point 3	4750	500	2	iii
Delivery Point 7	1700	4500	1	ii
Delivery Point 9	500	2580	3	i
DIM	**[m]**	**[m]**	**[kg]**	[#]

On the other hand, the deliverables which are above UAV payload capacity have to be carried by AGV. In Table 2.3, mandatory deliverables for AGV, $W_j > \rho \quad \forall j \in P$, are listed. Since the solution aims to minimize delivery time of vehicles, minimum deliverable numbers for each vehicle lead us to minimum delivery time. Accordingly, route generation is handled as dividing the problem into two subproblem to ensure minimizing delivery time. First subproblem focuses on minimum delivery time of AGV regarding AGV delivers at least mandatory deliverables and the second subproblem focuses on minimum delivery time of UAV for UAV-proper deliverables in accordance with shortest AGV route.

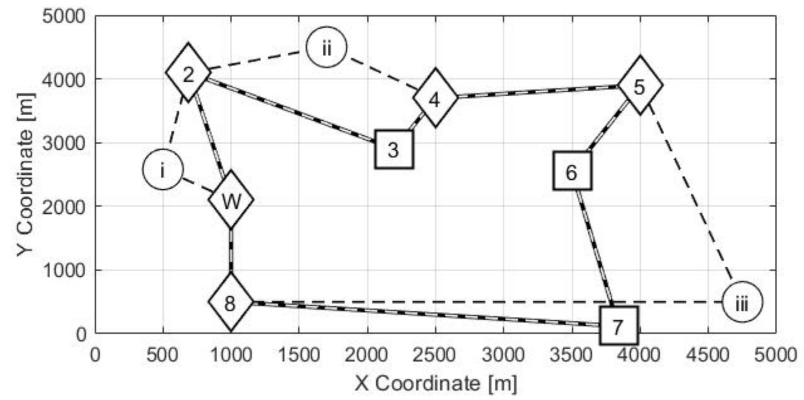
Table 2.3. Mandatory deliverables for AGV, $\Psi^T = \Psi_n(\chi_n(X_n, Y_n), W_n > \rho)$.

Delivery Point	X_n	Y_n	W_n	**Reordered Point # in Figure 2.3(a)**
Delivery Point 1	3500	2550	11	6
Delivery Point 2	1000	500	9	8
Delivery Point 4	2200	2900	5	3
Delivery Point 5	690	4100	7	2
Delivery Point 6	3840	120	6	7
Delivery Point 8	4000	3900	8	5
Delivery Point 10	2500	3700	7	4
DIM	**[m]**	**[m]**	**[kg]**	**[#]**

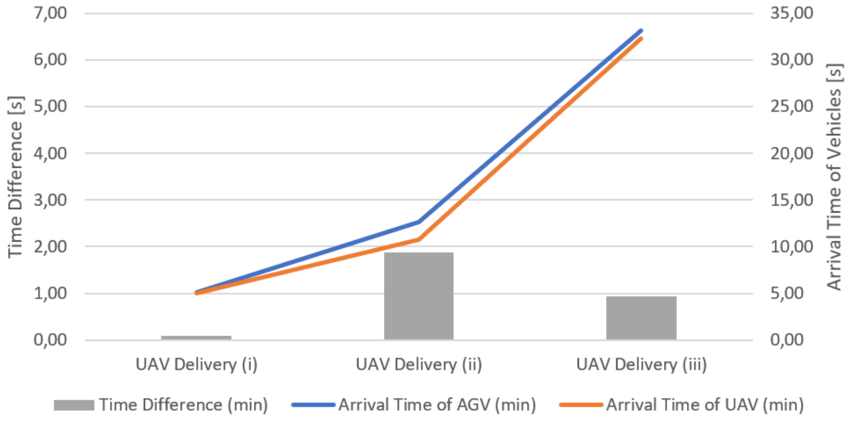
Solution of first subproblem generates the shortest delivery route for AGV regarding mandatory deliverables listed in Table 2.3 to ensure that minimum delivery time is satisfied. All delivery points are reordered arithmetically and first point is taken as $Warehouse$ and others are numbered in accordance with reorder. Generated shortest path solution for mandatory AGV deliveries is illustrated in Figure 2.3(a) as white striped black line with Arabic numeral.

Second subproblem solution finds the best sequential combination of UAV deliverable which satisfies all conditions such as delivery height, current battery status for each delivery, maximum flight distance etc. which is illustrated in Figure 2.3(a) as black striped line with Roman numeral. Then two found solutions are combined to obtain the minimized delivery time for cooperative delivery mission planning of UAV and AGV, illustrated in Figure 2.3(a).

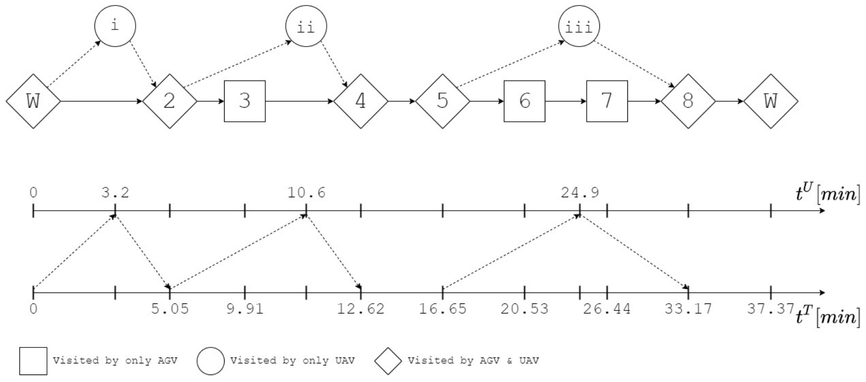
The model handles two subproblem in parallel but also considers effect of both subproblem to each other. Since UAV and AGV can be only split at delivery points and be united at delivery points, it is not always possible to reach same point at the same time. Model considers delay time of a vehicle and waiting time and conditions for the other one to find solution.



(a) Expected Cooperative Delivery Mission Planning of UAV and AGV.



(b) Time Differences between Arrival Time of UAV and AGV.



(c) Timeline of ECDMS in 1D.

Figure 2.3. Illustration of ECDMS.

In ECDMS, AGV and UAV start the delivery mission from Warehouse. First delivery which point i is conducted by UAV at $t_i^U = 3.2\ min$ and UAV turns back to

AGV on point 2 at $t_2^U = 4.96\ min$, however, AGV arrives point 2 at $t_2^T = 5.05\ min$ where t_2^U and t_2^T represents time of UAV and AGV at given point, respectively. As seen from arrival times, $t_2^T - t_2^U = 0.09\ min$ and UAV has to wait until AGV arrives to point 2. Time difference value of arrival time for UAV and AGV are given in Figure 2.3(b) and accordingly, timeline of ECDMS is illustrated as 1D in Figure 2.3(c). Recall that, battery consumption is a condition for UAV and if waiting time increases it causes more battery consumption for UAV and may cause to fail of performing all UAV delivery missions. In all sorties (UAV deliveries), UAV reaches reunion point before AGV and has to wait for arrival of AGV. If opposite situation occurs such as AGV reaches reunion point before UAV and waits till UAV came, then there is no negative effect on UAV for battery consumption but total delivery time increases since AGV stops moving to wait for UAV. The solution aims to minimize waiting and delay time of vehicles regarding battery status of UAV which is given in first row of Table 2.4.

Table 2.4. Battery Status of UAV in *min* at Reunion Points with and without Charging Capability.

β_f	β_0	β_2	β_4	β_8
β_f **with Charging**	18	13,99	9,49	0,32
β_f **w/out Charging**	18	13,99	9,49	-0,70
DIM	**[min]**	**[min]**	**[min]**	**[min]**

In ECDMS, UAV completes delivery mission with $\beta_8 = 0.32\ min$ on point 8. As noticed, remaining battery is too low and seems like UAV is almost failed to perform all deliveries. Recall that UAV is capable of charging while travelling on AGV. As seen in Figure 2.3(c), UAV is travelling on AGV from point 4 to point 5. During this time, battery of UAV is charged with a constant charging rate, κ. Also, UAV battery is charged between point 8 and *Warehouse*, but since all UAV deliverables are delivered before point 8, battery charging after point 8 does not make any sense for delivery mission planning. Since the β_8 is too low, 0.32 *min*, discussing the scenario, when UAV does not have charging capability, may lead to better understanding of charge effect. In the second row of Table 2.4, battery status of UAV at reunion points are given if UAV does not have charging capability and as noticed, battery status at point 8 is $\beta_8 = -0.7\ min$. Recall that UAV is capable of delivering if and only if the battery has enough capacity to perform delivery but negative value means UAV exceed current battery capacity and is unable to perform delivery then the ECDMS is failed. With charging capability, UAV charges its battery during travel between point 4 and point 5 and fulfills all delivery missions allocated to UAV.

2.3. Description of Mathematical Model

Mathematical model of the problem has been formulated as a modified version of the FSTSP model presented by Murray and Chu [50]. We cast the logic expression described in Section 2.1 and 2.1.1 into algebraic forms to be solvable by integer programming. Following equations present a cooperative delivery mission planning model consist of n delivery points, and point 0 and point $n+1$ refers to Warehouse as starting and finishing point of the route.

Cost function aims to minimize the time when the AGV is turned back to warehouse with UAV. Thus, cost function is defined as

$$min\ t^T_{n+1}. \tag{2.1}$$

The cost function is hereby equivalent to $min\{max\{t^T_{n+1}, t^U_{n+1}\}\}$, where t^T_{n+1} and t^U_{n+1} represents the time of AGV and UAV on point $n+1$, warehouse, respectively.

To ensure that each delivery point is visited only once by UAV or AGV according to Rule (7) and (8), $x_{i,j}$ and $y_{i,j,k}$ are specified as

$$\sum_{\substack{i \in P^d \\ i \neq j}} x_{i,j} + \sum_{\substack{i \in P^d \\ i \neq j}} \sum_{\substack{k \in P^a \\ <i,j,k> \in U}} y_{i,j,k} = 1, \tag{2.2}$$

where $\forall j \in P$. As described in Section 2.1, UAV and AGV start and end cooperative delivery mission together. Accordingly, start and end values for $x_{i,j}$ are defined as

$$\sum_{j \in P^a} x_{0,j} \quad = \quad 1 \tag{2.3}$$

$$\sum_{i \in P^d} x_{i,n+1} \quad = \quad 1 \tag{2.4}$$

to ensure simultaneous start and end. Addition to Equations (2.2), (2.3) and (2.4), sub-tour and loop of AGV are prevented to satisfy Rule (8) by

$$u_i - u_j + 1 \leq (n+2)(1 - x_{i,j}), \tag{2.5}$$

where $\forall i \in P,\ j \in P^a$ and $j \neq i$. On the other hand, route of AGV is desired to be continuous to satisfy uninterrupted loop for delivery mission. Accordingly, it is defined to provide continuous delivery of AGV, which necessitate that if AGV arrive to point j then AGV must depart from point j for next delivery, as

$$\sum_{\substack{i \in P^d \\ i \neq j}} x_{i,j} = \sum_{\substack{k \in P^a \\ j \neq k}} x_{j,k}, \tag{2.6}$$

where $\forall j \in P$. Addition to Equations (2.2) and (2.3), Rule (7) is satisfied by

$$\sum_{\substack{i \in P^U \\ j \neq i}} \sum_{\substack{k \in P^a \\ <i,j,k> \in U}} y_{i,j,k} \leq 1, \tag{2.7}$$

where $\forall i \in P^d$ by ensuring that UAV can depart from any point including warehouse at most once and it similarly satisfies Rule (7) with

$$\sum_{\substack{i \in P^d \\ i \neq k}} \sum_{\substack{j \in P^U \\ <i,j,k> \in U}} y_{i,j,k} \leq 1, \tag{2.8}$$

where $\forall k \in P^a$ by ensuring that UAV can arrive to any point including warehouse at most once. Rule (5) is satisfied by

$$2y_{i,j,k} \quad \leq \quad \sum_{\substack{h \in P^a \\ h \neq i}} x_{i,h} + \sum_{\substack{l \in P^d \\ l \neq k}} x_{l,k} \tag{2.9}$$

$$y_{0,j,k} \quad \leq \quad x_{h,k} \tag{2.10}$$

$$u_k \quad \geq \quad u_i + 1 - (n+2)\left(1 - \sum_{<i,j,k> \in U} y_{i,j,k}\right), \tag{2.11}$$

where $\forall i \in P^d,\ j \in P^U,\ k \in P^a$ and $k \neq i$. If UAV is departed from point i and arrive to AGV on point k, then AGV must visit both point i and k as provided by Equation (2.9). The special case is that if point i is the warehouse. Since AGV is already started from depot/warehouse, it is ensured that AGV visits point k to collect

UAV as provided by Equation (2.10). Since the AGV must visit point i and point k in sequence, Equation (2.11) ensures that AGV visits point i before point k.

For the synchronization of the time delay considered in Rule (6), following equations are defined as

$$t_i^U \geq t_i^T - \epsilon \left(1 - \sum_{\substack{j \in P^U \\ j \neq i}} \sum_{\substack{k \in P^a \\ <i,j,k> \in U}} y_{i,j,k} \right) \tag{2.12}$$

$$t_i^T \geq t_i^U - \epsilon \left(1 - \sum_{\substack{j \in P^U \\ j \neq i}} \sum_{\substack{k \in P^a \\ <i,j,k> \in U}} y_{i,j,k} \right) \tag{2.13}$$

$$t_k^U \geq t_k^T - \epsilon \left(1 - \sum_{\substack{i \in P^d \\ i \neq k}} \sum_{\substack{j \in P^U \\ <i,j,k> \in U}} y_{i,j,k} \right) \tag{2.14}$$

$$t_k^T \geq t_k^U - \epsilon \left(1 - \sum_{\substack{i \in P^d \\ i \neq k}} \sum_{\substack{j \in P^U \\ <i,j,k> \in U}} y_{i,j,k} \right), \tag{2.15}$$

where $\forall i \in P^d$ and $\forall k \in P^a$. Equations (2.12) and (2.13) ensure that UAV and AGV are time-coordinated when UAV is departed from any node i. Similarly, Equations (2.14) and (2.15) ensure that UAV and AGV are time-coordinated when UAV is arrived to any node k. Assumption (2) applies for both UAV and AGV. Operational time of the vehicles, $\mathit{\Delta} P$, is added to the delivery time of AGV as

$$t_k^T \geq t_h^T + \tau_{h,k}^T + \mathit{\Delta} P - \epsilon \left(1 - x_{h,k} \right), \tag{2.16}$$

where $\forall h \in P^d,\ k \in P^a$ and $k \neq h$. Differently from AGV, $\mathit{\Delta} P$ is applied for only going of UAV to delivery point. Thus, equations for UAV within the scope of Assumption (2) are defined as

$$t_j^U \ \geq \ t_i^U + \tau_{i,j}^U - \epsilon \left(1 - \sum_{\substack{k \in P^a \\ <i,j,k> \in U}} y_{i,j,k} \right) + \mathit{\Delta} P \tag{2.17}$$

$$t_k^U \geq t_j^U + \tau_{j,k}^U - \epsilon \left(1 - \sum_{\substack{i \in P^d \\ <i,j,k> \in U}} y_{i,j,k} \right), \tag{2.18}$$

where $\forall i \in P^d,\ j \in P^U,\ k \in P^a,\ i \neq j$ and $k \neq j$. Limitation of max. flight time of UAV, described in Constraint (2), is defined as

$$t_k^U \geq t_i^U + \zeta + \epsilon \left(1 - \sum_{\substack{j \in P^U \\ <i,j,k> \in U}} y_{i,j,k} \right), \tag{2.19}$$

where $\forall i \in P^d$ and $k \in P^a$. It is feasible if and only if UAV performs an delivery from point i to point j and turn back from point j to AGV on point k.

Independently from Assumptions, AGV needs to follow a route in a sequence. Accordingly, following equations are defined, to describe proper values for the sequence of points on AGV route, as

$$u_i \geq u_j + 1 - (n+2)p_{i,j} \tag{2.20}$$

$$u_i \leq u_j - 1 + (n+2)(1 - p_{i,j}) \tag{2.21}$$

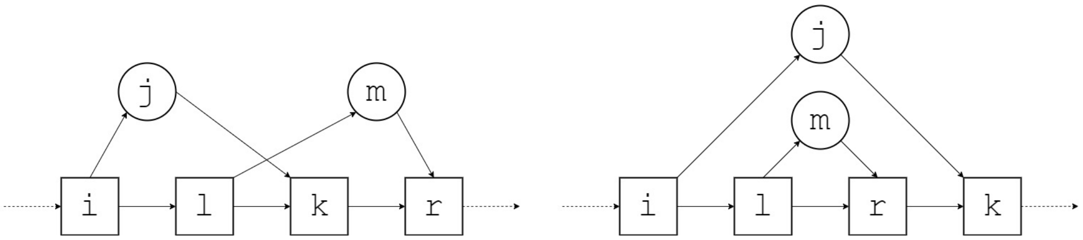
$$p_{i,j} + p_{j,i} = 1, \tag{2.22}$$

where $\forall i \in P,\ j \in P$ and $j \neq i$. Note that these equations are applicable for ordering of points visited by AGV only. The values are irrelevant for any point i and j which are visited by UAV only.

As described in Section 2.1, only one UAV is used for the model. Following equation is defined as

$$t_l^U \geq t_k^U - \epsilon \left(3 - \sum_{\substack{m \in P^U \\ m \neq i \\ m \neq k \\ m \neq l}} \sum_{\substack{r \in P^a \\ r \neq i \\ r \neq k \\ <l,m,r> \in U}} y_{l,m,r} - \sum_{\substack{j \in P^U \\ <i,j,k> \in U \\ j \neq l}} y_{i,j,k} - p_{i,l} \right), \tag{2.23}$$

where $\forall i \in P^d,\ k \in P^a,\ l \in P,\ k \neq i,\ l \neq i$ and $l \neq k$ to ensure that UAV conducts assigned delivery mission if and only if UAV is not conducting another mission, and to prevent multiple delivery possibilities. There are two prohibited flight scenarios of UAV in the model which are called as Θ_1 and Θ_2, and illustrated respectively in Figure 2.4(a) and Figure 2.4(b). Θ_1 occurs when UAV tries to depart from a point before it had arrived from former delivery mission.



(a) Prohibited Flight Scenario 1 of UAV (Θ_1). (b) Prohibited Flight Scenario 2 of UAV (Θ_2).

Figure 2.4. Illustration of Prohibited Flight Scenarios of UAV.

Θ_2 occurs when UAV tries to depart from a point and arrive to another point before it had arrived from former delivery mission. Both scenarios are prevented by means of parenthesis in Equation (2.23). For better understanding, UAV is considered as departing from AGV at point i and arrives to AGV at point k. Then, UAV departs again at point l and arrives to AGV at point r. That means AGV has to visit point i before point l, $p_{i,l} = 1$. $\sum\sum y_{l,m,r} = 0$ if the UAV tries to depart from point l before UAV has arrived to AGV at point k. If $\sum\sum y_{l,m,r} = 0$, that means inequality in Equation (2.23) is not satisfied.

Initial time values of UAV, t_0^U, and AGV, t_0^T, are defined as $t_0^T = 0$ and $t_0^U = 0$, respectively, to ensure that both UAV and AGV leave the warehouse and start to delivery mission in the same time.

$p_{0,j}$ is set as $p_{0,j} = 1$ where $\forall j \in P$ to ensure that AGV starts to delivery mission from Warehouse as described in Section 2.1. Time values of both UAV and AGV are constrained as

$$t_i^T \geq 0 \quad \forall i \in P^W \tag{2.24}$$

$$t_i^U \geq 0 \quad \forall i \in P^W \tag{2.25}$$

to ensure that both vehicles perform delivery. Initial battery status of UAV is set as $\beta_0 = \zeta$ to ensure that UAV starts to delivery mission with full battery capacity.

Current battery status of UAV is defined, regarding energy consumption of UAV and time consumed during delivery operation, $\mathit{\Delta P}$, to satisfy Constraint (4), as

$$\beta_k \geq \beta_i - \tau_{i,j}^U - \tau_{j,k}^U - \mathit{\Delta P} - \epsilon \left(1 - \sum_{\substack{j \in P^U \\ <i,j,k> \in U}} y_{i,j,k} \right), \tag{2.26}$$

where $\forall i \in P^d$, $k \in P^a$ and $j \neq i$. If battery status is not a positive value, it means that UAV is not able to perform the delivery, thus can not satisfy Constraint (4). Accordingly battery status of UAV is defined as

$$\beta_i \geq 0 \tag{2.27}$$

where $\forall i \in P^W$. Charging capability of UAV is modeled as

$$\beta_j \leq \beta_i + \kappa * \tau_{i,j}^T - \epsilon \left(3 - \sum_{\substack{i \in P^d \\ i \neq j}} x_{i,j} - p_{i,j} - q_{i,j} \right), \tag{2.28}$$

where $\forall j \in P^a$ to satisfy Constraint (2) and current battery status is modeled as

$$\beta_i = \zeta, \tag{2.29}$$

where $\forall i \in P^W$ and $\beta_i \geq \zeta$ to satisfy Constraint (3).

2.4. Differences from Original FSTSP Model

Presented model in this study differs from the original model presented by Murray and Chu [50]. Some of definitions and some of constraints are modified even new constraints and definitions have been added in the model. All major differences and additions are briefly explained in the following.

First modification was implemented to Equations (2.16), (2.17) and (2.18). Murray and Chu [50] consider SR and SL (package loading, battery changing, UAV recovery etc.) as additional consumed times and take them into account. Differently from this, there is no battery change or package loading concept in our model. Instead of SR and SL, $\mathit{\Delta} P$ was taken into account as the time consumed during delivery operation on delivery point. $\mathit{\Delta} P$ was added to t_k for AGV deliveries in Equation (2.16). In opposite to Murray and Chu [50], $\mathit{\Delta} P$ was added to Equation (2.17) as consumed time on delivery point for delivery operation. On the other hand, there is no additional time taken into account for Equation (2.18) since UAV is assumed as turning back to AGV and land on the vehicle without any time delay.

Second modification was applied to Equation (2.19), $t_j^U - \tau_{i,j}^U$ term was replaced with t_i^U under the inspiration from Ponza [65]. Main aim of replacing term is to simplify constraints and reduce variable numbers as much as possible to reduce computational time for large amount of delivery points.

Third modification is to add current battery status to the model. Murray and Chu [50] consider battery changing per each UAV delivery and the only constraint within the scope of battery is that battery capacity limits the maximum flight distance. In opposite to this, battery of UAV is not able to change but it is able to be charged during travelling on AGV with a constant charging rate, κ. Equation (2.26) defines battery consumption during delivery operation of UAV and on the other hand, Equation (2.28) defines battery status according to charging when UAV is located on AGV. Initial battery status, β_0, and Equations (2.27) and (2.29) are other related constraints and decision variables about battery status.

2.5. Further Considerations on Variables and Constants

$\mathit{\Delta} P$ is the time spent during delivery operation including dropping package and acceptance by the customer etc. It is assumed $\mathit{\Delta} P = 60s$ for the model. Since the distance between delivery points are assumed as air-distance and $\tau_{i,j}^T = \tau_{j,i}^T$, travel time of AGV, $\tau_{i,j}^T$, is calculated regarding the symmetrical distance matrix and average speed of AGV, v^T, is taken as 30 kph. Similarly, $\tau_{i,j}^U$ is calculated regarding same symmetrical distance matrix and velocity of UAV for horizontal travel in air, v^U, is taken as 60 kph. Both τ^U and τ^T are obtained from distance matrix, D. Travel time of AGV is calculated according to formula, $\tau^T_{i,j} = d_{i,j}/v^T$. Acceleration and deceleration effect of vehicle, traffic lights or road network effect are not considered in the model since such a precision about AGV is supposed to be unnecessary for this level of research. Definitely, more precise assumptions can be done but this level of approximation seems sufficient to create a model for FSTSP. On the other hand, the time calculation of UAV deliveries are more precise considering different take-off, landing and horizontal travelling speeds, effect of payload on speed.

If UAV is delivering a payload to delivery point, then delivery time of UAV from departure point to delivery point is calculated as

$$\tau^U_{i,j} = (\Upsilon/v^U_T + d_{i,j}/v^U + \Upsilon/v^U_L) * \lambda_j \tag{2.30}$$

regarding different speed for ascend, horizontal and descend movement of UAV. On the other hand, if UAV is turning back to AGV from delivery point, $\tau^U_{j,k}$ is defined as

$$\tau^U_{j,k} = \Upsilon/v^U_T + d_{j,k}/v^U + \Upsilon/v^U_L \tag{2.31}$$

since UAV is turning back to AGV without payload as defined in Rule (3). λ is a coefficient to represent payload effect on the UAV.

Since the speed of UAV is not same for different weights in real time, this effect was reflected to the model as

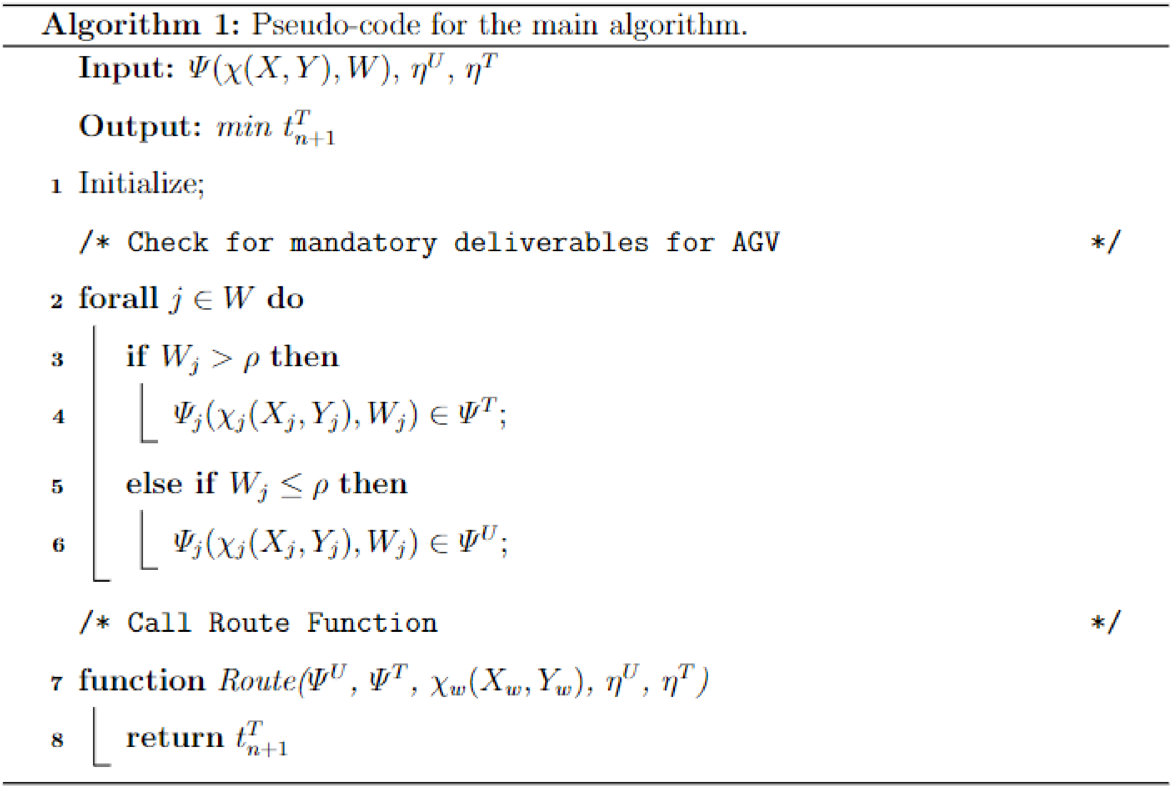
$$\lambda_j = \left(\frac{W_j + W_U}{W_U}\right)^{\frac{5}{3}} \tag{2.32}$$

to get more realistic results. λ_j is a varying coefficient depending on weight of deliverable which is W_j. The increase rate of λ is assumed as parabolic to obtain approximately double energy consumption of UAV when the payload exceeds half weight of UAV itself.

According to literature review in Section 1.1.1.2, battery capacity, ζ, is taken as 38 min which is a reasonable value since the values can vary from 10 to 55 min. Effectiveness of chosen ζ value is discussed in Section 3.4.

2.6. Solution via Dynamic Programming Algorithm

As the solution algorithm, dynamic programming and enumeration algorithms are used to solve the presented variant of FSTSP in Section 2.3 in accordance with flow diagram given in Figure B.1. Since FSTSP is a combined problem which includes AGV and UAV route inside, solution of the problem is divided into two main subproblem; shortest path for AGV route and shortest UAV delivery time regarding the AGV route. Accordingly, the problem is handled as a TSP first and the route is computed for AGV.



```
Algorithm 1: Pseudo-code for the main algorithm.
  Input: $\Psi(\chi(X,Y),W)$, $\eta^U$, $\eta^T$
  Output: $min\ t^T_{n+1}$
  Initialize;
  /* Check for mandatory deliverables for AGV                    */
  forall $j \in W$ do
      if $W_j > \rho$ then
          $\Psi_j(\chi_j(X_j,Y_j),W_j) \in \Psi^T$;
      else if $W_j \leq \rho$ then
          $\Psi_j(\chi_j(X_j,Y_j),W_j) \in \Psi^U$;
  /* Call Route Function                                          */
  function $Route(\Psi^U, \Psi^T, \chi_w(X_w,Y_w), \eta^U, \eta^T)$
      return $t^T_{n+1}$
```

Figure 2.5. Main Algorithm.

First, deliverables are classified as AGV deliverables and UAV deliverables regarding their weights according to Constraint (4) to find the shortest path for AGV.

Next, the *Route* function given as Algorithm 2 in Figure (2.7-2.12), generates a path for AGV regarding Ψ^U, Ψ^T, $\chi_w(X_w, Y_w)$, η^U, η^T and then, UAV route is calculated regarding generated AGV route. If the *Route* function finds a solution which satisfy all defined constraints and limitations then provides t^T_{n+1} as output which is the solution. Otherwise, the function provide a null array as output.

```
/* Check that Route function has found the solution */
if $t^T_{n+1} \neq []$ then
    return $t^T_{n+1}$ as $min\ t^T_{n+1}$
else if $t^T_{n+1} = []$ then
    /* Check solution with less number of UAVs */
    for $k \leftarrow 1$ to $h$ do
        /* $h$ is a parameter to tune permitted at least UAV numbers
           to plan cooperative delivery mission.  Equals to
           $(n^U - 2)$ as default. */
        forall $l \in perms(P^U, k)$ do
            $El = perms(P^U, k)$
            $\Psi^{U\prime} = \Psi^U - \Psi_{El}$;
            $\Psi^{T\prime} = \Psi^T + \Psi_{El}$;
            /* Call Route Function again */
            function $Route(\Psi^{U\prime}, \Psi^{T\prime}, \chi(X_w, Y_w), \eta^U, \eta^T)$
                return ${t^{T\prime}}_{n+1}$
            if ${t^{T\prime}}_{n+1} < {t^{T*}}_{n+1}$ then
                /* Update Best Delivery Time* with Delivery Time' */
                ${t^{T*}}_{n+1} = {t^{T\prime}}_{n+1}$;
    return ${t^{T*}}_{n+1}$ as $min\ t^T_{n+1}$
```

Figure 2.6. Main Algorithm. (cont.)

Algorithm then checks that *Route* function has found and provided a proper solution. If it is found then t^T_{n+1} equals to $min\ t^T_{n+1}$ as the exact solution. Otherwise, that means there is no solution that all UAV deliverables are delivered by UAV and then, algorithm starts to enumerate all the route possibilities to search for a proper solution with less UAV deliveries.

Beginning from Line 13 in Algorithm 1, all possible combination of UAV deliveries are started to be checked. ‘h’ is the lower limit of n^U for cooperative delivery mission. ‘h’ is discussed as an optimization parameter with details in Section 3.5.2. Algorithm eliminates one more UAV deliverable for each loop until it reaches the limit, ‘h’. For each elimination, all possible combination of rest of UAV deliverables are checked to find the solution. For each possibility, delivery points of UAV and AGV deliverables, and weights of UAV and AGV deliverables are updated as $\Psi^{U\prime}$ and $\Psi^{T\prime}$, respectively. *Route* function is called with the updated parameters to obtain a possible solution, ${t^{T\prime}}_{n+1}$. If the output is a better result than the current ‘Best’ solution, ${t^{T*}}_{n+1}$, then the ‘Best’ solution is updated with ${t^{T\prime}}_{n+1}$. This loop continues until the all possible delivery routes for UAV deliveries are checked and compared with each other. In the end of the loop, algorithm finds the solution, *min* t^T_{n+1}, as ${t^{T*}}_{n+1}$.

```
Algorithm 2: Pseudo-code for Route Function
    Data: $\Psi^U$, $\Psi^T$, $\chi_w(X_w, Y_w)$, $\eta^U$, $\eta^T$
    Result: $min\ t^T_{n+1}$
    Initialize;
    /* Create Distance Matrix, $D$                                  */
    forall $i \in \Psi^T$ do
        forall $j \in \Psi^T$ do
            $d_{i,j} \in D$
    /* First, create AGV route by DP Function                       */
    function $tsp\_dp1(\chi^T, D, \eta^T)$
        return $min\ \Delta L^T\ \&\ min\ \Delta T^T \in \tau^T$
```

Figure 2.7. Route Function.

Route function, which is given as Algorithm 2 in Figure (2.7-2.12), considers data such as Ψ^U, Ψ^T, $\chi_w(X_w, Y_w)$, η^U, η^T and provides $min\ t^T_{n+1}$ as output which is also the solution for the cooperative delivery mission planning for UAV and AGV. First of all, distance matrix, D is created which consist of distance between each AGV deliverable, $d_{i,j}$. Regarding to created D, χ^T and η^T, *tsp_dp*1 [66] function finds shortest route for AGV deliveries, ΔL^T, and the related route time, ΔT^T by dynamic programming method. *tsp_dp*1 is an open-source function published by Elad Kivelevitch on May 15, 2011 for MATLAB. *tsp_dp*1 is built by dynamic programming algorithm to find the shortest path for TSP. Input parameters of *tsp_dp*1 are distance matrix and delivery points, and output parameters are order and total distance of shortest route for the given delivery points. Since dynamic programming method divides the problem into 2^{n^T} subproblems, compared to enumeration method whose time complexity is $O(n^T!)$, dynamic programming has a time complexity of $O(2^{n^T}(n^T)^2)$ which provides much less computational times especially for high amount of AGV deliverables. According to the found route by *tsp_dp*1 function, algorithm starts to search for a proper UAV deliveries planning.

```
/* Find UAV tuples                                                    */
forall $i \in P^d$ do
    forall $k \in P^a$ do
        forall $j \in P^U$ do
            if $\tau^U_{i,j} + \tau^U_{j,k} \le \zeta$ then
                $< i, j, k > \in U$
/* Find Nearest $m$ Points of AGV Route to Each UAV Deliverable.
   Equals to 5 as default to decrease computational time.             */
forall $j \in P^U$ do
    $\sigma_j = min(D^U, m)$
```

Figure 2.8. Route Function. (cont.)

For the UAV deliveries, tuples for all UAV deliverables are found. A tuple is $< i, j, k >$ where $i \in P^d$ is departure point of UAV, $j \in P^U$ is UAV delivery point and $k \in P^a$ is arrival point of UAV. A tuple also has to satisfy battery constraint which is described in Equation (2.19), accordingly delivery time of UAV is also checked.

Normally, algorithm checks all AGV delivery points to make it a departure or arrival point for an UAV delivery but especially for far points, it is not feasible to use that point for delivery since it increases energy consumption of UAV and delivery time. To decrease computational time, unnecessary (far) points are eliminated by Line 12 and 13 of Algorithm 2 in Figure 2.8 as discussed in Section 3.5.1. 'm' is a parameter to tune numbers of nearest point of AGV route to each UAV. Decreasing of chosen number of nearest points decrease computational time and may cause to become distant from exact solution at the same time.

```
   /* Find all different route possibilities for UAV deliverables
      */
14 forall $j \in P^U$ do
15     forall $i \in P^d$ do
16         forall $k \in P^a$ do
               /* Prevent backward delivery                                    */
17             if $k > i$ then
18                 if $i \neq k \quad \forall i, k \in \sigma_j \ and < i, j, k > \in U$ then
19                     Set route as a possible UAV route, $PR$
```

Figure 2.9. Route Function. (cont.)

Thus, this parameter should be tuned well with respect to varying total number of delivery points. For the total delivery points of 10 to 15, $m = 5$. While finding nearest points of AGV for UAV deliveries, distance matrix of UAV deliverables, D^U is created first which consist of distance of each UAV deliverable to the each point on the AGV route. Accordingly, 'm' nearest points of D^U is chosen for each UAV deliverable. Effect of different values of 'm' is discussed in Section 3.5.1.

According to the found nearest points and tuples, possible UAV delivery routes are examined between Line 14 and 19 of Algorithm 2 in Figure 2.9. If departure point of UAV, i, precedes arrival point of UAV, k, and both i and k are member of found nearest points and the $< i, j, k >$ is a tuple, then the related UAV route is set as possible UAV route.

```
   /* Calculate delivery time of UAV for each UAV route
      possibility                                              */
20 function CostUAV($D^U$, $\Psi^U$, $\eta^U$)
21     forall Possible UAV Routes, PR do
22         Calculate $\tau^U_{i,j}$ & $\tau^U_{j,k} \in \tau^U$
```

Figure 2.10. Route Function. (cont.)

CostUAV function calculate the total delivery time, cost, of UAV for each possible UAV routes, PR, according to difference of travelling time of UAV described in Equations (2.17) and (2.18) with variables; D^U, Ψ^U, and η^U.

Clash function checks if any prohibited flight scenario of UAV occurred in the UAV route which is described as Θ_1 and Θ_2 in Figure 2.4(a) and Figure 2.4(b), respectively. Recall that the first prohibited flight scenario of UAV, Θ_1, is that when UAV tries to depart from a point before it had arrived from former delivery mission and the second prohibited flight scenario of UAV, Θ_2, is that when UAV try to depart from a point and arrive to another point before it had arrived from former delivery mission. Concurrently, battery status of UAV for the each step of UAV route, each UAV deliverable. If a negative β value is found or a Θ is occurred, then related possible

```
/* Check for $\Theta$ and battery status of UAV */
function Clash($\beta$, $PR$)
    forall $z \in PR$ do
        if $\beta_j < 0 \quad \forall j \in P^U$, or $\Theta_1$ or $\Theta_2$ occurred then
            Eliminate $PR_z$
    return Delivery time of UAV for each deliverable
```

Figure 2.11. Route Function. (cont.)

UAV route, PR_z is eliminated. The process is repeated for all possible routes of UAV and total delivery time of possible routes are obtained.

```
/* Synchronize Time of AGV and UAV on Reunion Points */
function SyncTime($t^U$, $t^T$, $\tau^U$, $\tau^T$, $PR$, $DR^T$)
    forall Possible Routes, $PR$ do
        forall $p \in DR^T$ do
            if $\tau_p^U > \tau_p^T$ then
                $t_p^T = t_p^U$
            else if $\tau_p^U < \tau_p^T$ then
                $t_p^U = t_p^T$
    return $t^U$ and $t^T$
return $t_{n+1}^T$
```

Figure 2.12. Route Function. (cont.)

SyncTime function synchronize the time of AGV and UAV for each step of cooperative delivery route, DR^T, regarding to Rule (6) as described in Equations (2.12-2.15). Time values of UAV, τ^U and AGV, τ^T, are compared to observe if there is any delay of a vehicle which causes other one to wait on reunion point, arrival point of UAV. If AGV reaches the reunion point before UAV, which means τ^U is greater than τ^T, then AGV has to be wait for UAV until it arrives and t^T is equalized to t^U. Vice versa, if UAV reaches the reunion point before AGV, which means τ^T is greater than τ^U, then UAV has to be wait for AGV until it arrives and t^U is equalized to t^T. This process is repeated for all possible route which are not eliminated. At the end of *Route* function, t^T_{n+1} is obtained as result. Since time of UAV and AGV is synchronized, it represents the cooperative delivery time of UAV and AGV at last point, warehouse.

3. RESULTS & DISCUSSIONS

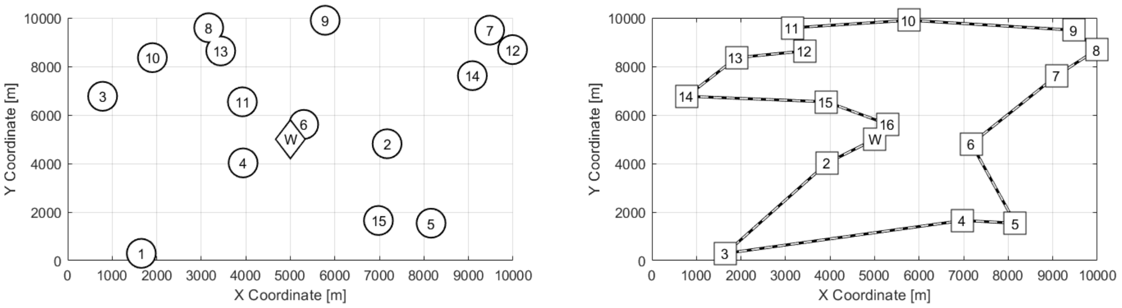
In this chapter, results of 140 different data sets, which is called instances hereafter, are investigated with respect to different tunable comparison parameters. Each instance consists of coordinates of delivery points and warehouse, weight value of each deliverable, battery capacity of UAV, ζ, payload limit of UAV, ρ, and charging rate of UAV, κ. In Section 3.1, cooperative delivery mission planning for UAV and AGV for Instance #140 is given. Section 3.2, contains comparison of computational time of enumeration and DP method for both cooperative and only AGV delivery planning. Effect of charging capability of UAV is investigated and results are given in Section 3.3. In Section 3.4, effect of battery capacity of UAV on cooperative delivery mission planning for UAV and AGV are investigated and results given.

Although main aim of the study is to find the exact solution, it is evaluated in Section 3.5 that what if some parameters are optimized to find a good enough result compared to the exact result for cooperative delivery mission planning. These parameters are the number of nearest points to UAV evaluated for UAV deliveries and number UAV deliveries in cooperative delivery mission planning.

3.1. Cooperative Delivery Route Planning for UAV and AGV

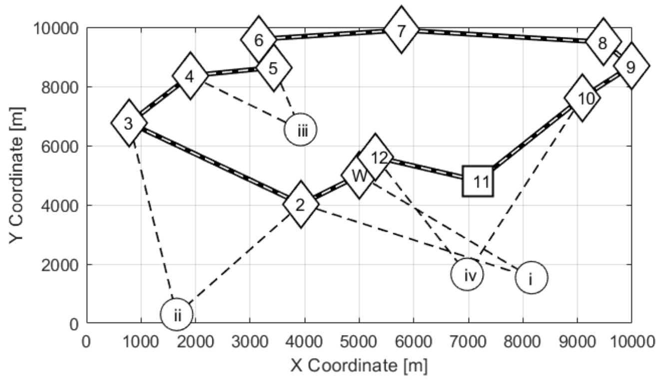
In this section, result of a sample cooperative delivery mission planning and its route visualization are given. For calculations, 5 nearest points and at least 3 UAV deliveries are considered as optimization parameters which are discussed in detail in Section 3.5.1 and 3.5.2, respectively.

In Figure 3.1(a), locations of delivery points and warehouse, $\chi(X,Y)$, are given. Regarding delivery points' coordinates and weights, $\Psi(\chi(X,Y),W)$, an only AGV deliver route is planned according to *tsp_dp*1 function in Algorithm 2 and the route is given Figure 3.1(b).



(a) Locations of Delivery Points and Warehouse for Instance #140.

(b) Only AGV Delivery Route Planning for Instance #140.



(c) Cooperative Delivery Route Planning for Instance #140.

Figure 3.1. Sub steps of Cooperative Delivery Mission Planning for Instance #140.

Then, a cooperative delivery route is planned. Resulting cooperative delivery mission and related route time (for cooperative delivery case), route time (for only AGV delivery case), β_f with charging capability and β_f without charging capability are given in Table 3.1. As seen in Table 3.1, route time of only AGV case is is approximately 19.5% more than cooperative delivery time. Cooperation between UAV and AGV decreases total delivery time. Moreover, β_f without charging capability value shows that if the UAV has no charging capability then it needs to have 46% more battery capacity to conduct the same cooperative delivery mission. The expected number of UAV deliveries is $n^U = 6$ since Instance #140 has 6 deliverables where $W_n \leq \rho$. Nevertheless, there is no possible route, which consists of 6 UAV deliveries, found by Algorithm 1. Then algorithm checks for less UAV deliveries and best solution is found as a cooperative delivery mission planning consisting of 11 AGV and 4 UAV deliveries. Cooperative delivery mission route is illustrated in Figure 3.1(c).

Table 3.1. Total Time of Cooperative Delivery Mission of UAV and AGV.

	n	n^T	n^U	**Only AGV Route Time**	**Coop. Route Time**	**β_f with Charging**	**β_f w/out Charging**
	15	11	4	91,01	76,12	0,11	-17,46
DIM	[-]	[-]	[-]	[min]	[min]	[min]	[min]

According to mathematical model expressed in Section 2.3, UAV and AGV have to wait each other on delivery point whichever arrive to the point first. This situation may cause delay of total cooperative delivery time. In Figure 3.2, time difference between UAV and AGV on reunion points are given. Positive time difference value means AGV delays and negative time difference value means UAV delays. Until both has arrived to reunion point, time is kept counting. Times of UAV and AGV are synchronized according to Rule (6) as described in the Line 28 – 35 of Algorithm 2 in Figure 2.12.

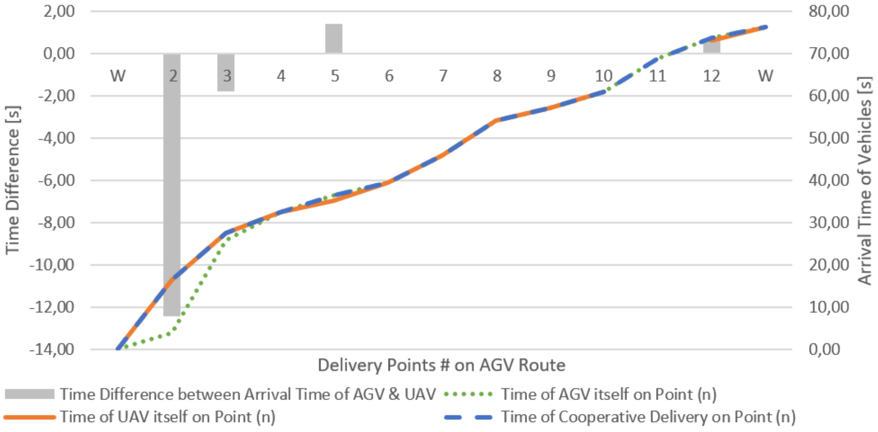


Figure 3.2. Time Differences between Arrival Time of UAV and AGV for Instance #140.

Recall that UAV is able to deliver if and only if its battery capacity is sufficient for delivery mission. In the last column of Table 3.1, it is seen that if UAV does not have charging capability then the delivery mission would be failed due to insufficient battery capacity. Details of battery status of UAV are given Table 3.2. As seen in Table 3.2, UAV charges itself between point 5 and point 10. Unless it is charged, remaining battery capacity on point 5, $2,60$ min, would not be sufficient to perform last UAV delivery mission. There is no β data for point 11 since UAV is performing its last UAV delivery mission while AGV is passing through point 11. No battery status data is obtained if UAV does not stop by related point.

Table 3.2. Battery Status Details of UAV.

Delivery Point (n)	**Time of Cooperative Delivery on Point (n)**	**Departure Point of UAV**	**Arrival Point of UAV**	β_n
W	0,00	Yes	No	38,00
2	16,32	Yes	Yes	21,68
3	27,49	No	Yes	7,70
4	32,38	Yes	No	10,64
5	36,50	No	Yes	2,60
6	39,48	No	No	4,39
7	45,75	No	No	6,18
8	54,18	No	No	7,97
9	57,09	No	No	9,76
10	60,89	Yes	No	11,56
11	68,69	No	No	-
12	73,76	No	Yes	0,11
W	76,12	No	No	1,53
DIM	[min]	[-]	[-]	[min]

Cooperative delivery route of UAV and AGV depends on varying constraint and limitations. Since most crucial constraint is battery specifications of UAV such as capacity and charge rate of battery for the presented model in this study, effect of different ζ and κ values for the battery is examined for Instance #140.

Table 3.3. Cooperative Delivery Route Time of Instance #140 with Varying Charging Rates.

Instance #	n	n^T	n^U	κ	**Coop. Route Time**	β_f **with Charging**	β_f **w/out Charging**
e15u6_5	15	11	4	0,0	81,18	3,26	3,26
	15	12	3	0,2	78,99	2,95	-2,89
	15	11	4	0,4	78,11	3,87	-11,58
	15	11	4	0,6	76,12	0,11	-17,46
	15	11	4	0,8	76,12	5,97	-17,46
DIM	[#]	[#]	[#]	[-]	[min]	[min]	[min]

First, κ is handled to observe effect on cooperative delivery route. Note that $\zeta = 38min$, nearest points parameter, $m = 5$ and at least UAV delivery parameter, $h = 3$, for the observation of κ case. Five different κ values which are 0, 0.2, 0.4, 0.6 and 0.8 are examined. As seen in Table 3.3, increasing κ value provides decreasing of total delivery time cooperative delivery route of UAV and AGV. Decreasing effect is

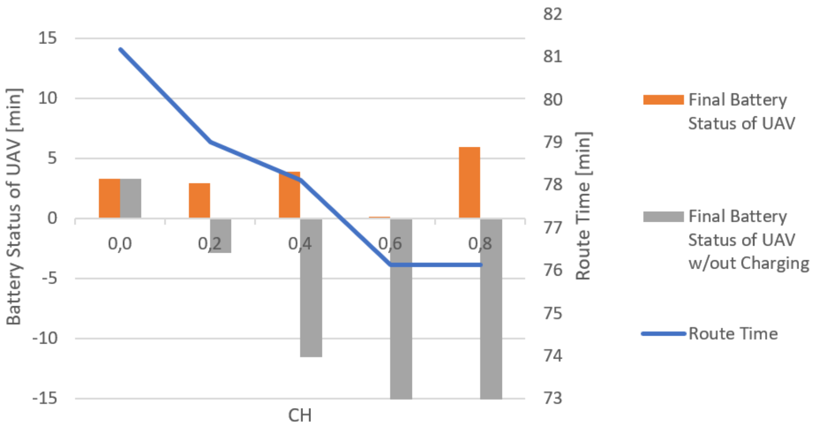


Figure 3.3. Cooperative Delivery Route Time of Instance #140 with Varying Charging Rates.

visualized in Figure 3.3. On the other hand, cooperative delivery times are same for $\kappa = 0.6$ *and* $\kappa = 0.8$. Increasing the value of κ by more than a certain value does not contribute to the solution. It shows that κ is not the only factor which constrain cooperative delivery time. Recall that κ presents the charging rate (gaining flight time) of UAV per minute. Increasing κ value to get it closer to 1 is not a realistic approach since the real-world applications do not have sufficient mobile charging infrastructure. Considering both examination values and real-world applications, it's obvious that taking κ value into account as nearly 0.6 is a sufficient choice for modelling cooperative delivery mission planning for UAV and AGV.

Table 3.4. Cooperative Delivery Route Time of Instance #140 with Varying Battery Capacities.

Instance #	n	n^T	n^U	ζ	**Route Time**	β_f **with Charging**	β_f **w/out Charging**
e15u6_5	15	11	4	28	80,07	4,06	-14,40
	15	12	3	33	78,99	9,63	-7,89
	15	11	4	38	76,12	0,11	-17,46
	15	11	4	43	76,12	5,11	-12,46
	15	10	5	53	75,81	1,78	-11,06
DIM	**[#]**	**[#]**	**[#]**	**[min]**	**[min]**	**[min]**	**[min]**

Second, ζ is handled to observe effect on cooperative delivery route. Note that $\kappa = 0.6$, nearest points parameter, $m = 5$ and at least UAV delivery parameter, $h = 3$ for the observation of ζ similar to κ case. Five different ζ values which are 28, 33, 38, 43 and 53 are examined. As seen in Table 3.4, increasing ζ value provides decreasing of total delivery time cooperative delivery route of UAV and AGV. Decreasing effect is visualized in Figure 3.4. On the other hand, cooperative delivery times are same for $\zeta = 38min\ and \zeta = 43min$ moreover the time for $\zeta = 53min$, $75,81min$, is decreased only for 0.4%. Increasing the value of ζ by more than a certain value does not contribute to the solution that much. It shows that ζ is not the only factor which constrain

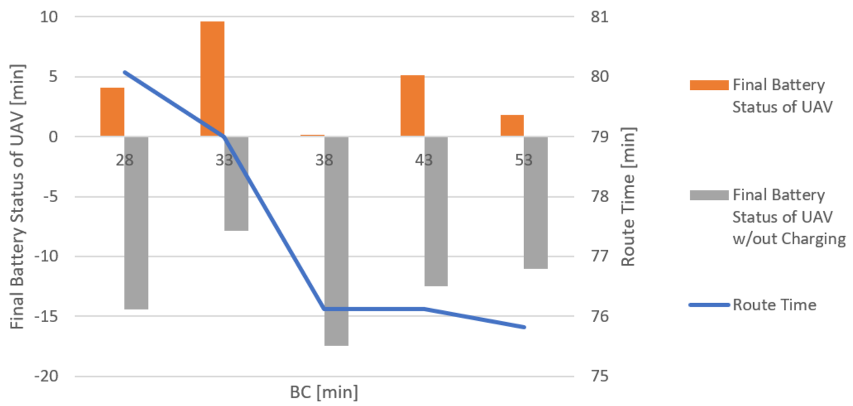


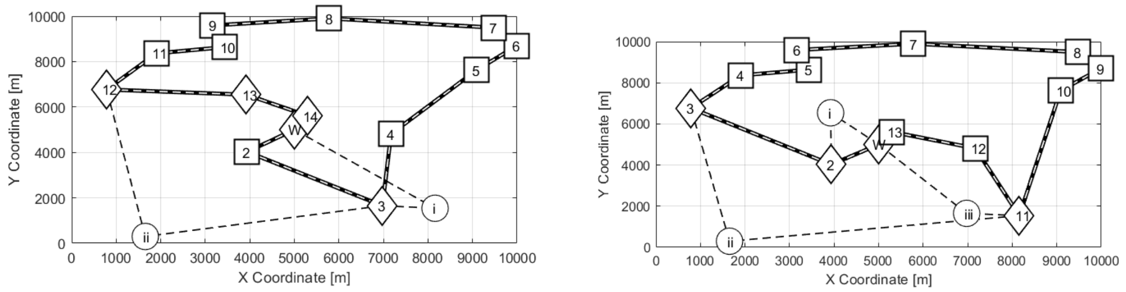
Figure 3.4. Cooperative Delivery Route Time of Instance #140 with Varying Battery Capacities.

cooperative delivery time similar to κ factor. Recall that increasing ζ value means increasing weight of battery and energy consumption of UAV relatively in real-world applications. Considering both examination values and real-world applications, it's obvious that taking ζ value into account as around $38min$ is a sufficient choice for modelling cooperative delivery mission planning for UAV and AGV.

Table 3.5. Cooperative Delivery Route Time of Instance #140 with Varying UAV Deliveries.

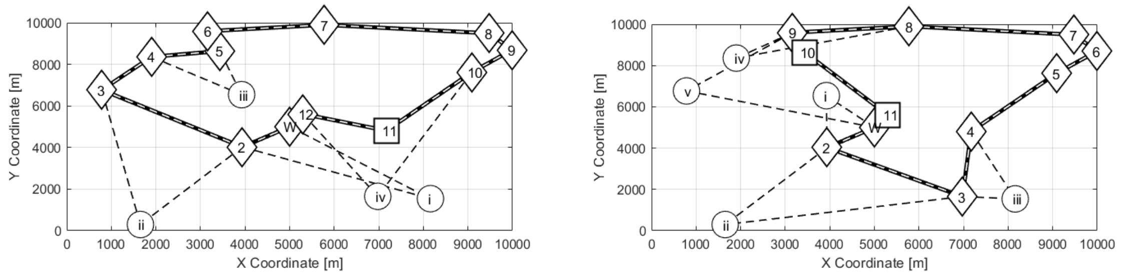
Instance #	n	n^T	n^U	**Route Time**	β_f **with Charging**	β_f **w/out Charging**
e15u6_5	15	13	2	75,13	0,41	0,41
	15	12	3	78,10	0,20	-5,42
	15	11	4	76,12	0,11	-17,46
	15	10	5	81,62	0,08	-13,69
DIM	[#]	[#]	[#]	**[min]**	**[min]**	**[min]**

As seen in Figure 3.1(c), cooperative delivery route for Instance #140 have only 4 UAV deliveries not 6 UAV deliveries as expected. Main reason of this situation is



(a) Cooperative Delivery Mission Planning for Instance #140 with 2 UAV Deliveries.

(b) Cooperative Delivery Mission Planning for Instance #140 with 3 UAV Deliveries.



(c) Cooperative Delivery Mission Planning for Instance #140 with 4 UAV Deliveries.

(d) Cooperative Delivery Mission Planning for Instance #140 with 5 UAV Deliveries.

Figure 3.5. Cooperative Delivery Mission Planning for Instance #140 with Varying UAV Deliveries.

that there is no proper UAV route for 6 UAV deliveries and algorithm finds a solution with less UAV deliveries as explained in Algorithm 1. Moreover, the solution does not consist of 5 UAV deliveries either. To observe the logic behind it, cases of different UAV deliveries for Instance #140 are examined and cooperative route time and final battery status of UAV are given in Table 3.5. Furthermore, cooperative mission route for each case is illustrated in Figure 3.5, respectively. As seen in Table 3.5, route time varies for each case. There is no linear connection between cooperative route time and performed UAV delivery number. Since the algorithm focuses on minimum cooperative delivery time, the examination results shows that it is possible to find the best solution as a case with less or more UAV deliveries. Note that, minimum cooperative delivery time according to data in Table 3.5 is $75,13min$ for 2 UAV deliveries case unlike the data given in Table 3.1. Recall that optimization parameter, at least ‘h’ UAV, is assumed as 3 for the calculations of Instance #140. Consequently, none of 2 UAV deliveries cases is evaluated for the solution. Main reason of this elimination is discussed in detail in Section 3.5.2.

3.2. Computational Time of Enum. and DP Methods

3.2.1. Only AGV Delivery Mission Planning

Computational time values of both Enumeration and Dynamic Programming methods considering only AGV delivery cases for 30 data sets chosen from Instance # 1-81 are given in Figure 3.6. Detailed computational time results of these data sets (Instance #1-81) according to enumeration and DP methods for only AGV cases are given in Table C.1. As seen, computational time are similar to each other for 8 and 9 delivery points even enumeration method times are less than DP method times for some instances which have 8 delivery points. Main reason of this, DP firstly divide the problem into subproblems and then calculate minimum delivery time. But in opposite, enumeration method starts directly to calculate each route possibility to find minimum delivery time. For small amount of delivery points, enumeration methods could be sufficient to be used.

On the other hand, especially beginning from instance 21, gap between computational times of enumeration method and DP method starts to widen exponentially by increasing each delivery point. Since computational times are limited by the capacity of computer, if the iteration numbers exceed the capacity of computer, it gives error for computing. Total iteration number is equal to $12! = 479.001.600$ (almost 480 million) for 12 delivery points which exceeds the capacity of computer. Accordingly, enumeration method results for the instances which contain 12 and more delivery points could not be obtained.

However, there is a linearity between computational time for enumeration method and number of delivery points, since the method computes all the route possibilities one by one in order, it is possible to hypothetically find the computational time values for enumeration method, taking into account this linearity. Hypothetical results are given

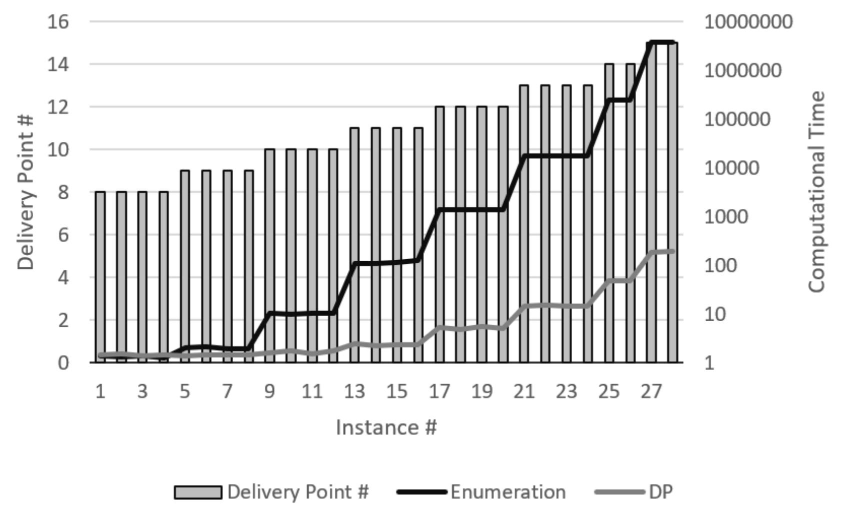


Figure 3.6. Computational Time Values for Only AGV Delivery Cases.

in Table 3.6. For the instance 81, hypothetical computational time of enumeration method is almost equal to 32 years. Since this value is not feasible for a computation, enumeration method becomes inefficient for more than 11 delivery points.

3.2.2. Cooperative Delivery Mission Planning

Computational time of Cooperative Delivery Mission Planning for UAV and AGV regarding all route possibilities are given in Figure 3.7 for both Enumeration and Dynamic Programming Methods. In the figure, results of 60 data sets (Instance #1-60) are visualized to observe increase of computational time of delivery mission planning with respect to increasing number of delivery points. Detailed computational time results of 81 data sets (Instance #1-81) according to enumeration and DP methods are given in Table D.1. Computational time for enumeration method for Instance #61-81 could not obtained since the computation exceeds memory capacity of the computer. There are five instance for each case to obtain average computational time since all delivery points and weights have been created randomly. Recall that Algorithm 1 firstly checks if there is a solution which all the the UAV deliverables are delivered by UAV. For initial samples, UAV deliverable numbers are selected relatively less according to total delivery points. That means algorithm catches a solution which all UAV deliverables

Table 3.6. Hypothetical Computational Time Values for only AGV Delivery Cases Exceeding Computer Capacity.

#	Instance #	n	Enumeration Time	DP Time
41	c12u4_1	12	1356,00	5,21
42 **⋮** **49**	c12u4_2 ⋮ c12u5_4	12 ⋮ 12	1356,00 ⋮ 1356,00	4,86 ⋮ 5,58
50	c12u5_5	12	1356,00	5,10
51	c13u5_1	13	17628,00	14,19
52 **⋮** **59**	c13u5_2 ⋮ c13u6_4	13 ⋮ 13	17628,00 ⋮ 17628,00	14,91 ⋮ 14,50
60	c13u6_5	13	17628,00	14,25
61 **⋮** **70**	c14u5_1 ⋮ c14u6_5	14 ⋮ 14	246792,00 ⋮ 246792,00	48,24 ⋮ 48,93
71 **⋮** **80**	c15u6_1 ⋮ c15u7_5	15 ⋮ 15	3701880,00 ⋮ 3701880,00	184,09 ⋮ 193,60
81	c17u7_1	17	1006911360,00	6173,53
DIM	#	#	[s]	[s]

are delivered by UAV at first try and computational time severely decreases since the algorithm does not try alternative solutions with less UAV deliverables delivered by UAV. Effect of less and more number of UAV deliverables for same number of delivery points are examined in Section 3.5.2.

Computational time for both Enumeration and DP methods are pretty close to each other up to 55th instance as seen in Figure 3.7. Beginning from instance 56, computational time gap between Enumeration and DP widens. Algorithm computes every

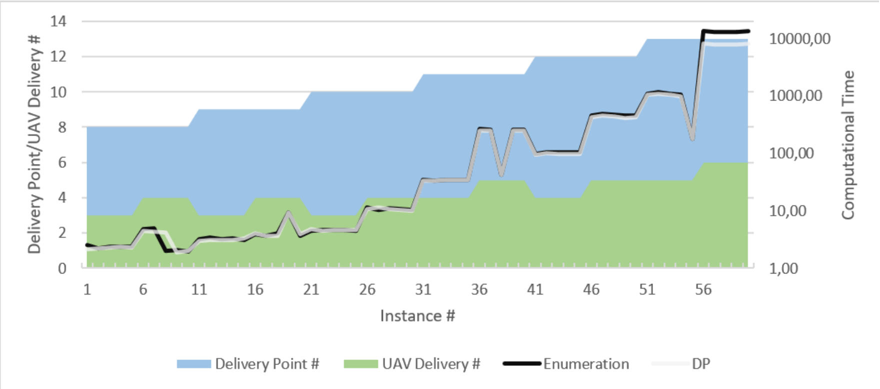


Figure 3.7. Computational Time Values for Cooperative Delivery Mission Planning.

possible route if the route planning contains at least two UAV deliveries. Accordingly, up to instance 55, both enumeration and DP method considers at most 10 AGV deliveries and 2 UAV deliveries. Since route possibilities depends on permutation of 'n' points, 10 AGV delivery points require 10! = 3628800 iterations to be solved. Thanks to memorization of distance between delivery points, enumeration method is able to solve iterations numbers under 10 million with similar time with DP. As known, DP divides problems into subproblems and while solving subproblems it also uses memorization. On the other hand, for the iterations numbers above 10 million, Enumeration method is not sufficient as much as DP method. Within this scope, beginning from instance 56, computational time differences can be seen in Figure 3.7. For instance 56, algorithm checks up to 11 AGV deliverable and 2 UAV deliverable option to find the best solution which gives minimum total delivery time. For solution, 11! = 39.916.800 iterations should be performed by computer. Since computing almost 40 million iterations and holding the results in memory is very challenging for an average computer, computational time starts to be longer and longer with increasing deliverable numbers. Advantage of using DP is being noticed after 10 AGV delivery points. DP method does not memorize worse solution and keeps searching for the best solution. In this way, DP is able to find the best AGV route with minimum delivery time under an acceptable time.

3.3. Effect of Charging Capability of UAV to Cooperative Delivery Mission Planning

Effect of charging capability of UAV is investigated for 50 data sets (Instances #1-50). For each data set, cooperative delivery time and β_f are examined with charging capability ($\kappa = 0.6$) and without charging capability ($\kappa = 0$). The time for if only AGV performs all deliveries, cooperative delivery time, and battery status of UAV after final UAV delivery are given in Figure 3.8. Detailed computational results of 50 data sets (Instance #1-50) according to charging capability are given in Table E.1. According to results, 74% of instances have less cooperative delivery time compared to only AGV delivery case. That shows that cooperative delivery mission planning, especially for higher amount of delivery points, provides less delivery time by allocating each deliverable to proper vehicle. Recall that algorithm checks if all of desired UAV deliverables are able to delivered by UAV then it does not check any other alternative cooperative delivery possibility. It is possible to obtain less cooperative delivery time for same instance but this scenario is discussed in Section 3.5.2.

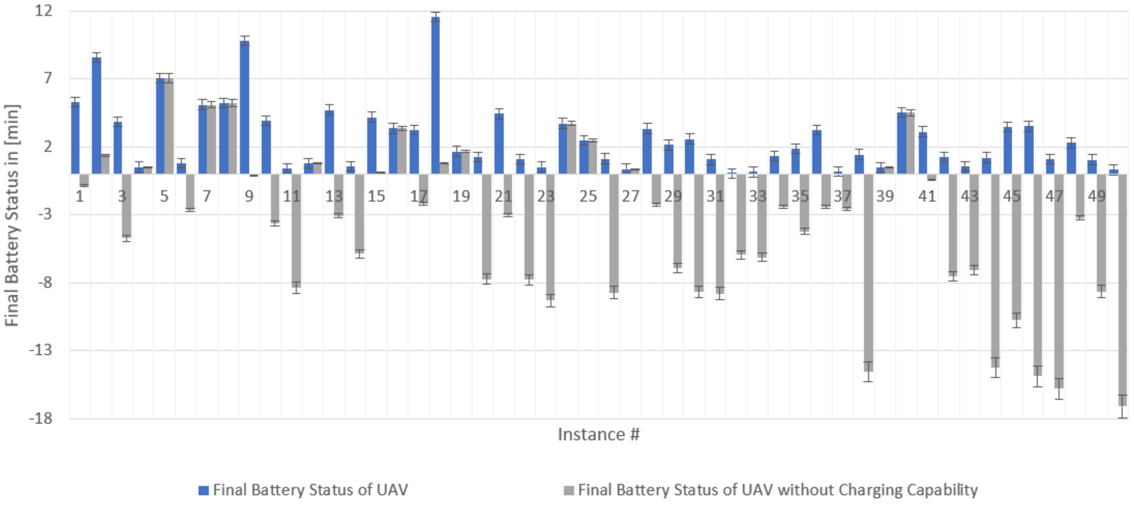
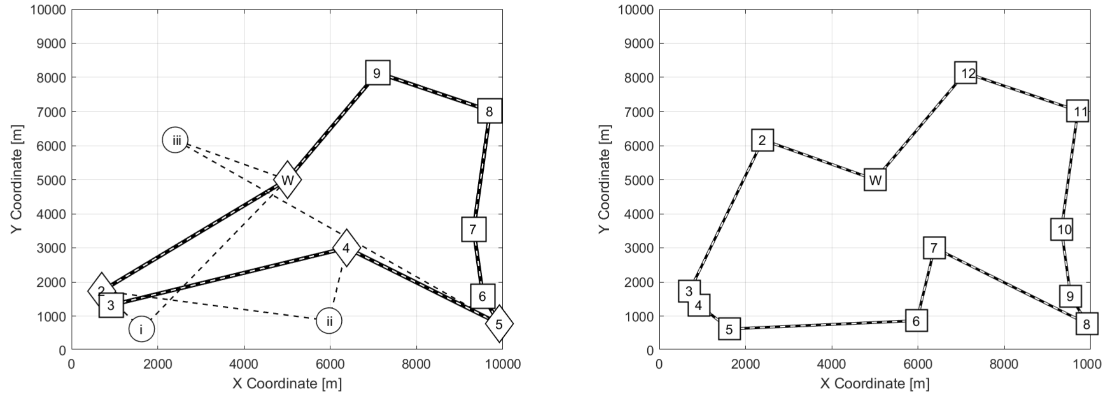


Figure 3.8. Effect of Charging Capability to Cooperative Delivery Mission Planning.

Charge status represents the current flyable time of UAV and if the value is positive UAV have that amount of time in [min] for flight but if the value is negative then UAV is not able to flight any more and the found delivery planning for the related instance is failed. According to results in Table E.1 which are illustrated in Figure 3.8, 58% of instances is not capable to perform found delivery route planning without charging capability since the final battery status values are negative. For the model, charging rate of UAV is assumed as $\kappa = 0.6$, and it is a controllable parameter by user in the algorithm. Charging rate means that UAV charges itself and gains flyable time of κ times travelling time on AGV. For better understanding of the effect of charging capability of UAV, charge status details of instance #36, $c11u5\_1$, are given in Figure 3.7, related cooperative route planning is given in Figure 3.9(a) and related only AGV delivery route planning is given in Figure 3.9(b).



(a) Cooperative Delivery Route Planning for Instance #36.

(b) Only AGV Delivery Route Planning for Instance #36.

Figure 3.9. Cooperative and Only AGV Delivery Route Planning for Instance #36.

As seen in Table 3.7, UAV starts to delivery mission with full battery capacity, ζ, and departs from Warehouse to deliver first UAV deliverable and lands on AGV at point 2. After that, UAV departs again from point 2 to perform second UAV delivery and lands on point 4. During the second delivery, UAV skips point 3, accordingly battery status for point 3 is not applicable. (Thus remarked as not a number, -). After second delivery UAV travels on AGV from point 4 to point 5 and it is charged and gains $0,6 * (35,55 - 26,16) = 5,63$ min of flyable time during this travel. In addition to this, UAV departs from point 5 to perform third and the last UAV delivery and lands on AGV at point W, Warehouse. Note that UAV spends $17,94$ min of battery capacity for the last delivery and if it is not charged during the travel between point 4 and point 5, it will not be able to perform last delivery since $15,55$ min of battery capacity is not enough for necessary flyable time.

Table 3.7. Charge Status Details of Instance #36.

Delivery Point (n)	**Time of Cooperative Delivery on Point (n)**	**Departure Point of UAV**	**Arrival Point of UAV**	β_n
W	0,00	Yes	No	38,00
2	11,79	Yes	Yes	27,58
3	13,74	No	No	-
4	26,16	No	Yes	15,55
5	35,55	Yes	No	21,18
6	38,36	No	No	-
7	43,31	No	No	-
8	51,30	No	No	-
9	57,98	No	No	-
W	66,52	No	Yes	3,24
DIM	**[min]**	**[-]**	**[-]**	**[min]**

In summary, charging capability of UAV has a major effect on cooperative delivery mission planning for UAV and AGV since UAV has to turn back to AGV for next delivery and it is possible to spending time on AGV for next delivery. It is mathematically logical to take advantage of this duration and charge UAV.

On the other hand, charging rate value is also important. For example, if the UAV has a less charging rate such as $\kappa = 0,2$ instead of $\kappa = 0,6$, the battery status on point 5 is calculated as

$$15,55 + 0,20 * (35,55 - 26,16) = 17,43min \tag{3.1}$$

and the cooperative delivery mission planning could be failed once again since the calculated battery status of UAV for $\kappa = 0, 2$ is under necessary flyable time, $17, 94\ min$.

Increasing charging rate may also provide less battery capacities since battery is capable to refill itself in short times. Less battery capacities means less battery weights and recall that less weight provides longer flyable time by decreasing energy consumption depending on decreasing weight of UAV.

Effect of decreasing charging rate of UAV to cooperative delivery mission planning are studied on 50 data sets and results are discussed in Section 3.3.1.

3.3.1. Effect of Charging Rate of UAV to Cooperative Delivery Mission Planning

Instance 1-50 are investigated within the scope of three different charging rate cases; $\kappa = 0.6$, $\kappa = 0.2$ and No κ (which means UAV does not have charging capability). Payload limit, ρ and battery capacity, ζ are same for each instance. Results are given in Figure 3.10.

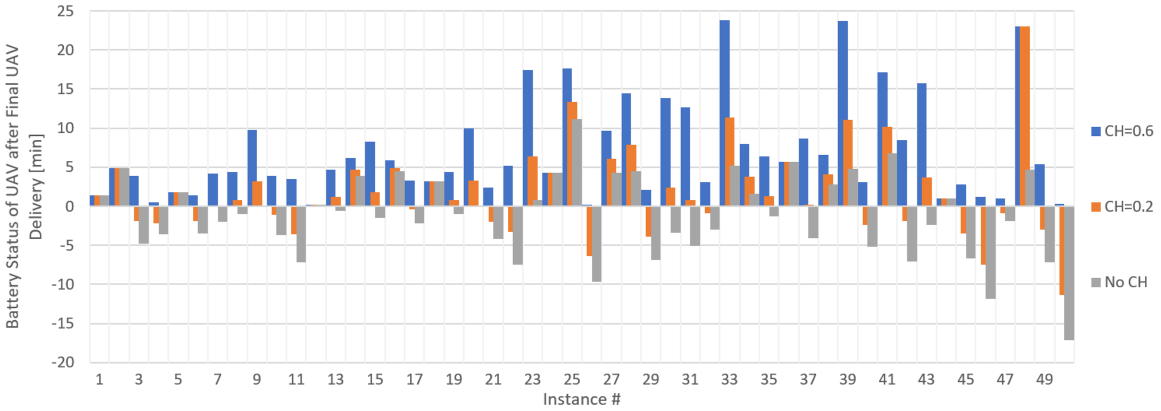


Figure 3.10. Effect of Different Charging Rate of UAV to Cooperative Delivery Mission Planning.

As seen in Figure 3.10, value of final battery status of UAV, decreases as κ decreases or stays same if the battery is not charged during delivery mission. The final battery status of UAV with $\kappa = 0.6$ is taken as reference values for comparison with $\kappa = 0.2$ and $\kappa = 0$. 36% of instances fails if the κ value is dropped to 0.2 from 0.6. Even if κ value is dropped to 0, which means UAV loses its charging capability, then 58% of instances fails since battery status of UAV falls below 0 as mentioned in Section 3.3. Note that nearest 4 points case mentioned in Section 3.5.1 and at least 3 UAV case mentioned in Section 3.5.2 are assumed for κ value comparison to work with less computational times. Therefore, final battery status for some instances differs from the values in Figure 3.8 because none of optimization cases is implemented on computations held for Section 3.3.

3.4. Effect of Battery Capacity of UAV

Since UAV flight range is limited by battery capacity, ζ, it is important to choose an optimal battery capacity for UAV. Delivery area is limited to an area of 10000 square meters. Accordingly, battery capacity of UAV is expected enough for UAV to perform UAV deliveries. If the battery capacity is too low then less deliveries will be held by UAV and cooperative delivery time is expected to be more. In opposite, if the battery capacity is too high then UAV will be able to held more UAV deliveries. Note that battery capacity is just a parameter for theoretical studies but in real world applications, more battery capacity are accompanied by more weight on UAV. Weight effect of varying battery capacities is neglected in this study. To observe the effect of battery capacity to cooperative delivery route time, two different battery capacities, 33 min and 38 min, have been evaluated for 35 data sets (Instance #106-140) and change of cooperative delivery time is given in Figure 3.11. Detailed computational results of 35 data sets (Instance #106-140) according to varying ζ value are given in Table F.1.

According to results, cooperative route time decreases in average with increasing battery capacity as expected but the decrease rate is very low which is around $2-3\%$ regarding an 15% capacity increase of battery. It shows that increasing battery capacity

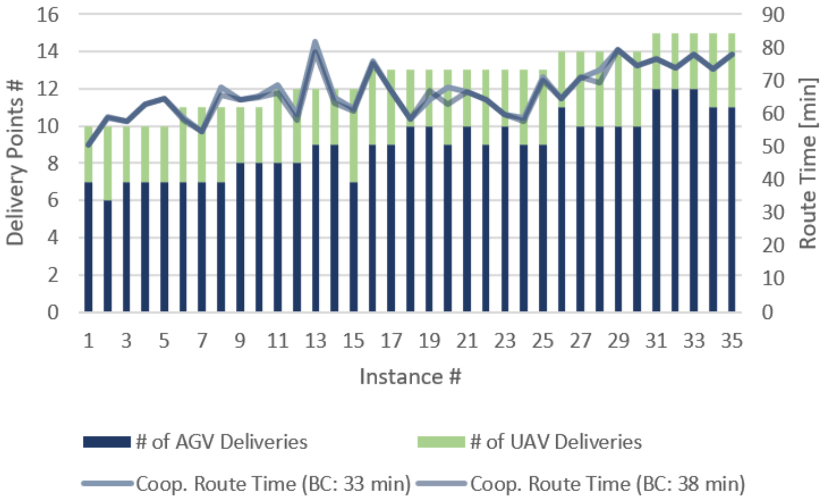


Figure 3.11. Effect of Battery Capacity of UAV to Cooperative Delivery Mission Planning.

more than necessary does not lead to decrease cooperative delivery time with same rate. Note that weight effect of battery capacity is neglected. If it is also included in the calculation, the results are expected to be even lower since increasing weight causes more energy consumption which means less flyable time for UAV with same capacity of battery.

3.5. Optimization of Decision Parameters

The main purpose of this study is to investigate the advantage of dynamic programming, herein after DP, to find exact solution for a modified FSTSP. Recall that, in Section 3.2, computational time of DP decreases compared to Enumeration method as the number of delivery points increase. On the other hand, increasing number of delivery points causes too long computational times since algorithm try to evaluate all the route possibilities. To decrease computational time and to observe efficiency of DP for high number of delivery points, optimization of some major decision parameters have been evaluated and the effects of them on the cooperative delivery mission planning have been studied under Subsections 3.5.1 and 3.5.2.

3.5.1. Consideration of 'm' Nearest Points of AGV Route to UAV

Algorithm takes at least '$(n^U - 2)$' UAV deliveries into account for a cooperative delivery mission planning as mentioned in Section 3.2.2. For each UAV deliverable, all the points on AGV route is evaluated as a departure and arrival points. Indeed, evaluating both nearest point to a deliverable and farthest point to same deliverable is not mathematically logical, it is just waste of computational time. To decrease computational time, far points to a UAV deliverable are eliminated. Since some points are out of evaluation, it is possible to miss some route alternatives which have less cooperative delivery time. To observe the effect of elimination of far points, cases which evaluates 4 of nearest points to UAV and 5 of nearest points to UAV are studied and cooperative delivery time and computational time of different cases are listed in Figure 3.12.

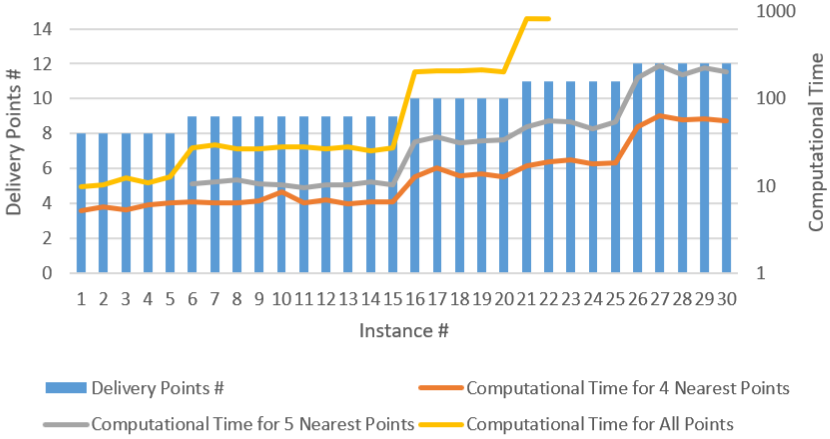


Figure 3.12. Effect of Considering 'm' Nearest Points to UAV.

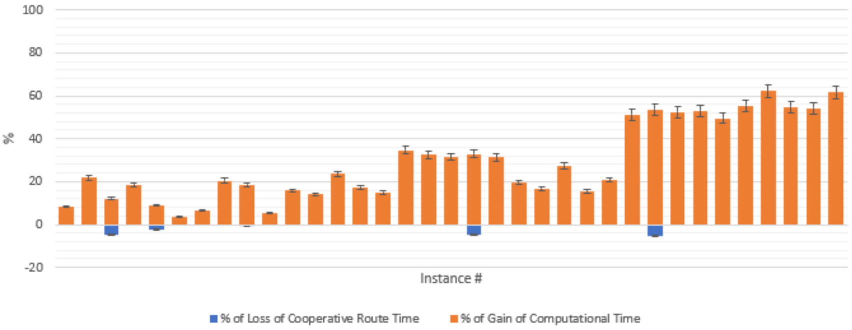
Detailed computational results of 30 data sets (Instance #91-120) according to varying 'm' value are given in Table G.1. Starting from Instance #95, it is obviously seen that decreasing number of evaluated nearest points to an UAV deliverable does not cause major time difference of cooperative delivery route time. On the other hand, decreasing number of evaluated nearest points to an UAV deliverable provides crucial decrease of computational time. For 9 delivery points, computational time reduced to about a quarter for 4 nearest points case. Furthermore, for 11 delivery points, computational time decreases to one in fifty for 4 nearest case.

There are some instances which 4 or 5 nearest points cases have less cooperative delivery route than all points case. It is expected that 4 or 5 nearest points cases should not have less time because if there is a better solution, algorithm should have already found the better solution. Main reason behind this confliction is about symmetrical route planning. Since route is being calculated regarding distances are air-distance and going to and turning back from a point is same distance and time for AGV, there are always two shortest path for AGV. One is clockwise and the other one is counter clockwise. Since tsp_dp1 function does not differenciate this situation, the function gives randomly the shortest path for different computations. Cooperative delivery time may change according to different direction of the route because UAV speed varies for going to and turning back from a point. Payload weight effects UAV speed and correspondingly delivery time. Therefore, total cooperative delivery time differs for clockwise and counterclockwise direction of same AGV route.

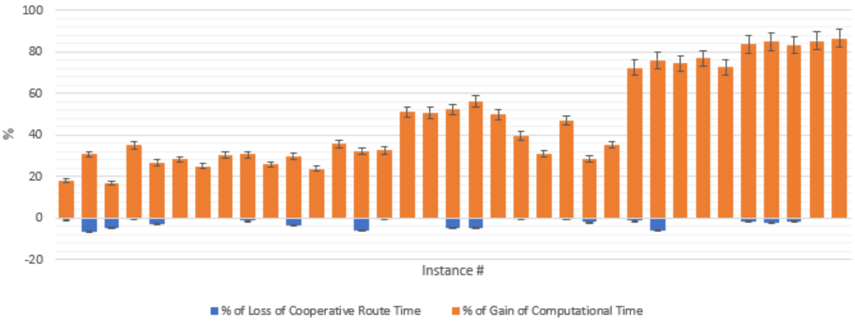
To sum up, cooperative delivery route time difference between all points case and 4 or 5 nearest points cases are between $1-5\%$ in general but computational time difference is about $50-95\%$. For a large number of delivery points, the accuracy of the result seems to be compromised in order to reduce the computational time by reducing number of considered points for the minimum cooperative delivery route time.

3.5.2. Consideration of at least ‘h’ UAV Deliveries for Cooperative Delivery Planning

Normally algorithm checks if there is a cooperative delivery planning solution with at least 2 UAV deliveries. Aim of looking for at least 2 UAV deliveries is to obtain an efficient and worth cooperative delivery planning for UAV and AGV. Efficient and worth mean that if a UAV is used for only one delivery of 15 delivery points then it does not like it worth to use the UAV. On the other hand, checking for every combination of different numbers of UAV for the best route planning may cause waste of computational time. To observe the effect of the minimum numbers of UAVs to be used for delivery, ‘h’, minimum number of 3 and 4 UAVs compared to number of 2 UAVs have been studied for 35 data set (Instances #106-140) and loss of route time in percentage and gain of computational time in percentage are given in Figure 3.13 for comparison. Detailed computational results of Instance #106-140 according to varying ‘h’ value are given in Table H.1.



(a) Case of Considering at least 3 UAV deliveries.



(b) Case of Considering at least 4 UAV deliveries.

Figure 3.13. Effect of Considering at least ‘h’ UAV Deliveries for Cooperative Delivery Mission Planning for UAV and AGV.

As seen in Figure 3.13, computational time has drastically decreased as minimum number of UAV increases for all instances. Nevertheless, cooperative delivery route time increases around $1-3\%$ in general. Moreover, as the total delivery points increase, the route time stays same or increases with a very small rates for both 3 and 4 UAV cases since there is no better solution with 2 UAV case for high number of delivery points. According to comparison between 2 UAV case and 3 UAV case, for 10 delivery points, route time changes(increases) for 40% of instances and for 11 delivery points, route time increase for 20% of instances. For the instances with 12 or more delivery points, route time increases for only 8% of instances. Moreover, these results decreases more for the comparison of 4 UAV and 2 UAV cases. According to these results, the accuracy of the result seems to be compromised in order to reduce the computational time by increasing minimum number of UAV deliveries for the minimum cooperative delivery route time especially for high number of delivery points.

4. CONCLUSION

Main objective of thesis was to develop a computation-time efficient algorithm by dynamic programming (DP) method to find the exact solution of cooperative delivery mission planning for UAV and AGV.

To accomplish this, a modified FSTSP was presented which considers battery charging with a defined constant charge rate for UAV and a weight-aware energy consumption model is applied for the UAV differently from original FSTSP. The problem was divided into two problem to find exact solution; first subproblem finds shortest AGV route by dynamic programming algorithm and second subproblem enumerates all possible UAV routes taking into account all limitations and constraints to find minimum cooperative delivery time. By combining two subproblems, exact solution for FSTSP was obtained. Results of examined data sets show that DP has time efficiency compared to Enumeration method for high number of delivery point since time complexity of DP is lower than enumeration method. On the other hand, for high number of delivery points, even DP cannot provide feasible computational time.

Tunable optimization parameters, 'h' and 'm', were defined to decrease computational times by eliminating inefficient iterations. Results of nearest points parameter, 'm', show that constraining departure and arrival points of UAV for each UAV deliverable as 'm' nearest points decreases computational time with a worthy loss of cooperative delivery time. Similarly, looking for small numbers of at least 'h' UAV deliveries for a cooperative delivery planning is not sufficient. Limiting 'h' with higher numbers decreases computational times with a worthy loss of cooperative delivery time.

Results of different ζ and κ shows that these parameters take a lead role for cooperative delivery time. Same ζ and κ values give varying results for different data sets. It shows that these parameters are needed to be well-tuned to get optimal results for different number of delivery points and varying size of delivery area, which is a different and in-depth study objective.

REFERENCES

1. Hahsler, M. and H. Kurt, “TSP – Infrastructure for the Traveling Salesperson Problem”, *Journal of Statistical Software*, Vol. 23, No. 2, pp. 1–21, 2007.

2. Singhal, G., B. Bansod and L. Mathew, “Unmanned Aerial Vehicle Classification, Applications and Challenges: A Review”, Preprints:2018110601, 2018.

3. Kingston, D., R. Beard, T. McLain, M. Larsen and W. Ren, “Autonomous Vehicle Technologies for Small Fixed Wing UAVs”, *Journal of Aerospace Computing, Information and Communication*, Vol. 2, No. 1, pp. 92–108, 2005.

4. Øyvind Magnussen, M. Ottestad and G. Hovland, “Multicopter Design Optimization and Validation”, *Modeling, Identification and Control*, Vol. 36, No. 2, pp. 67–79, 2015.

5. Cai, G., K. Peng, B. M. Chen and T. H. Lee, “Design and Assembling of a UAV Helicopter System”, *International Conference on Control and Automation*, Vol. 2, pp. 697–702, Budapest, 2005.

6. Vest, M. S. and J. Katz, “Aerodynamic Study of a Flapping-Wing Micro-UAV”, *37th Aerospace Sciences Meeting and Exhibit*, Nevada, 1999.

7. Boon, M. A., A. P. Drijfhout and S. Tesfamichael, “Comparison of a Fixed-Wing and Multi-Rotor UAV for Environmental Mapping Applications: A Case Study”, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLII, No. 2/W6, pp. 47–54, 2017.

8. Moschetta, J. M., “The Aerodynamics of Micro Air Vehicles: Technical Challenges and Scientific Issues”, *International Journal of Engineering Systems Modelling and Simulation*, Vol. 6, No. 3/4, pp. 134–148, 2014.

9. Hassanalian, M., *Wing Shape Design and Kinematic Optimization of Bio-inspired Nano Air Vehicles for Hovering and Forward Flight Purposes*, Ph.D. Thesis, New Mexico State University, 2016.

10. Ozdemir, U., Y. O. Aktas, A. Vuruskan, Y. Dereli, A. F. Tarhan, K. Demirbag, A. Erdem, G. D. Kalaycioglu, I. Ozkol and G. Inalhan, “Design of a Commercial Hybrid VTOL UAV System”, *Journal of Intelligent & Robotic Systems*, Vol. 74, No. 1, pp. 371–393, 2014.

11. Primicerio, J., S. F. D. Gennaro, E. Fiorillo, L. Genesio, E. Lugato, A. Matese and F. P. Vaccari, “A Flexible Unmanned Aerial Vehicle for Precision Agriculture”, *Precision Agriculture*, Vol. 13, No. 4, pp. 517–523, 2012.

12. Hassanalian, M. and A. Abdelkefi, “Classifications, Applications, and Design Challenges of Drones: A Review”, *Progress in Aerospace Sciences*, Vol. 91, pp. 99–131, 2017.

13. Macrina, G., L. D. P. Pugliese, F. Guerriero and G. Laporte, “Drone-aided Routing: A Literature Review”, *Transportation Research Part C: Emerging Technologies*, Vol. 120, p. 102762, 2020.

14. Otto, A., N. Agatz, J. Campbell, B. Golden and E. Pesch, “Optimization Approaches for Civil Applications of Unmanned Aerial Vehicles (UAVs) or Aerial Drones: A Survey”, *Networks*, Vol. 72, No. 4, pp. 411–458, 2018.

15. Smith, K. W., “The Use of Drones in Environmental Management”, *World Environmental and Water Resources Congress*, Virginia, 2015.

16. Roldán, J. J., G. Joossen, D. Sanz, J. Del Cerro and A. Barrientos, “Mini-UAV Based Sensory System for Measuring Environmental Variables in Greenhouses”, *Sensors*, Vol. 15, No. 2, pp. 3334–3350, 2015.

17. Villa, T. F., F. Salimi, K. Morton, L. Morawska and F. Gonzalez, “Development

and Validation of a UAV Based System for Air Pollution Measurements", *Sensors*, Vol. 16, No. 12, p. 2202, 2016.

18. Kavoosi, Z., M. H. Raoufat, M. Dehghani, J. Abdolabbas, S. A. Kazemeini and M. J. Nazemossadat, "Feasibility of Satellite and Drone Images for Monitoring Soil Residue Cover", *Journal of the Saudi Society of Agricultural Sciences*, Vol. 19, No. 1, pp. 56–64, 2020.

19. Capolupo, A., S. Pindozzi, C. Okello, N. Fiorentino and L. Boccia, "Photogrammetry for Environmental Monitoring: The Use of Drones and Hydrological Models for Detection of Soil Contaminated by Copper", *Science of The Total Environment*, Vol. 514, pp. 298–306, 2015.

20. Corbane, C., F. Jacob, D. Raclot, J. Albergel and P. Andrieux, "Multitemporal Analysis of Hydrological Soil Surface Characteristics Using Aerial Photos: A Case Study on a Mediterranean Vineyard", *International Journal of Applied Earth Observation and Geoinformation*, Vol. 18, pp. 356–367, 2012.

21. Velez, F., A. Nadziejko, A. Christensen, S. Oliveira, T. Rodrigues, V. Costa, M. Duarte, F. Silva and J. Gomes, "Wireless Sensor and Networking Technologies for Swarms of Aquatic Surface Drones", *IEEE 82nd Vehicular Technology Conference*, pp. 1–2, Boston, 2015.

22. Tauro, F., M. Porfiri and S. Grimaldi, "Surface Flow Measurements from Drones", *Journal of Hydrology*, Vol. 540, pp. 240–245, 2016.

23. Sieg, M., "Drone Wars: Armed Unmanned Aerial Vehicles", 2010, `https://t.ly/53m0`, accessed on August 22, 2022.

24. Dunn, D. H., "Drones: Disembodied Aerial Warfare and the Unarticulated Threat", *International Affairs*, Vol. 89, No. 5, pp. 1237–1246, 2013.

25. Chan, C. and T. Y. Kam, "A Procedure for Power Consumption Estimation of

Multi-rotor Unmanned Aerial Vehicle", *Journal of Physics: Conference Series*, Vol. 1509, No. 1, p. 012015, 2020.

26. Zhang, J., J. Campbell, D. Sweeney and A. Hupman, "Energy Consumption Models for Delivery Drones: A Comparison and Assessment", *Transportation Research Part D: Transport and Environment*, Vol. 90, p. 102668, 2021.
27. DJI, "Matrice 300 RTK - Specifications", `https://www.dji.com/matrice-300/specs`, accessed on August 22, 2022.
28. DJI, "Agras T20 - Specifications", `https://www.dji.com/t20/specs`, accessed on August 22, 2022.
29. Dantzig, G. B. and J. H. Ramser, "The Truck Dispatching Problem", *Management Science*, Vol. 6, No. 1, pp. 80–91, 1959.
30. Yeun, L. C., W. R. Ismail, K. Omar and M. Zirour, "Vehicle Routing Problem: Models and Solutions", *Journal of Quality Measurement and Analysis*, Vol. 4, No. 1, pp. 205–218, 2008.
31. Wang, X., S. Poikonen and B. Golden, "The Vehicle Routing Problem with Drones: Several Worst-case Results", *Optimization Letters*, Vol. 11, No. 4, pp. 679–697, 2017.
32. Poikonen, S., X. Wang and B. Golden, "The Vehicle Routing Problem with Drones: Extended Models and Connections", *Networks*, Vol. 70, No. 1, pp. 34–43, 2017.
33. Daknama, R. and E. Kraus, "Vehicle Routing with Drones", ArXiv:1705.06431, 2017.
34. Lysgaard, J., A. N. Letchford and R. W. Eglese, "A New Branch-and-cut Algorithm for the Capacitated vehicle Routing Problem", *Mathematical Programming 100:2*, Vol. 100, No. 2, pp. 423–445, 2003.

35. Augerat, P., J. M. Belenguer, E. Benavent, A. Corberán, D. Naddef and G. Rinaldi, *Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem*, Tech. rep., 1995.

36. Baldacci, R., N. Christofides and A. Mingozzi, “An Exact Algorithm for the Vehicle Routing Problem Based on the Set Partitioning Formulation with Additional Cuts”, *Mathematical Programming*, Vol. 115, No. 2, pp. 351–385, 2008.

37. Fukasawa, R., H. Longo, J. Lysgaard, M. P. d. Aragão, M. Reis, E. Uchoa and R. F. Werneck, “Robust Branch-and-Cut-and-Price for the Capacitated Vehicle Routing Problem”, *Mathematical Programming*, Vol. 106, No. 3, pp. 491–511, 2006.

38. Balas, E. and M. W. Padberg, “Set Partitioning”, *Combinatorial Programming: Methods and Applications*, Vol. 19, pp. 205–258, Dordrecht, 1975.

39. Ellabib, I., O. A. Basir and P. Calamai, “An Experimental Study of a Simple Ant Colony System for the Vehicle Routing Problem with Time Windows”, *International Workshop on Ant Algorithms*, pp. 53–64, Berlin, 2002.

40. Desrosiers, J., Y. Dumas, M. M. Solomon and F. Soumis, “Time Constrained Routing and Scheduling”, *Handbooks in Operations Research and Management Science*, Vol. 8, pp. 35–139, 1995.

41. Cordeau, J. F., G. Laporte and A. Mercier, “A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows”, *Journal of the Operational Research Society*, Vol. 52, No. 8, pp. 928–936, 2001.

42. Campbell, A. M. and M. Savelsbergh, “Efficient Insertion Heuristics for Vehicle Routing and Scheduling Problems”, *Transportation Science*, Vol. 38, No. 3, pp. 369–378, 2004.

43. Bent, R. and P. V. Hentenryck, “A Two-stage Hybrid Algorithm for Pickup and Delivery Vehicle Routing Problems with Time Windows”, *Computers & Operations*

Research, Vol. 33, No. 4, pp. 875–893, 2006.

44. Ropke, S. and D. Pisinger, “An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows”, *Transportation Science*, Vol. 40, No. 4, pp. 455–472, 2006.

45. Charlotte, K. and P. Aasen, *Developing a Vans-and-Drones System for Last-Mile Delivery*, Master’s Thesis, The Arctic University of Norway, 2021.

46. Kóczy, L. T., P. Földesi and B. Tüű-Szabó, “Enhanced Discrete Bacterial Memetic Evolutionary Algorithm - An Efficacious Metaheuristic for the Traveling Salesman Optimization”, *Information Sciences*, Vol. 460-461, pp. 389–400, 2018.

47. Dahiya, C. and S. Sangwan, “Literature Review on Travelling Salesman Problem”, *International Journal of Research*, Vol. 5, No. 16, pp. 1152–1155, 2018.

48. Zhang, J. C., “Comparison of Various Algorithms Based on TSP Solving”, *Journal of Physics: Conference Series*, Vol. 2083, No. 3, p. 032007, 2021.

49. Agatz, N., P. Bouman and M. Schmidt, “Optimization Approaches for the Traveling Salesman Problem with Drone”, *Transportation Science*, Vol. 52, No. 4, pp. 965–981, 2018.

50. 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*, Vol. 54, pp. 86–109, 2015.

51. Ha, Q. M., Y. Deville, Q. D. Pham and M. H. Hà, “A Hybrid Genetic Algorithm for the Traveling Salesman Problem with Drone”, *Journal of Heuristics*, Vol. 26, No. 2, pp. 219–247, 2020.

52. Ha, Q. M., Y. Deville, Q. D. Pham and M. H. Hà, “On the Min-cost Traveling Salesman Problem with Drone”, *Transportation Research Part C: Emerging Technologies*, Vol. 86, pp. 597–621, 2018.

53. Kim, S. and I. Moon, “Traveling Salesman Problem With a Drone Station”, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 49, No. 1, pp. 42 – 52, 2019.

54. Mataija, M., M. R. Šegić and F. Jozić, “Solving the Travelling Salesman Problem Using the Branch and Bound Method”, *Zbornik Veleučilišta u Rijeci*, Vol. 4, No. 1, pp. 259–270, 2016.

55. Cormen, T. H., C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms, Second Edition*, MIT Press, Cambridge, 2001.

56. Cook, S. A., “An Overview of Computational Complexity”, *Communications of the ACM*, Vol. 26, No. 6, pp. 400–408, 1983.

57. Nilsson, C., “Heuristics for the Traveling Salesman Problem”, *Linkoping University*, Vol. 38, pp. 00085–9, 2003.

58. Potvin, J.-Y., “Genetic Algorithms for the Traveling Salesman Problem”, *Annals of Operations Research*, Vol. 63, No. 3, pp. 339–370, 1996.

59. Ferrandez, S. M., T. Harbison, T. Weber, R. Sturges and R. Rich, “Optimization of a Truck-drone in Tandem Delivery Network Using k-means and Genetic Algorithm”, *Journal of Industrial Engineering and Management*, Vol. 9, No. 2, pp. 374–388, 2016.

60. Miner, G., J. Elder, A. Fast, T. Hill, R. Nisbet and D. Delen, *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*, Academic Press, Cambridge, 2012.

61. Helsgaun, K., “An Effective Implementation of the Lin–Kernighan Traveling Salesman Heuristic”, *European Journal of Operational Research*, Vol. 126, No. 1, pp. 106–130, 2000.

62. Rosenkrantz, D. J., R. E. Stearns and P. M. Lewis, II, “An Analysis of Several

Heuristics for the Traveling Salesman Problem", *SIAM Journal on Computing*, Vol. 6, No. 3, pp. 563–581, 1977.

63. Beckers, R., J. L. Deneubourg and S. Goss, "Trail Laying Behaviour during Food Recruitment in the antLasius Niger (L.)", *Insectes Sociaux*, Vol. 39, No. 1, pp. 59–72, 1992.

64. Dorigo, M. and L. M. Gambardella, "Ant Colonies for the Travelling Salesman Problem", *Biosystems*, Vol. 43, No. 2, pp. 73–81, 1997.

65. Ponza, A., *Optimization of Drone-assisted Parcel Delivery*, Master's Thesis, University of Padua, 2016.

66. Kivelevitch, E., "Dynamic Programming Solution to the TSP.", 2011, `https://t.ly/tqwu`, accessed on August 22, 2022.

67. Luo, Z., Z. Liu and J. Shi, "A Two-Echelon Cooperated Routing Problem for a Ground Vehicle and Its Carried Unmanned Aerial Vehicle", *Sensors*, Vol. 17, No. 5, p. 1144, 2017.

68. Marinelli, M., L. Caggiani, M. Ottomanelli and M. Dell'Orco, "En Route Truck–Drone Parcel Delivery for Optimal Vehicle Routing Strategies", *IET Intelligent Transport Systems*, Vol. 12, No. 4, pp. 253–261, 2018.

69. Carlsson, J. G. and S. Song, "Coordinated Logistics with a Truck and a Drone", *Management Science*, Vol. 64, No. 9, pp. 4052–4069, 2017.

70. Bouman, P., N. Agatz and M. Schmidt, "Dynamic Programming Approaches for the Traveling Salesman Problem with Drone", *Networks*, Vol. 72, No. 4, pp. 528–542, 2018.

71. Yurek, E. E. and H. C. Ozmutlu, "A Decomposition-based Iterative Optimization Algorithm for Traveling Salesman Problem with Drone", *Transportation Research Part C: Emerging Technologies*, Vol. 91, pp. 249–262, 2018.

72. Tu, P. A., N. T. Dat and P. Q. Dung, “Traveling Salesman Problem with Multiple Drones”, *Proceedings of the Ninth International Symposium on Information and Communication Technology*, p. 46–53, Association for Computing Machinery, Danang City, 2018.

73. Poikonen, S., B. Golden and E. A. Wasil, “A Branch-and-Bound Approach to the Traveling Salesman Problem with a Drone”, *Informs Journal on Computing*, Vol. 31, No. 2, pp. 335–346, 2019.

74. Jeong, H. Y., B. D. Song and S. Lee, “Truck-Drone Hybrid Delivery Routing: Payload-Energy Dependency and No-Fly Zones”, *International Journal of Production Economics*, Vol. 214, pp. 220–233, 2019.

75. Salama, M. and S. Srinivas, “Joint Optimization of Customer Location Clustering and Drone-based Routing for Last-mile Deliveries”, *Transportation Research Part C: Emerging Technologies*, Vol. 114, pp. 620–642, 2020.

76. Dayarian, I., M. Savelsbergh and J. P. Clarke, “Same-day Delivery with Drone Resupply”, *Transportation Science*, Vol. 54, No. 1, pp. 229–249, 2020.

77. Moshref-Javadi, M., A. Hemmati and M. Winkenbach, “A Truck and Drones Model for Last-mile Delivery: A Mathematical Model and Heuristic Approach”, *Applied Mathematical Modelling*, Vol. 80, pp. 290–318, 2020.

78. Moshref-Javadi, M., S. Lee and M. Winkenbach, “Design and Evaluation of a Multi-trip Delivery Model with Truck and Drones”, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 136, p. 101887, 2020.

79. Gonzalez-R, P. L., D. Canca, J. L. Andrade-Pineda, M. Calle and J. M. Leon-Blanco, “Truck-Drone Team Logistics: A Heuristic Approach to Multi-drop Route Planning”, *Transportation Research Part C: Emerging Technologies*, Vol. 114, pp. 657–680, 2020.

80. Agárdi, A., L. Kovács and T. Bányai, “Vehicle Routing in Drone-based Package Delivery Services”, *Solutions for Sustainable Development - Proceedings of the 1st International Conference on Engineering Solutions for Sustainable Development, ICESSD*, pp. 151–159, Miskolc, 2019.

APPENDIX A: FSTSP MODELS AND CHARACTERISTICS ACCORDING TO LITERATURE REVIEW

Table A.1. FSTSP Models and Characteristics according to Literature Review

Reference Study	**Problem Model**	**# of UAV**	**Mathematical Model**	**Cost Function**	**UAV Energy**	**UAV Recharge**	**UAV Capacity**	**Synchronization**	**Solution Algorithm**	**Instances (Data Set)**
Murray and Chu [50]	FSTSP	1	Yes	Completion time	No	No	No	Yes	Heuristic / Clarke–Wright savings Nearest Neighbor Sweep	up to 10 customers
Ferrandez [59]	FSTSP	1/n	No	Delivery time	Est.	No	No	Yes	Heuristic / Genetic Algorithm	up to 100 customers
Ponza [65]	FSTSP	1	Yes	Completion time	No	No	No	Yes	Metaheuristic / Simulated Annealing	up to 200 customers
Luo [67]	FSTSP	1	Yes	UAV routing time	No	No	No	Yes	Heuristic / Greedy Algorithm Clarke–Wright savings	up to 20 customers
Marinelli [68]	FSTSP	1	Yes	Operations costs	No	No	No	Yes	Heuristic / Greedy Randomised Adaptive Search Procedure (GRASP)	Bouman [70]
Carlsson and Song [69]	FSTSP	1	No	Completion time	No	No	No	Yes	Heuristic / Convex Hull Insertion 2-Opt and 3-Opt Algorithm	up to 500 customers
Agatz [49]	FSTSP	1	Yes	Operational costs	No	No	No	Yes	Heuristic / Greedy Algorithm Exact / Dynamic Programming	up to 100 customers
Bouman [70]	FSTSP	1	No	Completion time	No	No	No	Yes	Exact / Dynamic Programming	up to 20 customers
Yurek and Ozmutlu [71]	FSTSP	1	Yes	Completion time	No	No	No	Yes	Heuristic / Nearest Neighbour	up to 20 customers
Phan [72]	FSTSP	n	Yes	Routing cost	No	No	No	Yes	Heuristic / GRASP Adaptive Large Neighborhood Search (ALNS)	Ha [52]
Chang and Lee	FSTSP	n	No	Delivery time	No	No	No	Yes	Heuristic / K-means Clustering	up to 30 customers
Ha [52]	FSTSP	1	Yes	Operational costs	No	No	Yes	Yes	Heuristic / GRASP TSP-LS	up to 100 customers
Ha [51]	FSTSP	1	Yes	Routing cost & UAV waiting	No	No	Yes	Yes	Heuristic / Hybrid Genetic Algorithm	Murray and Chu [50]
Poikonen [73]	FSTSP	1	No	Operational costs	No	No	No	No	Heuristic / Lower Bound Divide-and-Conquer Exact / Branch-and-Bound	up to 200 customers
Jeong [74]	FSTSP	n	Yes	Delivery time	No	No	Yes	Yes	Heuristic / Genetic Algorithm 2-Opt Algorithm	up to 10 customers
Salama and Srinivas [75]	FSTSP	n	Yes	Completion time	No	No	No	No	Heuristic / Unsupervised Machine Learning-based Algorithm	up to 35 customers
Dayarian [76]	FSTSP	n	No	Completion time & routing cost	No	No	No	Yes	Heuristic / Insertion	up to 60 customers
Moshref-Javadi [77]	FSTSP	1	Yes	Maximize the orders	No	No	Yes	Yes	Heuristic / ALNS	up to 101 customers
Moshref-Javadi [78]	FSTSP	n	Yes	Customers waiting time	No	No	No	Yes	Heuristic / Hybrid Tabu Search-Simulated Annealing	up to 159 customers
Gonz´alez-R [79]	FSTSP	1	Yes	Completion time	Est.	No	No	Yes	Heuristic / Simulated Annealing	up to 250 customers
Agardi [80]	FSTSP	1	Yes	Distance	No	No	Yes	Yes	Heuristic / Nearest Neighbour Arbitrary Insertion Genetic Algorithm Hill Climbing	-

APPENDIX B: FLOW DIAGRAM OF THE ALGORITHM

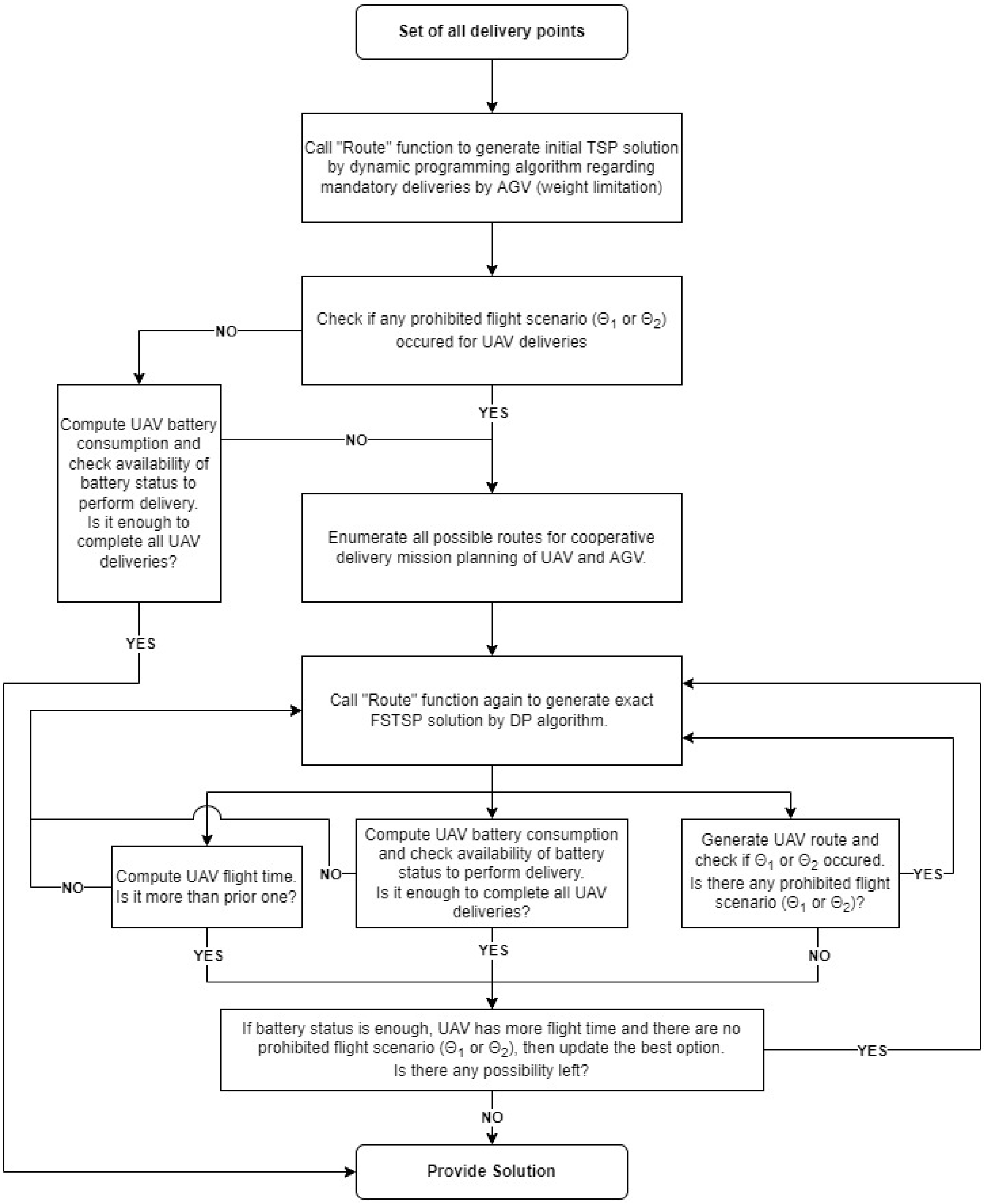


Figure B.1. Flow diagram of the algorithm

APPENDIX C: COMPUTATIONAL TIME VALUES FOR ONLY AGV DELIVERY CASES

Table C.1. Computational Time Values for only AGV Delivery Cases.

#	Instance #	n	Enumeration Time	DP Time
1	c8u3_1	8	1,392	1,448
2	c8u3_2	8	1,314	1,518
⋮	⋮	⋮	⋮	⋮
9	c8u4_4	8	1,345	1,408
10	c8u4_5	8	1,217	1,462
11	c9u3_1	9	2,038	1,388
12	c9u3_2	9	2,118	1,472
⋮	⋮	⋮	⋮	⋮
19	c9u4_4	9	1,912	1,418
20	c9u4_5	9	1,923	1,415
21	c10u3_1	10	10,106	1,560
22	c10u3_2	10	9,895	1,711
⋮	⋮	⋮	⋮	⋮
29	c10u4_4	10	10,225	1,537
30	c10u4_5	10	10,450	1,721
31	c11u4_1	11	108,690	2,428
32	c11u4_2	11	108,808	2,179
⋮	⋮	⋮	⋮	⋮
39	c11u5_4	11	112,246	2,278
40	c11u5_5	11	122,316	2,324
41	c12u4_1	12	-	5,209
42	c12u4_2	12	-	4,864
⋮	⋮	⋮	⋮	⋮
49	c12u5_4	12	-	5,576
50	c12u5_5	12	-	5,097

Table C.1. Computational Time Values for only AGV Delivery Cases. (cont.)

51	c13u5_1	13	-	14,194
52	c13u5_2	13	-	14,905
⋮	⋮	⋮	⋮	⋮
59	c13u6_4	13	-	14,496
60	c13u6_5	13	-	14,250
61	c14u5_1	14	-	48,241
⋮	⋮	⋮	⋮	⋮
70	c14u6_5	14	-	48,926
71	c15u6_1	15	-	184,094
⋮	⋮	⋮	⋮	⋮
80	c15u7_5	15	-	193,598
81	c17u7_1	17	1006911360,00	6173,53
DIM	#	#	[s]	[s]

APPENDIX D: COMPUTATIONAL TIME VALUES FOR DIFFERENT CASES

Table D.1. Computational Time Values for Different Cases.

#	**Instance #**	n	n^U	ρ	ζ	**Enum. Time**	**DP Time**
1	c8u3_1	8	3	5,5	38	2,573	2,120
2	c8u3_2	8	3	5,5	38	2,250	2,202
3	c8u3_3	8	3	5,5	38	2,345	2,324
4	c8u3_4	8	3	5,5	38	2,409	2,400
5	c8u3_5	8	3	5,5	38	2,398	2,278
6	c8u4_1	8	4	5,5	38	4,818	4,443
7	c8u4_2	8	4	5,5	38	5,019	4,367
8	c8u4_3	8	4	5,5	38	4,017	4,255
9	c8u4_4	8	4	5,5	38	2,050	1,919
10	c8u4_5	8	4	5,5	38	1,923	2,023
11	c9u3_1	9	3	5,5	38	3,172	3,040
12	c9u3_2	9	3	5,5	38	3,475	3,109
13	c9u3_3	9	3	5,5	38	3,256	3,132
14	c9u3_4	9	3	5,5	38	3,316	3,113
15	c9u3_5	9	3	5,5	38	3,098	3,263
16	c9u4_1	9	4	5,5	38	3,883	4,081
17	c9u4_2	9	4	5,5	38	3,626	3,637
18	c9u4_3	9	4	5,5	38	4,096	3,670
19	c9u4_4	9	4	5,5	38	9,472	9,291
20	c9u4_5	9	4	5,5	38	3,720	3,963
21	c10u3_1	10	3	5,5	38	4,553	4,793
22	c10u3_2	10	3	5,5	38	4,644	4,552
23	c10u3_3	10	3	5,5	38	4,643	4,689
24	c10u3_4	10	3	5,5	38	4,600	4,579
25	c10u3_5	10	3	5,5	38	4,437	4,648

Table D.1. Computational Time Values for Different Cases. (cont.)

26	c10u4_1	10	4	5,5	38	11,449	10,834
27	c10u4_2	10	4	5,5	38	10,440	11,399
28	c10u4_3	10	4	5,5	38	11,032	10,651
29	c10u4_4	10	4	5,5	38	10,615	10,378
30	c10u4_5	10	4	5,5	38	10,516	10,020
31	c11u4_1	11	4	5,5	38	34,728	33,875
32	c11u4_2	11	4	5,5	38	33,083	34,043
33	c11u4_3	11	4	5,5	38	35,100	35,000
34	c11u4_4	11	4	5,5	38	34,700	34,600
35	c11u4_5	11	4	5,5	38	34,800	34,500
36	c11u5_1	11	5	5,5	38	266,199	254,371
37	c11u5_2	11	5	5,5	38	260,000	248,579
38	c11u5_3	11	5	5,5	38	42,000	42,200
39	c11u5_4	11	5	5,5	38	255,000	249,441
40	c11u5_5	11	5	5,5	38	257,000	252,492
41	c12u4_1	12	4	5,5	38	98,189	94,458
42	c12u4_2	12	4	5,5	38	105,000	100,583
43	c12u4_3	12	4	5,5	38	103,800	99,140
44	c12u4_4	12	4	5,5	38	103,700	98,288
45	c12u4_5	12	4	5,5	38	104,000	98,648
46	c12u5_1	12	5	5,5	38	459,000	432,000
47	c12u5_2	12	5	5,5	38	487,000	465,000
48	c12u5_3	12	5	5,5	38	471,000	448,000
49	c12u5_4	12	5	5,5	38	453,000	421,000
50	c12u5_5	12	5	5,5	38	462,000	429,000
51	c13u5_1	13	5	5,5	38	1099,109	1065,909
52	c13u5_2	13	5	5,5	38	1156,000	1106,000
53	c13u5_3	13	5	5,5	38	1083,000	1045,000
54	c13u5_4	13	5	5,5	38	1045,000	998,000

Table D.1. Computational Time Values for Different Cases. (cont.)

55	c13u5_5	13	5	5,5	38	180,000	178,000
56	c13u6_1	13	6	5,5	38	13515,814	8234,000
57	c13u6_2	13	6	5,5	38	12985,000	7955,000
58	c13u6_3	13	6	5,5	38	12881,000	7841,000
59	c13u6_4	13	6	5,5	38	13211,000	8007,000
60	c13u6_5	13	6	5,5	38	13347,000	8097,000
61	c14u5_1	14	5	5,5	38	-	4163,229
62	c14u5_2	14	5	5,5	38	-	4000,000
63	c14u5_3	14	5	5,5	38	-	4200,000
64	c14u5_4	14	5	5,5	38	-	4150,000
65	c14u5_5	14	5	5,5	38	-	4232,000
66	c14u6_1	14	6	5,5	38	-	15000,000
67	c14u6_2	14	6	5,5	38	-	15320,000
68	c14u6_3	14	6	5,5	38	-	14862,000
69	c14u6_4	14	6	5,5	38	-	15326,000
70	c14u6_5	14	6	5,5	38	-	14698,000
71	c15u6_1	15	6	5,5	38	-	26049,123
72	c15u6_2	15	6	5,5	38	-	26853,000
73	c15u6_3	15	6	5,5	38	-	27003,000
74	c15u6_4	15	6	5,5	38	-	25830,000
75	c15u6_5	15	6	5,5	38	-	26431,000
76	c15u7_1	15	7	5,5	38	-	95431,000
77	c15u7_2	15	7	5,5	38	-	94899,000
78	c15u7_3	15	7	5,5	38	-	95897,000
79	c15u7_4	15	7	5,5	38	-	96012,000
80	c15u7_5	15	7	5,5	38	-	95193,000
DIM	#	#	#	**[kg]**	**[min]**	[s]	[s]

APPENDIX E: EFFECT OF CHARGING CAPABILITY TO COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

Table E.1. Effect of Charging Capability of UAV to CDMP.

#	**Instance #**	n	**Desired** n^U	ρ	ζ	n^T	n^U	**Only AGV Route Time**	**Coop. Route Time**	β_f **with Charging**	β_f **w/out Charging**
1	c8u3_1	8	3	5,5	38	5	3	66,60	59,87	5,30	-0,86
2	c8u3_2	8	3	5,5	38	5	3	59,66	58,27	8,57	1,37
3	c8u3_3	8	3	5,5	38	5	3	61,72	69,24	3,86	-4,74
4	c8u3_4	8	3	5,5	38	5	3	60,24	53,28	0,51	0,51
5	c8u3_5	8	3	5,5	38	5	3	55,28	44,72	7,03	7,03
6	c8u4_1	8	4	5,5	38	6	2	65,31	58,31	0,76	-2,66
7	c8u4_2	8	4	5,5	38	6	2	75,01	61,49	5,09	5,09
8	c8u4_3	8	4	5,5	38	6	2	63,90	49,67	5,20	5,20
9	c8u4_4	8	4	5,5	38	4	4	58,59	54,57	9,78	-0,08
10	c8u4_5	8	4	5,5	38	4	4	67,47	60,90	3,91	-3,65
11	c9u3_1	9	3	5,5	38	6	3	62,98	59,52	0,42	-8,39
12	c9u3_2	9	3	5,5	38	6	3	58,74	58,35	0,77	0,77
13	c9u3_3	9	3	5,5	38	6	3	70,53	63,54	4,71	-3,08
14	c9u3_4	9	3	5,5	38	6	3	60,85	65,48	0,54	-5,89
15	c9u3_5	9	3	5,5	38	6	3	68,52	64,20	4,18	0,10
16	c9u4_1	9	4	5,5	38	5	4	46,60	45,09	3,36	3,36
17	c9u4_2	9	4	5,5	38	5	4	62,34	51,37	3,25	-2,21
18	c9u4_3	9	4	5,5	38	5	4	66,57	66,72	11,53	0,77

Table E.1. Effect of Charging Capability of UAV to CDMP. (cont.)

19	c9u4_4	9	4	5,5	38	6	3	69,42	57,29	1,66	1,66
20	c9u4_5	9	4	5,5	38	5	4	61,16	65,55	1,26	-7,74
21	c10u3_1	10	3	5,5	38	7	3	75,07	76,41	4,42	-3,03
22	c10u3_2	10	3	5,5	38	7	3	83,52	78,22	1,10	-7,78
23	c10u3_3	10	3	5,5	38	7	3	68,13	67,45	0,51	-9,31
24	c10u3_4	10	3	5,5	38	7	3	55,91	50,96	3,72	3,72
25	c10u3_5	10	3	5,5	38	7	3	56,71	51,03	2,48	2,48
26	c10u4_1	10	4	5,5	38	6	4	71,19	69,78	1,13	-8,72
27	c10u4_2	10	4	5,5	38	6	4	54,74	45,74	0,37	0,37
28	c10u4_3	10	4	5,5	38	6	4	64,24	62,82	3,34	-2,29
29	c10u4_4	10	4	5,5	38	6	4	71,47	61,50	2,13	-6,90
30	c10u4_5	10	4	5,5	38	6	4	79,84	73,05	2,58	-8,69
31	c11u4_1	11	4	5,5	38	7	4	76,98	72,65	1,06	-8,80
32	c11u4_2	11	4	5,5	38	7	4	78,39	73,85	0,03	-5,97
33	c11u4_3	11	4	5,5	38	7	4	75,42	64,41	0,15	-6,13
34	c11u4_4	11	4	5,5	38	7	4	74,68	68,60	1,34	-2,42
35	c11u4_5	11	4	5,5	38	7	4	70,90	66,94	1,86	-4,23
36	c11u5_1	11	5	5,5	38	8	3	76,54	66,52	3,24	-2,40
37	c11u5_2	11	5	5,5	38	9	2	74,36	67,62	0,17	-2,57
38	c11u5_3	11	5	5,5	38	6	5	62,56	79,26	1,43	-14,55
39	c11u5_4	11	5	5,5	38	9	2	78,34	64,96	0,47	0,47
40	c11u5_5	11	5	5,5	38	9	2	73,56	54,20	4,51	4,51
41	c12u4_1	12	4	5,5	38	8	4	68,83	62,17	3,10	-0,39
42	c12u4_2	12	4	5,5	38	8	4	69,89	78,55	1,26	-7,53
43	c12u4_3	12	4	5,5	38	8	4	79,63	77,51	0,56	-7,05
44	c12u4_4	12	4	5,5	38	8	4	84,70	85,84	1,20	-14,25
45	c12u4_5	12	4	5,5	38	8	4	74,91	82,92	3,46	-10,75
46	c12u5_1	12	5	5,5	38	7	5	81,67	83,55	3,51	-14,89
47	c12u5_2	12	5	5,5	38	7	5	73,24	82,21	1,09	-15,81

Table E.1. Effect of Charging Capability of UAV to CDMP. (cont.)

48	c12u5_3	12	5	5,5	38	9	3	87,42	74,12	2,28	-3,24
49	c12u5_4	12	5	5,5	38	7	5	71,51	82,18	1,05	-8,64
50	c12u5_5	12	5	5,5	38	7	5	74,70	75,55	0,30	-17,11
DIM	[-]	[#]	[#]	[#]	[#]	[#]	[#]	**[min]**	**[min]**	**[min]**	**[min]**

co-op-er-a-tive

APPENDIX F: EFFECT OF BATTERY CAPACITY OF UAV TO COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

Table F.1. Effect of Battery Capacity of UAV to CDMP.

#	Instance #	n	Desired n^U	n^T	n^U	ζ	Coop. Route Time	β_f
106	e10u6_1	10	6	7	3	33	50,42	1,06
				7	3	38	50,42	6,06
107	e10u6_2	10	6	6	4	33	58,53	0,29
				7	3	38	59,04	11,50
108	e10u6_3	10	6	7	3	33	57,54	5,85
				7	3	38	57,54	10,85
109	e10u6_4	10	6	7	3	33	62,80	20,69
				7	3	38	62,80	25,69
110	e10u6_5	10	6	7	3	33	64,45	15,24
				7	3	38	64,45	20,24
111	e11u6_1	11	6	7	4	33	58,65	11,33
				7	4	38	58,03	2,46
112	e11u6_2	11	6	7	4	33	54,51	0,15
				7	4	38	54,51	1,79
113	e11u6_3	11	6	7	4	33	67,90	0,29
				7	4	38	65,68	3,79
114	e11u6_4	11	6	8	3	33	64,11	3,07
				8	3	38	64,07	5,87
115	e11u6_5	11	6	8	3	33	65,36	6,69
				7	4	38	64,98	3,38

Table F.1. Effect of Battery Capacity of UAV to CDMP. (cont.)

116	e12u7_1	12	7	8	4	33	68,65	3,28
				9	3	38	66,40	2,04
117	e12u7_2	12	7	8	4	33	59,72	15,65
				8	4	38	57,91	1,19
118	e12u7_3	12	7	9	3	33	81,65	15,54
				8	4	38	78,53	1,53
119	e12u7_4	12	7	9	3	33	64,94	2,02
				9	3	38	63,30	2,39
120	e12u7_5	12	7	7	5	33	61,54	6,58
				9	3	38	60,72	0,50
121	e13u6_1	13	6	9	4	33	76,01	3,83
				8	5	38	75,07	1,82
122	e13u6_2	13	6	9	4	33	67,08	13,70
				9	4	38	67,08	18,70
123	e13u6_3	13	6	10	3	33	58,24	3,69
				10	3	38	58,24	2,88
124	e13u6_4	13	6	10	3	33	64,29	3,38
				9	4	38	66,96	18,52
125	e13u6_5	13	6	9	4	33	68,04	1,13
				9	4	38	62,66	2,61
126	e13u7_1	13	7	10	3	33	66,75	6,37
				10	3	38	66,75	11,37
127	e13u7_2	13	7	9	4	33	64,17	3,72
				9	4	38	64,17	8,72
128	e13u7_3	13	7	10	3	33	59,79	4,69
				10	3	38	59,79	9,69
129	e13u7_4	13	7	9	4	33	58,68	7,89
				10	3	38	57,57	0,20

Table F.1. Effect of Battery Capacity of UAV to CDMP. (cont.)

130	e13u7_5	13	7	9	4	33	71,10	6,16
				9	4	38	70,06	1,50
131	e14u6_1	14	6	11	3	33	64,58	13,13
				11	3	38	64,58	18,13
132	e14u6_2	14	6	10	4	33	70,81	1,71
				11	3	38	70,89	10,16
133	e14u6_3	14	6	10	4	33	73,26	10,40
				10	4	38	69,26	1,80
134	e14u6_4	14	6	10	4	33	79,33	20,17
				10	4	38	79,33	25,17
135	e14u6_5	14	6	10	4	33	74,32	5,37
				10	4	38	74,32	0,51
136	e15u6_1	15	6	12	3	33	76,65	6,00
				12	3	38	76,65	11,00
137	e15u6_2	15	6	12	3	33	73,63	11,93
				12	3	38	73,63	16,93
138	e15u6_3	15	6	12	3	33	78,00	2,01
				12	3	38	78,00	2,95
139	e15u6_4	15	6	11	4	33	73,46	3,49
				11	4	38	73,46	8,49
140	e15u6_5	15	6	11	4	33	77,79	6,65
				11	4	38	77,79	8,20
DIM	[#]	[#]	[#]	[#]	[#]	[min]	[min]	[min]

APPENDIX G: EFFECT OF CONSIDERING ‘m’ NEAREST POINTS TO UAV TO COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

Table G.1. Effect of Considering ‘m’ Nearest Points to UAV to CDMP.

#	Instance #	n	Desired n^U	n^T	n^U	Coop. Route Time	‘m’	DP Time
91	e8u5_1	8	5	5	3	41,67	All	9,73
				5	3	40,73	4	5,26
92	e8u5_2	8	5	6	2	50,10	All	10,39
				6	2	51,88	4	5,70
93	e8u5_3	8	5	5	3	48,98	All	12,40
				5	3	48,98	4	5,28
94	e8u5_4	8	5	6	2	49,59	All	10,76
				6	2	47,02	4	6,04
95	e8u5_5	8	5	6	2	45,33	All	12,69
				6	2	43,39	4	6,38
96	e9u5_1	9	5	6	3	61,91	All	27,45
				6	3	61,91	5	10,52
				6	3	61,91	4	6,54
97	e9u5_2	9	5	6	3	55,08	All	29,66
				6	3	55,08	5	11,21
				7	2	56,51	4	6,42
98	e9u5_3	9	5	6	3	55,27	All	26,93
				7	2	51,66	5	11,70
				6	3	54,41	4	6,44
99	e9u5_4	9	5	7	2	64,70	All	26,74
				6	3	67,09	5	10,52
				7	2	67,19	4	6,71

Table G.1. Effect of Considering ‘m’ Nearest Points to UAV to CDMP. (cont.)

100	e9u5_5	9	5	6	3	61,74	All	28,24
				6	3	61,74	5	10,40
				7	2	62,61	4	8,65
101	e9u5_6	9	5	6	3	52,34	All	27,78
				6	3	52,34	5	9,59
				6	3	52,34	4	6,47
102	e9u5_7	9	5	6	3	49,20	All	26,91
				7	2	49,32	5	10,31
				7	2	49,32	4	6,85
103	e9u5_8	9	5	7	2	63,00	All	28,32
				7	2	63,00	5	10,19
				7	2	70,52	4	6,24
104	e9u5_9	9	5	5	4	46,16	All	25,25
				6	3	46,15	5	11,22
				5	4	46,16	4	6,53
105	e9u5_10	9	5	6	3	46,34	All	27,30
				6	3	46,34	5	10,31
				6	3	47,32	4	6,54
106	e10u6_1	10	6	8	2	51,32	All	199,92
				7	3	51,77	5	32,06
				7	3	50,42	4	12,61
107	e10u6_2	10	6	7	3	57,90	All	207,61
				7	3	59,04	5	36,11
				7	3	59,04	4	16,33
108	e10u6_3	10	6	8	2	56,36	All	209,32
				8	2	56,36	5	30,86
				8	2	56,36	4	13,15

Table G.1. Effect of Considering ‘m’ Nearest Points to UAV to CDMP. (cont.)

109	e10u6_4	10	6	7	3	62,24	All	214,63
				6	4	61,97	5	32,59
				7	3	62,80	4	13,88
110	e10u6_5	10	6	7	3	59,35	All	200,93
				8	2	60,58	5	33,63
				8	2	63,00	4	12,86
111	e11u6_1	11	6	8	3	54,77	All	824,25
				8	3	56,14	5	47,63
				7	4	58,03	4	16,94
112	e11u6_2	11	6	7	4	54,51	All	818,19
				7	4	54,51	5	55,65
				7	4	54,51	4	18,70
113	e11u6_3	11	6	-	-	-	All	-
				8	3	67,63	5	54,19
				7	4	65,68	4	19,96
114	e11u6_4	11	6	-	-	-	All	-
				9	2	61,46	5	45,20
				9	2	63,98	4	18,09
115	e11u6_5	11	6	-	-	-	All	-
				8	3	63,08	5	54,10
				7	4	64,98	4	18,25
116	e12u7_1	12	7	-	-	-	All	-
				9	3	66,40	5	174,68
				9	3	66,40	4	47,36
117	e12u7_2	12	7	-	-	-	All	-
				8	4	58,48	5	240,51
				8	4	57,91	4	64,28

Table G.1. Effect of Considering ‘m’ Nearest Points to UAV to CDMP. (cont.)

118	e12u7_3	12	7	-	-	-	All	-
				9	3	80,12	5	186,02
				8	4	78,53	4	57,80
119	e12u7_4	12	7	-	-	-	All	-
				9	3	63,30	5	223,85
				9	3	63,30	4	59,13
120	e12u7_5	12	7	-	-	-	All	-
				9	3	58,32	5	201,11
				9	3	60,72	4	55,33
DIM	[#]	[#]	[#]	[#]	[#]	[min]	[#]	[s]

APPENDIX H: EFFECT OF CONSIDERING AT LEAST ‘h’ UAV DELIVERIES TO COOPERATIVE DELIVERY MISSION PLANNING FOR UAV AND AGV

Table H.1. Effect of at least ‘h’ UAV deliveries for CDMP.

#	**Instance #**	n	**Desired** n^U	n^T	n^U	**Coop. Route Time**	**‘h’**	**DP Time**
106	e10u6_1	10	6	7	3	50,42	2	12,61
				7	3	50,42	3	11,55
				6	4	50,75	4	10,37
107	e10u6_2	10	6	7	3	59,04	2	16,33
				7	3	59,04	3	12,77
				6	4	63,04	4	11,33
108	e10u6_3	10	6	8	2	56,36	2	13,15
				7	3	59,04	3	11,57
				6	4	59,08	4	10,96
109	e10u6_4	10	6	7	3	62,80	2	13,88
				7	3	62,80	3	11,31
				6	4	63,08	4	9,02
110	e10u6_5	10	6	8	2	63,00	2	12,86
				7	3	64,45	3	11,70
				6	4	64,81	4	9,44
111	e11u6_1	11	6	7	4	58,03	2	16,94
				7	4	58,03	3	16,31
				7	4	58,03	4	12,17
112	e11u6_2	11	6	7	4	54,51	2	18,70
				7	4	54,51	3	17,46
				7	4	54,51	4	14,04

Table H.1. Effect of at least ‘*h*’ UAV Deliveries for CDMP. (cont.)

113	e11u6_3	11	6	7	4	65,68	2	19,96
				7	4	65,68	3	15,88
				7	4	65,68	4	13,94
114	e11u6_4	11	6	9	2	63,98	2	18,09
				8	3	64,07	3	14,74
				7	4	64,88	4	12,56
115	e11u6_5	11	6	7	4	64,98	2	18,25
				7	4	64,98	3	17,24
				7	4	64,98	4	13,56
116	e12u7_1	12	7	9	3	66,40	2	47,36
				9	3	66,40	3	39,85
				8	4	68,65	4	33,30
117	e12u7_2	12	7	8	4	57,91	2	64,28
				8	4	57,91	3	55,17
				8	4	57,91	4	49,09
118	e12u7_3	12	7	8	4	78,53	2	57,80
				8	4	78,53	3	44,16
				8	4	78,53	4	37,25
119	e12u7_4	12	7	9	3	63,30	2	59,13
				9	3	63,30	3	48,90
				8	4	67,03	4	40,04
120	e12u7_5	12	7	9	3	60,72	2	55,33
				9	3	60,72	3	47,03
				8	4	61,00	4	37,39
121	e13u6_1	13	6	8	5	75,07	2	41,82
				8	5	75,07	3	27,30
				8	5	75,07	4	20,50

Table H.1. Effect of at least ‘*h*’ UAV Deliveries for CDMP. (cont.)

122	e13u6_2	13	6	9	4	67,08	2	42,90
				9	4	67,08	3	28,94
				9	4	67,08	4	21,17
123	e13u6_3	13	6	10	3	58,24	2	38,23
				10	3	58,24	3	26,20
				9	4	61,04	4	18,22
124	e13u6_4	13	6	11	2	63,88	2	41,74
				9	4	66,96	3	28,01
				9	4	66,96	4	18,20
125	e13u6_5	13	6	9	4	62,66	2	42,57
				9	4	62,66	3	29,23
				9	4	62,66	4	21,35
126	e13u7_1	13	7	10	3	66,75	2	85,22
				10	3	66,75	3	68,59
				8	5	66,95	4	51,40
127	e13u7_2	13	7	9	4	64,17	2	99,53
				9	4	64,17	3	82,77
				9	4	64,17	4	68,88
128	e13u7_3	13	7	10	3	59,79	2	67,64
				10	3	59,79	3	49,03
				9	4	59,97	4	35,95
129	e13u7_4	13	7	10	3	57,57	2	100,46
				10	3	57,57	3	84,89
				9	4	58,68	4	71,69
130	e13u7_5	13	7	9	4	70,06	2	98,16
				9	4	70,06	3	77,83
				9	4	70,06	4	63,64

Table H.1. Effect of at least ‘*h*’ UAV Deliveries for CDMP. (cont.)

131	e14u6_1	14	6	11	3	64,58	2	103,62
				11	3	64,58	3	50,44
				10	4	65,43	4	28,60
132	e14u6_2	14	6	12	2	67,15	2	94,01
				11	3	70,89	3	43,51
				10	4	71,08	4	22,66
133	e14u6_3	14	6	10	4	69,26	2	104,58
				10	4	69,26	3	49,85
				10	4	69,26	4	26,84
134	e14u6_4	14	6	10	4	79,33	2	90,69
				10	4	79,33	3	42,52
				10	4	79,33	4	21,05
135	e14u6_5	14	6	10	4	74,32	2	88,25
				10	4	74,32	3	44,51
				10	4	74,32	4	24,28
136	e15u6_1	15	6	12	3	76,65	2	289,04
				12	3	76,65	3	128,71
				11	4	77,97	4	47,61
137	e15u6_2	15	6	12	3	73,63	2	300,32
				12	3	73,63	3	113,68
				11	4	75,31	4	45,40
138	e15u6_3	15	6	12	3	78,00	2	292,06
				12	3	78,00	3	132,31
				11	4	79,37	4	49,17
139	e15u6_4	15	6	11	4	73,46	2	261,40
				11	4	73,46	3	119,32
				11	4	73,46	4	38,49

Table H.1. Effect of at least ‘*h*’ UAV Deliveries for CDMP. (cont.)

140	e15u6_5	15	6	11	4	77,79	2	303,92
				11	4	77,79	3	116,84
				11	4	77,79	4	40,97
DIM	[#]	[#]	[#]	[#]	[#]	[min]	[#]	[s]