Southern Methodist University

SMU Scholar

Civil and Environmental Engineering Theses and Dissertations

Civil Engineering and Environmental Engineering

Spring 2020

Hybrid Vehicle-drone Routing Problem For Pick-up And Delivery Services Mathematical Formulation And Solution Methodology

Aline Karak
akarak@smu.edu

Follow this and additional works at: https://scholar.smu.edu/engineering_civil_etds

Part of the Operations Research, Systems Engineering and Industrial Engineering Commons, and the Transportation Engineering Commons

Recommended Citation

Karak, Aline, "Hybrid Vehicle-drone Routing Problem For Pick-up And Delivery Services Mathematical Formulation And Solution Methodology" (2020). *Civil and Environmental Engineering Theses and Dissertations*. 9.
https://scholar.smu.edu/engineering_civil_etds/9

This Dissertation is brought to you for free and open access by the Civil Engineering and Environmental Engineering at SMU Scholar. It has been accepted for inclusion in Civil and Environmental Engineering Theses and Dissertations by an authorized administrator of SMU Scholar. For more information, please visit http://digitalrepository.smu.edu.

HYBRID VEHICLE-DRONE ROUTING PROBLEM FOR PICK-UP AND DELIVERY SERVICES MATHEMATICAL FORMULATION AND SOLUTION METHODOLOGY

Dissertation Committee:

Prof. Khaled Abdelghany (Advisor)

Prof. Barbara Minsker

Prof. Janille Smith-Colin

Prof. Eli Olinick

Prof. Aurelie Thiele

HYBRID VEHICLE-DRONE ROUTING PROBLEM FOR PICK-UP AND DELIVERY SERVICES MATHEMATICAL FORMULATION AND SOLUTION METHODOLOGY

A Dissertation Presented to the Graduate Faculty of

Lyle School of Engineering

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Civil and Environmental Engineering

by

Aline Karak

B.Sc., Rafik Hariri University, Mechref, Lebanon, 2014
M.Sc., University of Balamand, Koura, Lebanon, 2016

May 16, 2020

Copyright (2020)

Aline Karak

All Rights Reserved

ACKNOWLEDGMENTS

I wish to express my sincere appreciation to my advisor, Professor Khaled Abdelghany, he convincingly guided and encouraged me to be professional and do the right thing even when the road got tough. Without his persistent help, the goal of this project would not have been realized. I would like also to pay my special regards to my committee members. I am grateful to all my colleagues at the Transportation Lab and who have supported me and had to put up with my stresses. In particular, I would like to thank Dr. Eli Olinick and Dr. Ahmed Hassan for their many helpful discussions and useful feedback. Last but not least, I wish to acknowledge the support and great love of my family, my husband, Fady Tadrous; my parents, Toni and Layla; my brothers, Elie, Naji, and Pierre; my sisters in law, Vicky, Diana, and Layal; my mother in law, Samira; and my brother in law, Pola. They kept me going on and this work would not have been possible without their input.

Karak, Aline B.Sc., Rafik Hariri University, Mechref, Lebanon, 2014
M.Sc., University of Balamand, Koura, Lebanon, 2016

The Hybrid Vehicle-Drone Routing Problem for Pick-up and Delivery Services

Advisor: Professor Khaled Abdelghany

Doctor of Philosophy conferred May 16, 2020

Dissertation completed March 6, 2020

The fast growth of online retail and associated increasing demand for same-day delivery have pushed online retail and delivery companies to develop new paradigms to provide faster, cheaper, and greener delivery services. Considering drones' recent technological advancements over the past decade, they are increasingly ready to replace conventional truck-based delivery services, especially for the last mile of the trip. Drones have significantly improved in terms of their travel ranges, load-carrying capacity, positioning accuracy, durability, and battery charging rates. Substituting delivery vehicles with drones could result in $50M of annual cost savings for major U.S. service providers.

The first objective of this research is to develop a mathematical formulation and efficient solution methodology for the hybrid vehicle-drone routing problem (HVDRP) for pick-up and delivery services. The problem is formulated as a mixed-integer program, which minimizes the vehicle and drone routing cost to serve all customers. The formulation captures the vehicle-drone routing interactions during the drone dispatching and collection processes and accounts for drone operation constraints related to flight range and load carrying capacity limitations. A novel solution methodology is developed which extends

the classic Clarke and Wright algorithm to solve the HVDRP. The performance of the developed heuristic is benchmarked against two other heuristics, namely, the vehicle-driven routing heuristic and the drone-driven routing heuristic.

Anticipating the potential risk of using drones for delivery services, aviation authorities in the U.S. and abroad have mandated necessary regulatory rules to ensure safe operations. The U.S. Federal Aviation Administration (FAA) is examining the feasibility of drone flights in restricted airspace for product delivery, requiring drones to fly at or below 400-feet and to stay within the pilot's line of sight (LS).

Therefore, a second objective of this research is considered to develop a modeling framework for the integrated vehicle-drone routing problem for pick-up and delivery services considering the regulatory rule requiring all drone flights to stay within the pilot's line of sight (LS). A mixed integer program (MIP) and an efficient solution methodology were developed for the problem. The solution determines the optimal vehicle and drone routes to serve all customers without violating the LS rule such that the total routing cost of the integrated system is minimized. Two different heuristics are developed to solve the problem, which extends the Clarke and Wright Algorithm to cover the multimodality aspects of the problem and to satisfy the LS rule. The first heuristic implements a comprehensive multimodal cost saving search to construct the most efficient integrated vehicle-drone routes. The second heuristic is a light version of the first heuristic as it adopts a vehicle-driven cost saving search.

Several experiments are conducted to examine the performance of the developed methodologies using hypothetical grid networks of different sizes. The capability of the

developed model in answering a wide variety of questions related to the planning of the vehicle-drone delivery system is illustrated. In addition, a case study is presented in which the developed methodology is applied to provide pick-up and delivery services in the downtown area of the City of Dallas. The results show that mandating the LS rule could double the overall system operation cost especially in dense urban areas with LS obstructions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1-1. Background

The evolution of drone technology during the past decade has opened the door for numerous innovative applications in transportation/logistics (Troudi et al., 2017; Kunze, 2016; Menouar et al., 2017), defense (Paust 2010; Schneiderman, 2012), public safety and security (Chowdhury et al., 2017; Clarke and Moses, 2014; Vattapparamban et al., 2016; Merwaday and Guvenc, 2015), healthcare (Thiels et al., 2015; Kim et al., 2017; Balasingam, 2017), and forestry and agriculture (Getzin et al., 2012), to name a few. In particular, the use of drones for product delivery has received considerable attention following Amazon's recently announced plan to use drones for product delivery (Rose, 2013). A leading U.S. delivery company estimates an annual cost saving of about $50M if drones replaced its trucks for the last mile of the delivery trip (Rash, 2017).

Drone usage for delivery applications is expected to grow significantly in the next few years. Several contributing factors to this growth include: (1) the expanding online retail industry; (2) improved capability, reliability, and cost effectiveness of drones; and (3) high competition among pick-up and delivery service providers. Therefore, there are increasing calls to develop innovative pick-up and delivery systems that integrate drones in order to meet growing demand and reduce service costs pick-up and delivery service providers.

Figure 1-1: Drones applications (source: collected from the Internet).

Using drones for delivery services may have potential risk, thus aviation authorities in the U.S. and abroad are mandating neccessary regulatory rules to ensure safe operations (Jones, 2017). The U.S. Federal Aviation Administration (FAA) is examining the feasibility of drone flights in restricted airspace for product delivery, requiring drones to fly at or below 400-feet and to stay within the pilot's line of sight (LS) (Clarke and Moses, 2014; Locklear, 2017; Dorr, 2018).

1-2. The Mothership System

Effort is underway to develop a technology that meets the requirements of product delivery applications. Drone manufacturers are developing the next-generation drones with increased travel ranges, load carrying capacity, positioning accuracy, durability, and battery charging rates (Floreano and Wood, 2015). A parallel effort is devoted to studying the logistical aspects of adopting drones for delivery services, taking into consideration regulatory rules and operational constraints. For example, the flying side-kick system, in which one drone is mounted on a vehicle and used to visit selected customers, has been developed to address these logistical constraints (Murray and Chu, 2015). However, this system does not take full advantage of drone capabilities in terms of visiting multiple customers per dispatch nor the possibilities for more efficient vehicle-drone integration.

In this context, a novel system recently conceptualized for using drones to provide product delivery services is the integrated vehicle-drone system (a.k.a. the "mothership" system). The system generally consists of vehicles (trucks or vans) that carry unmanned vehicles (robots and/or drones), as shown in Figure 1-2, from their depots to neighborhoods where the unmanned vehicles are dispatched to perform pick-up and delivery tasks (McFarland, 2016). The system adopts a "swarm" dispatching approach which allows dozens of pick-ups and deliveries to be performed simultaneously (PYMNTS, 2016; Petersen, 2016). Such a system is estimated to double the average number of packages delivered in a typical working shift as compared to the conventional system in which a vehicle completes one delivery at a time (Lockridge, 2017).

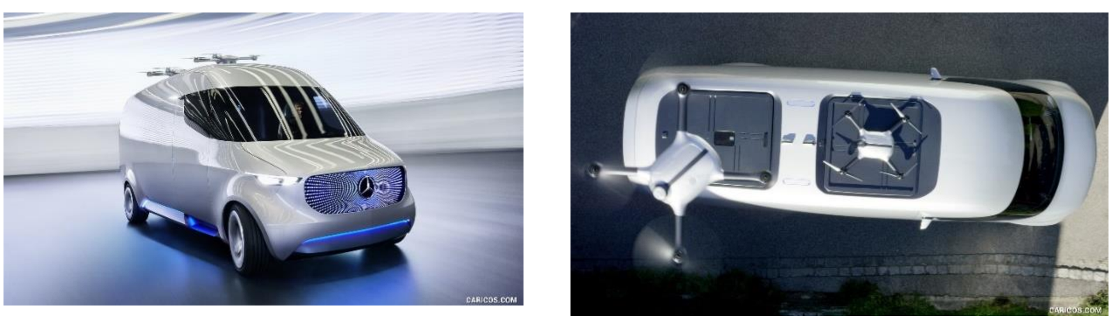


Figure 1-2: Mothership vision by MERCEDES-BENZ (**Hsu, 2017**).

This system can be viewed as a version of the pick-up and delivery problem, which can be classified into three different problem categories: one-to-one, one-to-many-to-one, and many-to-many (Berbeglia et al. 2010). In the one-to-one problem, each commodity is transported directly from an origin to a destination. In the one-to-many-to-one problem, commodities are transported from a single depot to customers, and commodities picked up from the customers are transported to that depot. Finally, the many-to-many problem involves transporting commodities from multiple depots to multiple customers, and vice versa. The mothership system studied in this paper is a one-to-many-to-one problem, as the vehicle and the drones are dispatched from one depot to deliver the commodities to customers, and to pick up the commodities from the customers and transport them back to the depot.

Integrating drones with a vehicle in the form of the mothership system presents several advantages as compared to previously proposed systems such as the flying side-kick delivery system. For example, the mothership system considers the dispatching of multiple drones simultaneously, and each drone can serve multiple customers per dispatch. On the other hand, the side-kick system assumes that only one drone is used, which serves one customer per dispatch. Furthermore, the mothership system offers flexibility in terms of the drone dispatching and collection locations (i.e., these could be the same or different). The side-kick system forces the drone collection location to be different from its dispatching location, as the vehicle does not wait at the dispatching location. Also, in the side-kick system, drones are used for package delivery only without the option to provide package pick-up services along their tours. Thus, the superior and flexible configuration of the mothership system provides the capability to perform pick-up and delivery services more efficiently.

The mothership system is also envisioned to reduce congestion caused by trucking in urban areas as it increases dependence on drones and reduces the number of required vehicle stops as compared to the side-kick system, where most customers are served by the vehicle. In addition, the mothership system is expected to reduce the workload on the driver as it limits her/his tasks to driving between specified stops and loading/unloading packages from the drones. Thus, the driver is not involved in any door-to-door pick-up or delivery tasks, which enhances her/his working conditions and safety. Finally, while the side-kick system assumes that drone dispatches and collections occur at a customer location, the mothership system allows the vehicle to dispatch and collect the drones at special locations that can be sensibly selected to limit any inconvenience (e.g., noise, safety, and aesthetic) to the customers.

Previous research work on studying the integrated vehicle-drone systems has completely ignored the effect of regulatory rules (those requiring all drones to stay within the pilot's LS as shown in Figure 1-3) on the operation performance of these systems by assuming a clear LS between the drones' dispatching and target locations. This assumption significantly precludes the use of the developed models for real-world applications. Therefore this research not only studies the mothership system but also focuses on studying the effect of the LS regulatory rule on the performance of the integrated vehicle-drone systems for pick-up and delivery services.

Figure 1-3: Drones satisfying LS rule (source: collected from the Internet).

1-3. Challenges of Developing the Mothership System

Designing a hybrid vehicle-drone system for pick-up and delivery services entails determining the optimal setting of several system parameters including: a) vehicle and drone resources required for the pick-up and delivery tasks; b) locations (stations) for drone dispatching and collection; c) tactics used to dispatch and collect the drones; d) number of customers visited per drone dispatch; and e) optimal vehicle and drone routing decisions.

For example, the number of vehicles, the number of drones mounted on each vehicle, and the capabilities of the drones in terms of their flying ranges and load carrying capacities should be determined for each operation. The locations for dispatching and collecting the drones should be selected to ensure that all customers can be reached by the drones. Furthermore, two tactics may be considered for drone dispatching and collection. First, a vehicle could dispatch its drone(s) at a location and wait at the same location to collect the drone(s). This dispatch-wait-collect tactic is suitable in cases where drones must remain within sight for safety considerations. Alternatively, the dispatch-move-collect tactic allows the vehicle to move after dispatching the drones. The drones could be collected at another location by the same vehicle or by another available vehicle. Finally, the optimal route for each drone should be determined in terms of its dispatching and collection stations and the sequence of customers visited. Optimal vehicle routes should be determined in terms of the sequence of customers to be served, if any, and the sequence of stations used for drone dispatching and collection.

Several sources of complexity characterize the hybrid vehicle-drone routing problem (HVDRP). First, even for small size problems, the HVDRP involves a large number of decision variables including vehicle and drone resources/capabilities, locations of drone dispatching and collection, and routing decisions for the vehicles and the drones. The problem can be generally viewed as an extension of the classic vehicle routing problem (VRP) which is known to be an NP-hard problem (Golden et al., 2008). Thus, the execution time required to obtain an exact optimal solution grows exponentially as the problem size increases. Second, most decision variables involved in this problem are highly interdependent and cannot be optimized separately. For example, the locations for drone dispatching and collection depend on the drone's flying range and load carrying capabilities, and vice versa.

Furthermore, optimizing vehicle routes and drone routes independently could result in a sub-optimal solution, because the vehicle routes determine the stations for dispatching and collecting the drones, which in turn define the origins and destinations of the drones' routes. The locations of the dispatch and collection stations are simultaneously impacted by the sequence of customers visited by the drones. Finally, because such a system has not yet been deployed in the real-world, developing a model to study the HVDRP requires making several assumptions related to defining the configuration of the system and its operational parameters.

1-4. Challenges of Developing the Mothership System Satisfying the line of sight (LS) Rule

Mandating the LS rule for the mothership system is expected to result in significant changes to its basic configuration. We refer to this problem as the integrated vehicle-drone routing problem with the LS rule (IVDRP-LS). For example, for a basic mothership system that is not satisfying the LS rule, all customers are assumed to be visited only by drones that are dispatched from the customers' nearest vehicle stops (stations). This assumption might not hold when the LS rule is mandated. Customers who do not fall within the pilot's LS from any of the drones' possible dispatching locations are visited by the vehicle.

Allowing customers to be visited by the vehicle converts the mothership system into a system similar to the flying sidekick system with multiple customers per drone tour. In addition, it is not guaranteed that customers are always served from their nearest stations as the LS from these stations might be obstructed. As such, mandating the LS is expected to affect locations used for drone dispatching and collection, and routing decisions for the vehicle and drones, respectively.

Furthermore, the mothership system implements two tactics for drone dispatching and collection: (1) the dispatch-wait-collect tactic in which the vehicle dispatches its drones at a location and waits to collect them; and (2) the dispatch-move-collect tactic which allows the vehicle to move to another location after dispatching the drones. Satisfying the

LS regulatory rule requires the vehicle to always wait for the drones at the dispatching station until the drones return or land at a station that is visible from the dispatching station.

1-5. Research Approach

Two mathematical formulations in the form of a mixed-integer program (MIP) are developed for both problems (basic mothership system and mothership system satisfying LS rule). The first formulation, developed for the HVDRP, solves for the optimal drone and vehicle routes to serve all customers such that the total cost of the pick-up and the delivery operation is minimized. The formulation considers operational constraints for the vehicle and drones and captures their interdependence. Due to the NP-hard nature of the HVDRP, its optimal solution can only be obtained in a reasonable execution time for small problem instances. Thus, there is a need to develop efficient heuristics that can be used to obtain a good solution for large problem instances such as those anticipated in real-world applications. To achieve this goal, we introduce a novel solution methodology that extends the classic Clarke and Wright (CW) algorithm, the hybrid Clarke and Wright heuristic (HCWH) (Clarke and Wright, 1964).

The heuristic considers the cost savings for both the vehicle and the drones while solving for the optimal vehicle route, thus generating an efficient multimodal vehicle-drone network. The performance of the HCWH is benchmarked against two other heuristics that are developed as part of this research, which are the vehicle-driven heuristic (VDH) and the drone-driven heuristic (DDH). In the VDH, the optimal vehicle route is obtained first and then the drones are routed, assuming a fixed vehicle route. A reverse approach is considered for the DDH: given the optimal drone routes, the vehicle is routed to enable the dispatching and collection of the drones. The performance of these heuristics is compared in terms of the solution quality and the required execution time, considering several randomly generated networks of different sizes and configurations.

The second formulation, developed for the IVDRP-LS, determines the optimal vehicle and drone routes to serve all customers such that the total travel cost of both modes is minimized and the LS regulatory rule is satisfied. The IVDRP-LS considerably extends the VRP which is known to be an NP-hard problem (Golden et al., 2008). In order to solve large problem instances in reasonable execution times, we introduce a novel solution methodology that adopts an updated version of the classic CW algorithm to consider the multimodality aspects of the integrated vehicle-drone routing problem and to satisfy the LS rule (Clarke and Wright, 1964). The solution methodology implements a Multimodal-Based Heuristic (MBH) with randomization procedure to construct near optimal vehicle and LS-mandated drone routes. The performance of the MBH is benchmarked by comparing its performance against that of a Single-Mode-Based Heuristic (SBH). The SBH is a lighter version of the MBH as it adopts a vehicle-driven search procedure.

1-6. Research Contributions

This research contributes to the existing literature in several ways. First, to the author's knowledge, this research is among the first to develop a model that studies the mothership system at a high realism and the impact of LS rule on this system. Most existing models fall short of representing drones' capabilities in terms of flight range and load carrying capacity, and consequently, misrepresent their impact on routing decisions.

Moreover, this model considers advanced features for the HVDRP that are technically feasible and could result in significant cost savings, such as allowing the drones to visit multiple customers in a single dispatch and allowing them to be dispatched and collected at two different locations.

Second, the research presents two comprehensive mathematical formulations that can be used to obtain the optimal solution for small size problems in order to benchmark the solution quality of the developed heuristics. The first formulation, considers the main operational constraints defined for the problem, including interdependence between the vehicle and the drones and the limitations of the drones in terms of flight range and load carrying capacity. The second formulation, explicitly captures key operational aspects of the integrated vehicle-drone system considering the LS rule.

Third, this research presents a novel extension of the classic CW algorithm to solve the HVDRP and the IVDRP-LS. The cost savings computed at each iteration account for both vehicle and drone routing costs. Thus, the solution simultaneously optimizes the routing decisions for the multimodal vehicle-drone network. Heuristics solving the IVDRP-LS not only suit the multimodality nature of the problem but also the LS constraints.

Fourth, the performances of the developed heuristics in terms of solution quality and execution time are examined considering several grid networks of different sizes and configurations. A sensitivity analysis to examine the effect of several system parameters on the overall performance of the network is also presented. Finally, the research is the first to quantify the impact of the LS rule on the overall system performance considering real-world urban settings. The results are presented for a case study that illustrates the application of the developed methodology in the downtown area of the City of Dallas, Texas, considering different customer spatial distributions.

1-7. Research Objectives

This research is motivated by the need to advance the theory and practice of the multimodal mothership system, which are capable of finding the optimal drone and vehicle routes to serve all customers such that the total cost of the pick-up and the delivery operation is minimized. Several objectives are considered for this research.

First, the comprehensive literature review discusses existing work related to the subject of this research. It covers several versions of the classic VRP that share features of the problem on hand, such as the green vehicle routing problem (GVRP), the two-echelon location and routing problem (2E-LRP), and the truck and trailer routing problem (TTRP). Literature that covers the previous research that studied the vehicle drone routing problem is also considered.

The second objective is to develop a modeling framework for the HVDRP. The problem is formulated in the form of the mixed integer linear program with the goal of finding the optimal routes for the two modes to serve all the customers in the network. The constraints of the model should consider operational challenges for the vehicle and drones and capture their interdependence.

The third objective is to develop heuristics that can solve large problem instances in a reasonable execution time. The heuristics extend the CW algorithm to consider the multimodality of the integrated vehicle-drone routing problem (Clarke and Wright, 1964), namely the hybrid Clarke and Wright heuristic (HCWH). The performance of the HCWH is benchmarked against a vehicle-driven heuristic (VDH) and a drone-driven heuristic (DDH).

The fourth objective is to develop an MIP for the IVDRP-LS. The formulation will determine the optimal vehicle and drone routes to serve all customers such that the total travel cost of both modes is minimized. The constraints of the model should not only capture the interdependence of the vehicle and drones but also ensure that the LS regulatory rule is satisfied.

The fifth objective is to develop a solution methodology that can solve large problem instances of a mothership system satisfying the LS rule. The solution methodology implements a Multimodal-Based Heuristic (MBH) with randomization procedure to construct near optimal vehicle and LS-mandated drone routes. The performance of the MBH is benchmarked by comparing its performance against that of a Single-Mode-Based Heuristic (SBH).

The sixth objective is to conduct several experiments (1) to examine the performance of the three heuristics developed, (2) to illustrate the capability of the developed model in answering a wide variety of questions related to the planning of the basic mothership delivery system, and (3) to allow the service providers decide on the most suitable equipment configuration (vehicle-only system vs. integrated vehicle-drone system) for the service area under consideration based on the level of LS restrictions.

Finally, this research will present a case study that illustrates the application of the developed methodology in the downtown area of the City of Dallas, Texas, considering different customer spatial distributions.

1-8. Organization of the Dissertation

The dissertation is organized as follows. Chapter 2 presents a review of previous models developed for studying the integrated vehicle-drone routing problem. It provides a review of the different approaches used to develop such systems and other complementary problems related to the integrated vehicle-drone routing problem. Chapter 3 provides a formal definition and formulation of the HVDRP that studies the basic mothership system and of the IVDRP-LS that can solve optimally the mothership delivery system satisfying the LS rule. Chapter 4 presents the hybrid Clarke and Wright heuristic along with the vehicle-driven and drone-driven heuristics to solve large problem instances of a basic mothership system. Chapter 5 extends Chapter 4 by presenting a novel solution methodology that not only captures the multimodality of the problem, but also considers the LS constraints. Chapter 6 describes the experiments designed (1) to evaluate the developed solution methodologies, (2) to answer questions related to the planning of the basic mothership delivery system, and (3) to answer questions related to the impact of the LS rule on overall system performance. Chapter 7 provides the results of the case study, describing the application of the developed methodology in Dallas's downtown area. Finally, Chapter 8 provides concluding remarks and presents possible research extensions.

Chapter 2

BACKGROUND REVIEW

2-1. Introduction

This chapter reviews the literature related to the hybrid vehicle-drone routing problem (HVDRP). It starts with Section 2-2, which provides a review of the classical vehicle routing problem (VRP) and its common solution methodologies. Section 2-3 describes main extensions of the VRP, considering aspects shared with the HVDRP problem. Section 2-4 reviews the two-echelon location and routing problem (2E-LRP) and its suggested solution approaches. Similar to the HVDRP, the 2E-LRP considers two level trips. The upper-level trips start from the main depot to distribute goods to a set of satellite depots and return to the main depot. The lower-level trips serve the end customers. Another problem related to the HVDRP is the truck and trailer routing problem (TTRP), which is reviewed in Section 2-5. The problem requires a subset of customers to be visited by a truck-trailer pair, while other customers are visited by the truck alone. Section 2-6 reviews previous research work focusing on the drone routing problem and its different applications (e.g., surveillance applications, area coverage and delivery). Section 2-7 presents different models that take into consideration vehicle-drone integration for delivery services. Finally, Section 2-8 concludes this review and highlights main research gaps identified in the literature.

2-2. The Classical Vehicle Routing Problem

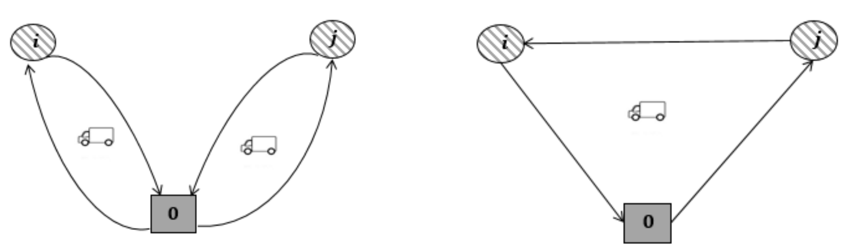
The vehicle routing problem (VRP) is a well-studied optimization problem that determines the optimal routes of one or more vehicles used to serve a set of customers. The problem was introduced in the pioneer work of Dantzig (1959), which is considered as a generalization of the traveling salesman problem (TSP) (Dantzig, 1954; Lawler, 1985). The solution of the problem entails determing the shortest tour among several customers, where the tour starts and ends at a fixed depot. Since then, the problem has been extensively studied and hundreds of papers studying different aspects of this problem have been published (Balinski and Quandt, 1964; Fisher and Jaikumar, 1978; Altinkemer and Gavish, 1991).

Lenstra and Kan (1981) studied the complexity of the VRP and concluded that the problem is NP hard as it cannot be solved in polynomial time. Many publications have considered this issue and proposed efficient algorithms to solve the problem. These algorithms are generally classified into three categories: exact algorithms, classic heuristic algorithms and metaheuristic algorithms.

Exact algorithms are designed to obtain optimal solutions for the problem, which are based on branch-and-bound and dynamic programming techniques. However, these algorithms can only be applied on small problems because they require high computation time. Examples of the exact algorithms include: set partitioning and column generation

(Balinski and Quandt, 1964), dynamic programing (Eilon et al., 1971), the k-degree center tree (Christofides et al., 1981), and the assignment lower bound and a related branch-and-bound algorithm (Laporte et al., 1986). Fisher and Jaikumar (1978) proposed a method for deterministic VRPs. They assumed that the VRP can be reduced to K TSPs. Since the TSP can be viewed as a linear program, the VRP may be solved optimally with Benders' decomposition (Benders, 1962).

Heuristic methods produce good quality solutions (close to optimal) in a reasonable running time. Clarke and Wright (1964) presented a widely-used algorithm for solving the VRP that was based on the saving concept. While the saving algorithm does not guarantee finding the optimal solution, it often yields a good solution that is close to the optimal solution. The saving concept is built on the idea that the cost saving is obtained from merging two routes into one route as shown in Figure 2-1 where node 0 is the depot. Excessive research has focused on improving the solution quality and the computation time of the saving algorithm (Paessens, 1988; Altinkemer and Gavish, 1991; Wark and Holt, 1994; Reimann et al., 2004).



(a) (b)

Figure 2-1: Saving concept illustration.

Christofides and Eilon (1969) applied the 2-opt and 3-opt to improve the solution where each method starts with a certain tour and improves it by applying small changes to the given route. In general, the algorithm of k-opt local search starts with an initial solution for a route and try to improve it by choosing k edges and trying to reconnect them in a different way. The way the k-opt is implemented is that it goes over all k-edges and over all ways of reconnecting them until no better solution can be obtained. Cullen and Jarvis (1981) introduced interactive heuristic for solving a broad class of routing problems. The heuristic adopts the cluster first route second approach, where a set partitioning formulation solved by means of column generation is considered. Since this approach is heuristic-based, the location-allocation subproblem are only solved approximately and not optimally.

Unlike the local optimization heuristic, the metaheuristics succeeded in leaving the local optimum by temperedly accepting the moves that worsen the objective function value. The drawback of the metaheuristics is that they do not guarantee finding the optimal solution. The probability of finding the global optimum increase with the increase in the computation time. Examples of metaheuristics used to solve the VRP include: tabu search (Fred Glover, 1986) simulated annealing (Corana et al., 1987), constraint programming (Shaw, 1988), genetic algorithms (Goldberg, 1989), and ant search algorithms (Bullnheimer et al., 1997).

2-3. Extensions of the Vehicle Routing Problem

Many problems have branched from the original VRP. A summary of different extensions of the TSP and VRP, their formulations and solution methodologies can be found in Golden et al. (2008), Eksioglu et al. (2009), and Braekers et al. (2014). These extensions include: vehicle routing problem with time window (VRPTW) where there are specific time windows to meet the demands (Solomon, 1987), inventory routing problem (IRP) that includes decisions on when to serve customers (Campbell, 1998), the period vehicle routing problem (PVRP) where vehicles tend to serve the customers over a specified period of time (Francis, 2007), and the consistent vehicle routing problem (ConVRP) that ensures that same vehicles serve same customers at the same time every day (Groër et al., 2008). Other versions of the vehicle routing problem that share some similarity with the HVDRP include the green vehicle routing problem (GVRP) (Erdogan, 2012) and the capacitated vehicle routing problem (CVRP) (Toth and Vigo, 2002).

The GVRP presented by Erdogan (2012), Schneider (2014), and Hiermann (2016) entails scheduling efficient routes for electric vehicles that need to stop at charging stations distributed in the network to recharge their batteries so they can extend the vehicles' travel distance. Failing to schedule proper stops for battery charging precludes the vehicles from completing their scheduled tour and/or returning to their depot. As such, scheduling stops for battery charging is considered as a hard constraint for the vehicles in the GVRP. Similar constraint should also be considered for the drones in the HVDRP as they need to be adequately charged to complete their tours and return back to the vehicle.

The CVRP is an extension of the VRP with additional vehicle capacity constraint (Christofides, 1976). Its similarity with the HVDRP is that the HVDRP involves capacity constraints that limit the travel distance and carrying capacity for the drones. The CVRP extends the TSP, and hence many exact approaches were inherited from the work done for the TSP. Some approaches extended the direct tree search with branch-and-bound algorithms to column generation and branch-and-cut algorithm. These exact algorithms can only solve small problems with limited number of customers. A review of exact algorithms based on the branch-and-bound approach is presented in Toth and Vigo (2002).

2-4. Two-Echelon Location Routing Problem (2E-LRP)

The 2E-LRP considers two trip levels. The upper-level trips are performed by large vehicles that start from the main depot to distribute goods and travel to a set of satellite depots before returning to the main depot. The lower-level trips are performed by small vehicles serving the end customers. These trips start and end at the satellite depots. The 2E-LRP was first used in applications of newspapers distribution and city logistics. In these applications, large trucks arriving from outside may be required to unload their goods at platforms located on the periphery of a city, from which smaller and more environmentally-friendly vehicles are allowed to continue downtown (Prodhon and Prins, 2014).

Jacobsen and Madsen, (1980) and Madsen, (1983) were the first to apply the 2E-LRP to the process of newspapers distribution. Three decisions are considered as part of the problem solution: number and location of transfer points, the structure of the first-level trips from the printing office to transfer points, and the structure of the second-level trips from the transfer points to the retailers.

Lin and Lie (2009) proposed a model that consists of a set of distribution centers, plants, big clients and small clients. The design decisions consider determining the location and number of distribution centers, the first level routing between plants, distribution centers and big clients, and the second level routing between distribution centers and small clients. They proposed a hybrid genetic algorithm embedded with routing heuristics. The chromosome of the genetic algorithm specified only the open satellites and big clients that are served in the first level trip. The routing heuristic consists of a cluster-based routing heuristic, followed by a local search heuristic. The heuristic starts by the second level trip to know the quantity shipped by each satellite, which then becomes demand for the first-level trip. The computational results showed that the difference between the suggested heuristic and the optimal solution is slightly less than 0.01%. Also, it was found that including some of the big clients in the first level trip might induce important savings.

Nguyen et al. (2012a) considered the 2E-LRP with a single central depot with known location, a set of capacitated satellites, and a set of customers. Unlike Lin and Lie (2009), all customers were served in the second level trip. The authors presented four constructive heuristics and a hybrid metaheuristic: a greedy randomized adaptive search procedure (GRASP) complemented by a learning process (LP) and path relinking (PR). The GRASP and the LP executed three randomized constructive heuristics to create trail solutions and applied a variable neighborhood descent (VND) to improve them. Then, the metaheuristic was implemented with PR, which can be applied to the main loop, as a post optimization step, or both. The numerical results showed that the suggested heuristic outperforms the previously published heuristics.

In Nguyen et al. (2012b), the same authors extended their work by considering the multi-start iterated local search (MS-ILS) that is reinforced by a path-relinking procedure (PR), used internally for intensification. The initial solutions were generated using three greedy randomized heuristics. The first heuristic built the second level routes by randomizing the extended Clarke and Wright algorithm (ECWA) described in Prins et al. (2006) for the location routing problem with capacitated depot (CLRP). The second heuristic was inspired by the nearest neighborhood heuristic for the TSP, where one satellite was opened at random and set of routes were constructed for it. The third heuristic is an insertion heuristic that constructed second-level routes one by one. Each ILS ran alternates between two search spaces which are the 2E-LRP solutions, and TSP solutions covering the main depot and the customers. Giant tours were converted into feasible solutions using three-phase splitting procedure by inserting satellites, partitioning the sub-sequence assigned to each satellite into second-level routes, and adding first-level routes to supply the selected satellites. The experiments reused the same two sets of instances used in their previous work. It was found that the MS-ILS + PR outperforms the previous GRASP by 0.8% on average but with longer running time.

The basic, most studied problem among the 2E-LRPs is the capacitated 2E-LRP (2E-CLRP) where both the first-level and second-level vehicles are capacitated. The fleet of both vehicles was assumed to be unlimited. The first-level trips visited the opened satellite, where each open satellite was visited exactly once. The second-level trips started from the opened satellite to serve the customers, and each customer was served only once. Contardo et al. (2012) introduced two algorithms to solve the 2E-CLRP. The first algorithm was a branch-and-cut algorithm that was strengthened using several families of valid inequalities that included first-echelon inequalities, second-echelon inequalities, separation algorithm, node selection strategy, branching strategy, and separation strategy. This algorithm is based on the decomposition of the 2E-CLRP into two CLRPs.

The second algorithm was the adaptive large neighborhood (ALNS). The ALNS was first proposed by Ropke and Pisinger (2006) to solve pickup and delivery problems with time windows. ALNS was proposed to be used with the 2E-CLRP based on the decomposition of the 2E-CLRP into two CLRPs. The algorithm starts by a search space that allows the exploration of infeasible solutions. Satellites that yield the lowest cost and can serve the total customer demand are opened. Then, the "destroy and repair" operators that remove, open, or swap the satellites of the initial solution are applied. Finally, a local search is applied to improve the CLRP solution. The computational results showed that ALNS outperformed the previously published heuristics. The branch-and-cut method provides tight lower bounds and is able to solve small- and medium-size instances to optimality within reasonable computing times.

Govindan et al. (2014) introduced a two echelon location-routing problem with time window (2E-LRPTW). A multi-objective optimization model that integrated sustainability in decision making for distribution in a perishable food supply chain network (SCN) was proposed. 2E-LRPTW aims to reduce carbon footprint and greenhouse gas emission in addition to determining the number and locations of facilities and optimizing the amount of products delivered to lower stages and routes at each level. The proposed heuristic, MHPV, consists of a hybrid of two algorithms named multi-objective particle swarm optimization (MOPSO) and adapted multi-objective variable neighborhood search

(AMOVNS). The results showed that the hybrid approaches outperform the existing models.

2-5. Truck and Trailer Routing Problem (TTRP)

The truck and trailer routing problem (TTRP) extends the VRP where a fleet of trucks and trailers, each with fixed capacity Q_k and Q_i, respectively, serve a set of customers. A complete vehicle that consists of a truck and a trailer is assumed to have a capacity of Q_k+ Q_i. The number of trucks should be greater than or equal to the number of trailers. Customers are assumed to be served from the main depot and are divided into two sets: vehicle customers who are reachable by either a complete vehicle or by a truck only, and truck customers who are reachable by a truck only. The solution of the TTRP consists of three types of routes: pure truck route, pure vehicle route, and complete vehicle route. A pure truck route is traveled by the truck alone, while a vehicle route is traveled by a complete vehicle only. A complete vehicle route consists of a complete vehicle's main tour with one or more sub tours traveled by truck only.

The HVDRP is similar to the TTRP in the sense that both require routing two types of vehicles. The TTRP integrates the truck and the trailer, while the HVDRP integrates the vehicle and the drones. Similarly, the solution for the HVDRP consists of three types of routes: vehicle only routes, drone only routes, and vehicle-drone routes. However, while the drones and vehicles move independently, a trailer can only be moved by connecting it to a truck. Considering these similarities, we review main research work devoted to solving the TTRP considering different operational conditions.

Gerdessen (1996) discussed a related problem named the vehicle routing problem with trailer (VRPT) where there were two assumptions entranced to simplify the problem: 1) each customer possesses unit demand; and 2) the trailer is parked exactly once. In this work, three construction heuristics are proposed followed by an improvement heuristic. Two real world applications for the VRPT are considered. The first one is the distribution of the dairy products in large cities with a heavy traffic and limited parking space. These conditions made delivering with complete vehicle (truck-trailer) very difficult, and required parking the trailer in order for the truck to be able to serve certain customers. Another real world example is the delivery of compound animal feed to farmers where there might be narrow roads or small bridges that cannot be traversed by truck pulling a trailer.

Chao (2002) developed a solution methodology to solve the TTRP. In this work, several assumptions were made on the cost and demand to simplify the problem. From the cost perspective, the cost is assumed to be proportional to the distance traveled by the fleet. The difference in the travel cost between the complete vehicle and the truck alone is ignored. Also, the work ignored the cost of trailer parking, the cost of shifting demands between the truck and its pulling trailer, and the fixed cost of maintaining the fleet.

From the demand perspective, the total demand load carried in a pure truck route or in a sub tour cannot exceed the truck capacity. However, the sum of all sub tours' demand load in a complete tour is allowed to exceed the truck capacity under the assumption that shifting the demand load from the truck to the trailer is acceptable, but the sum of the demand load in the main tour and all sub-tours cannot exceed the capacity of a complete vehicle. In addition, a tour length restriction is assumed for each route. The depot and every complete vehicle customer location can be considered a trailer-parking location.

The solution approach suggested in this paper starts by construction steps. These steps consisted of rate assignments where customers are allocated to routes by solving relaxed generalization assignments, route construction where routes are constructed using the cheapest insertion heuristic, and descent improvement where customers are moved among routes with the purpose of converting an infeasible solution to a feasible one. Finally, the solution is improved by applying a tabu search coupled with the deviation concept found in deterministic annealing. After applying this heuristic on 21 test problems, it was shown that the suggested heuristic can solve the TTRP effectively and efficiently.

Scheuerer (2006) adopted Chao's TTRP model and constructed two new construction heuristics, T-Cluster and T-Sweep, accompanied by a tabu search heuristic for solving it. The T-Cluster heuristic is a cluster-based sequential insertion procedure where routes are constructed by inserting customers one by one until the vehicle is fully utilized. The T-Sweep heuristic extends the approach of the classic sweep algorithm introduced by Gillett and Miller (1974). The heuristic constructs feasible routes by rotating a ray centered at the depot and including customers in the vehicle route gradually until the vehicle capacity is reached. Then, a new vehicle is used. The results presented in this work, which consisted of 21 benchmark problems, showed that the T-Cluster heuristic outperformed the T-Sweep heuristic and the construction heuristic presented in Chao (2002) in terms of solution quality.

Lin et al. (2009) proposed a simulated annealing (SA) heuristic for solving the TTRP. The SA heuristic is based on a local search heuristic that avoids local optimums by accepting the worst solution in some iterations. The heuristic consists of a list of customers that are classified as either vehicle-serviced customers or truck-serviced customers. Dummy zeroes define the first-level routes and sub-tours, and different types of vehicles that are represented by a vector of binary variables. The solution ensures that the capacity of the vehicle is not violated but the number of used vehicles might exceed the number of available vehicles. In such cases, the route combination approach is applied where a penalty term is added to the objective function to guide the search towards a feasible region. Three neighborhoods are used: two that randomly relocate and exchange customers and one that flips the type of vehicle used to serve the randomly selected customer.

Caramia and Guerriero (2009) developed an approach based on mathematical programming and local search. MIP is used to assign customers to a first-level trip with the objective of minimizing the fleet size that is used to serve them. A second IP is solved to build second-level routes. In case the second IP can produce disconnected sub-tours, a local search based on edge insertion is applied to repair the solution.

Several publications have also focused on problems that branched out from the TTRP. Lin et al. (2010) studied a relaxed TTRP (RTTRP) that ignores the constraint forcing a certain number of trucks and trailers. Lin et al. (2011) also extended their SA heuristic to apply it on TTRP with time window (TTRPTW). Another problem that branched from the TTRP is the single truck and trailer routing problem with satellite depot (STTRPSD) that assumes a single truck with trailer based at main depot that must serve customers accessible only by truck (Villegas et al., 2010).

2-6. The Drone Routing Problem

Research focusing on the drone routing problem has been expedited over the past few years. Applications that involve drone routing can generally be classified into surveillance applications and product delivery applications. A summary of models developed for the drone routing problem is given in Table 2-1. For example, the work of Grochlsky et al. (2006) focused on surveillance applications in which drones equipped with sensors cooperate with an unmanned a ground vehicle (UAG) to accurately locate a ground target. Shetty et al. (2008) considered a problem in which a fleet of drones is routed to serve a set of predetermined locations with different priorities. The drone routes are constrained by their flight range and payload capacity. A modeling framework is developed which decomposes the problem into a target assignment problem and a vehicle routing problem. A solution methodology that adopts a tabu search heuristic is developed to coordinate both problems.

Sundar and Rathinam (2014) studied a single drone routing problem where multiple depots are available for refueling it. They assumed that the drone can be refueled from any depot. The objective of their problem is to optimize the amount of fuel used by the drone by finding the drone's route where each customer is visited at least once and the fuel constraint is not violated. They proposed an approximation algorithm for the problem with a fast construction and improvement heuristics.

Table 2-1: Summary of drone routing research.

Authors	Problem: Multi-drones	Problem: multi customers served per drone route	Problem: Drone application	Problem: Multi-trip	Solution Method
Shetty et al. (2008)	✓	✓	Delivery	✓	Mixed-integer linear programing
Sundar and Rathinam (2014)	✓	✓	Delivery	✓	Route construction and improvement heuristic
Avellar et al. (2015)	✓	NA	Area Coverage		Mixed-integer linear programing
Grochlsky et al. (2006)	✓	NA	surveillance applications	NA	Search and localization algorithms
Fargeas et al. (2015)	✓	NA	surveillance applications	NA	Mathematical analysis
Dorling et al. (2016)	✓	✓	Delivery	✓	Simulated annealing heuristic
San et al. (2016)	✓		Delivery	✓	Genetic algorithm
Choi and Schonfeld (2017)	✓	✓	Delivery	✓	Mathematical analysis

Avellar et al. (2015) developed an optimization model for a minimum time area coverage using a fleet of drones taking into consideration the maximum flight time and the setup time. The number of drones used is chosen as a function of the size and the format of the area. The framework assumed that each drone can be used only once, ignoring the possibility of re-dispatching after battery recharging.

Fargeas et al. (2015) formulated the path planning problem for a group of drones patrolling a network of roads and pursuing intruders using unattended ground sensors (UGSs). They also presented a heuristic algorithm since the formulated problem was shown to be an NP hard problem. The suggested heuristic predicts intruder’s locations by using detections from the sensors. It also optimizes the vehicles' path by minimizing a linear combination of missed deadlines and the probability of not intercepting intruders.

Dorling et al. (2016) were among the first to study the drone delivery problem (DDP). A model was proposed which constructs drone routes that account for battery and payload weight limitations and allows for multiple deliveries per route. However, all drones were assumed to be dispatched and collected at a single depot. They introduced two multi trip VRPs, one that minimizes the cost subject to a limit and another that minimizes the overall delivery time subject to budget constraints. Certain assumptions were made for both problems which are: (a) the drones can fly as fast as they can at a constant speed; (b) the demand at each location can be served by one drone; and (c) there is enough fully charged batteries and hence no need to recharge the used ones.

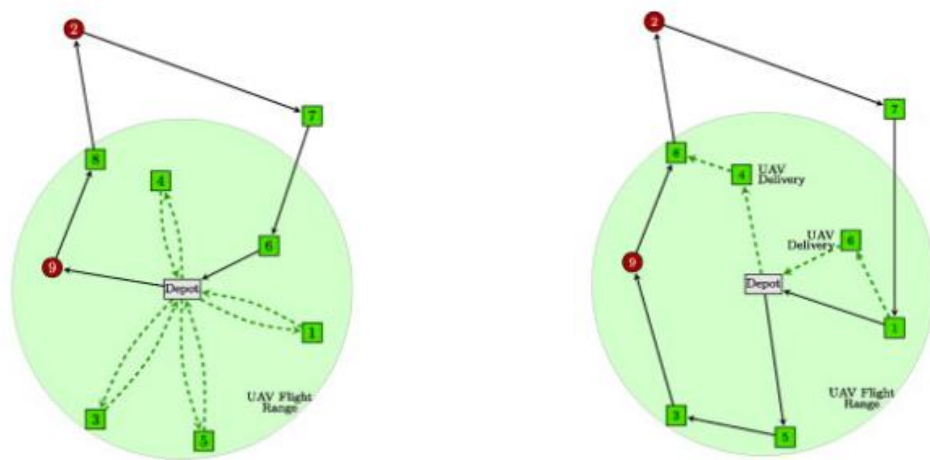
Since the problem is an NP hard, a simulated annealing (SA) heuristic is used for finding suboptimal solutions to practical scenarios. To balance cost and delivery time of the drone delivery process, the SA heuristic is used to show that the minimum cost has an inverse exponential relationship with the delivery time limit, and the minimum overall delivery time has an inverse exponential relationship with the budget. The drawback of the SA algorithm is that it does not take advantage of characteristics inherent to the VRP. For example, it does not benefit from the geographical information to avoid infeasible routes with two locations at opposite ends of the area of interest. The results showed that it is important for a drone delivery operation to consider optimizing battery weight and reusing drones. Optimizing battery weights resulted in improvements of over 10% as compared to solutions where each drone had an identical battery weight. Also, reusing drones led to considerable cost savings.

San et al. (2016) presented the implementation steps used to assign a swarm of drones to perform deliveries for targeted locations. They considered constraints related to the delivery process including flight range, carrying capacity, and volume of packages. It assumed that the drone can perform one delivery per dispatch. Their solution is based on Genetic Algorithm (GA) of multidimensional genes to solve multi-objective constraints. The proposed algorithm was capable of generating acceptable solutions quickly when dealing with a large amount of data in a real operation.

Finally, Choi and Schonfeld (2017) studied an automated drone delivery system that assumes that a drone can lift multiple packages within its maximum payload and serve recipients in a service area of given radius. Main assumptions considered for this automated system include: (a) a set of identical drones travel on a 3-dimensional Euclidean network, (b) the demand is uniformly distributed temporally and spatially assuming one package per customer, (c) the entire demand is served within a predetermined time period. The delivery vehicles traveled a round-trip line haul distance from the distribution center to demand points at a specified operating speed. Finally, the researchers conducted sensitivity analysis to explore how the system reacts to variations in the inputs.

2-7. The Vehicle-Drone Routing Problem

Research that takes into consideration vehicle-drone integration for delivery services has recently received considerable attention. A pioneer study on vehicle-drone integration for delivery services is presented in Murray and Chu (2015). They introduced the flying sidekick traveling salesman problem (FSTSP) and parallel drone scheduling TSP (PDSTSP) and both aim at minimizing the total travel time of the truck and the drone.



(a) Optimal PDSTSP solution (b) Optimal FSTSP solution

Figure 2-2: Comparison between PDSTSP and FSTSP solution (Murray and Chu, 2015).

The FSTSP considers a set of customers who can be served by either the drone or the truck. The deliveries that require a signature or the deliveries that exceed the carrying load capacity of the drones are served by the truck only. Certain operation conditions were assumed for the FSTSP. The drone can visit only one customer per dispatch but the truck can serve multiple customers while the drone is in flight. The drone is assumed to remain in constant flight which means that the truck should arrive at the collection location before the drone. The drone cannot be dispatched and collected at the same location. The truck cannot revisit any customer to collect a drone and cannot collect the stations at intermediate locations; it can only collect the drone at a location of a customer it is serving. The drone cannot visit any customer once it returns to the depot.

The PDSTSP is a combination of two classical problems. First, the TSP sequences customers who are assigned to be visited by the truck. Second, the parallel identical machine scheduling problem with a minimal makespan objective is used to schedule the remaining customers to a fleet of drones. The PDSTSP ignores the truck-drone integration. As such, the truck never carries, dispatches or collects the drones. They assume that both the truck and drones are dispatched from the depot where the truck serves customers along a TSP route, while the drones serve customers directly from the depot. They proposed an MIP formulation for both problems and two simple heuristics were developed and tested on small problem instances of up to 10 customers.

2-7-1. Research Work Extending the FSTSP

Ha et al. (2015) extended the FSTSP presented in Murray and Chu (2015) by considering the time span which represents the maximum allowable time that either the truck or the drone can wait for each other at the customer node. They introduced two methods to solve the problem: route-first-cluster-second and cluster-first-route-second. In more recent work, Ha et al. (2018) built on the FSTSP, but instead of minimizing the delivery completion time, they minimized the total operational cost in a problem they called traveling salesman with drone (TSP-D). The problem is formulated in the form of an MIP which was solved using a heuristic that adopts a greedy randomized adaptive search procedure (GRASP). GRASP is based on a new split procedure that optimally splits a TSP tour into a TSP-D solution. Then, the TSP-D solution is improved through a local search procedure. The results show that GRASP outperformed the methods presented in Murray and Chu (2015). Thus, GRASP demonstrates that not only it solves min-cost TSP-D but also min-time TSP-D.

Mathew et al. (2015) considered an integrated truck-drone system that consists of a carrier truck and a carried drone. The problem assumes that all the deliveries are performed by the drone while the role of the truck is to carry the drone and the delivery packages closer to the customers' location. It also assumes that the drone can perform only one delivery per dispatch and that the truck, unlike the FSTSP, can wait for the drone either in the same location it was dispatched from or at a different location. The problem was formulated as an optimal path-planning problem on a graph. Two algorithms were proposed, which are based on enumeration and a reduction to the traveling salesman problem.

The work presnted in Ferrandez et al. (2016) extended the FSTSP to meet two goals. First, it compares the time and energy savings between truck-drone delivery system and truck only system. Second, an optimization algorithm is developed that determines the number of optimal truck stops to dispatch/collect drones given the deliveries requirement and the number of drones on board of the truck. In this work, a k-means clustering algorithm is used to find the truck stops and a genetic algorithm is used to construct the truck TSP.

Carlson and Song (2017) used theoretical analysis to show that with the FSTSP presented in Murray and Chu (2015) the improvement in efficiency is proportional to the square root of the speed ratio of the truck and the drone. Marinelli et al. (2017) extended the FSTSP by relaxing the constraints that ensure that the drone must be dispatched and collected at a depot or a customer node. The authors maximized the drone usage in parcel delivering by allowing the drone to be dispatched and collected not only at a node but also along a route arc. A greedy randomized adaptive search procedure was developed to solve the problem.

Pugliese and Guerriero (2017) extended FSTSP to consider the time window constraint. The problem is modeled as a vehicle routing problem with time window with the objective of serving all the customers within their time window. The results show that the use of drones does not reduce the cost of delivery but is environmentally convenient and improves the service quality. Moshref-Javadi and Lee (2017) used the truck-drone delivery problem to minimize the waiting time of the customers in order to maximize customer satisfaction. They assumed that the truck waits at one stop until all drones perform their deliveries and return. Then, the truck can move to another stop. The results show that increasing the number of drones onboard of the vehicle and allowing for multiple delivery per drone dispatch can effectively reduce the customers' waiting time.

Luo et al. (2017) considered a similar problem but the drone can serve multiple customers per dispatch and the truck must dispatch and collect the drone at different locations. They proposed two heuristics: the first one constructed a complete tour for all the customers and split the drone routes, while the second heuristic constructed the truck's tour and assigned the drone routes to it. Although this work enhances the configuration of the flying side-kick system by allowing the drones to serve multiple customers per dispatch, it prevents the drone from returning to its dispatching node which could impact the overall system efficiency.

Chang and Lee (2018) extended the FSTSP to consider multiple drones that can be dispatched simultaneously from the truck while the truck cannot leave before collecting all drones. The drones are assumed to be dispatched and collected at the same location. They developed a nonlinear mathematical program combined with a clustering technique to determine the optimal stop locations for the vehicle to dispatch the drones.

Tu et al. (2018) and Murray and Raj (2020) suggested problems that assume multiple drones carried by a single truck, where the drone dispatches and collections are forced to be at different locations. Tu et al. (2018) developed the TSP with multiple drones (TSP-mD) that was solved by an adaptive large neighborhood search heuristic, while Murray and Raj (2020) proposed the multiple flying sidekick traveling salesman problem (mFSTSP) that was solved by a three-phased heuristic solution approach. Kitjacharoenchai et al. (2019) consider an mTSP with multiple drones per truck which is solved by an adaptive insertion heuristic. Jeong et al. (2019) extended the FSTSP to consider the energy consumption of drones and restricted flying areas. An MIP is presented along with a two-phase constructive and search heuristic that is used to solve real-world problem instances.

Agatz et al. (2018) studied a similar problem to the FSTSP, namely the TSP with drones, which allows the truck to wait at the collecting location for the drone to arrive. They provided an MIP which was solved using the truck-first-drone-second heuristic. The heuristic implements dynamic programming (DP) and local search techniques to determine efficient drone routes. Bouman et al. (2018) proposed an exact solution approach for the problem presented in Agatz et al. (2018) aiming to solve larger instances.

Yurek et al. (2018) developed a two-stage approach for the problem suggested by Agatz et al. (2018). The first stage generates the TSP tours by DP and determines the customers assigned by the drones; the second stage uses MIP to generate the complementary drone routes that obey the truck route constructed in the first stage. The algorithm starts with the shortest truck route and iteratively improves the assignment and routing decisions. The results show that the proposed algorithm is efficient as it was able to solve the uniform instances with a problem size of 12 customers in a reasonable amount of time, whereas existing studies (Agatz et al. (2018) and Murray and Chu (2015)) optimally solved problems with a maximum of 10 customers for the same execution time.

Poikonen et al. (2019) developed an approximate branch-and-bound algorithm for the same problem that was able to optimally solve 29 out of 30 instances of up to 10 nodes with a maximum gap of 0.05%. El-Adle et al. (2019) developed an enhanced MIP for the same problem with valid inequalities and processing schemes which can solve instances of 24 nodes optimally. Freitas and Penna (2020) proposed a hybrid heuristic named HGVNS to solve TSP with drones. The computational results show that the proposed approach is faster than the approach proposed by Agatz et al. (2018) for instances larger than 100 customers.

Finally, Poikonen and Golden (2020) introduced a k-multi-visit drone routing problem (k-MVDRP) that extends the FSTSP to consider single truck and multiple drones where each drone can serve multiple customers per dispatch. The drones are allowed to return to predefined locations other than the customers' locations. This work also considers the drone energy drain function that takes into account each package weight. They conducted several sensitivity analyses that show that drone speed and the number of drones carried by the truck have high impact on the objective value.

2-7-2. Research Work Extending the PDSTSP

Ham (2018) extended the PDSTSP presented in Murray and Chu (2015) by assuming the problem can apply not only to deliveries but also to pickups. Here, the drone can perform a pickup after delivering to a customer or can return directly to the depot. A constraint programing (CP) method is proposed to consider a multi-truck, multi-drone, and multi-depot problem constrained with a time window with the objective of minimizing the time required to perform all the deliveries.

Kim and Moon (2019) extended the PDSTSP, considering a single vehicle and multiple drones where the drones could be dispatched not only from the depot but also from pre-specified drone stations. A drone station can store and utilize a number of drones. However, drones stored at the station cannot be dispatched before the vehicle arrives at that station. An MIP is developed for the problem which is solved using an efficient decomposition approach that divides the problem into traveling salesman and parallel identical machine scheduling problems.

Table 2-2: Summary of research extending Murray & Chu (2015).

Authors	Problem					Solution Method
	Multi-trucks	Multi-drones	Multi customers served per drone route	Multi-trip	Considers LS rule	
Murray & Chu (2015)				✓		TSP route and re-assign route
Ha et al. (2015)		✓		✓		Route first cluster second and cluster first route second
Ha et al. (2018)		✓		✓		Heuristic that adopts greedy randomized adaptive search procedure
Bouman et al. (2017)				✓		Dynamic programing
Marinelli et al. (2017)				✓		Heuristic that adopts greedy randomized adaptive search procedure
Pugliese & Guerriero (2017)	✓	✓	✓	✓		Mixed Integer Program
Agatz et al. (2018)				✓		Route first cluster second
Ham (2018)	✓	✓	✓	✓		Constraint programing, and variable ordering heuristic
Mathew et al. (2015)				✓		Reduce to TSP and use TSP solver
Luo et al. (2017)			✓	✓		TSP route and split; route and reassign
Ferrandez et al. (2016)		✓		✓		K-means clustering and genetic algorithm
Moshref-Javadi & Lee (2017)		✓		✓		Mixed Integer Program
Yurek et al. (2018)				✓		Decomposition-based iterative optimization algorithm
Poikonen et al. (2019)				✓		Branch-and-bound approach
El-Adle et al. (2019)				✓		Enhanced MIP
Freitas and Penna (2020)				✓		Hybrid heuristic named HGVNS
Poikonen and Golden (2020)		✓	✓	✓		Mixed Integer Program

2-7-3. Other Problem Configurations

Problem configurations other than FSTSP and PDSTSP are also considered. Savuran and Karakaya (2016) developed a model that assumes a system consisting of a single truck and a single drone with the objective of minimizing the total travel distance of the drone by determining optimal vehicle stops. The problem is solved using a metaheuristic in the form of a Genetic Algorithm (GA).

Wang et al. (2017) introduced a more general problem called the vehicle routing problem with drone (VRP-D) that considers multiple trucks and drones with the objective of minimizing the total duration of the delivery mission. While no optimization framework is provided for the problem, the work focused on testing several worst-case scenarios to develop bounds on the best possible time savings for truck-drone integration compared to the truck-alone case. Each drone is assigned multiple customers per dispatch and the drone is set to return to its dispatching truck, which waits for the drone at the dispatching location. The work was later extended by Poikonen et al. (2017) to consider the limitation of the drones' battery life and extend the worst-case bounds to a more generic distance/cost matrix.

Carlsson and Song (2017) introduced the horsefly routing problem that assumes a single drone and a single vehicle. Unlike previous problems, in the horsefly problem, the drones' dispatching and collecting locations are not restricted at customers' locations. A continuous approximation model is used to replace combinatorial approaches which is known to be computationally expensive. Li et al. (2018) also used the continuous approximation approach to study the economic impact of using a truck and drone delivery system.

Boysen et al. (2018) evaluated the benefits of having multiple drones versus a single drone. They studied the complexity of a problem that considers a fixed vehicle route and determines a set of drone routes each defining a drone's dispatching and collecting locations and the customer serviced. Drones can serve multiple customers per dispatch. They introduced two MIP that can be integrated in a straight forward metaheuristic framework. One of the MIP was able to optimally solve instances of up to 100 customers.

Schermer et al. (2019) introduced an MIP solved using a metaheuristic for the VRP-D. The metaheuristic partitions the VRP-D into sub problems, starting with allocation and sequencing, and followed by assignment and scheduling. The metaheuristic was able to optimally solve 90% of 10 nodes instances. Wang and Sheu (2019) studied a variant of the VRP-D where drones may visit multiple customers per dispatch and could be exchanged between vehicles at certain hub nodes. They developed an MIP that is solved using branch-and-price algorithm that was able to find an optimal solution for instances of up to 15 nodes.

Finally, Karak and Abdelghany (2019) presented an integrated vehicle-drone routing system in the form of the mothership system, which (1) considers the dispatching of multiple drones simultaneously with each drone serving multiple customers as long as the drones' flight range and load-carrying capacity are not violated; (2) allows the drones to be collected from the dispatching location or any subsequent stop; and (3) minimizes the number of stops made by the vehicle by forcing the drone to serve all customers.

Table 2-3: Summary of research related to vehicle-drone integration.

Authors	Problem					Solution Method
	Multi-trucks	Multi-drones	Multi customers served per drone route	Multi-trip	Considers LS rule	
Carlson & Song (2017)				✓		Mathematical analysis
Wang et al. (2017)	✓	✓		✓		Worst case analysis
Poikonen et al. (2017)	✓	✓		✓		Worst case analysis
Savuran and Karakaya (2016)				✓		Genetic algorithm (GA).
Schermer et al. (2019)	✓	✓		✓		Metaheuristic partitions the VRP-D into sub problems
Wang and Sheu (2019)	✓	✓	✓	✓		Branch-and-price algorithm
Boysen et al. (2018)		✓	✓	✓		Mixed Integer Program
Li et al. (2018)				✓		Continuous approximation approach
Karak and Abdelghany (2019)		✓	✓	✓		Hybrid heuristic named HCWH

2-8. Summary

This chapter reviewed several topics related to the integrated truck-drone routing problem. The problems presented can be viewed as a generalization of the classical vehicle routing problem (VRP) in which a vehicle uses the shortest route to visit several customers and returns back to its depot. The literature discussed several versions of the classical VRP that share features of the HVDRP problem, such as the green vehicle routing problem (GVRP) and the capacitated vehicle routing problem (CVRP). The two-echelon location and routing problem (2E-LRP) and the truck and trailer routing problem (TTRP) are other studied problems that share similarities with the problems studied in this dissertation. The review covers the applications and methodologies developed for solving these problems.

The literature review reveals that previous research work regarding the usage of drones for delivery applications has focused primarily on either the drone routing problem or the integrated vehicle drone routing problem, which uses the drones only for the last mile of the trip. Models that consider full vehicle-drone routing integration in the form of a mothership system have not been developed. Most existing models focus on using the drones in the form of a flying side-kick system. In addition, these models are limited in terms of representing the main features and operational constraints that characterize the integrated vehicle-drone routing problem. For example, they force certain routes for the drones (e.g., drones are prohibited for returning to their dispatch locations for collection) and they force a certain strategy for drone dispatch and collection (e.g., either the dispatch-wait-collect or the dispatch-move-collect, but not both). In addition, they fall short of representing the operational limitations of the drones in terms of flight range and load carrying capacity in the context of the mothership delivery system. They also limit the drone usage to package delivery only for a pre-determined set of customers without the option to provide package pick-up services along their tours. Finally, existing models assume a homogenous drone fleet in terms of operation cost, flight range, and load carrying capacity, which might not be the case in real-world applications.

Moreover, all existing models fall short of considering the LS rule and its impact on the performance of these proposed systems. Incorporating regulatory rules, mandated by the aviation authorities, in models used for configuring drone-based delivery systems is vital for enabling their real-world deployment. Although the ultimate goal of using drones in performing delivery tasks is to promote a vehicle-free delivery system with reduced operation cost, frameworks that can examine the effect of mandating the LS rule on achieving this goal do not exist yet. This research extends the existing literature by developing a framework to study vehicle-drone integration for pick-up and delivery services considering the LS rule. The framework provides a platform for policy makers and service providers to design and evaluate the performance of drone-based delivery systems that ensure safe operations.

Chapter 3

PROBLEMS DEFINITION AND FORMULATION

3-1. Introduction

This Chapter formally defines the HVDRP that considers a basic mothership system and IVDRP-LS that considers a mothership system that obeys LS rule. Section 3-2 presents the list of variables and other notations used to formulate both problems. Section 3-3 presents all the assumptions considered by the HVDRP, which specify the operation scenarios of the proposed mothership system; presents the mathematical formulation of the HVDRP; and discusses the complexity of the HVDRP. Section 3-4 presents the additional assumptions considered by the IVDRP-LS that are related to the flying LS rule; presents the mathematical formulation of the IVDRP-LS; and discusses the complexity of the IVDRP-LS. Finally, Section 3-5 gives a summary of the chapter.

3-2. Problem Definition

This section presents the notations that describe data sets, model parameters, and decision variables used to develop the modeling framework for the HVDRP and the IVDRP-L.

Notations:

Sets:

G	Directed multimodal network
N	Set of nodes, indexed by i, j, k, l, m and $n \in N$
N_D	Subset of nodes that includes the depot node
N_V	Subset of station nodes
N_C	Subset of customer nodes
A	Set of links, indexed by node pair (i, j), where $i \in N$ and $j \in N$
D	Set of drones, indexed by $d \in D$

General Parameters:

q_m	Delivery weight of customer located at node $m \in N_C$
p_m	Pick-up weight of customer located at node $m \in N_C$
r_d	Maximum flight range of drone $d \in D$
w_d	Load-carrying capacity of drone $d \in D$
l_{ij}	The length of link $(i, j) \in A$
cv_{ij}	Average travel cost from node $i \in N$ to node $j \in N$ for the vehicle
cd_{ij}^{d}	Average travel cost from node $i \in N$ to node $j \in N$ for drone $d \in D$
$M1$	Very large positive number – the maximum possible distance travelled by the vehicle. One possible value for $M1$ is the vehicle traveled distance obtained using the basic TSP as it provides a good upper bound on the vehicle traveled distance.

$M2$ Very large positive number – the maximum possible load-carrying capacity by a drone $M2$= max (w_d $\forall\ d \in D$)

$M3$ Very large positive number – the maximum possible flight range by a drone $M3$= max (r_d $\forall\ d \in D$)

LS Parameters:

a_{ij} = 1 if node $j \in N$ is within the LS from station $i \in N_v$, and 0 otherwise

s_i = 1 if customer $i \in N_C$ is within LS from at least one station and reachable by the drones, and 0 otherwise

Decision variables:

x_{dij} = 1 if drone $d \in D$ traverses link $(i, j) \in A$ on-board of the vehicle, and 0 otherwise

y_{ij} = 1 if the vehicle traverses link $(i, j) \in A$, and 0 otherwise

z_{jdml} = 1 if drone $d \in D$ dispatched from station $j \in N_V$ travels on link $(m, l) \in A$, and 0 otherwise

b_{ijd} = 1 if drone d$\in D$ dispatched from station $j \in N_V$ is collected at station $i \in N_V$, and 0 otherwise

f_{jd} = 1 if drone $d \in D$ is dispatched from station $j \in N_V$, and 0 otherwise

dw_{ijd} = Delivery load carried by drone $d \in D$ after visiting node $i \in N$ and heading to node $j \in N$

pw_{ijd} = Pick-up load carried by drone $d \in D$ after visiting node $i \in N$ and heading to node $j \in N$

dst_{ijd} = Remaining flight range of drone $d \in D$ after visiting node $i \in N$ and heading to node $j \in N$

u_i = Specifies the order of node $i \in N$ in the vehicle route

d_{ij} = Total distance traveled by the vehicle after traveling on link $(i, j) \in A$

Consider a multimodal vehicle-drone network $G(N, A)$, where N is the set of nodes and A is the set of links. A set of drones D mounted on a vehicle are assumed to provide pick-up and delivery services for customers distributed in this network. The vehicle starts and ends its tour at a single depot. The set of nodes $N = N_D \cup N_V \cup N_C$ includes the depot node in $N_D = \{0\}$, the station nodes $N_V = \{1,2, \dots, |N_V|\}$ where the vehicle can stop to dispatch and collect the drones, and the customer nodes $N_C = \{|N_V| + 1, \dots, |N_V| + |N_C|\}$. The pick-up weight, p_m, and the delivery weight, q_m, are assumed to be given for each customer node $m \in N_C$. Each link, $(i, j) \in A$, is defined in terms of its length, l_{ij}, the average travel cost by the vehicle, cv_{ij}, and the average travel cost by each drone, cd_{ij}^d. These costs are assumed to be a function of the length of the link (i, j) and the cost per unit distance for each mode. The travel cost per unit distance for all drones is assumed to be less than that of the vehicle. Each drone, $d \in D$, is defined in terms of its maximum load-carrying capacity, w_d, and maximum flight range, r_d, which depends on its battery lifespan. A drone cannot exceed its maximum flight range or its load-carrying capacity. These sets and parameters are used to formulate both the HVDRP and the IVDRP-LS.

Since the IVDRP-LS extends the HVDRP to consider the LS rule extra parameters are used to formulate this problem. A two-dimensional visibility graph is constructed to determine visible drone destinations from the different dispatching stations (Frontera et al., 2017). Assuming that drones follow the Euclidian trajectory from their origins to destinations, if the straight line connecting any origin-destination pair is obstructed by any obstacle, then the destination is assumed to be out of sight from the origin. We use the parameter $a_{ij} \in \{0,1\}$ to define visibility between node pair ij, which is equal to one if node $j \in N$ is within the LS from station $i \in N_v$, and zero otherwise. The parameter $s_i \in \{0,1\}$ is used to represent if a customer location can be seen and reachable by a drone from any of the station nodes. Customers that are not within LS of any station (i.e., $s_i = 0$) are assumed to be served by the vehicle. The vehicle is assumed to have unconstrained load-carrying capacity. The problem requires determining the optimal route for the vehicle and the drones to serve all customers in the network such that the total travel cost for the vehicle and the drones is minimized and all drones stay within the LS from their dispatching stations until they return to their collection stations.

Several decision variables are defined for the HVDRP and the IVDRP-LS. To represent the vehicle route, we define $y_{ij} \in \{0,1\}$, which is equal to one if the vehicle travels on link $(i,j) \in A$, and zero otherwise. The variable $x_{dij} \in \{0,1\}$ is equal to one if drone $d \in D$ is mounted on the vehicle while traveling on link $(i,j) \in A$, and zero otherwise. The drone route is defined by the binary variable $z_{jdlm} \in \{0,1\}$, which is equal to one if drone $d \in D$ dispatched from station $j \in N_V$ travels on link $(l,m) \in A$, and zero otherwise. The delivery and pick-up load carried by drone $d \in D$ after departing from node $i \in N$ and heading to node $j \in N$ is given by $dw_{ijd} \geq 0$ and $pw_{ijd} \geq 0$, respectively. The variable $dst_{ijd} \geq 0$ defines the remaining flight range of drone $d \in D$ after traveling on link $(i,j) \in A$ and heading to node $j \in N$. This decision variable is used to determine if the drone has the adequate flight range to travel on link $(i, j) \in A$. Additional variables are also used to define the vehicle-drone interaction. We define the variable $f_{jd} \in \{0,1\}$, which is equal to one if drone $d \in D$ is dispatched from the vehicle at node $j \in N_V$. Also, the variable $b_{ijd} \in \{0,1\}$ is equal to one if drone $d \in D$ dispatched at node $j \in N_V$ is collected by the vehicle at node $i \in N_V$. To track the vehicle's traveled distance, we introduce the variable d_{ij}, which is defined as the total distance traveled by the vehicle after traveling on link $(i, j) \in A$. Finally, the variable u_i is used to ensure sub-tour elimination for the vehicle such that $1 \leq u_i \leq |N_V \cup N_C| + 2$.

3-3. Hybrid Vehicle Drone Routing Problem (HVDRP)

This section is organized as follows; subsection 3.3.1 presents all the assumptions considered by the HVDRP, subsection 3.3.2 presents the mathematical formulation of the HVDRP, and subsection 3.3.3 discuss the complexity of the formulation.

3-3-1. HVDRP Assumptions

The following assumptions are considered by the HVDRP, which specify the operation scenarios of the proposed mothership system:

1. Multiple drones are mounted on a single vehicle.
2. Each drone can serve more than one customer per dispatch as long as its flight range and load carrying capacity are not violated.
3. Drones can return to any station along the vehicle route, which could be the same as or different from the dispatching one.

4. Multiple drones can be dispatched simultaneously from any station, which allows the use of a swarm of drones to enhance the overall productivity of the system.
5. Each station can be visited by the vehicle only once.
6. Customers are served only by drones.
7. Vehicles are used only as mobile depots for the drones in order to reduce the required number of vehicle stops.
8. Drones can be dispatched and collected several times from the same station.
9. The vehicle cannot move from a station before collecting all drones that are planned to return to that station.
10. Drones that arrive to a collection station early are assumed to wait for the vehicle in idle conditions before being assigned a new tour. This assumption in conjunction with assumption 9, ensures a proper visitation sequence for the vehicle and the drones.
11. Drone batteries are replaced with fully charged batteries each time they are collected by the vehicle.
12. Packages are loaded and unloaded from the drones once the drones have been collected by the vehicle.

Each drone is defined in terms of its maximum flying range and load carrying limitation. The vehicle starts and ends its route at a depot and stops at selected stations to dispatch and/or collect the drones. The stations are locations where the vehicle and drones may wait for each other for collection. This configuration allows the vehicle to accommodate multiple dispatches of the same drone from a certain station to serve a dense customer population around that station. As the system involves multiple drones, these drones are expected to arrive at their collection stations at different times. Drones that arrive early wait in idle condition until the vehicle arrives. At any of these stations, drones could be dispatched such that each drone visits one or more customers to pick-up and/or deliver their packages. Each package is defined in terms of its weight.

The drone route must ensure that its maximum flying range and load carrying capacity are not violated. A drone may return to any station along the vehicle's route for collection. After battery replacement and package loading/unloading, the drone can be dispatched again to serve a new set of customers. The process is repeated until all customers in the service area have been reached. This configuration takes advantage of the expected reduced drone operation cost, compared to the vehicle cost, and provides more flexibility in routing the drones and the vehicle. Thus, the system is able to provide efficient integration between the vehicle and the drones to reduce dependence on the vehicle and increase the use of drones in performing the pick-up and delivery services. The resulting system is expected to reduce the total system operation cost, alleviate congestion associated with urban trucking, and enhance the drivers' work conditions.

3-3-2. HVDRP Mathematical Formulation

This section presents the mathematical formulation developed for the HVDRP. This formulation presents a first attempt to model the mothership system. While the formulation presents a set of variables and constraints that capture the unique aspects of the problem, it also takes advantage of the similarities that exist between the HVDRP with the TTRP and CVRP. For example, it extends the TTRP to represent the vehicle transportation of the drones along the different links by using variables y_{ij} and x_{dij}, which describe the movement of the vehicle and any mounted drones. It also borrows features from existing formulations of the CVRP such as variables dw_{ijd}, pw_{ijd}, and dst_{ijd} that track the drones' load carrying capacity and flying range. In addition, a new dimension is added to the decision variable used to describe the drone (i.e., capacitated vehicles) routing decisions, z_{jdlm}, in order to match the drones with their dispatching stations and hence capture the vehicle-drone interactions aspect of the problem.

As presented below the problem is modeled in the form of an MIP. Considering an objective function that minimizes the total operation cost for the vehicle and the drones, four main sets of constraints are defined as follows:

- Depot constraints,
- Vehicle constraints,
- Drone constraints, and
- Vehicle-drone interaction constraints.

The expression in (1) and equations (2)-(43) describe the MIP for the HVDRP.

Objective Function:

$$\text{Minimize} \quad \sum_{i \in N_1} \sum_{j \in N_1} y_{ij} \cdot cv_{ij} + \sum_{j \in N_v} \sum_{d \in \mathrm{D}} \sum_{m \in \mathrm{N}} \sum_{l \in \mathrm{N}} z_{jdml} \cdot cd_{ml}^{d} \tag{1}$$

Depot Constraints:

$$\sum_{j \in N_v} \mathrm{y}_{kj} = 1 \qquad \forall\, k \in N_D \tag{2}$$

$$\sum_{j \in N_v} y_{jk} = 1 \qquad \forall\, k \in N_D \tag{3}$$

$$\sum_{j \in N_v} \sum_{d \in D} \sum_{k \in N_D} x_{dkj} = \sum_{j \in N_v} \sum_{d \in D} \sum_{k \in N_D} x_{djk} \tag{4}$$

Vehicle Constraints:

$$\sum_{\substack{j \in N_1 \\ j \neq i}} y_{ij} = \sum_{\substack{j \in N_1 \\ j \neq i}} y_{ji} \qquad \forall\, i \in N_1 \tag{5}$$

$$d_{ki} \leq l_{ki} + M1 \times (1 - y_{ki}) \qquad \forall\, i \in N_1, \forall\, k \in N_D \tag{6}$$

$$d_{ki} \geq l_{ki} - M1 \times (1 - y_{ki}) \qquad \forall\, i \in N_1, \forall\, k \in N_D \tag{7}$$

$$d_{ij} \leq d_{ki} + l_{ij} + M1 \times \left(2 - y_{ij} - y_{ki}\right) \qquad \forall\, i \in N_V, \forall\, j \in \{N_1: j \neq i\}, \forall\, k \in \{N_1: k \neq i\} \tag{8}$$

$$d_{ij} \geq d_{ki} + l_{ij} - M1 \times \left(2 - y_{ij} - y_{ki}\right) \qquad \forall\, i \in N_V, \forall\, j \in \{N_1: j \neq i\}, \forall\, k \in \{N_1: k \neq i\} \tag{9}$$

$$d_{ij} \leq M1 \times y_{ij} \qquad \forall\, i \in N_1, \forall\, j \in N_1 \tag{10}$$

Drone Constraints:

$$z_{jdlm} \leq \sum_{k \in N_C} z_{jdjk} \qquad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N, \forall\, j \in N_V \tag{11}$$

$$z_{jdik} = 0 \qquad \forall\, d \in D, \forall\, k \in N, \forall\, i \in N_1, \forall\, j \in \{N_1: j \neq i\} \tag{12}$$

$$\sum_{j \in N_v} \sum_{d \in D} \sum_{l \in N} z_{jdlm} = 1 \qquad \forall\, m \in N_C \tag{13}$$

$$\sum_{\substack{l \in N \\ l \neq m}} z_{jdlm} = \sum_{\substack{n \in N \\ n \neq m}} z_{jdmn} \qquad \forall\, d \in D, \forall\, m \in N_C, \forall\, j \in N_V \tag{14}$$

$$dw_{ikd} + pw_{ikd} \leq w_d \qquad \forall\, d \in D, \forall\, i \in N, \forall\, k \in N \tag{15}$$

$$\sum_{k \in N} pw_{jkd} = M2 \times (1 - z_{jdjm}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in N_C \tag{16}$$

$$dw_{mkd} \geq dw_{lmd} - q_m - M2 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{17}$$

$$dw_{mkd} \leq dw_{lmd} - q_m + M2 \times (2 - z_{jdlm} - z_{jdmk}) \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{18}$$

$$pw_{mkd} \geq pw_{lmd} + p_m - M2 \times (2 - z_{jdlm} - z_{jdmk}) \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{19}$$

$$pw_{mkd} \leq pw_{lmd} + p_m + M2 \times (2 - z_{jdlm} - z_{jdmk}) \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{20}$$

$$dw_{lmd} \leq \sum_{j \in N_v} M2 \times z_{jdlm} \quad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{21}$$

$$pw_{lmd} \leq \sum_{j \in N_v} M2 \times Z_{jdlm} \quad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{22}$$

$$dst_{jkd} = r_d \times z_{jdjk} \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N \tag{23}$$

$$dst_{ikd} \geq l_{ik} \times z_{jdik} \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, (i,k) \in A \tag{24}$$

$$dst_{mkd} \geq dst_{lmd} - l_{lm} - M3 \times (2 - z_{jdlm} - z_{jdmk}) \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{25}$$

$$dst_{mkd} \leq dst_{lmd} - l_{lm} + M3 \times (2 - z_{jdlm} - z_{jdmk}) \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N: m \neq j\}, \forall\, l \in \{N: l \neq m\}, \forall\, k \in \{N: k \neq m\} \tag{26}$$

$$dst_{lmd} \leq \sum_{j \in N_v} M3 \times z_{jdlm} \quad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{27}$$

Vehicle-Drone Integrating Constraints:

$$b_{ijd} \leq \sum_{l \in N_c} z_{jdli} \quad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{28}$$

$$b_{ijd} \geq z_{jdki} \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N_C, \forall\, i \in N_V \tag{29}$$

$$f_{jd} \leq \sum_{m \in N_c} z_{jdjm} \quad \forall\, d \in D, \forall\, j \in N_V \tag{30}$$

$$f_{jd} \geq z_{jdjk} \quad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N_C \tag{31}$$

$$\sum_{\substack{k \in N_1 \\ k \neq i}} x_{dki} + \sum_{j \in N_1} \sum_{m \in N_c} z_{jdmi} = \sum_{\substack{k \in N_1 \\ k \neq i}} x_{dik} + \sum_{m \in N_c} z_{idim} \quad \forall\, d \in D, \forall\, i \in N_V \tag{32}$$

$$x_{dij} \leq y_{ij} \quad \forall\, d \in D, \forall\, i \in N_1, \forall\, j \in \{N_1: j \neq i\} \tag{33}$$

$$\sum_{k \in N_1} y_{ki} \geq b_{ijd} \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_1 \tag{34}$$

$$y_{ji} \geq y_{ij} + M1 \times (b_{ijd} - 1) \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{35}$$

$$\sum_{k \in N_1} d_{ki} \geq \sum_{m \in N_1} d_{mj} - M1 \times (1 - b_{ijd}) \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{36}$$

$$\sum_{k \in N_1} x_{dki} + \sum_{\substack{j \in N_1 \\ j \neq i}} b_{ijd} \geq f_{id} \qquad \forall\, d \in D, \forall\, i \in N_V \tag{37}$$

$$b_{ijd} \leq f_{jd} \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{38}$$

$$u_i - u_j + N_1 \times y_{ij} \leq N_1 - 1 \qquad \forall\, i \in N_1, \forall\, j \in \{N_v\colon\, j \neq i\} \tag{39}$$

$$u_0 = 1 \tag{40}$$

$$y_{ij} \in \{0,1\},\ x_{dij} \in \{0,1\},\ z_{jdlm} \in \{0,1\},\ b_{ijd} \in \{0,1\}, f_{jd} \in \{0,1\},\ dw_{lmd} \geq 0, \\ pw_{lmd} \geq 0, dst_{lmd} \geq 0, d_{ij} \geq 0, 1 \leq u_i \leq Nv\,(size) + 2 \quad \forall\, d \in D, \forall\, i \in N_1, \forall\, j \in N_1, \forall\, l \in N, \forall \\ \in N \tag{41}$$

The objective function given in (1) minimizes the total operation cost for the multimodal network. The first term represents the operation cost of the tour constructed for the vehicle to dispatch and collect the drones. The second term represents the operation cost of the tours constructed for the drones to visit all customers. Constraints (2) and (3) ensure that the vehicle starts and ends its tour at the depot. Constraint (4) ensures that all drones return back to the depot.

Constraint (5) guarantees path continuity for the vehicle. Constraints (6) to (9) track the distance traveled by the vehicle as it moves out from the depot or any intermediate station. These constraints are nonbinding if the vehicle does not travel on link (i, j). Thus, constraint (10) ensures that $d_{ij}\ \forall\ (i, j) \in A$ is equal to zero if the vehicle does not traverse link (i, j).

Constraints (11) to (27) ensure the feasibility of the tours constructed for each drone. Constraints (11) and (12) state that each drone starts its tour from its dispatching station. Constraint (13) ensures that each customer is served by one drone. Constraint (14) guarantees the continuity of the tour constructed for each drone. Drones cannot be loaded beyond their maximum carrying load capacity as described in constraint (15). Constraint (16) ensures that each drone leaves the vehicle carrying the required delivery load to serve its designated customers. Constraints (17) to (20) update the delivery and pick-up carrying load for each drone at each customer node. Constraints (21) and (22) ensure that dw_{lmd} and pw_{lmd} are equal to zero, if drone $d \in D$ does not travel on link $(l, m) \in A$. Constraints (23) to (27) ensure that the flight range for each drone is not violated. Constraint (23) mandates that each drone starts its tour with a fully charged battery (i.e., full flight range). Constraint (24) ensures that the drone's flight range is sufficient for the drone to reach its destination. Constraints (25) and (26) update the remaining flight range based on the traveled distance for each drone. Constraint (27) ensures that the available flight range (battery lifespan) for a drone is not decremented if a link is not traveled by that drone.

The remaining constraints capture the interactions between the vehicle and drones for dispatch and collection. Constraint (28) states that the decision variable b_{ijd}, which defines the dispatching and collection stations for drone $d \in D$, is equal to one if the drone dispatched from station $j \in N_V$ is collected at station $i \in N_V$. Note that the same station could be used for dispatching and collecting the drone, that is $i = j$. Constraint (29) requires that if a drone returns to a station, the vehicle must pick-up that drone from that station. Constraint (30) and (31) ensure that the value of the decision variable f_{jd}, which is used to specify the dispatching station of the drone, is equal to one if drone $d \in D$ is dispatched from station $j \in N_V$. Constraint (32) guarantees drone flow conservation at all stations. Constraint (33) ensures that the vehicle can carry a drone a long a link only if the vehicle is scheduled to travel on that link. Constraint (34) requires that the vehicle visits the collection station where the drone is scheduled to return. Constraints (35) and (36) ensure that if a drone is dispatched and collected from two different stations, the vehicle must visit the dispatching station before the collection station. In constraint (37), the drone must be collected by the vehicle to replace/recharge its battery and load it with a new set of packages before it is dispatched. Constraint (38) states that a drone cannot be collected if it was not dispatched in the first place. Sub-tour elimination is provided by constraints (39) and (40). Finally, constraint (41) forces the binary and non-negativity conditions for the variables.

3-3-3. HVDRP Formulation Complexity

To understand the complexity of the HVDRP, one can view its formulation as an extension of the conventional formulation of the VPR (1) to construct the vehicle route, (2) to construct the drone routes, and (3) to ensure correct integration of the two modes. Thus, the complexity for the HVDRP depends on the complexity of these three problem components. For the vehicle routing decisions, in addition to determining the set of visited stations and their sequence in the tour, the problem also entails deciding on the dispatching and collection of drones at different stations and the transportation of drones by the vehicle along different links. The vehicle routing component of the HVDRP is an NP-hard problem where the number of solutions grows exponentially with the number of stations in the network.

For the drone routing decisions, the problem is an extension of the CVRP, where each drone is modeled as a vehicle with limited capacity (i.e., flying range and the load carrying capacity). It requires determining feasible routes for all drones, where each drone route is defined in terms of its dispatching and collecting stations, the set of customers to be visited, and the sequence by which these customers are inserted in the drone route. The drone routing component is also NP-hard as the number of solutions grows exponentially with the number of stations and customers in the network. Thus, to appreciate the level of complexity of the HVDRP, one should try to answer the following question: do the vehicle-drone integration constraints specifically introduced to model the mothership system reduce the search space for the vehicle and drone routing decisions? Considering the interdependence between the vehicle and the drone routing decisions, their solution spaces cannot be reduced a priori. The HVDRP requires examining the combinations of the vehicle and the drone routing decisions, while ensuring the feasibility of the integrated solution with respect to station visitation sequencing, to determine the optimal integrated vehicle-drone routing scheme. This additional check has a combinatorial complexity considering the routing combinations for the vehicle and the drones in constructing integrated solutions.

As such, the HVDRP is an NP-hard problem with a higher level of complexity compared to the conventional VRP and the CPRP. Thus, obtaining provably optimal solutions using this formulation for the HVDRP are limited to small problem instances, and efficient methodologies are needed in order to provide good solutions within a practical running time to suit real-world applications of the mothership system.

3-4. Integrated vehicle-drone routing problem with the LS rule (IVDRP-LS)

This section is organized as follows; subsection 3.3.1 presents all the assumptions considered by the IVDRP-LS, subsection 3.3.2 presents the mathematical formulation of the IVDRP-LS, and subsection 3.3.3 discusses the complexity of the formulation.

3-4-1. IVDRP-LS Assumptions

The IVDRP-LS assumes multiple drones mounted on a single vehicle. Each drone is defined in terms of its maximum flying range and load-carrying capacity. The vehicle starts and ends its route at a depot and stops at selected stations to dispatch and/or collect the drones. At any of these stations, drones could be dispatched such that each drone visits one or more customers to pick up and/or deliver their packages. If a customer is not reachable by any of the drones, this customer should be visited by the vehicle. The vehicle is allowed to visit a station/customer only once. Also, the vehicle cannot dispatch/collect drones at any customer location. Each customer package is defined in terms of its weight. The maximum flying range and load-carrying capacity of all drones must not be violated. Drones that arrive early at a collection station are assumed to wait for the vehicle in idle conditions before being assigned a new tour. After each drone collection, the drone's battery is replaced, preparing it for a new dispatch. Drones can be dispatched and collected several times from the same station. The process is repeated until all customers are served. The following additional assumptions are related to the flying LS rule:

1. All customers visited by a drone must be within the LS from the drone's dispatching station.
2. A drone may return to any station along the vehicle's route for collection as long as the collecting station is within the LS from the dispatching station.
3. The vehicle cannot depart a station before all drones that are dispatched at this station finish their tours.
4. Customers that are not within any station's LS nor reachable by drones are served by the vehicle.
5. For drones to stay within the pilot's LS, it is assumed that they do not land at any customer location (e.g., using ropes to drop/pick-up the packages as described in Vanian (2016)).

3-4-2. IVDRP-LS Mathematical Formulation

The problem is modeled in the form of an MIP as presented below. Considering an objective function that minimizes the total operation cost of the vehicle and the drones, the solution described in terms of the vehicle tour, denoted as VT, and the set of tours constructed for the drones, denoted as DT, must satisfy the following constraints:

A. The vehicle and drones that left the depot must return to the depot.
B. The drones cannot carry beyond their load-carrying capacity.
C. The drones cannot violate their maximum flight range.

D. The vehicle and drone tours must be compatible.

E. The vehicle and drone tours must meet the LS regulatory rule.

The MIP formulation:

$$\text{Minimize} \quad \sum_{i \in N} \sum_{j \in N} y_{ij} \cdot cv_{ij} + \sum_{j \in N_v} \sum_{d \in \mathrm{D}} \sum_{m \in \mathrm{N}} \sum_{l \in \mathrm{N}} z_{jdml} \cdot cd_{ml}^{d} \tag{42}$$

$$\sum_{j \in N} \mathrm{y}_{kj} = 1 \qquad \forall\, k \in N_D \tag{43}$$

$$\sum_{j \in N} \mathrm{y}_{jk} = 1 \qquad \forall\, k \in N_D \tag{44}$$

$$\sum_{j \in N} \sum_{d \in D} \sum_{k \in N_D} x_{dkj} = \sum_{j \in N} \sum_{d \in D} \sum_{k \in N_D} x_{djk} \tag{45}$$

$$\sum_{\substack{j \in N \\ j \neq i}} y_{ij} = \sum_{\substack{j \in N \\ j \neq i}} y_{ji} \qquad \forall\, i \in N \tag{46}$$

$$d_{ki} \leq l_{ki} + M1 \times (1 - y_{ki}) \qquad \forall\, i \in N, \forall\, k \in N_D \tag{47}$$

$$d_{ki} \geq l_{ki} - M1 \times (1 - y_{ki}) \qquad \forall\, i \in N, \forall\, k \in N_D \tag{48}$$

$$d_{ij} \leq d_{ki} + l_{ij} + M1 \times \left(2 - y_{ij} - y_{ki}\right) \qquad \forall\, i \in (N_V \cup N_C), \forall\, j \in \{N: j \neq i\}, \forall\, k \in \{N: k \neq i\} \tag{49}$$

$$d_{ij} \geq d_{ki} + l_{ij} - M1 \times \left(2 - y_{ij} - y_{ki}\right) \qquad \forall\, i \in (N_V \cup N_C),, \forall\, j \in \{N: j \neq i\}, \forall\, k \in \{N: k \neq i\} \tag{50}$$

$$d_{ij} \leq M1 \times y_{ij} \qquad \forall\, i \in N, \forall\, j \in N \tag{51}$$

$$u_i - u_j + (N+1) \times y_{ij} \leq N \qquad \forall\, i \in N, \forall\, j \in \{N_v \cup N_C: j \neq i\} \tag{52}$$

$$u_0 = 1 \tag{53}$$

$$z_{jdlm} \leq \sum_{k \in N_C} z_{jdjk} \qquad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N, \forall\, j \in N_V \tag{54}$$

$$z_{jdik} = 0 \qquad \forall\, d \in D, \forall\, k \in N, \forall\, i \in N_V, \forall\, j \in \{N_V: j \neq i\} \tag{55}$$

$$\sum_{\substack{l \in N \\ l \neq m}} z_{jdlm} = \sum_{\substack{n \in N \\ n \neq m}} z_{jdmn} \qquad \forall\, d \in D, \forall\, m \in N_C, \forall\, j \in N_V \tag{56}$$

$$dw_{ikd} + pw_{ikd} \leq w_d \qquad \forall\, d \in D, \forall\, i \in N, \forall\, k \in N \tag{57}$$

$$\sum_{k \in N} pw_{jkd} = M2 \times (1 - z_{jdjm}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in N_C \tag{58}$$

$$dw_{mkd} \geq dw_{lmd} - q_m - M2 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{59}$$

$$dw_{mkd} \leq dw_{lmd} - q_m - M2 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{60}$$

$$pw_{mkd} \geq pw_{lmd} + p_m - M2 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{61}$$

$$pw_{mkd} \leq pw_{lmd} + p_m + M2 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N_C : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{62}$$

$$dw_{lmd} \leq \sum_{j \in N_v} M2 \times z_{jdlm} \qquad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{63}$$

$$pw_{lmd} \leq \sum_{j \in N_v} M2 \times z_{jdlm} \qquad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{64}$$

$$dst_{jkd} = r_d \times z_{jdjk} \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N \tag{65}$$

$$dst_{ikd} \geq l_{ik} \times z_{jdik} \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, (i,k) \in A \tag{66}$$

$$dst_{mkd} \geq dst_{lmd} - l_{lm} - M3 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{67}$$

$$dst_{mkd} \leq dst_{lmd} - l_{lm} + M3 \times (2 - z_{jdlm} - z_{jdmk}) \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, m \in \{N : m \neq j\}, \forall\, l \in \{N : l \neq m\}, \forall\, k \in \{N : k \neq m\} \tag{68}$$

$$dst_{lmd} \leq \sum_{j \in N_v} M3 \times z_{jdlm} \qquad \forall\, d \in D, \forall\, l \in N, \forall\, m \in N \tag{69}$$

$$b_{ijd} \leq \sum_{l \in N_c} z_{jdli} \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{70}$$

$$b_{ijd} \geq z_{jdki} \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N_C, \forall\, i \in N_V \tag{71}$$

$$f_{jd} \leq \sum_{m \in N_c} z_{jdjm} \qquad \forall\, d \in D, \forall\, j \in N_V \tag{72}$$

$$f_{jd} \geq z_{jdjk} \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, k \in N_C \tag{73}$$

$$\sum_{\substack{k \in N \\ k \neq i}} x_{dki} + \sum_{j \in N_V} \sum_{m \in N_c} z_{jdmi} = \sum_{\substack{k \in N \\ k \neq i}} x_{dik} + \sum_{m \in N_c} z_{idim} \qquad \forall\, d \in D, \forall\, i \in N_V \tag{74}$$

$$\sum_{d \in D} \sum_{k \in N} x_{dkj} = \sum_{d \in D} \sum_{k \in N} x_{djk} \qquad \forall\, d \in D, \forall\, j \in N_C \tag{75}$$

$$x_{dij} \leq y_{ij} \qquad \forall\, d \in D, \forall\, i \in N, \forall\, j \in \{N: j \neq i\} \tag{76}$$

$$\sum_{k \in N} y_{ki} \geq b_{ijd} \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{77}$$

$$\sum_{k \in N} x_{dki} + \sum_{\substack{j \in N_V \\ j \neq i}} b_{ijd} \geq f_{id} \qquad \forall\, d \in D, \forall\, i \in N_V \tag{78}$$

$$y_{ji} \geq y_{ij} + M1 \times (b_{ijd} - 1) \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{79}$$

$$\sum_{k \in N} d_{ki} \geq \sum_{m \in N} d_{mj} - M1 \times (1 - b_{ijd}) \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{80}$$

$$b_{ijd} \leq f_{jd} \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V \tag{81}$$

$$\sum_{j \in N_v} \sum_{d \in D} \sum_{l \in N} z_{jdlm} = s_m \qquad \forall\, m \in N_C \tag{82}$$

$$z_{jdlm} \leq a_{jm} \qquad \forall\, d \in D, \forall\, j \in N_V, \forall\, l \in N, \forall\, m \in N \tag{83}$$

$$\sum_{\substack{j \in N \\ j \neq m}} y_{jm} = 1 - s_m \qquad \forall\, m \in N_C \tag{84}$$

$$y_{lm} \in \{0,1\},\ x_{dlm} \in \{0,1\},\ z_{jdlm} \in \{0,1\},\ b_{ijd} \in \{0,1\},\ f_{jd} \in \{0,1\},\ dw_{lmd} \geq 0,\ pw_{lmd} \geq 0, dst_{lmd} \geq 0,\ d_{lm} \geq 0,\ 1 \leq u_i \leq (N_v \cup N_C)\,(size) + 2 \qquad \forall\, d \in D, \forall\, i \in N_V, \forall\, j \in N_V, \forall\, l \in N, \forall\, m \in N \tag{85}$$

The objective function given in (42) minimizes the total operation cost for the multimodal network. The first term represents the operation cost of the tour constructed for the vehicle to dispatch and collect the drones and to visit the subset of customers that are not reachable by any of the drones due to LS restrictions. The second term represents the operation cost of the tours constructed for the drones to visit the subset of drone-reachable customers.

- Constraints (43) and (44) ensure that the vehicle starts and ends its tour at the depot. **(A)**
- Constraint (45) ensures that all drones return back to the depot. **(A)**
- Constraint (46) guarantees path continuity for the vehicle. **(Modeling VT)**
- Constraints (47) to (50) track the distance traveled by the vehicle as it moves out of the depot or any intermediate station. **(Modeling VT)**
- Constraint (51) ensures that d_{ij} $\forall\ (i,j) \in A$ is equal to zero if the vehicle does not traverse link (i,j). **(Modeling VT)**
- Constraints (52) and (53) guarantee sub-tour elimination for the vehicle. **(Modeling VT)**
- Constraints (54) and (55) ensure that each drone starts its tour from its dispatching station. **(Modeling DT)**
- Constraint (56) guarantees the continuity of the tour constructed for each drone. **(Modeling DT)**
- Constraint (57) ensures that the drones do not carry beyond their load-carrying capacity. **(B)**
- Constraint (58) guarantees that each dispatched drone carries the required delivery load to serve its designated customers. **(B)**
- Constraints (59) to (62) update the delivery and pick-up carrying load for each drone at each customer node. **(B)**

- Constraints (63) and (64) ensure that dw_{lmd} and pw_{lmd} are equal to zero, if drone $d \in D$ does not travel on link $(l, m) \in A$. **(B)**
- Constraint (65) guarantees that each drone initiates its tour with a fully charged battery. **(C)**
- Constraint (66) mandates that the drone has sufficient flight range to reach its destination. **(C)**
- Constraints (67) and (68) update the remaining flight range for each drone. **(C)**
- Constraint (69) ensures that dst_{lmd} is equal to zero, if drone $d \in D$ does not travel on link $(l, m) \in A$. **(C)**
- Constraint (70) and (71) require that if a drone returns to a station, the vehicle must pick up that drone from that station. **(D)**
- Constraint (72) and (73) ensure that the value of the decision variable f_{jd} is equal to one if drone $d \in D$ is dispatched from station $j \in N_V$. **(D)**
- Constraint (74) ensures the drone flow conservation at all stations. **(D)**
- Constraint (75) ensures that the vehicle does not dispatch or collect drones at any customer location. **(D)**
- Constraint (76) states that the vehicle can only carry a drone along a link if it is scheduled to travel on that link. **(D)**
- Constraint (77) and (78) mandate the vehicle to visit the collection station where the drone is scheduled to return. **(D)**
- Constraints (79) and (80) guarantee that if a drone is dispatched and collected from two different stations, the vehicle must visit the dispatching station before the collection station. **(D)**

- Constraint (81) states that a drone cannot be collected if it was not dispatched in the first place. **(D)**
- Constraint (82) ensures that each customer that is reachable by drones from a visible station is served by one drone. **(E)**
- Constraint (83) ensures that each node visited by a drone must be within the LS of the drone's dispatching station. **(E)**
- Constraint (84) ensures that each customer, not visible from any reachable station, is served by the vehicle. **(E)**
- Finally, constraint (85) forces the binary and non-negativity conditions for the variables.

3-4-3. Formulation Complexity

Similar to the HVDRP, the IVDRP-LS consists of three main problem components: (1) constructing the vehicle route, (2) constructing the drone routes, and (3) ensuring correct integration of the two modes while satisfying the LS rule. The first and second components were proven in subsection 3.3.3 to be NP hard problems. The third component requires examining the combinations of the vehicle and the drone routing decisions, while ensuring not only the feasibility of the integrated solution with respect to station visitation sequencing but also ensuring that the drone routes obey the LS rule. This additional check has a combinatorial complexity considering the routing combinations for the vehicle and the drones in constructing integrated solutions.

The LS rule could affect the complexity of the problem in two opposite directions. The complexity of the problem increases in cases where customers are obstructed but still can be served by the drones as more complex routing decisions need to be made for the drones to stay in the pilot's LS. On the other hand, as the problem includes more customers that cannot be reached by the drones, these customers are visited by the vehicle, thus reducing the computation effort for vehicle-drone routing integration. Nonetheless, in both cases, the IVDRP-LS is an NP hard problem, and hence obtaining provably optimal solutions using the mathematical formulation presented above is limited to small problem instances. Therefore, the next section presents efficient methodologies developed to provide good solutions within a practical running time to suit real-world applications.

3-5. Summary

This chapter presents a model for the hybrid vehicle-drone routing problem (HVDRP) for pick-up and delivery services and a model for the integrated vehicle-drone routing problem with the LS rule (IVDRP-LS) for pick-up and delivery services. Both problems formulated in the form of a mixed integer program which solves for the optimal vehicle and drone routes to serve all customers such that the total operational cost of the pick-up and delivery services is minimized. The HVDRP formulation captures the vehicle-drone routing interactions and considers the drones' operational constraints including flight range and load carrying capacity limitations. The IVDRP-LS formulation not only captures the vehicle-drone routing interactions and considers the drones' operational constraints including flight range and load carrying capacity limitations but also satisfies the LS regulatory rule.

Chapter 4

SOLUTION METHEDOLEGY FOR THE BASIC MOTHERSHIP SYSTEM

4-1. Introduction

This chapter presents a novel solution methodology that extends the classic Clarke and Wright algorithm to solve the HVDRP (Clarke and Wright, 1964), namely the hybrid Clarke and Wright heuristic (HCWH). The heuristic takes into consideration the cost savings for both the vehicle and the drones while solving for the optimal vehicle route. Thus, it generates an efficient multimodal vehicle-drone network. The performance of the HCWH is benchmarked against two other heuristics that are developed as a part of this research, which are the vehicle-driven heuristic (VDH) and the drone-driven heuristic (DDH). This Chapter is organized as follows. Section 4-2 presents an overview of the heuristics developed in this chapter. Sections 4-3, 4-4, and 4-5 describe the three heuristics, HCWH, VDH, and the DDH, respectively. Section 4-6 gives an example to describe the performance of the three heuristics. Finally, Section 4-7 summarizes the chapter.

4-2. Overview

As mentioned earlier, the HVDRP defined above is an NP-hard problem. As such, finding the exact optimal solution in a reasonable execution time is limited only to small-size problems. In this section, we present three heuristics-based solution methodologies that are developed to find a near-optimal solution for the problem. The heuristics extend the Clarke and Wright (CW) algorithm to consider the multimodality of the integrated vehicle-drone routing problem (Clarke and Wright, 1964). The CW algorithm uses the saving matrix concept to rank the merging process of two sub-routes into one large route. For the two nodes i and j with distance l_{ij}, the saving ϑ_{ij} is calculated as follows:

$$\vartheta_{ij} = l_{i0} + l_{0j} - l_{ij} \tag{86}$$

where l_{i0} and l_{0j} are the distances from nodes i and j to the depot node 0, respectively. This concept can be applied to the HVDRP to construct the vehicle route and the drone routes. For the vehicle, nodes i and j represent dispatching and collection stations while node 0 represents the depot. For a drone route, nodes i and j represent customers. The depot(s) for the drone is the closest station to the first visited customer and the closest station to the last visited customer. For each customer i, the closest station s_i is determined such that:

$$l_{is_i} = \min(l_{is_k} \, \forall \, s_k \in N_V) \tag{87}$$

Thus, the saving expression for a drone becomes:

$$\vartheta_{ij} = l_{is_i} + l_{s_j j} - l_{ij} \tag{88}$$

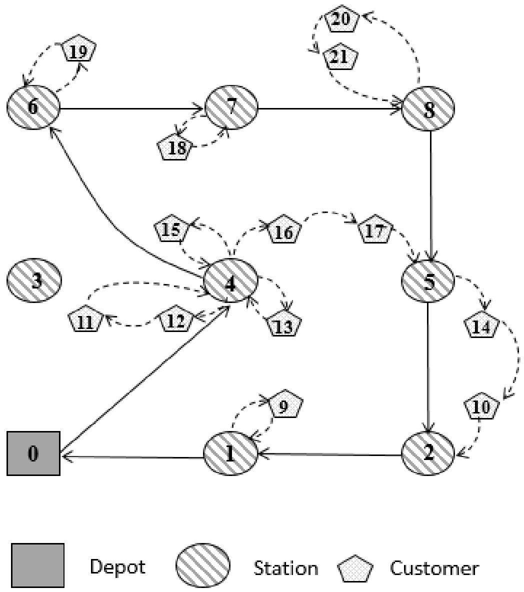
In Equation (88), s_i and s_j could be the same station if it is the closest to both customers, or two different stations if the closest station to customer i is different from that of customer j. For example, consider the network in Figure 4-1, customers 16 and 17 are close to stations 4 and station 5, respectively. Thus, their saving is calculated as follows:

$$\vartheta_{16-17} = l_{16-4} + l_{5-17} - l_{16-17} = 2.83 + 2.83 - 1 = 4.66 \text{ miles}$$

On the other hand, customers 20 and 21 are both close to station 8 resulting in the following saving:

$$\vartheta_{20-21} = l_{20-8} + l_{8-21} - l_{20-21} = 1.80 + 1.41 - 0.5 = 2.71 \text{ miles}$$

The following subsections describe the three heuristics developed for the HVDRP, namely the hybrid Clarke and Wright heuristic (HCWH), the vehicle-driven heuristic (VDH), and the drone-driven heuristic (DDH). The three heuristics consist of two route building procedures implemented in an iterative framework, one for the vehicle and one for the drones. For simplicity, we assume an identical set of drones in terms of their flight range, r, and load carrying capacity, w.



Node	X (mile)	Y (mile)	PW (lbs.)	DW (lbs.)
0	0.0	0.0	-	-
1	5.0	0.0	-	-
2	10.0	0.0	-	-
3	0.0	5.0	-	-
4	5.0	5.0	-	-
5	10.0	5.0	-	-
6	0.0	10.0	-	-
7	5.0	10.0	-	-
8	10.0	10.0	-	-
9	6.0	2.0	5.0	2.0
10	12.0	2.0	0.0	4.0
11	2.0	4.0	1.0	3.0
12	3.0	4.0	2.0	4.0
13	7.0	3.0	2.0	3.0
14	12.0	3.0	2.0	3.0
15	3.0	7.0	1.0	2.0
16	7.0	7.0	2.0	3.0
17	8.0	7.0	1.0	3.0
18	3.0	8.0	5.0	1.0
19	2.0	12.0	2.0	2.0
20	9.0	11.5	0.0	2.0
21	9.0	11.0	1.0	3.0

Vehicle cost = $2.0/mile
Drone cost = $1.0/mile

Figure 4-1: An example of a multimodal vehicle-drone network.

The HCWH constructs vehicle route and drone routes such that the total network cost is minimized. It adopts a multimodal cost-reduction greedy strategy that combines vehicle and drone cost savings to construct efficient intermodal routes that minimize the total operation cost for the vehicle and drones. In the VDH, the vehicle route is first optimized ignoring the drone routes, resulting in a set of stations that can be used to dispatch and collect the drones. Given this set of stations, efficient drone routes are then determined. The process is iterated to eliminate stations of high cost from the vehicle route while ensuring solution feasibility. The DDH, on the other hand, first optimizes the drone routes to determine the set of dispatching and collection stations. An efficient vehicle route is then constructed to visit these stations, taking into consideration that the dispatching station is visited before the collection station for each drone.

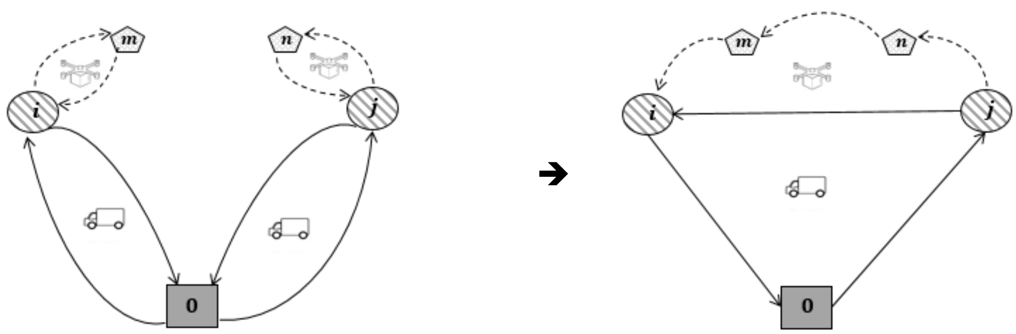
As such, both the VDH and DDH implement single-mode cost-reduction greedy strategies with respect to the vehicle and the drones, respectively, while the HCWH implements a multimodal cost-reduction strategy that simultaneously minimizes the cost of both modes. Even though the VDH and the DDH provide platforms to benchmark the performance of the HCWH, these two heuristics could be valuable for certain problems. For example, in problem instances where the vehicle's operation cost is much higher than that of the drone's cost, the solution generated by the VDH, which gives high priority to the optimization of the vehicle route, is expected to be close to the optimal solution. On the other hand, for problem instances in which the drone's operation cost is relatively high, the DDH is likely to generate near-optimal solutions, as it optimizes the routes of multiple drones over the route of one vehicle.

4-3. The Hybrid Clarke and Wright Heuristic (HCWH)

As mentioned above, the HCWH simultaneously optimizes the operation cost of both the vehicle and the drones to minimize the operation cost of the entire multimodal network. The savings associated with merging a pair of stations in the vehicle route is calculated such that it considers a) the saving in the vehicle's routing cost and b) the saving in the drones' routing cost. Thus, a term is added to the saving described in Equation (86). This additional term computes the maximum saving associated with serving two customers through constructing a feasible drone route that starts and ends at these two stations. Including these two terms in the cost saving determines the two stations with the highest cost reduction for the multimodal network. It reduces the routing cost of the vehicle and allows the construction of an efficient drone route that starts and ends at these two stations. The drone savings term is not considered, in case no feasible drone route can be constructed between the two stations due to limitation of the drone's flight range and/or its load carrying capacity. As illustrated in Figure 4-2, the integrated vehicle and drone savings can be calculated as follows:

$$\vartheta_{ij} = (l_{j0} + l_{0i} - l_{ij}) \cdot c_v + (l_{im} + l_{nj} - l_{mn}) \cdot c_d \tag{89}$$

where, nodes m and n are customer nodes. Stations i and j are the closest station to nodes m and n, respectively. Furthermore, c_v and c_d are the vehicle's and drones' operation cost per unit distance, respectively.



$$Cost = \begin{pmatrix} c_v \times (l_{i0} + l_{0i} + l_{0j} + l_{j0}) + \\ c_d \times (l_{jn} + l_{nj} + l_{mi} + l_{im}) \end{pmatrix} \qquad Cost\ 2 = \begin{pmatrix} c_v \times (l_{i0} + l_{ij} + l_{j0}) + \\ c_d \times (l_{nj} + l_{mn} + l_{mi}) \end{pmatrix}$$

$$\Rightarrow Saving = Cost\ 1 + Cost\ 2 = (l_{j0} + l_{0i} - l_{ij}) \cdot c_v + (l_{im} + l_{nj} - l_{mn}) \cdot c_d$$

Figure 4-2: Illustration of cost saving for the hybrid vehicle-drone routing problem.

H1: The Hybrid Clarke and Wright Heuristic

Input: Network topology and customer information
Result: S_V and S_D
repeat
  Set the closest station for each customer considering the stations in N_V
  ψ_D = Calculate pair saving for customers
  ψ_V = Calculate_Multimodel_Saving_For_Stations(ψ_D, w, r)
  S_V = Call the CW algorithm using $N_V \cup N_D$ and ψ_V as input
  S_D = Build_Drone_Routes (ψ_D, w, r)
  Check if reversed vehicle route reduces total cost
  stn= Determine_Station_with_Highest_Multimodel_Saving($N_V, N_D, S_V, S_D, \psi_V, w, r$)
  if ($stn \neq \emptyset$) **then**
    $N_V = N_V - stn$; $S_V = \emptyset$; $S_D = \emptyset$
  else
    Stop
  end if
until (Stop)
return S_V and S_D

Figure 4-3: Main steps of the hybrid Clarke and Wright heuristic.

The main steps of the HCWH are described in Figure 4-3. The heuristic (H1) starts by determining the closest station for every customer using Equation (87). Then, the elements of the drone cost savings list, ψ_D, are calculated using Equation (88) and the resulting list is sorted in a descending order. The multimodal cost savings list, ψ_V, is calculated as described in heuristic (H2) and also sorted in a descending order. Next, ψ_V is used as an input for the CW algorithm to construct an efficient vehicle route, S_V, considering all stations in the network and the depot, $N_V \cup N_D$. Assuming an identical set of drones, ψ_D is used as an input for the CW algorithm to construct the drone routes given their load carrying capacity, w, and maximum flight range, r. In this step, the drones could be dispatched or collected from any station in the network. Heuristics (H3) and (H4), presented in Figure 4-5 and Figure 4-7 respectively, are used to construct the drone routes for a given set of dispatching and collection stations. More details on this step are given hereafter.

The heuristic then checks if reversing the vehicle route improves the total cost. In this step, the vehicle route is reversed and the corresponding drone routes are constructed. If the total cost decreases, S_V and S_D are updated. This step is important, as the order by which the stations are visited by the vehicle could affect the feasibility of some drone routes with respect to their load carrying capacity limitations. Next, the heuristic iteratively searches for expensive stations for possible elimination from the vehicle route. The station with the highest multimodal cost saving is determined. This station is eliminated after ensuring that all customers can be served using a feasible set of drone routes that do not start from or end at the eliminated station. An efficient vehicle route is again constructed using the CW algorithm after excluding this station. The corresponding drone routes are constructed considering the reduced set of stations. The procedure is repeated until no further stations can be eliminated from the vehicle route. A station cannot be eliminated if its elimination prevents the drones from reaching any of the customers or it results in an increase in the total routing cost.

The calculation of the multimodal cost savings is presented in Figure 4-4. The heuristic starts by computing the vehicle's cost saving using Equation (86). The savings are ranked in the descending ordered list, ψ_V. Next, the heuristic constructs initial drone routes, where each route includes one customer such that the drone is dispatched and collected from its nearest station. The heuristic then loops over the elements of the drone saving list, ψ_D. For each saving element in this list, ϑ_{mn}, customers m and n are merged into one drone route considering the following two conditions: (a) drone's maximum flight range and its load carrying capacity are not violated, and (b) the absolute difference between the number of collected and dispatched drones at the stations does not exceed the maximum number of drones allowed on the vehicle, $MaxD$. For this drone route, if the origin station i is different from the destination station j, then their saving element, ϑ_{ij} is modified by adding to it the saving of the merged customers, ϑ_{mn}, as explained in Equation (89). The counter of the number of drone routes between station i and station j, $cntr_{ij}$, is incremented by one to ensure that condition (b) is satisfied.

The heuristic also checks the saving for element ϑ_{ji} by reversing the drone route from customer n to costumer m. If the reverse drone route (i.e. the origin station i becomes the destination and the destination station j becomes the origin) does not violate the two conditions described above, then the saving element, ϑ_{ji}, is modified by adding to it the saving of the merged customers, ϑ_{nm}. The counter of the number of drone routes between these stations, $cntr_{ji}$, is incremented. Finally, the elements in ψ_V are sorted in a descending order to be used as an input for the construction of the vehicle route.

H2: Calculate_Multimodal_Saving_For_Stations

Input: ψ_D, w, r

Result: ψ_V

ψ_V = Calculate pair saving for stations

S_D = Construct initial drone routes

$cntr_{ij} = 0 \ \forall\, i \in N_V, \forall\, j \in N_V$

while $\psi_D \neq \emptyset$ **do**

  Starting from the first element, ϑ_{mn}, of ψ_D

  Get *route1* that contains customer m, and *route2* that contains customer n from S_D

  if (*route1* ≠ *route2* & customers *m* and *n* are not intermediate nodes) **then**

    merged_drone_route = Merge customers *m* and *n* in a new route

    if ($\sum_k (pw_k + dw_k) \leq w$ & $\sum_{\acute{m}\acute{n}} l_{\acute{m}\acute{n}} \leq r$ & $cntr_{ij} \leq MaxD$) **then**

      Remove *route1* and *route2* from S_D and add *merged_drone_route* to S_D

      if ($i \neq j$) **then**

        $cntr_{ij} = cntr_{ij} + 1$; $\vartheta_{ij} = \vartheta_{ij} \cdot c_v + \vartheta_{mn} \cdot c_d$; overwrite ϑ_{ij} in ψ_V

        reversed_merged_drone_route = Reverse *merged_drone_route*

        if ($\sum_{\acute{k}} (pw_{\acute{k}} + dw_{\acute{k}}) \leq w$ & $\sum_{\acute{n}\acute{m}} l_{\acute{n}\acute{m}} \leq r$ & $cntr_{ji} \leq MaxD$) **then**

          $cntr_{ji} = cntr_{ji} + 1$; $\vartheta_{ji} = \vartheta_{ji} \cdot c_v + \vartheta_{nm} \cdot c_d$; overwrite ϑ_{ji} in ψ_V

        end if

      end if

    end if

  end if

  Eliminate ϑ_{mn} from ψ_D

End

Sort ψ_V in descending order

return ψ_V

Figure 4-4: Procedure for calculating multimodal cost savings for stations.

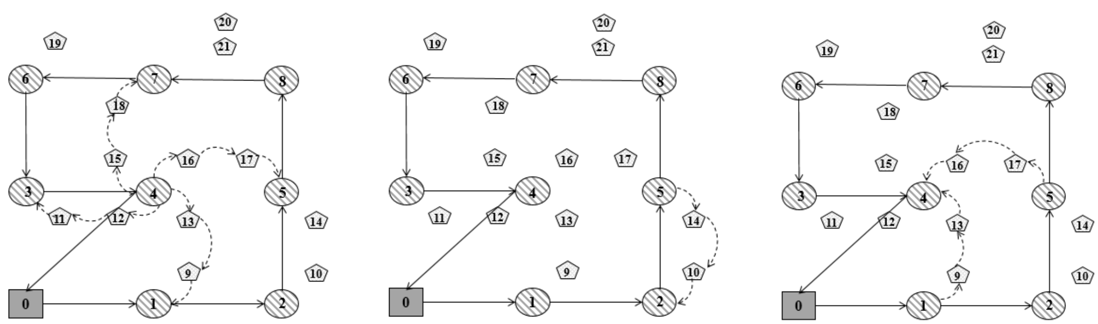
Building efficient drone routes is slightly more challenging than building the vehicle route. Similar to the vehicle routing step, the CW algorithm is used to build the drone routes. However, this step requires implementing two additional constraints to ensure the feasibility of merging two customers into one drone route. The first constraint ensures that the drone’s maximum flight range and load carrying capacity are not violated. The second constraint ensures the feasibility of the drone routes with respect to the vehicle route. This constraint involves three operation rules. First, the absolute difference between the number of collected and dispatched drones at any station does not exceed the maximum number of drones allowed on the vehicle. Second, the drone's dispatching station must be visited by the vehicle before visiting the collection station. Finally, at every station, the vehicle must have at least one drone to serve customers around that station.

H3: Build_Drone_Routes

Input: ψ_D, w, r
Result: S_D
S_D = Construct initial drone routes
while $\psi_D \neq \emptyset$ **do**
  Starting from the first element, ϑ_{ij}, of ψ_D
  Get *route1* that contains customer i, and *route2* that contains customer j from S_D
  if (*route1* ≠ *route2* & customers i and j are not intermediate nodes) **then**
    merged_route = Merge customers i and j in a new route
    if ($\sum_k (pw_k + dw_k) \leq w$ & $\sum_{mn} l_{mn} \leq r$ &
    merged_route is feasible from the vehicle's perspective) **then**
      if (the nearest station of any customer in *merged_route* is neither O nor D) **then**
        merged_route = Improve_Drone_Route(*merged_route*)
      end if
      Remove *route1* and *route2* from S_D
      Add *merged_route* to S_D
    end if
  end if
  Eliminate ϑ_{ij} from ψ_D
End
return S_D

Figure 4-5: Procedure for building the drone routes.

To illustrate these rules, consider the network example in Figure 4-6, which shows the vehicle route and the routes of two drones mounted on this vehicle. Figure 4-6(a) provides a case in which the first rule is violated. At station 4, the absolute difference between the number of collected and dispatched drones is four, which is greater than the number of drones carried by the vehicle. Figure 4-6(b) shows a case that violates the second rule. Because the vehicle visits station 2 before station 5, the drone route that starts at station 5 and ends at station 2 is infeasible. Figure 4-6(c) gives an example of the violation of the third rule, since the vehicle arrives at station 8 with no drones onboard, which makes it infeasible to serve customers at nodes 20 and 21.



(a) Rule 1 Violation (b) Rule 2 Violation (c) Rule 3 Violation

Figure 4-6: Example of a multimodal vehicle-drone network with operation violations.

After completing the step of merging two customers in a drone route, the heuristic checks if the drone route could be further improved by re-ordering the customers along the route. As shown in Figure 4-7, the heuristic (H4) checks if the nearest station to any of the customers in set N'_C served in that route, as determined in Equation (87), is neither the origin, O, nor the destination, D, of the route. If a customer has another station as its nearest station, the nearest station of that customer is over-written to be the closest of the origin or the destination of the route. The cost saving matrix, ψ_{New}, for the customers served in the route is recalculated after over-writing their nearest station. Then, the route is rebuilt based on the new saving matrix resulting in a more efficient sequence of customer nodes. The new route is checked against the drone's flight range and load carrying capacity limits. If the constraint is satisfied, the algorithm returns the modified route. Otherwise, it maintains the original route as the most efficient route.

H4: Improve_Drone_Route

Input: *route*
Result: *improved_route*
for all ($j \in N'_c$) **do**
  if (s_j is neither O nor D) **then**
    $s_j = \begin{cases} O \; if \; (l_{Oj} \leq l_{Dj}) \\ D \quad otherwise \end{cases}$
  end if
end for
Recalculate ψ_{New} based on the new closest station for customers in N'_C
improved_route = Rebuild the route based on the new saving matrix ψ_{New}
if ($\sum_k (pw_k + dw_k) \leq w$ & $\sum_{mn} l_{mn} \leq r$) **then**
  return *improved_route*
end if
return *route*

Figure 4-7: Procedure for improving the drone route.

As mentioned above, at each iteration, a search procedure is implemented to determine the station along the vehicle route with the highest cost saving. Following the steps of heuristic (H5) presented in Figure 4-8, the procedure applies a simple linear search to determine the station, stn, with the highest cost saving, ϑ_{stn}. The cost calculation in this step requires calculating the vehicle routing cost savings and the drone extra cost associated with removing this station. A feasibility check is implemented to ensure that customers close to a potentially eliminated station are reachable by a drone from any of the remaining stations, N'_V, in the vehicle route. For example, consider the network presented in Figure 4-1, assuming the drone's maximum flight range is 7.0 miles, removing station 6 is infeasible as customer 19 cannot be served from station 7. A round trip of 7.2 miles to reach customer 19 from station 7 violates the drone's maximum flight range. However, removing station 1 is feasible as customers 9 can still be served from station 4. The length of the round trip to serve customer 9 from station 4 is 6.3 miles.

H5: Determine_Station_with_Highest_Multimodel_Saving

```
Input: N_V, N_D, S_V, S_D, ψ_V, w, r
Result: stn
N′_V = N_V
for all (j ∈ N_V) do
   Remove Station j from N′_V
   if (customers close to station j are reachable by drone from any station in N′_V) then
      Set the closest station for each customer considering the stations in N′_V
      ψ′_D = Calculate pair saving for customers
      S′_V = Call the CW algorithm using N′_V ∪ N_D and ψ_V as input
      S′_D = Build_Drone_Routes (ψ′_D, w, r)
      ϑ_j = cost (S_V) – cost (S′_V) + cost (S_D) – cost (S′_D)
      if (ϑ_j > 0) then
         Add the saving element ϑ_j to ψ
      end if
      S′_V = Ø; S′_D = Ø; ψ′_D = Ø
   end if
   N′_V = N_V
end for
Sort the element in ψ in a descending order
stn = station corresponding to the first element, ϑ_stn, in the sorted list ψ
return stn
```

Figure 4-8: Procedure for determining the station with the highest cost along the vehicle route.

4-4. The Vehicle-Driven Heuristic (VDH)

The steps of the VDH are similar to the ones described in Figure 4-3, with the exception of the method used to calculate the cost savings for routing the vehicle. Here, the cost savings for the vehicle do not consider the drone cost savings. Thus, the VDH gives priority to reducing the routing cost of the vehicle over that of the drones. Similar to the HCWH, the heuristic starts by generating an initial vehicle route in which the vehicle visits all stations in the network. In this step, the CW algorithm is used to generate an efficient route for the vehicle using the vehicle cost saving matrix calculated using Equation (86). Drone routes are then constructed assuming that the drones can be dispatched and collected from any station in the network. Here, the drone savings are calculated using Equation (88). One should note that this heuristic does not include the step of checking if reversing the vehicle route will reduce the total cost as it focuses on reducing the vehicle routing cost.

Next, as the heuristic adopts a greedy strategy with respect to the vehicle routing cost, it iteratively searches for stations which are expensive for the vehicle to visit while ignoring any extra drone routing cost associated with removing this station. The station with the highest vehicle cost saving is eliminated after ensuring that all customers can be visited from the remaining stations. The vehicle route and the drone routes are reconstructed considering the reduced set of stations. The process is iterated until no other stations can be eliminated from the vehicle route.

4-5. The Drone-Driven Heuristic (DDH)

Unlike the VDH, the DDH gives priority to reducing the routing cost of the drones over the vehicle. Hence, it constructs the drone routes before the vehicle route. The main steps of the DDH are presented in Figure 4-9. The heuristic (H6) incrementally constructs drone routes while ensuring that the resulting set of drone dispatching and collection stations can be served by one vehicle. The CW algorithm using the drone cost saving matrix calculated with Equation (88) is used to construct the drone routes. The newly generated drone route is added to the existing set of drone routes resulting in an updated set of drone dispatching and collection stations. The CW algorithm is then activated to generate an efficient vehicle route. If the vehicle routing problem becomes infeasible, this new drone route is ignored and the next most efficient drone route is generated instead. The process continues until all customers are served and a feasible vehicle route is constructed to dispatch and collect all drones.

Similar to the HCWH, building a feasible drone route requires satisfying the three operation rules described in Figure 4-6. However, in the DDH, only the first rule is considered as part of the drone route building procedure. The other two rules are moved to the vehicle route building procedure. In other words, the first rule, which ensures that the absolute difference between the number of drones dispatched and number of drones collected at any station is less than the maximum number of drones mounted on the vehicle, is mandated while constructing the drone routes. The second and the third rules are enforced while building the vehicle route, which require visiting the dispatching stations before the collection stations and ensuring that at any station there is at least one drone to serve nearby customers.

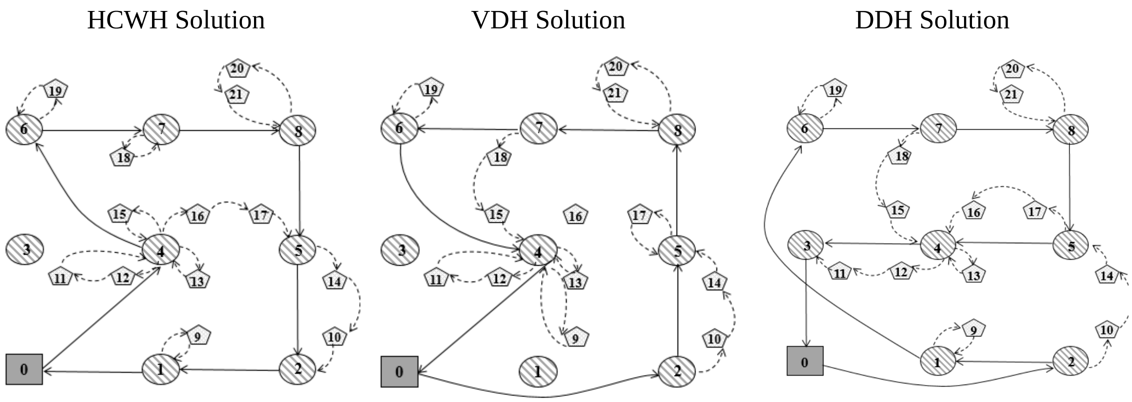
H6: The Drone-Driven Heuristic

Input: Network topology and customers information
Result: S_V and S_D
Set the closest station for each customer considering the stations in N_V
ψ_D = Calculate pair saving for customers
ψ_V = Calculate pair saving for stations
S_D = Build_Drone_Routes (ψ_D, w, r)
S_V = Construct initial vehicle routes
while $\psi_V \neq \emptyset$ **do**
  Starting from the first element, ϑ_{ij}, of ψ_V
  Get *route1* that contains station i, and *route2* that contains station j from S_V
  if (*route1* ≠ *route2* & stations i and j are not intermediate nodes) **then**
    merged_vehicle_route = Merge station i and station j in a new route
    if (*merged_vehicle_route* is feasible from the drones' perspective) **then**
      Remove *route1* and *route2* from S_V
      Add *merged_vehicle_route* to S_V
    end if
  end if
  Eliminate ϑ_{ij} from ψ_V
End
return S_V and S_D

Figure 4- 9: Main steps of the drone-driven heuristic.

4-6. Example to Demonstrate the Performance of the HCWH, VDH and DDH

The problem given in Figure 4-1 is solved using these three heuristics. The obtained solutions are presented in Figure 4-10, which illustrates the difference in the solutions as they adopt different strategies for routing cost optimization. The HCWH provides the solution with the least total routing cost. A total cost of $138.81 is recorded for the HCWH solution compared to $140.66 and $157.30 for the VDH solution and the DDH solution, respectively. As the VDH adopts a vehicle-based cost reduction strategy, it gives a solution in which station 1 is eliminated from the vehicle route. The HCWH solution kept this station as part of the vehicle route, as its elimination causes an increase in the multimodal cost. Finally, while the DDH significantly reduces the drone routing cost, the corresponding vehicle routing cost is the highest among all three heuristics, causing the total cost to be the highest.



Vehicle Cost	=	\$88.28	Vehicle Cost	=	\$88.28	Vehicle Cost	=	\$112.36
Drone Cost	=	\$50.53	Drone Cost	=	\$52.38	Drone Cost	=	\$44.94
Total Cost	=	\$138.81	Total Cost	=	\$140.66	Total Cost	=	\$157.30

Figure 4-10: Example to demonstrate the performance of the HCWH, VDH and DDH.

4-7. Summary

This chapter covers the heuristics developed to solve the basic mothership system. They can be used to obtain a good solution for large problem instances such as those anticipated in real-world applications. A novel solution methodology that extends the classic Clarke and Wright algorithm is introduced, named the hybrid Clarke and Wright heuristic (HCWH) (Clarke and Wright, 1964). The HCWH considers the cost savings for both the vehicle and the drones while solving for the optimal vehicle route, thus generating an efficient multimodal vehicle-drone network. The performance of the HCWH is benchmarked against two other heuristics that are developed as part of this research, which are the vehicle-driven heuristic (VDH) and the drone-driven heuristic (DDH). In the VDH, the optimal vehicle route is obtained first and then the drones are routed, assuming a fixed vehicle route. A reverse approach is considered for the DDH: given the optimal drone routes, the vehicle is routed to enable the dispatching and collection of the drones.

Chapter 5

SOLUTION METHEDOLEGY FOR THE MOTHERSHIP SYSTEM CONSIDERING THE LINE OF SIGHT RULE

5-1. Introduction

This chapter presents a novel solution methodology that extends the hybrid Clarke and Wright heuristic (HCWH) presented in Chapter 4 namely the Multimodal-Based Heuristic (MBH) to satisfy the LS constraints. The MBH iterates between two main procedures for constructing the drone routes and the vehicle route, respectively, while sharing information on routing cost and routing feasibility of both modes. The performance of the MBH is benchmarked against another heuristic that is developed as a part of this research, named the Single-Mode-Based Heuristic (SBH). The SBH is a light version of the MBH as it adopts a vehicle-driven approach which aims to prioritize the vehicle cost savings over the drone cost savings. This Chapter is organized as follows. Section 5-2 presents an overview of the heuristics developed in this chapter. Sections 5-3 and 5-4 describe the MBH and SBH, respectively. Finally, Section 5-5 summarizes this chapter.

5-2. Overview

This section presents an overview of the heuristic-based solution methodology developed to determine a near optimal solution for the IVDRP-LS described in Chapter 3. Figure 5-1 illustrates an example of the routes of the vehicle and two drones constructed to serve two customers with and without considering the LS constraints. As shown in the figure, the vehicle is used to serve one of the customers (C_2) as this customer cannot be visited by any of the drones. A drone cannot be dispatched from neither station s_1 or s_2 to customer C_2 due to flying range limitation and LS obstruction, respectively. In this case, the CW distance (cost) saving formula for the vehicle is calculated as given in Equation (86), where l_{i0} and l_{0j} are the distances from nodes i and j to the depot node 0, respectively. Nodes i and j represent the dispatching and collection stations as well as any customer node that cannot be reached by a drone from any station.

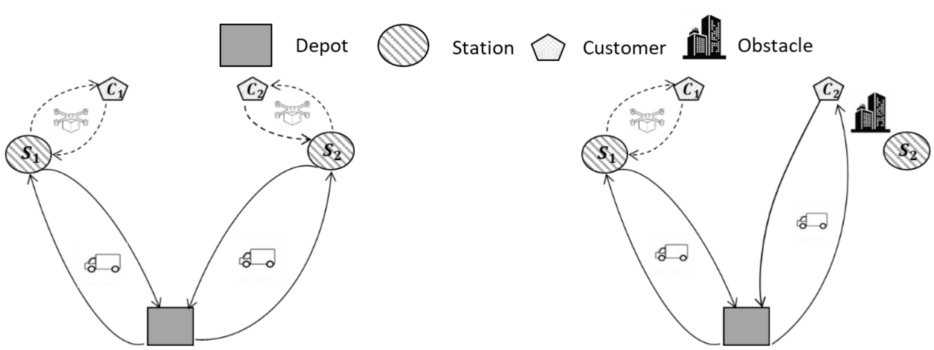


Figure 5-1: Illustration of the solution with and without LS consideration.

The calculation of drone savings is more complicated than the vehicle savings. Figure 5-2 illustrates three different cases that should be considered while calculating the drone savings. In Case A, the customer is within the LS of its closest station. In Cases B and C, an obstacle exists between the customer and its closest station. In Case B, the customer can be served from another station (s_2). Case C assumes that the drone's battery is not sufficient to return to the dispatching station s_2. Instead, it returns to station s_1 under the assumption that the collection station s_1 is within the LS of the dispatching station s_2. In order to calculate the drone savings, we first determine the closest station s_i with clear LS for each customer i such that:

$$l_{is_i} = \min\left(l_{is_k} \forall s_k \in N_V \mid a_{is_k} = 1\right) \tag{90}$$

Then, the saving expression for a drone, if both customers i and j are served with a tour that starts and ends from the same station can be calculated as follows:

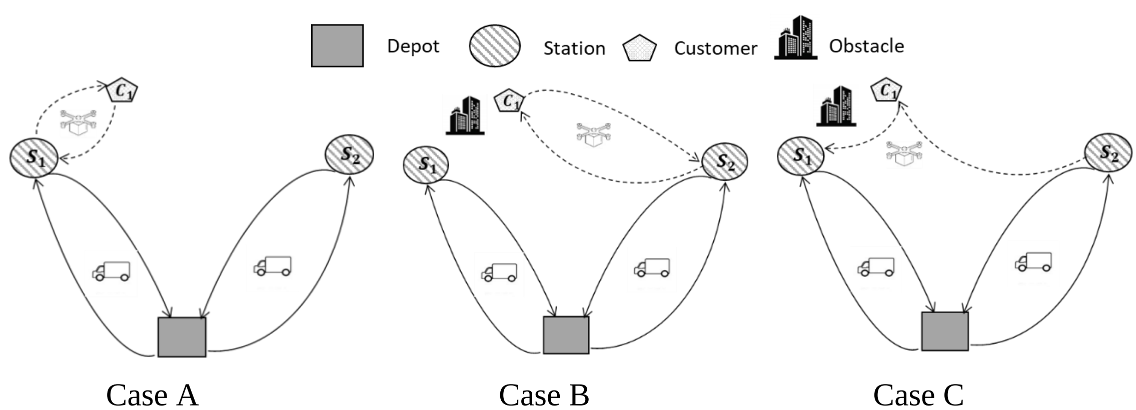


Figure 5-2: Illustration of the drone's initial solution.

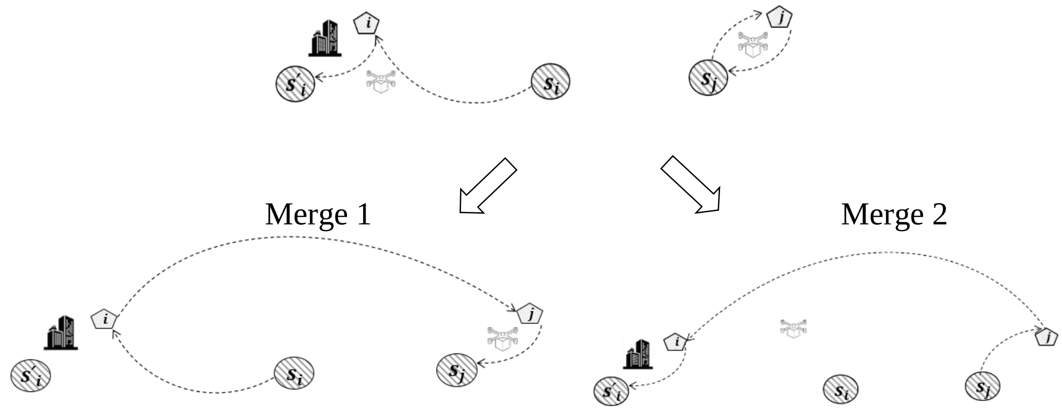


Figure 5-3: Illustration of merging two drone routes serving two customers from different stations.

Consider the case in which customer i is served by a drone that is dispatched and collected at two different stations (i.e., dispatching station s_i and collection station s'_i as shown in Figure 5-3), and another customer j is served by another drone that is dispatched and collected at the same station s_j. Constructing a new tour that merges customers i and j could occur through either s_i or s'_i. As illustrated in Figure 5-3, Merge 1 constructs a tour that starts at dispatching station s_i, while Merge 2 constructs a tour that ends at collection station s'_i. As the structure of the merge is not known a priori, we use the average distance of l_{is_i} and $l_{is'_i}$ as an approximation of the saving value associated with merging customers i and j in one tour. Hence, an approximated value for the drone saving can be calculated as follows.

$$\vartheta_{ij} = \frac{l_{is_i} + l_{is'_i}}{2} + l_{s_j j} - l_{ij} \tag{91}$$

If both customers i and j are served by tours that start and end at different stations, the average saving associated with merging these two customers can be calculated as follows.

$$\vartheta_{ij} = \frac{l_{is_i} + l_{is'_i}}{2} + \frac{l_{js_j} + l_{js'_j}}{2} - l_{ij} \tag{92}$$

Two heuristics are developed to solve the IVDRP-LS. The heuristics use the saving formulas described above to calculate the vehicle and drone savings. To simplify the presentation of these heuristics, we assume a fleet of identical drones in terms of flight range, r, and load-carrying capacity, w. The first heuristic, named the Multimodal-Based Heuristic (MBH), implements a multimodal cost-reduction greedy strategy that combines vehicle and drone cost savings. It iteratively eliminates stations of high cost to visit while ensuring solution feasibility. The heuristic implements a solution improvement procedure by repetitively randomizing the savings lists of both the vehicle and the drones until no better solution can be obtained after a pre-defined number of iterations. The second heuristic, named the Single-Mode-Based Heuristic (SBH), is a light version of the MBH as it adopts a vehicle-driven approach which aims to prioritize the vehicle cost savings over the drone cost savings. The following sections describe these two heuristics in more details.

5-3. Multimodal-Based Heuristic (MBH)

Figure 5-4 describes the main steps of the MBH. The MBH is a construction heuristic that iterates between two main procedures for constructing the drone routes and the vehicle route, respectively, while sharing information on routing cost and routing feasibility of both modes. The heuristic starts by determining the set of customers, $\grave{N}_V$, that cannot be served by a drone due to LS obstruction and/or due to limitation of the drones' flight ranges. The heuristic also determines the set of customers, $\grave{N}_C$, that can only be served by a drone that is dispatched and collected at two different stations due to the drones' flight range limitation, as illustrated in Figure 5-2 (Case C). Next, the heuristic iteratively executes a block of seven steps to generate an efficient vehicle route, $\mathcal{R}_V$ and its associated drone routes, $\mathcal{R}_D$. First, the closest station for each customer is determined such that the LS is not obstructed, as explained in Equation (90). Second, the cost savings list for routing the vehicle, ψ_V, is calculated as given in Equation (86). Third, the heuristic checks if any of the customers need to be served by a drone tour that stars and ends at two different stations due to LS obstruction and/or limitation of drones' flight ranges.

If $\grave{N}_C$ is not empty, the initial set of feasible drone routes, $\mathcal{R}_D^{initial}$, are constructed for all customers in $\grave{N}_C$. Each initial drone route consists of three nodes representing a dispatching station, the customer and a collection station that is different from the dispatching station. More details of this step are given in Heuristic (H1). Given $\mathcal{R}_D^{initial}$, the heuristic constructs a vehicle route, $\mathcal{R}_V$, that enables at least one drone route (i.e., the vehicle stops to dispatch and collect the drone serving that route) for every customer in $\grave{N}_C$. Heuristics (H2) and (H3) provide the details of constructing the vehicle route in this step.

If $\mathcal{R}_D^{initial}$ contains multiple drone routes serving any customer in $\grave{N}_C$, the least expensive route is determined and $\mathcal{R}_D^{initial}$ is updated to eliminate all expensive ones. If $\grave{N}_C$ is empty, the conventional CW algorithm is used to construct an efficient vehicle route, $\mathcal{R}_V$.

Heuristic H: The MBH

Input: Network topology and customer information
Result: $\mathcal{R}_V$ and $\mathcal{R}_D$
$\grave{N}_V \leftarrow$ Determine the set of customers that cannot be served by drones due to LS obstruction and/or due to the drones' flight range limitation
$\grave{N}_C \leftarrow$ Determine the set of customers that have to be served by drones from two different stations due to the drones' flight range limitation
repeat
  Determine the closest visible station for each customer considering the stations in N_V
  $\psi_V \leftarrow$ Calculate pair vehicle savings for $N_V \cup \grave{N}_V$
  if ($\grave{N}_C \neq \emptyset$) **then**
    $\mathcal{R}_D^{initial} \leftarrow$ Determine_Initial_Feasible_Drone_Routes($N_V, \grave{N}_C, r$)
    $\mathcal{R}_V \leftarrow$ Construct_Feasible_Vehicle_Route($\psi_V, N_V \cup \grave{N}_V \cup N_D, \grave{N}_C, \mathcal{R}_D^{initial}$)
    $\mathcal{R}_D^{initial} \leftarrow$ Update $\mathcal{R}_D^{initial}$ by eliminating all expensive initial drone routes for each customer in $\grave{N}_C$
  else
    $\mathcal{R}_V \leftarrow$ Call the CW algorithm using $N_V \cup \grave{N}_V \cup N_D$ and ψ_V as input
  end if
  $\psi_D \leftarrow$ Calculate pair savings for customers using the initial drone route, $\mathcal{R}_D^{initial}$, as an input
  $\mathcal{R}_V \leftarrow$
  Improve_Route_Through_Considering_The_Multimodal_Savings_For_Stations($\mathcal{R}_V, \psi_V, \psi_D, w, r$)
  $\mathcal{R}_D \leftarrow$ Call the CW algorithm using N_C, ψ_D, w, and r as input to build drone routes that satisfy $\mathcal{R}_V$
  $stn \leftarrow$ Determine station with the highest multimodal savings
  if ($stn \neq \emptyset$) **then**
  $N_V \leftarrow N_V - stn$; $\mathcal{R}_V \leftarrow \emptyset$; $\mathcal{R}_D \leftarrow \emptyset$
  else
  Stop
  end if
until (Stop)
Improve_Solution_Through_Vehicle_and_Drones_Savings_Lists_Randomization(
$\mathcal{R}_V, \mathcal{R}_D, \psi_V, \psi_D, w, r, \grave{N}_C$)
return $\mathcal{R}_V$ and $\mathcal{R}_D$

Figure 5-4: Main steps of the MBH.

In the fourth step of this iterative block, the drone cost savings list ordered in a descending order, ψ_D, is calculated using Equations (88), (91), and (92). Fifth, the heuristic improves the vehicle route through considering the multimodal savings of the stations. The details of this step are given in Heuristic (H4). Sixth, using the drone savings list ψ_D, the CW algorithm is again activated to construct efficient drone routes, $\mathcal{R}_D$, considering the vehicle route obtained in the previous step. These drone routes are constructed while satisfying the drones' maximum flight range and load-carrying capacity, respectively. The last step in this block determines the station with the highest multimodal cost savings and checks if this station can be eliminated from the vehicle route. A station is eliminated from N_V only if its elimination does not cause a customer to be unreachable by a drone nor cause an increase in the total routing cost. The closest station that does not violate the LS constraint is again determined for each customer considering the reduced set of stations. The vehicle route and corresponding drone routes are reconstructed. The procedure is repeated until no further stations can be eliminated from the vehicle route. Finally, a post-processing step is implemented to check if any reduction in the total cost can be obtained by randomizing the savings list for both the vehicle and the drones. This step is described in more details in Heuristic (H5).

Heuristic (H1) is used to build the feasible set of initial drone routes for every customer $i \in \grave{N}_C$. All stations in N_V are scanned to check their eligibility as dispatching stations of a drone serving customer i. A station j is marked as a feasible dispatching station if the LS to customer i is unobstructed (i.e., a_{ji}=1) and the distance from station j to customer i is less than the drone's maximum flight range (i.e., $l_{ji} < r$). All stations in N_V

are scanned again to determine eligible collecting stations. If station k is within the dispatching station's LS (i.e., a_{jk}=1) and the length of the route $j - i - k$ is less than the drone's maximum flight range (i.e. $(l_{ji} + l_{ik}) \leq r$), this route is added to the set of all feasible drone routes, $\mho_i$, for customer i. The set $\mathcal{R}_D^{initial}$ includes the sets of initial feasible drone routes for all customers in $\grave{N}_C$.

H1: Determine_Initial_Feasible_Drone_Routes

Input: $N_V, \grave{N}_C, r$
Result: $\mathcal{R}_D^{initial}$
for all ($i \in \grave{N}_C$) **do**
    for all ($j \in N_V$) **do**
        if (a_{ji}=1 & $l_{ji} \leq r$) **then**
            for all ($k \in N_V$) **do**
                if (a_{jk}=1 & $(l_{ji} + l_{ik}) \leq r$) **then**
                    route ← $\{j - i - k\}$
                    Add *route* to $\mho_i$
                end if
            end for
        end if
    end for
    Add $\mho_i$ to $\mathcal{R}_D^{initial}$
end for
return $\mathcal{R}_D^{initial}$

Figure 5-5: Construction of initial drone routes that start and end at different stations.

Obtaining the initial feasible drone routes, $\mathcal{R}_D^{initial}$, Heuristic (H2) is activated to construct a vehicle route that enables at least one feasible drone route for every customer $i \in \grave{N}_C$. H2 starts by constructing a set of initial vehicle routes, $\mathcal{R}_V$, where each route, starting and ending at the depot, includes one node. Next, the heuristic loops over the elements of the vehicle savings list, ψ_V. For each saving element ϑ_{kj}, nodes k and j are merged into one vehicle route, *route*, following CW procedure after checking the feasibility of *route* with respect to the constructed drone routes for customers in $\grave{N}_C$.

H2: Construct_Feasible_Vehicle_Route

Input: $\psi_V, N_V \cup \grave{N}_V \cup N_D, \grave{N}_C, \mathcal{R}_D^{initial}$
Result: $\mathcal{R}_V$
$\mathcal{R}_V$ ← Construct initial vehicle routes
while $\psi_V \neq \emptyset$ **do**
  Starting from the first element ϑ_{kj} in ψ_V
  Get *route1* that contains node k, and *route2* that contains node j from $\mathcal{R}_V$
  if (*route1* ≠ *route2* & nodes k and j are not intermediate nodes) **then**
    route ← Merge nodes k and j in a new route
    if (Checking_Vehicle_Route_Feasibility_from_Drone_Perspective(*route*, $\mathcal{R}_D^{initial}$)=*true*) **then**
      Remove *route1* and *route2* from $\mathcal{R}_V$ and add *route* to $\mathcal{R}_V$
      update $\mathcal{R}_D^{initial}$ to remove infeasible drone routes not enabled by $\mathcal{R}_V$
    end if
  end if
  Eliminate ϑ_{kj} from ψ_V
End
return $\mathcal{R}_V$

Figure 5-6: Construction of a feasible vehicle route.

H3: Checking_Route_Feasibility_from_Drone_Perspective

Input: *route*, $\mathcal{R}_D^{initial}$
for all ($\mho_i \in \mathcal{R}_D^{initial}$) **do**
  drone_route ← get *drone_route* from $\mho_i$
  if (*route* violates *drone_route*) **then**
    remove *drone_route* from $\mho_i$
  end if
  if ($|\mho_i| = 0$) **then**
    return *false*
  end if
end for
return $true$

Figure 5-7: Checking vehicle route feasibility with respect to drone routes.

The steps used to perform this feasibility check are given in Heuristic (H3). For each possible merge in the vehicle route, H3 scans the initial drone routes $\mho_i$ for each customer $i \in \grave{N}_C$. If for any customer $i \in \grave{N}_C$, *route* does not enable at least one drone route in $\mho_i$, *route* is marked as infeasible, not allowing the merge of *route1* and *route2*. The set of drone routes, $\mathcal{R}_D^{initial}$, is updated to eliminate any drone routes not enabled by *route*.

As mentioned earlier, the MBH includes a procedure to improve the vehicle route through considering the multimodal cost savings. The procedure uses information on the drone routes to improve the vehicle route. The details of this procedure are presented in Heuristic (H4). H4 starts by calculating the multimodal savings for the stations. It loops over the elements of the drone savings list, ψ_D. For each saving element ϑ_{mn} in ψ_D, customers m and n are merged into one drone route, *drone_route*, considering the following two conditions: (a) the drone's maximum flight range and its load-carrying capacity are not violated, and (b) the absolute difference between the number of collected and dispatched drones at each station does not exceed the maximum number of drones, θ, allowed on the vehicle. If the dispatching station i of *drone_route* is different from its collection station j, their saving element, ϑ_{ij}, is updated to $\vartheta_{ij} \cdot c_v + \vartheta_{mn} \cdot c_d$. In addition, the counter of the number of drone routes, $cntr_{ij}$, between station pair ij is incremented. The counter $cntr_{ij}$ is tracked to ensure that condition (b) above is not violated.

The reverse drone route from customer n to costumer m is also considered. If the reversed drone route (i.e. the dispatching station i becomes the collection station, and the collection station j becomes the dispatching station) does not violate the two conditions mentioned above, the corresponding saving element and the number of drone routes are updated as follows: $\vartheta_{ji} = \vartheta_{ji} \cdot c_v + \vartheta_{nm} \cdot c_d$ and $cntr_{ji} = cntr_{ji} + 1$. Finally, the elements in ψ_V are sorted in a descending order and used as an input for constructing a new vehicle route, $\mathcal{R}'_V$. If the cost of $\mathcal{R}'_V$ is less than that of $\mathcal{R}_V$, $\mathcal{R}'_V$ replaces $\mathcal{R}_V$.

H4: Improve_Vehicle_Route_Through_Considering_The_Multimodal_Savings_For_Stations

Input: $\mathcal{R}_V,\ \psi_V, \psi_D, w, r$
Results: $\mathcal{R}_V$
$cntr_{ij} \leftarrow 0\ \ \forall\ ij \in N_V$
$\mathcal{R}'_D$ ← construct initial drone routes
while $\psi_D \neq \emptyset$ **do**
  Starting from the first element ϑ_{mn} in ψ_D
  Get *route1* that contains customer m, and *route2* that contains customer n from $\mathcal{R}'_D$
  if (*route1* ≠ *route2* & customers *m* and *n* are not intermediate nodes) **then**
    drone_route ← Merge customers *m* and *n* in new route with origin i and destination *j*
    if ($\sum_k (pw_k + dw_k) \leq w$ & $\sum_{\acute{m}\acute{n}} l_{\acute{m}\acute{n}} \leq r$ & $cntr_{ij} \leq \theta$ & $\sum_{\acute{m}} a_{i\acute{m}}$=1) **then**
      Remove *route1* and *route2* from $\mathcal{R}'_D$ and add *drone_route* to $\mathcal{R}'_D$
      if ($i \neq j$) **then**
        $cntr_{ij} \leftarrow\ cntr_{ij} + 1$; $\vartheta_{ij} \leftarrow \vartheta_{ij} \cdot c_v + \vartheta_{mn} \cdot c_d$; update ψ_V
        reversed_drone_route = Reverse *drone_route*
        if ($\sum_{\acute{k}} (pw_{\acute{k}} + dw_{\acute{k}}) \leq w$ & $\sum_{\acute{n}\acute{m}} l_{\acute{n}\acute{m}} \leq r$ & $cntr_{ji} \leq \theta$ & $\sum_{\acute{m}} a_{j\acute{m}}$=1) **then**
          $cntr_{ji} \leftarrow\ cntr_{ji} + 1$; $\vartheta_{ji} \leftarrow \vartheta_{ji} \cdot c_v + \vartheta_{nm} \cdot c_d$; update ψ_V
        end if
      end if
    end if
  end if
  Eliminate ϑ_{mn} and ϑ_{nm} from ψ_D
End
Sort ψ_V in descending order
$\mathcal{R}'_V$ ← Rebuild the vehicle route based on the new vehicle savings list ψ_V
if (cost of ($\mathcal{R}'_V$) < cost of ($\mathcal{R}_V$) **then**
  $\mathcal{R}_V \leftarrow\ \mathcal{R}'_V$
end if
return $\mathcal{R}_V$

Figure 5-8: Improve vehicle route through considering the multimodal savings for stations.

The last step of the MBH is a post-processing step that checks if any solution improvement in terms of the multimodal routing cost can be obtained by randomizing the savings lists ψ_V and ψ_D, respectively. Figure 5-9 describes the steps of Heuristic (H5) used to post-process the solution. The iterative heuristic starts by randomizing the vehicle savings list, ψ_V, associated with the latest solution. The first n elements in ψ_V are selected and randomly rearranged. The process is repeated for the next n elements until the entire list is randomized.

The randomized savings list ψ'_V is used to construct a new vehicle route $\mathcal{R}'_V$. The drones savings list, ψ_D, is also randomized and used to construct the corresponding drone routes $\mathcal{R}'_D$. If no better drone routes are found for a pre-specified number of iterations, $MaxD$, the drone routes $\mathcal{R}'_D$ are marked as the best drone routes that satisfy the constructed vehicle route $\mathcal{R}'_V$. The multimodal cost of the new solution ($\mathcal{R}'_V$ and $\mathcal{R}'_D$) is compared against that of the current solution ($\mathcal{R}_V$ and $\mathcal{R}_D$). If the new solution is found to reduce the multimodal routing cost, the solution is updated, that is $\mathcal{R}_V = \mathcal{R}'_V$, $\psi_v = \psi'_V$, $\mathcal{R}_D = \mathcal{R}'_D$, and $\psi_D = \psi'_D$. The latest vehicle savings list is again randomized searching for a better vehicle route. If no better solution considering the total multimodal routing cost is found for a pre-specified number of iterations, $MaxV$, the heuristic terminates after reporting the best solution recorded in all iterations.

H5: Improve_Solution_Through_Vehicle_and_Drones_Savings_Lists_Randomization

Input: $\mathcal{R}_V, \mathcal{R}_D, \psi_V, \psi_D, w, r, \grave{N}_C$
$itr_V \leftarrow 0$
while $(itr_V < MaxV)$ **do**
  $\psi'_V \leftarrow$ Randomize ψ_V
  if $(\grave{N}_C \neq \emptyset)$ **then**
    $\mathcal{R}_D^{initial} \leftarrow$ Reset all possible initial drone routes
    $\mathcal{R}'_V \leftarrow$ Construct_Feasible_Vehicle_Route($\psi'_V, N_V \cup \grave{N}_V \cup N_D, \grave{N}_C, S_D^{initial}$)
    $\mathcal{R}_D^{initial} \leftarrow$ Update $\mathcal{R}_D^{initial}$ by eliminating all expensive initial drone routes for each customer in $\grave{N}_C$
  else
    $\mathcal{R}'_V \leftarrow$ Call the CW algorithm using $N_V \cup \grave{N}_V \cup N_D$ and ψ'_V as input
  end if
  $itr_D \leftarrow 0$
  $\psi'_D \leftarrow$ Randomize ψ_D
  $\mathcal{R}'_D \leftarrow$ Call the CW algorithm using N_C, ψ'_D, w and r as input to build drone routes
  while $(itr_D < MaxD)$ **do**
    $\psi''_D \leftarrow$Randomize ψ'_D
    $\mathcal{R}''_D \leftarrow$ Call the CW algorithm using N_C, ψ''_D, w and r as input to build drone routes
    if (cost $(\mathcal{R}''_D) <$ cost ($\mathcal{R}'_D$)) **then**
      $\psi'_D \leftarrow \psi''_D$; $\mathcal{R}'_D \leftarrow \mathcal{R}''_D$; and $itr_D \leftarrow 0$
    else
      $itr_D \leftarrow itr_D + 1$
    end if
  End
  if (cost $(\mathcal{R}'_V + \mathcal{R}'_D) <$ cost $(\mathcal{R}_V + \mathcal{R}_D)$) **then**
    $\psi_V \leftarrow \psi'_V$; $\psi_D \leftarrow \psi'_D$; $\mathcal{R}_V \leftarrow \mathcal{R}'_V$; $\mathcal{R}_D \leftarrow \mathcal{R}'_D$; and $itr_V \leftarrow 0$
  else
    $itr_V \leftarrow itr_V + 1$
  end if
End

Figure 5-9: Improving the solution through randomizing the vehicle and drones savings lists.

5-4. Single-Mode-Based Heuristic (SBH)

As mentioned above, the Single-Mode-Based Heuristic (SBH) is a lighter version of the MBH as it adopts a vehicle-driven approach rather than a multimodal-driven approach. It follows the main steps of the MBH with two differences. First, in the SBH, the calculation of the cost savings used to construct the vehicle route are based entirely on the vehicle savings. Second, in the post-processing procedure, the SBH only randomizes ψ_V and fixes ψ_D. Similar to the MBH presented above, the SBH starts by determining customers that cannot be served by the vehicle. In addition, the list of customers, $\grave{N}_C$ are determined. Next, the heuristic calculates the savings list for the vehicle using Equation (86). This savings list is then used to construct the vehicle route that does not violate the initial drone routes for customer in $\grave{N}_C$, if any. Equations (88), (91), and (92) are used to build the savings list for the drones and the resulting list is sorted in a descending order. The drones' routes are then constructed assuming that the drones can be dispatched and collected from any visible station in the network in order to satisfy the LS rule. The heuristic removes the station with the highest cost savings for the vehicle after ensuring that all customers can be served from the remaining set of stations. The vehicle's and drones' routes are reconstructed considering the reduced set of stations. This step is repeated until there are no more stations can be eliminated from the vehicle route.

For the post-processing step, the SBH randomizes ψ_V only. A new vehicle route, $\mathcal{R}'_V$, is obtained based on the randomized cost savings list of the vehicle. The new vehicle route must not violate the initial drone routes of customers in $\grave{N}_C$. If a better solution is found, the new savings list, ψ'_V, is obtained. Otherwise, the current savings list is again randomized and used to determine a new vehicle route. If a pre-defined number of iterations, $MaxV$, is reached without any solution improvement, the heuristic terminates after producing the best vehicle route in all iterations. Finally, the heuristic constructs the drone routes based on the resulted $\mathcal{R}_V$. Although the SBH is greedy in terms of the vehicle routing cost, the cost of the vehicle is typically much higher than that of the drones. Therefore, it is expected that the SBH will obtain a good solution for the IVDRP-LS with faster execution time compared to the MBH as it skips two cumbersome steps, as mentioned above. A comparison of these two heuristics in terms of solution quality and running time is presented in the next section.

H6: Improve_Solution_Through_Vehicle_ Savings_Lists_Randomization

Input: $\mathcal{R}_V, \mathcal{R}_D, \psi_V, \psi_D, w, r, \grave{N}_C$
$itr_V \leftarrow 0$
while $itr_V < MaxV$ **do**
  $\psi'_V \leftarrow$ Randomize ψ_V
  if ($\grave{N}_C \neq \emptyset$) **then**
    $\mathcal{R}_D^{initial} \leftarrow$ Reset all possible initial drone routes
    $\mathcal{R}'_V \leftarrow$ Construct_Feasible_Vehicle_Route ($\psi'_V, N_V \cup \grave{N}_V \cup N_D, \grave{N}_C, \mathcal{R}_D^{initial}$)
    $\mathcal{R}_D^{initial} \leftarrow$ Update $\mathcal{R}_D^{initial}$ by eliminating all expensive initial drone routes for each customer in $\grave{N}_C$
  else
    $\mathcal{R}'_V \leftarrow$ Call the CW algorithm using $N_V \cup \grave{N}_V \cup N_D$ and ψ'_V as input
  end if
  if (cost ($\mathcal{R}'_V$) $<$ cost ($\mathcal{R}_V$)) **then**
    $\psi_V \leftarrow \psi'_V$; $\mathcal{R}_V \leftarrow \mathcal{R}'_V$; and $itr_V \leftarrow 0$
  else
    $itr_V \leftarrow itr_V + 1$
  end if
End
$\mathcal{R}_D \leftarrow$ Call the CW algorithm using $N_C, \psi_D, w,$ and r as input to build drone routes satisfying $\mathcal{R}_V$

Figure 5-10: Improve solution through vehicle savings lists randomization.

5-5. Summary

This chapter introduces a novel solution methodology that adopts an updated version of the classic CW algorithm to consider the multimodality aspects of the integrated vehicle-drone routing problem and to satisfy the LS rule (Clarke and Wright, 1964). The solution methodology implements a Multimodal-Based Heuristic (MBH) with a randomization procedure to construct near optimal vehicle and LS-mandated drone routes. The performance of the MBH is benchmarked by comparing its performance against that of a Single-Mode-Based Heuristic (SBH). The SBH is a lighter version of the MBH as it adopts a vehicle-driven search procedure.

Chapter 6

RESULTS AND ANALYSES

6-1. Introduction

This chapter presents the results of a set of experiments that are conducted to examine the performance of the heuristics described in Chapter 4 and Chapter 5. To avoid bias related to the data generation, a common grid network is used with randomly generated demand in terms of location and pick-up/delivery loads. Networks with different numbers of stations, numbers of customers, and density levels are considered. To avoid solution infeasibility, it was assured that a) the distance between any two stations was less than the drone's maximum flight range, and b) the pick-up/delivery load of any customer was less than the drone's load carrying capacity.

6-2. Experiments Setup

Seven roadway networks of a grid structure covering service areas that range from 25.0 to 400.0 square miles are considered. Stations for drone dispatching and collection are assumed to be located at the intersection nodes in these networks. These intersection nodes are spaced at a 5.0 mile distance. Customers are randomly distributed in the area with densities that range from 0.125 customer per square mile to 1.0 customer per square mile. The number of customers ranges from six customers in the smallest network to 100 customers in the largest network. Each customer is associated with a pick-up weight and/or a delivery weight that are randomly generated following the uniform distribution $U(0.0 \text{ lbs}, 5.0 \text{ lbs})$.

Table 6-1: Summary of network configurations used to test the performance of the heuristics

Network	Number of Customer	Number of Stations	Area (mile^2)	Customer Density (Customer/ mile^2)
A1 to A5	6	3	25 (5x5)	0.240
A6 to A10	8	3	25 (5x5)	0.480
B1 to B10	50	8	100 (10x10)	0.500
C1 to C10	50	15	225 (15 x15)	0.222
D1 to D10	50	24	400 (20 x20)	0.125
E1 to E10	100	8	100 (10x10)	1.000
F1 to F10	100	15	225 (15 x15)	0.444
G1 to G10	100	24	400 (20 x20)	0.250

One vehicle equipped with two drones is used to serve these customers, unless specified otherwise. The vehicle operation cost is assumed to be twice that of the drones. The vehicle depot was assumed to be located at the southwest corner of the grid networks. Both drones are assumed to have a maximum flight range of 7.0 miles and a load carrying capacity of 10.0 lbs. Such values are in the range of the drone specifications used by UPS in their drone delivery field experiment (McFarland , 2017). Table 6-1 summarizes the configuration of these seven networks. For each network, 10 random instances are generated representing different spatial distributions of the customers. All runs were carried out on a Dell workstation with 72 logical processors of 3.1 GHz and 192 GB memory. All heuristics are implemented in Java and the exact solution is obtained by CPLEX 12.6.1 Java callable libraries (IBM, 2009), which is used to solve the MIP presented in Chapter 4.

6-3. Results for the Basic Mothership System

This section's experiments are designed to evaluate the solution methodologies developed in Chapter 4 and presents a sensitivity analysis to examine the effect of several system parameters on the overall performance of the network.

6-3-1. Comparison with the Exact Optimal Solution

The performance of the HCWH, VDH and DDH, which are implemented in Java, are compared against the exact solution obtained by CPLEX 12.6.1 Java callable libraries (IBM, 2009), which is used to solve the MIP presented in Chapter 4. The total routing cost and the execution time are reported for all tested cases. Considering the large execution time required to obtain the exact solution using CPLEX, these results are reported only for the small networks A-1 to A-10 as their solutions can be obtained within a reasonable timeframe (less than six hours). Table 6-2 gives a summary of the performance comparison results. As shown in the table, the three heuristics produce the exact optimal solution for most tested networks. While the HCWH generates the exact solutions for all networks, the VDH and DDH generate the exact solutions for nine and seven networks out of the ten tested networks, respectively. In addition, for cases in which the optimal solution is not obtained, the optimality gaps recorded for the VHD are generally lower than those of the DDH.

Table 6-2: The heuristics' performance comparison with the optimal solution.

Instance	Exact Solution (CPLEX)		HCWH			VDH			DDH		
	Cost ($)	Runtime (sec)	Cost ($)	Runtime (sec)	Gap (%)	Cost ($)	Runtime (sec)	Gap (%)	Cost ($)	Runtime (sec)	Gap (%)
A-1	46.8	11.710	46.8	0.019	0.0	46.8	0.016	0.0	46.8	0.016	0.0
A-2	50.3	99.674	50.3	0.021	0.0	50.3	0.015	0.0	50.3	0.016	0.0
A-3	50.6	12.095	50.6	0.035	0.0	50.6	0.026	0.0	55.7	0.016	9.0
A-4	46.2	59.108	46.2	0.038	0.0	46.2	0.026	0.0	60.1	0.037	23.0
A-5	50.0	62.296	50.0	0.034	0.0	50.0	0.024	0.0	50.8	0.037	1.0
A-6	60.5	17056.50	60.5	0.031	0.0	60.5	0.016	0.0	60.5	0.031	0.0
A-7	50.5	1817.953	50.5	0.032	0.0	50.5	0.020	0.0	50.5	0.031	0.0
A-8	55.2	4903.824	55.2	0.032	0.0	56.1	0.031	2.0	55.2	0.016	0.0
A-9	53.2	2722.409	53.2	0.062	0.0	53.2	0.031	0.0	53.2	0.031	0.0
A-10	56.5	2113.717	56.5	0.059	0.0	56.5	0.031	0.0	61.2	0.018	8.0

The three heuristics significantly outperform CPLEX in terms of the execution time. For example, for network A-1, the exact optimal solution using CPLEX is obtained in about 11.7 seconds. The execution times for the three heuristics are less than 0.02 seconds for that network. One can also observe the large increase in the execution time using CPLEX when the number of customers is increased from six customers (networks A-1 to A-5) to eight (networks A-6 to A-10). For example, the execution time jumped to 17,056.5 seconds for A-6 compared to 11.7 seconds for A-1. Such substantial increase in the execution time with the increase in the number of customers is not observed for any of the three heuristics. The highest execution time for A-6 to A-10 networks is less than 0.10 seconds.

The results in Table 6-2 show that the HCWH was able to find the optimal solution for all studied instances. Additionally, a randomly generated problem instance of 8 customers and 5 stations is considered. For this problem instance, we used the solution obtained from the HCWH as a warm start (initial solution) for CPLEX. While CPLEX's optimal solution was not obtained within an execution time of up to 24 hours, CPLEX was able to find four incumbent solutions that are better than the one obtained by the HCWH with an improvement in the objective function of 7.35%. The results of this test illustrates that there could be cases in which CPLEX produces solutions with better performance than those obtained by the HCWH.

6-3-2. Performance Comparison for Large Network Instances

The performance of the three heuristics is again compared considering large problem instances. Six different networks are used in this comparison which includes 50 customers (networks B, C and D) and 100 customers (networks E, F and G), respectively. As given in Table 6-3 and Table 6-4, each network is tested for 10 random instances. For each instance, the routing cost is reported for the vehicle (C_{V}), the drones (C_{D}), and the entire network (C_{Total}). The number of stops made by the vehicle to dispatch and collect the drones, η, and the total number of drone dispatches, ή, to serve the customers also are given. Finally, the execution time for each problem instance is recorded. The average of the 10 random instances is given for each network.

As shown in the Table 6-3 and Table 6-4, the solution obtained using the HCWH provides the best cost performance for the majority of the tested problem instances. The

VDH outperformed the HCWH in a few problem instances, as the CW algorithm does not always guarantee optimality, especially when the routed drones are constrained by a limited flight range and load carrying capacity. While the total cost obtained by the VDH and the DDH for almost all problem instances is higher than that obtained using the HCWH, the VDH and DDH provide the lowest vehicle and drone costs, respectively.

These results are expected, since the VDH adopts a vehicle cost-reduction strategy and the DDH adopts a drone cost-reduction strategy. For example, the average total cost recorded for the D network using the HCWH is $346.30 with a vehicle cost of $208.80 and a drone cost of $137.60. For the VDH, the total cost increased to $347.10, while the vehicle cost was reduced to $205.00. For the DDH, the total cost increased to $394.90, while the drone cost was reduced to $131.20.

The three heuristics generally show comparable results in terms of the number of stops made by the vehicle and the number of drone dispatches. However, a closer look at some problem instances reveals that the VDH tends to reduce the number of stops made by the vehicle, while the DDH tends to reduce the number of drone dispatches. However, the number of stops recorded by VDH is associated with an increase in the number of drone dispatches. Similarly, the number of drone dispatches recorded by the DDH is associated with an increase in the number of stops made by the vehicle. The result is consistent with the cost-reduction strategies adopted for the two heuristics. It also resembles the split of the vehicle cost and the drone cost recorded for the solutions obtained by the VDH and the DDH, respectively. As the VDH aims at reducing the vehicle cost, it eliminates expensive stops for the vehicle at the expense of scheduling more drone dispatches. Similarly, the

DDH cuts the drone cost by reducing the number of drone dispatches at the expense of more vehicle stops.

For example, for network G, the VDH solution results in 21 vehicle stops and 48 drone dispatches. For the DDH solution, the number of drone dispatches decreased to 45 while the number of vehicle stops increased to 23. The HCWH produced a balanced solution in terms of number of stops made by the vehicles and the number of drone dispatches, which are recorded to be 22 and 46, respectively.

Two observations can be made regarding the execution times recorded for the three heuristics. First, the execution time generally increases as the problem size increases. For example, for network B, which includes 50 customers and 8 stations, average execution times of 2.068, 0.857, and 1.386 seconds were recorded for the HCWH, VDH, and DDH, respectively. For network G, which includes 100 customers and 24 stations, the average execution times jumped to 75.639, 27.247, and 27.211 seconds, respectively. Second, the execution times of the VDH and DDH are less than that of the HCWH. Computing the multimodal savings at each iteration for the HCWH increases the required execution time for that heuristic.

The execution time generally increases for HCWH and VDH in problem instances in which they continue to examine the possibility of eliminating more stations from the vehicle route. In problem instances in which customers are concentrated around a fewer number of stations and/or can be served by dispatching drones from multiple stations, they tend to examine the possibility of eliminating more stations, which increases its execution time. For example, as shown in Table 6-4, problem instances G-4 and G-6 recorded the highest execution times for the HCWH, and had vehicle routes with fewer stations (20 stations).

Table 6-3: Performance of the heuristics for 50 customer instances.

Instance	HCWH						VDH						DDH					
	C_{Total}	C_V	C_D	η	ή	T	C_{Total}	C_V	C_D	η	ή	T	C_{Total}	C_V	C_D	η	ή	T
B-1	190.7	80.0	110.7	7	20	4.440	190.7	80.0	110.7	7	20	1.517	222.6	114.1	108.5	8	20	1.666
B-2	206.5	94.1	112.4	8	21	1.990	206.5	94.1	112.4	8	21	0.823	233.1	120.0	113.1	8	22	1.572
B-3	225.1	94.1	131.0	8	23	1.936	225.1	94.1	131.0	8	25	0.861	242.7	112.4	130.3	8	24	1.220
B-4	221.3	94.1	127.3	8	23	1.840	219.8	94.1	125.7	8	22	0.827	249.5	120.0	129.5	8	25	1.847
B-5	199.8	94.1	105.6	8	22	1.693	201.3	94.1	107.2	8	22	0.736	216.1	108.3	107.8	8	23	1.335
B-6	194.3	94.1	100.2	8	19	1.687	195.2	94.1	101.1	8	19	0.783	215.6	114.1	101.5	8	20	1.188
B-7	221.0	94.1	126.9	8	23	1.712	221.0	94.1	126.9	8	23	0.752	235.9	108.3	127.6	8	23	1.145
B-8	193.9	94.1	99.8	8	20	1.721	196.9	94.1	102.8	8	19	0.739	224.8	126.5	98.3	8	18	1.156
B-9	207.5	94.1	113.4	8	20	1.834	210.1	94.1	116.0	8	20	0.781	206.9	94.1	112.8	8	20	1.229
B-10	198.1	94.1	104.0	8	21	1.824	198.1	94.1	104.0	8	21	0.750	228.1	120.6	107.5	8	21	1.498
B mean	205.8	92.7	113.1	8	21	2.068	206.5	92.7	113.8	8	21	0.857	227.5	113.8	113.7	8	22	1.386
C-1	254.0	136.6	117.4	11	22	11.806	254.0	136.6	117.4	11	22	3.421	305.0	198.1	106.8	14	21	3.005
C-2	273.2	148.3	124.9	13	28	6.649	284.9	148.3	136.6	12	27	2.999	289.9	166.5	123.4	14	25	2.261
C-3	280.7	160.0	120.7	15	26	3.936	280.9	148.3	132.6	13	25	3.072	314.7	194.8	119.9	15	23	1.999
C-4	264.1	154.1	110.0	14	27	3.611	266.2	142.4	123.8	12	23	3.211	276.5	166.5	110.0	14	20	2.349
C-5	269.2	140.6	128.6	11	23	16.294	268.2	142.4	125.8	11	23	4.867	340.6	218.9	121.7	15	25	2.867
C-6	256.6	128.3	128.3	11	24	8.475	261.7	128.3	133.4	11	25	2.718	356.7	234.5	122.2	13	22	2.267
C-7	278.4	154.1	124.3	14	25	6.620	286.0	150.7	135.3	12	27	3.766	320.3	200.7	119.6	15	23	3.738
C-8	275.5	154.2	121.4	13	23	8.060	286.9	154.2	132.7	12	25	4.016	329.2	210.5	118.7	15	23	3.348
C-9	260.5	142.4	118.1	12	22	10.216	255.2	128.3	126.9	11	22	3.769	311.7	197.2	114.5	14	22	2.530
C-10	262.5	136.6	125.9	11	24	8.400	262.5	136.6	125.9	11	24	2.735	277.8	160.7	117.1	13	23	1.199
C mean	267.5	145.5	122.0	11	24	8.406	270.7	141.6	129.1	11	24	3.457	312.2	194.8	117.4	14	23	2.556
D-1	326.4	202.4	124.0	18	25	16.182	335.8	208.3	127.5	17	26	7.649	383.3	263.1	120.2	20	26	3.812
D-2	355.7	216.5	139.2	18	28	30.226	361.7	216.6	145.1	17	30	11.993	415.3	279.5	135.8	22	29	3.401
D-3	347.9	202.4	145.5	18	28	33.708	348.8	202.4	146.4	17	29	11.935	414.9	277.2	137.7	22	29	2.799
D-4	376.1	222.4	153.7	19	30	25.122	356.5	196.5	160.0	17	31	12.840	442.2	293.1	149.1	22	30	5.601
D-5	336.6	202.4	134.2	17	25	15.615	345.3	202.4	142.9	14	27	9.352	372.0	240.6	131.4	19	27	3.948
D-6	320.1	193.0	127.1	15	25	22.202	313.6	186.5	127.1	15	25	8.273	369.3	250.7	118.6	19	24	2.266
D-7	363.0	208.9	154.1	17	29	33.180	363.9	208.9	155.0	15	28	10.824	394.2	254.8	139.4	20	27	3.047
D-8	356.2	222.4	133.8	20	26	12.937	347.7	206.5	141.2	17	28	9.051	397.9	269.0	128.9	21	27	3.461
D-9	334.0	194.8	139.2	16	26	15.640	348.7	199.0	149.7	14	28	8.145	386.1	254.0	132.1	18	26	5.284
D-10	347.4	222.4	125.0	19	25	12.463	348.5	222.4	126.1	19	26	7.248	374.1	254.8	119.3	21	25	3.377
D mean	346.3	208.8	137.6	18	27	21.728	347.1	205.0	142.1	16	28	9.731	394.9	263.7	131.2	20	27	3.699

C_{Total}: Total cost C_V: Vehicle cost C_D: Drone cost η: Number of vehicle stops ή: Number of drone dispatches T: Execution time

Table 6-4: Performance of the heuristics for 100 customer instances.

Instance	HCWH						VDH						DDH					
	C_{Total}	C_V	C_D	η	ή	T	C_{Total}	C_V	C_D	η	ή	T	C_{Total}	C_V	C_D	η	ή	T
E-1	257.7	94.1	163.6	8	33	12.512	257.7	94.1	163.6	8	33	4.132	288.7	124.7	164.0	8	33	6.762
E-2	278.5	94.1	184.4	8	35	13.721	278.6	94.1	184.5	8	35	4.848	284.1	100.0	184.1	8	37	9.567
E-3	266.0	94.1	171.9	8	37	13.470	269.7	94.1	175.6	8	35	4.754	298.9	126.5	172.3	8	36	9.650
E-4	276.3	100.0	176.3	8	35	12.622	272.9	94.1	178.8	8	35	4.317	285.8	106.5	179.3	8	35	10.768
E-5	259.4	94.1	165.3	8	37	12.530	260.7	94.1	166.6	8	38	4.348	286.9	120.7	166.2	8	37	7.974
E-6	254.9	94.1	160.8	8	34	13.294	254.9	94.1	160.8	8	34	4.960	281.9	120.0	161.9	8	34	6.800
E-7	277.6	94.1	183.5	8	39	15.332	277.6	94.1	183.5	8	39	5.272	307.0	126.5	180.5	8	39	10.259
E-8	271.3	94.1	177.2	8	34	12.445	272.3	94.1	178.2	8	34	4.770	306.3	128.3	178.0	8	34	6.585
E-9	288.1	94.1	194.0	8	37	14.381	289.6	94.1	195.5	8	38	5.224	310.6	112.4	198.2	8	38	12.629
E-10	262.7	94.1	168.6	8	36	12.573	257.2	94.1	163.1	8	35	4.427	299.1	132.4	166.7	8	33	12.478
E mean	269.2	94.7	174.5	8	36	13.288	269.1	94.1	175.0	8	36	4.705	294.9	119.8	175.1	8	36	9.347
F-1	384.4	160.0	224.4	15	43	20.967	381.5	154.1	227.4	14	41	10.526	407.7	188.3	219.5	15	41	21.384
F-2	372.2	160.0	212.2	15	42	13.461	366.0	160.0	205.9	15	40	5.396	407.0	198.9	208.1	15	41	19.333
F-3	360.0	160.0	200.0	15	39	20.040	358.9	154.1	204.8	14	41	10.197	395.6	197.1	198.5	15	40	14.644
F-4	373.0	168.3	204.7	15	39	14.181	368.0	160.0	208.0	15	40	5.632	424.7	218.9	205.8	15	40	18.300
F-5	369.2	162.4	206.8	14	41	31.731	371.2	162.4	208.8	14	41	10.06	417.2	214.1	203.1	15	39	14.889
F-6	361.8	154.1	207.7	14	38	46.113	368.4	148.3	220.1	13	42	14.403	417.6	212.3	205.3	15	39	15.201
F-7	386.6	168.3	218.3	15	42	13.736	382.0	160.0	222.0	15	43	5.619	436.0	217.7	218.3	15	43	18.313
F-8	378.3	154.2	224.1	14	44	33.654	381.3	154.1	227.2	14	43	11.136	453.6	226.7	226.9	15	43	21.351
F-9	381.5	160.0	221.5	15	44	13.708	383.4	160.0	223.4	15	44	5.756	381.5	160.0	221.5	15	44	20.127
F-10	369.7	160.0	209.7	15	39	13.424	377.7	154.1	223.6	14	41	10.260	432.0	222.4	209.6	15	38	15.306
F mean	383.5	169.0	214.6	15	42	22.102	384.0	165.5	218.5	15	42	10.472	417.3	205.6	211.7	15	41	17.885
G-1	480.1	234.1	246.0	21	46	70.449	492.3	242.4	249.9	21	47	22.865	510.3	273.0	237.3	23	47	23.482
G-2	501.0	254.1	246.9	24	47	16.390	510.5	254.1	256.4	24	48	9.337	580.2	334.2	246.0	24	46	35.149
G-3	450.2	232.4	217.8	21	45	102.881	458.3	233.0	225.3	19	45	40.653	503.3	294.8	208.5	24	45	24.259
G-4	483.9	230.7	253.2	20	49	207.618	490.3	224.9	265.5	19	51	42.464	554.6	314.0	240.6	24	47	32.429
G-5	508.4	250.7	257.7	22	48	40.781	508.6	250.7	257.9	22	49	16.861	587.9	341.9	246.0	23	46	16.502
G-6	475.2	222.4	252.8	20	47	113.370	480.6	228.3	252.3	20	47	33.637	523.4	274.3	249.1	23	46	36.320
G-7	445.1	228.3	216.8	21	43	43.609	455.9	236.6	219.3	21	44	16.189	490.8	276.6	214.2	22	42	32.231
G-8	508.2	270.7	237.5	24	46	35.030	507.5	258.9	248.6	22	49	25.679	545.1	308.9	236.2	24	45	27.257
G-9	487.6	248.3	239.3	23	46	69.989	526.8	263.7	263.1	20	53	38.790	544.5	310.7	233.8	24	45	17.224
G-10	468.4	242.4	226.0	22	43	56.276	479.6	242.4	237.2	21	45	25.997	551.3	326.0	225.3	23	44	27.263
G mean	480.8	241.4	239.4	22	46	75.639	491.0	243.5	247.5	21	48	27.247	539.1	305.4	233.7	23	45	27.211

C_{Total}: Total cost C_V: Vehicle cost C_D: Drone cost η: Number of vehicle stops ή: Number of drone dispatches T: Execution time

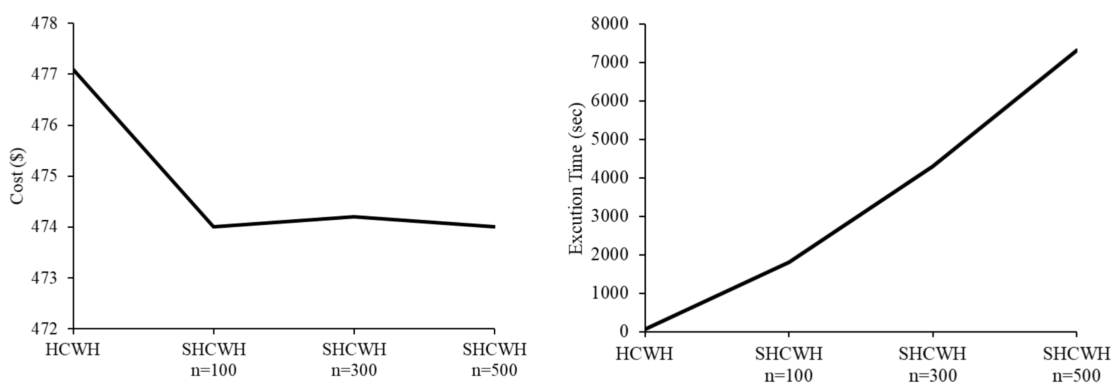
6-3-3. Deterministic HCWH versus Stochastic HCWH

A stochastic version of the deterministic HCWH is implemented, the SHCWH. The SHCWH starts by randomizing the descending-ordered savings list. The first E elements in the savings list are selected and randomly rearranged. The process is repeated for the next T elements until the entire list is randomized. The problem is again solved using the randomized savings list. If a better solution is found, the new savings list is updated and randomly rearranged as described above. Otherwise, the current savings list is again randomized and used to determine a new solution. If no better solution is found for a pre-specified number of iterations, n, the heuristic terminates, producing the best solution in all iterations.

Table 6-5: Comparison between the HCWH and the SHCWH.

Instance	HCWH			
	C_{Total}($)	T (seconds)	Δ(%)	Time Ratio
G-1	480.1	81.767	-	-
G-2	501.0	23.867	-	-
G-3	450.2	110.063	-	-
Instance	SHCWH $n = 100$			
	C_{Total}($)	T (seconds)	Δ(%)	Time Ratio
G-1	475.5	1983.698	0.96	24.26
G-2	501.0	663.630	0.00	27.81
G-3	445.5	2794.250	1.04	25.38
Instance	SHCWH $n = 300$			
	C_{Total}($)	T (seconds)	Δ(%)	Time Ratio
G-1	476.1	3390.732	0.83	41.47
G-2	501.0	2358.453	0.00	98.81
G-3	445.5	7170.176	1.04	65.15
Instance	SHCWH $n = 500$			
	C_{Total}($)	T (seconds)	Δ(%)	Time Ratio
G-1	475.5	7913.467	0.96	96.78
G-2	501.0	3150.885	0.00	132.02
G-3	445.5	10886.589	1.04	98.91

We compared the performance of the HCWH to that of the SHCWH. The two heuristics were used to obtain the solution for three different instances of network G. The number of iterations, n, considered for the SHCWH are 100, 300 and 500, and E is randomly generated such that it ranges from zero to six, respectively. The results gives the percentage improvement, Δ, in the total network cost and the magnitude by which the execution time increased (as multiples of the HCWH's execution time), as compared to those of the deterministic HCWH.



(a) Cost ($) (b) Execution Time (sec)

Figure 6-1: Comparison between the HCWH and the SHCWH.

As presented in Table 6-5 and Figure 6-1, the SHCWH is able to achieve a slight solution improvement of about 1.0% for instances G-1 and G-3. No improvement is recorded with the increase in the number of iterations. For example, for the 100 iteration case, the SHCWH's execution time for G-3 is recorded to be about 25 times that recorded for HCWH. No improvement is recorded for instance G-2. The slight improvement in the cost is associated with exponential increase in the execution time as shown in Figure 6-1.

6-3-4. Mothership versus Vehicle-Only Operation

In this set of experiments, we evaluate the potential cost savings associated with using the mothership system rather than depending only on the vehicle, as is in the current practice. Two scenarios are compared in this set of experiments. In the first scenario, two drones were dispatched and collected from one vehicle to serve all customers. The HCWH is used to obtain the solution for all test cases that adopt the mothership system. In the second scenario, one vehicle with no drones was used to serve all customers. An optimal solution (Applegate et al., 2008) and CW algorithm-based solution (Clarke and Wright, 1964) that includes all customers are obtained and used to benchmark the effectiveness of the mothership system.

Table 6-6: Impact of different cost-ratio for 50 customer instances.

Drone-Vehicle Cost Ratio	Network B		Network C		Network D	
	$\rho_{Optimal}$	ρ_{CW}	$\rho_{Optimal}$	ρ_{CW}	$\rho_{Optimal}$	ρ_{CW}
1:2	1.66	1.66	1.35	1.31	1.34	1.26
1:5	1.09	1.09	0.98	0.95	1.02	0.96
1:10	0.90	0.90	0.83	0.80	0.92	0.86
1:25	0.79	0.79	0.78	0.75	0.85	0.80
1:50	0.75	0.75	0.75	0.73	0.83	0.78

$\rho_{Optimal}$: Optimal traveling salesman solution ρ_{cw}: Vehicle routing solution obtained using CW algorithm

Table 6-7: Impact of different cost-ratio for 100 customer instances.

Drone-Vehicle Cost Ratio	Network E		Network F		Network G	
	$\rho_{Optimal}$	ρ_{CW}	$\rho_{Optimal}$	ρ_{CW}	$\rho_{Optimal}$	ρ_{CW}
1:2	1.45	1.41	1.37	1.32	1.40	1.32
1:5	0.89	0.86	0.89	0.86	0.99	0.94
1:10	0.70	0.68	0.73	0.70	0.85	0.80
1:25	0.59	0.57	0.64	0.61	0.77	0.72
1:50	0.55	0.53	0.61	0.58	0.73	0.69

$\rho_{Optimal}$: Optimal traveling salesman solution ρ_{cw}: Vehicle routing solution obtained using CW algorithm

Comparing the mothership system with the optimal solution provides a real evaluation of how beneficial the introduction of drones may be. The comparison with the CW solution allows the mothership system and the vehicle-only system to be compared when their solutions are obtained using the same technique. When constructing the vehicle-only route, all customers are assumed to be accessible by the vehicle, and a direct link is assumed to be between any two customers. Networks B to G described above are used to compare these two scenarios. In addition, drone/vehicle cost ratios that range from 1:2 to 1:50 are considered. Table 6-6 and Table 6-7 give the results for the network instances with 50 and 100 customers, respectively. The tables give the operation cost ratios between the mothership and the vehicle-only solutions obtained using the optimal TSP solution ($\rho_{Optimal}$) and the CW algorithm (ρ_{CW}).

As shown in the tables, the mothership system is generally more cost effective than the vehicle-only system, especially when the drone's operation cost is significantly less than the vehicle cost. For example, when the drone operation cost is only half the vehicle operation cost, the vehicle-only scenario outperforms the mothership system under the assumption that the vehicle has access to all customers through direct links. As the drone operation cost decreases compared to the vehicle operation cost, the mothership system is shown to significantly outperform the vehicle-only scenario.

For example, considering a drone-vehicle cost ratio of 1:25 and comparing with the CW vehicle routing solution, cost savings of 20% and 28% are recorded for network D with 50 customers and network G with 100 customers, respectively. These cost saving percentages are recorded at 15% and 23% for the same drone-vehicle cost ratio when the optimal vehicle route is obtained for the vehicle-only scenario.

Another interesting observation is related to the pattern by which the operation cost of the mothership system improves as the drone operation cost decreases as compared to that of the vehicle. For example, considering network F, $\rho_{Optimal}$ decreases by 9% when the drone-vehicle operation cost ratio changes from 1:10 to 1:25. This ratio decreases by only 3% when the drone-vehicle operation cost ratio changes from 1:25 to 1:50. Thus, it is worth investing to reduce the drone-vehicle cost ratio from 1:10 to 1:25 as it yields significant savings in the operation cost of the mothership system. Additional investment to further reduce the drone cost does not yield the same level of overall cost improvement. These results are comparable with the savings results reported in Ha et al. (2018), in which one drone is used in the form of flying side-kick delivery system.

The analysis presented above is extended by conducting an experiment in which we compare the performance of the mothership system with that of the vehicle-only system, which is solved considering three different algorithms: a) branch-and-bound; b) Clarke and Wright and c) nearest neighbor. Network instances B, C, and D with drone-to-vehicle operation cost ratio of 1:25 are considered in this experiment. The results of this experiment are presented Table 6-8, which gives the optimal solution obtained using the branch and bound algorithm for the vehicle-only system. The table also gives the corresponding solutions obtained using the CW algorithm and the nearest neighbor algorithm along with their gaps, respectively. In addition, the performance of the mothership system using the HCWH is given along with its ratio, ρ, to the optimal solution obtained using the branch and bound algorithm.

Table 6-8: Comparing the performance of the mothership system and the vehicle-only system solved using different solution methodologies.

Instance	Vehicle Only					Mothership System	
	Branch and Bound	Clarke and Wright		Nearest Neighbor		HCWH	
	Cost ($)	Cost ($)	Gap (%)	Cost ($)	Gap (%)	Cost ($)	ρ
B-1	1580.01	1582.76	0.17	1676.93	5.78	1289.12	0.82
B-2	1727.96	1730.11	0.12	1865.01	7.35	1307.77	0.76
B-3	1713.91	1714.44	0.03	1960.46	12.58	915.23	0.53
C-1	2360.46	2452.20	3.74	2701.15	12.61	1824.52	0.77
C-2	2252.65	2334.21	3.49	2721.88	17.24	1731.89	0.77
C-3	2518.30	2689.89	6.38	3086.07	18.4	2048.52	0.81
D-1	3386.45	3727.97	9.16	4111.62	17.64	2846.29	0.84
D-2	3038.19	3132.92	3.02	3767.28	19.35	2675.78	0.88
D-3	2976.86	3174.51	6.23	3283.27	9.33	2595.82	0.87

Based on the obtained results, for the vehicle-only system, a maximum gap of less than 10% is recorded when Clarke and Wright heuristic is used, compared to the solution obtained using the branch-and-bound algorithm. This maximum gap increased to about 20% when the nearest neighbor is applied. Comparing the performance of the mothership system with the optimal solution obtained by the branch-and-bound algorithm, the improvement in the total cost is recorded to range from 47% to 12%, which demonstrates the benefits of the mothership system compared to the vehicle-only system.

6-3-5. Effect of Number of Drones Carried by the Vehicle

In all of the experiments described above, the vehicle is assumed to carry two drones onboard. In this set of experiments, we examine the effect of the number of drones on overall network performance. Scenarios with a vehicle with one, two, and three drones are considered. The vehicle's operation cost is assumed to be twice the drones' cost. The results of this set of experiments is given in Table 6-9.

Table 6-9: The performance of the heuristics considering different number of drone.

instance	One Drone			Two Drones			Three Drones		
	HCWH	VDH	DDH	HCWH	VDH	DDH	HCWH	VDH	DDH
B	209.0	209.0	240.2	207.4	207.4	232.6	206.4	206.4	232.6
C	271.1	274.4	308.9	269.3	273.3	303.2	269.3	272.3	311.3
D	343.7	345.2	405.4	343.3	348.8	404.5	343.5	348.8	419.0
E	272.1	272.1	269.7	267.4	268.7	298.9	266.7	267.6	301.6
F	372.1	373.3	427.6	372.2	368.8	403.4	369.4	368	399.8
G	481.7	491.6	525.4	480.4	487.0	531.3	475.3	483.7	556.8

The results indicate that the effect of increasing the number of drones on the total network cost is not the same across the three heuristics. Increasing the number of drones resulted in a cost reduction when the HCWH and the VDH are used to solve the hybrid routing problem. On the contrary, the cost of the DDH solution increases with an increase in the number of drones. For example, for network G, the HCWH resulted in a cost of $481.70 when one drone is used. This cost decreased to $475.30 for the three-drone scenario. A similar pattern is observed for the VDH. For the DDH, the network cost was recorded at $525.40, $531.30, and $556.80 for one, two, and three drones, respectively.

Using limited number of drones constrains the structure of the drone's routes in order to be able to visit all customers. The drones are forced to make more returns to their dispatching stations. However, as more drones are included, more efficient drone routes could be constructed which, reduces the total cost as observed in the results of the HCWH and VDH. For the DDH, the drone routes are further optimized in a greedy way, which causes significant inefficiencies to the vehicle route as more stops are required for the drones. The increase in the cost of the vehicle route leads to an increase in the total cost of the network as reported above.

6-3-6. Trade-off between Flight Range and Load Carrying Capacity

Carrying a heavier load requires a drone to have a large battery and strong drone frame, which in turn adds weight to the drone and shortens its range. Thus, planning an efficient vehicle-drone delivery service requires examining the trade-off between the drones' flight range and load carrying capacity (Flynt, 2017). For that purpose, a set of experiments are conducted in which drones with different flight ranges and load carrying capacities are considered. The total network operation cost is recorded for several networks.

As illustrated in Figure 6-1, using drones with a small flight range (the left side of the x-axis), irrespective of the load carrying capacity, resulted in networks with high operation costs. Similarly, using drones with limited load carrying capacity (the right side of the x-axis), irrespective of the flight range, increased the total operation cost. For example, an operation cost of \$573.00 is recorded for network G for the scenario in which drones with a flight range of 5.0 miles and a load carrying capacity of 12.0 lbs. are used. Increasing the drone's flight range to 12.0 miles and reducing their load carrying capacity to 5.0 lbs. resulted in an operation cost of \$544.00. The results in the figure show that there is an optimal combination of the drone's flight range and load carrying capacity that minimizes the total operation cost of the network. With the exception of networks E and D, the least operation cost is recorded for drones with a flight range of 8.0 miles and a load carrying capacity of 9.0 lbs. For network E, which has the highest customer density (1.0 customer/mile^2), the least operation cost is recorded for drones with a relatively higher load carrying capacity. A cost of \$267.00 is recorded for drones with flight range of 7.0 miles and load carrying capacity of 10 lbs. On the other hand, for network D, which has a low customer density of 0.10 customer/mile^2, drones with a relatively long flight range (9.0 miles) are required to efficiently serve the sparsely distributed customers.

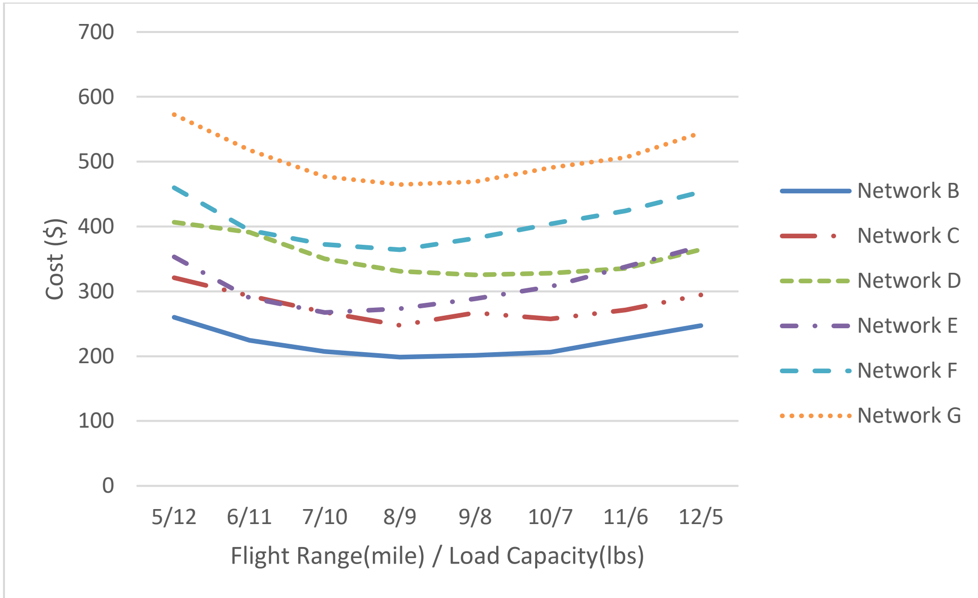


Figure 6-2: Flight range versus load capacity.

6-4. Results for the Mothership System Satisfying LS Rule

This section presents experiments that are designed to evaluate the solution methodologies developed in Chapter 5 and provides a sensitivity analysis to examine the effect of the LS rule on the overall performance of the network. In this set of experiments, the LS parameter a_{im} is randomly generated for every station node, i, and node, m such that $a_{im} = 1$ if $p(x \sim U(0,1) \geq 0.5)$, and zero otherwise. The vehicle's operation cost is assumed to be 25 times that of a drone's. For the post-processing step, the parameters $MaxV$ and $MaxD$ are set to be equal to 100 and 50, respectively. The parameter n is randomly generated such that it ranges from zero to 10.

6-4-1. Comparison with the Exact Optimal Solution

The performance of the MBH and the SBH are compared against the exact solution obtained by solving the MIP developed for the IVDRP-LS using CPLEX 12.6.1 Java callable libraries (IBM, 2009). This comparison is conducted only for networks A-1 to A-10 as CPLEX failed to generate its solution in a reasonable time (< 6 hours) for the larger networks. Table 6-10 summarizes the results of this performance comparison. The table gives the total (multimodal) routing cost and the execution time for each tested case. As shown in the table, the MBH and SBH are able to generate the optimal solution for seven and six of the 10 cases, respectively. For cases in which the heuristics failed to obtain the optimal solution, gaps of less than 2% are recorded. The optimality gaps recorded for the MBH are generally lower than those of the SBH.

Table 6-10: The heuristics' performance compared to the optimal solution.

Instance	Exact Solution (CPLEX)		MBH			SBH		
	Cost ($)	Runtime (Sec.)	Cost ($)	Runtime (Sec.)	Gap (%)	Cost ($)	Runtime (Sec.)	Gap (%)
A-1	525.12	1.031	525.12	0.173	0.0	525.12	0122	0.0
A-2	498.47	1.032	502.89	0.917	0.9	504.50	0.413	1.2
A-3	495.59	1.141	495.59	0.193	0.0	495.59	0.127	0.0
A-4	513.07	1.531	513.07	0.998	0.0	513.07	0.187	0.0
A-5	535.92	1.828	535.92	0.430	0.0	535.92	0.106	0.0
A-6	531.23	468.108	531.62	5.768	0.1	531.62	0.360	0.1
A-7	528.67	4.016	529.93	1.021	0.2	529.93	0.538	0.2
A-8	526.53	817.088	526.53	2.208	0.0	530.05	0.288	0.7
A-9	542.63	5.782	542.63	3.028	0.0	542.63	1.124	0.0
A-10	603.20	83.242	603.20	0.846	0.0	603.20	0.488	0.0

The heuristics' execution times are much less than those recorded by CPLEX. For example, for network A-10, CPLEX's execution time is recorded at 83.242 seconds. The corresponding execution times of the heuristics are less than one second. One can also notice the CPLEX's excessive execution time as the number of customers increases. For example, CPLEX's execution time jumps from 1.828 seconds for A-5 to 468.108 seconds for A-6. Much lower increases in the corresponding heuristics' execution times are recorded. The corresponding execution time for the MBH increases from 0.430 seconds to 5.768 seconds, and the corresponding execution time for the SBH increases from 0.106 seconds to 0.360 seconds. Moreover, it can be noticed that the execution time of CPLEX is not consistent across network instances with the same number of customers. For example, the execution time of network A-7 is 4.016 seconds, while the execution time of network A-8 is 817.008 seconds. The reason for the inconsistency in the execution times is due to the random settings of the LS parameter. As more customers are served by the vehicle due to LS restriction, the drone-related constraints become non-binding and thus CPLEX is able to generate the optimal solution in much less execution time.

6-4-2. Comparing the Performance of the MBH and the SBH

In this set of experiments, we compare the performances of the MBH and the SBH using ten random instances of networks B, C and D, respectively. Table 6-11 summarizes the results of this comparison. For each network instance, the table gives (1) the routing cost of the vehicle (C_V), the drones (C_D), and the entire network (C_{Total}); (2) the number of stops made by the vehicle to dispatch and collect the drones, η; (3) the total number of drone dispatches, ή, to serve the customers; and (4) the execution time, T.

For the majority of the tested instances, the MBH outperforms the SBH in terms of the total routing cost. For example, for the instances of network B, an average total cost of \$1475.60 is recorded for the MBH (\$1380.80 for the vehicle and \$94.80 for the drones). The SBH's corresponding average routing cost is recorded at \$1485.30 (\$1385.10 for the vehicle and \$100.20 for the drone). One might expect this result since the MBH implements two additional procedures to further optimize the total routing cost compared to the SBH. As explained above, the MBH uses a list of multimodal cost savings to construct the vehicle route, while the SBH uses a list that computes the savings for the vehicle only. Furthermore, in the post-processing step, the MBH randomizes the savings lists of both the vehicle and the drones to improve the total cost, while the SBH randomizes the vehicle's savings list only.

Although these two additional procedures reduce the total routing cost, they contribute significantly to its execution time. As shown in the table, the SBH's execution time is always less than that of the MBH. For example, for the instances of network B, which includes 50 customers and 8 stations, average execution times of 78.057 and 16.280 seconds are recorded for the MBH and the SBH, respectively. One can also notice the variation in the execution times recorded by both heuristics across different instances of the same network. For example, execution times of 41.654 and 212.604 seconds were recorded for the MBH to solve instances B-8 and B-9, respectively. This variation is due to the implementation of the post-processing step, where the number of iterations with no solution improvement varies in the different runs. Finally, the two heuristics produce comparable results in terms of the number of stops made by the vehicle and number of drone dispatches. However, the SBH tends to generate solutions with a larger number of drone dispatches compared to the MBH. As the SBH aims to prioritize savings in the vehicle cost over the drone cost, it generates the vehicle route without considering the expense of scheduling more drone dispatches. For example, for the instances of network B, the numbers of drone dispatches recorded by the SBH are always equal to or greater than those of the MBH.

Table 6-11: Performance of the heuristics.

Instance	MBH						SBH					
	C_{Total}	C_V	C_D	η	ή	T	C_{Total}	C_V	C_D	η	ή	T
B-1	1341.8	1261.3	80.5	23	13	53.412	1347.6	1261.3	86.3	23	15	18.976
B-2	1418.6	1329.0	89.6	21	16	136.250	1428.8	1333.8	95.0	21	18	14.522
B-3	1586.9	1516.8	70.1	34	14	62.739	1588.2	1516.8	71.4	34	15	21.199
B-4	1505.7	1397.6	108.1	25	20	33.807	1515.6	1397.6	118.0	25	22	17.554
B-5	1419.3	1306.7	112.6	24	19	37.717	1419.3	1306.7	112.6	24	19	14.013
B-6	1445.4	1349.8	95.6	24	19	50.067	1449.6	1349.8	99.8	24	19	17.651
B-7	1527.3	1414.5	112.8	22	19	24.703	1549.5	1432.8	116.7	22	21	13.304
B-8	1528.5	1439.5	89.0	28	17	41.654	1554.9	1458.8	96.1	28	18	18.119
B-9	1461.1	1356.4	104.7	23	18	212.604	1471.9	1356.4	115.5	23	19	13.978
B-10	1521.3	1436.8	84.5	25	16	66.318	1527.3	1436.8	90.5	25	17	13.486
B mean	1475.6	1380.8	94.8	25	17	78.057	1485.3	1385.1	100.2	25	18	16.280
C-1	2108.8	2020.7	88.1	34	18	63.445	2124.4	2020.7	103.7	34	20	37.680
C-2	2318.9	2196.0	122.9	29	22	37.675	2327.2	2196.0	131.2	29	24	36.717
C-3	2396.8	2272.2	124.6	28	25	48.580	2398.0	2272.2	125.8	28	25	22.162
C-4	2317.6	2208.4	109.2	32	22	63.358	2317.6	2208.4	109.2	32	22	24.837
C-5	2191.1	2071.7	119.4	28	21	29.981	2170.9	2045.6	125.3	28	22	24.269
C-6	2193.9	2075.2	118.7	30	23	49.729	2245.7	2117.7	128.0	30	25	23.302
C-7	2420.2	2313.6	106.6	33	21	95.645	2420.2	2313.6	106.6	33	21	36.001
C-8	2377.6	2260.5	117.1	27	24	85.369	2381.2	2260.5	120.7	27	24	21.306
C-9	2356.1	2236.5	119.6	31	21	53.346	2362.5	2236.5	126.0	31	22	26.967
C-10	2309.0	2173.7	135.3	30	25	51.747	2255.8	2120.5	135.3	30	25	36.358
C mean	2299.0	2182.9	116.1	30	22	57.887	2300.3	2179.1	121.2	30	23	28.960
D-1	3021.9	2921.1	100.8	32	20	67.886	3027.1	2921.1	106.0	32	21	39.137
D-2	3217.0	3116.1	100.9	37	22	65.036	3219.0	3116.1	102.9	37	23	56.415
D-3	3276.5	3150.2	126.3	35	25	73.847	3276.9	3149.6	127.3	34	25	38.701
D-4	3217.8	3094.2	123.6	37	23	46.579	3240.2	3115.3	124.9	37	23	39.776
D-5	3002.9	2894.2	108.7	31	21	52.384	3018.6	2908.9	109.7	31	22	72.184
D-6	2875.7	2778.3	97.4	32	21	63.905	2883.2	2778.3	104.9	32	23	38.216
D-7	3062.7	2935.3	127.4	33	24	53.189	3087.4	2959.2	128.2	33	25	54.248
D-8	3258.5	3120.5	138.0	32	27	92.200	3248.6	3110.6	138.0	32	27	52.912
D-9	3170.5	3051.7	118.8	34	22	55.490	3177.9	3051.7	126.2	34	23	35.435
D-10	3422.2	3300.7	121.5	37	24	105.515	3491.0	3369.2	121.8	37	25	63.115
D mean	3152.6	3036.2	116.4	34	23	67.603	3167.0	3048.0	119.0	34	24	49.014

C_{Total}: Total cost C_V: Vehicle cost C_D: Drone cost η: Number of vehicle stops ή: Number of drone dispatches T: Execution time

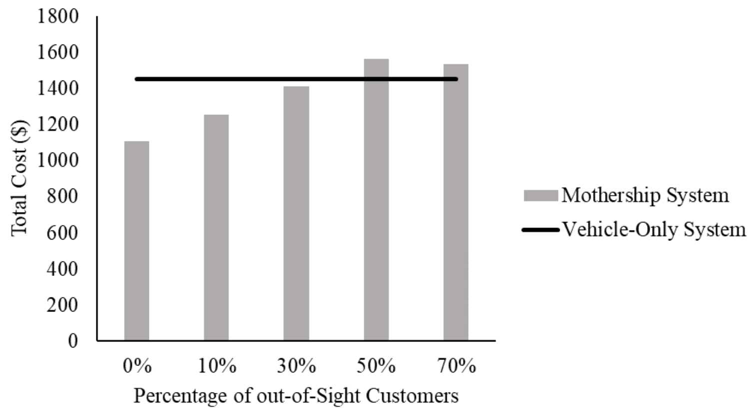
6-4-3. Effect of LS Rule on the Performance of the Mothership System

The results of this set of experiments quantify the additional cost resulting from mandating the LS rule for drone flights. The experiments are conducted using networks B, C, and D where the percentage of out-of-sight customers from any station is assumed to range from zero to 100%. A high percentage of out-of-sight customers represents dense urban areas with obstructions (e.g., high rise buildings). The results are presented for the two scenarios with and without mandating the LS rule. As explained earlier, mandating the LS rule requires customers who are out of sight to be served by the vehicle. Ignoring the LS rule, all customers are assumed to be served by drones. The results of these experiments are shown in Table 6-12 and Figure 6-4. Table 6-12 gives the percentage increase in the cost, ρ, associated with mandating the LS rule, and the corresponding execution time T. Figure 6-3 demonstrates the extra cost associated with having more percentage of obstructed customers and compares the cost of the mothership system with that of the vehicle only system.

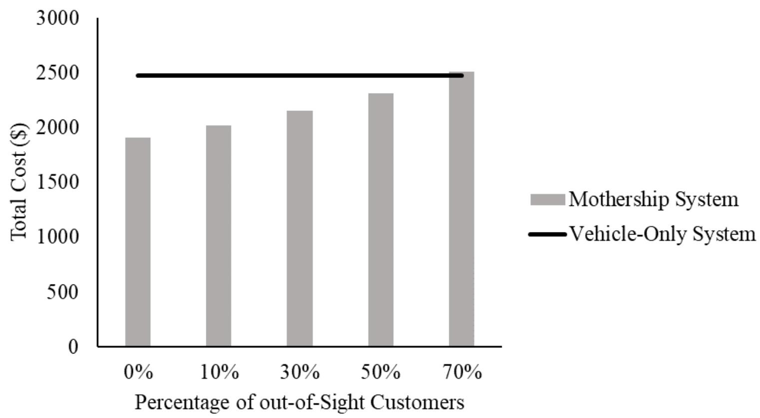
Table 6-12: Impact of the LS regulatory rule.

Percentage of out-of-Sight Customers	Network B			Network C			Network D		
	Cost ($)	ρ (%)	T (sec)	Cost ($)	ρ (%)	T (sec)	Cost ($)	ρ (%)	T (sec)
0%	1106.33	0.00	1806.28	1904.88	0.00	226.69	2754.33	0.00	11.09
10%	1253.50	13.30	533.16	2023.02	6.20	143.91	2802.03	1.73	157.38
30%	1409.60	27.41	157.32	2154.07	13.08	63.31	2858.48	3.78	80.60
50%	1562.05	41.19	61.96	2309.55	21.24	31.83	2962.31	7.55	76.12
70%	1535.06	38.75	47.83	2511.45	31.84	32.85	3157.57	14.64	39.09
100%	**1450.92**	**31.15**	**15.36**	**2474.62**	**29.91**	**23.27**	**3310.65**	**20.20**	**37.52**

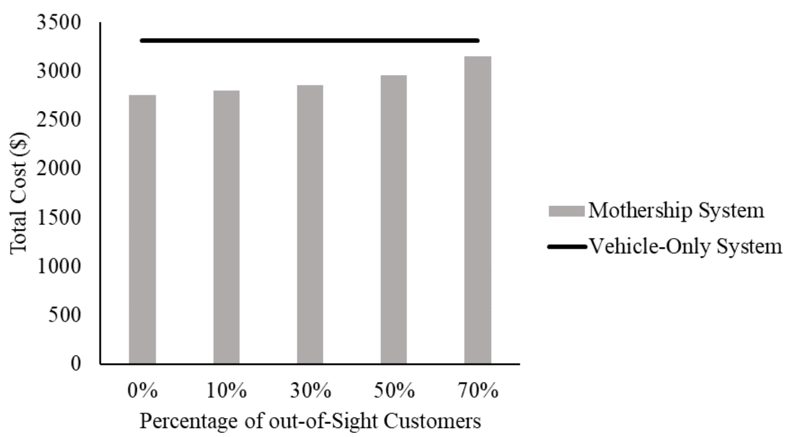
$\rho = \ (Cost\ with\ LS\ \ - Cost\ wihtout\ LS\ )/Cost\ without\ LS$



(a) B Network



(b) C Network



(c) D Network

Figure 6-3: Mothership system versus vehicle-only system.

As show in the table, the cost tends to increase with the increase in the percentage of out-of-sight customers. For example, for network B, the cost increases from $1253.50 to $1409.60 when the percentage of out-of-sight customers increases from 10% to 30%. The results also demonstrate the effectiveness of the mothership system compared to the vehicle-only delivery system as shown in Figure 6-4. For instance, considering network B which has a density of 0.5 customers per square mile, it is cheaper to use the vehicle-only system rather than the mothership system for cases in which the percentage of out-of-sight customers exceeds 30%. For example, a percentage cost increase of 41.19% is recorded when 50% of the customers are within sight. This percentage is higher than the one recorded when all customers are out of sight (ρ = 31.15%) and served only by the vehicle. For network D with 0.10 customers per square mile density, the mothership system is more cost effective than the vehicle-only system, even if the percentage of out-of-sight customers reaches 70%. The results of this experiment allow service providers to decide on the most suitable equipment (vehicle-only system vs. integrated vehicle-drone system) for each service area based on its LS restrictions.

The effect of having a higher percentage of customers with obstructed sight distance varies across networks. The execution time for networks B and C, characterized by dense customer distributions, decreases with the increase in the percentage of out-of-sight customers as more customers are served by the vehicle. For example, the execution time of network B decreases from 1806.28 seconds in the case where all customers have clear LS to 15.36 seconds for the case in which all customers are obstructed. A different pattern is observed for network D with sparse customers. As the percentage of customers with obstructed LS increases to 10%, the execution time jumped from 11.09 seconds to

157.38 seconds. As the percentage of obstructed-customers further increases, the execution time gradually decreases. When all customers are blocked, requiring them to be served by the vehicle, an execution time of 37.52 seconds is recorded.

6-4-4. Effect of Increasing the Drones' Flying Range

In this set of experiments, we examine the effect of using drones with enhanced capabilities, in terms of increased flight range, on the performance of the system. Networks B, C, and D are again considered in this set of experiments. The percentage of customers with obstructed LS from their nearest drone dispatching stations is assumed at 10%, 30% and 50%, respectively. For each network, the total routing cost and the percentage of customers served by drones are recorded, considering different flight ranges for the drones. The results of these experiments are given in Figure 6-2.

As shown in the figure, the total routing cost generally decreases as the drones' flying range increases. Also, an increase is recorded in the number of customers who are served by drones as the drones' flying range increases up to about nine miles at which distance all customers are served by drones. A higher cost is recorded as the percentage of customers with obstructed LS from their closest stations increases. This pattern is observed for all networks. However, the percentage increase with the increase of the flight range decreased for network C and D but not for network B. For example, for network C with drones' flying range that is equal to seven miles, the total cost increases from \$1824.52 to \$2114.52 (percentage increase of 15.89%) as the percentage of customers with obstructed LS increases from 0% to 50%. However, this percentage increase in the cost is recorded at

10.35% when the flight range increased from seven to 12 miles. For network B which has the highest customer density, increasing the drones' flying range has no effect on reducing the reported increased in the routing cost associated with having more customers with obstructed LS.

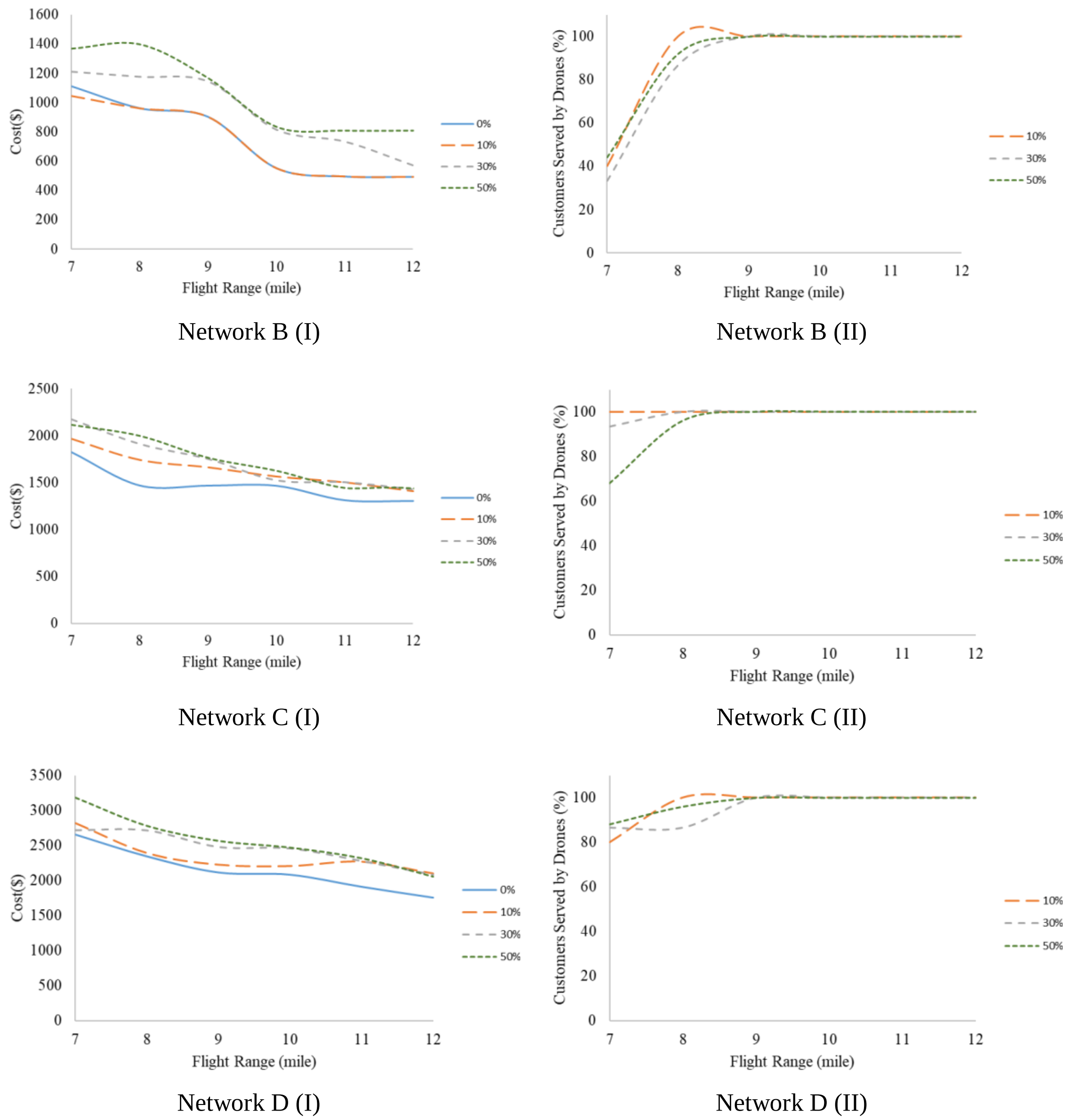


Figure 6-4: Effect of increasing the drones' flight range.

6-5. Summary

In this chapter, a set of experiments were conducted. Based on the obtained results, the following can be concluded:

(a) The developed heuristics produce high quality solutions that are comparable to the exact optimal solution for small problem instances.

(b) The heuristics are able to solve large problem instances in shorter execution times.

(c) The stochastic version of the HCWH is able to achieve a slight solution improvement.

(d) The mothership system is generally more cost effective than the vehicle-only system.

(e) The effect of increasing the number of drones on the total network cost is not the same across the HCWH, VDH, and DDH.

(f) The network operation cost is minimum when the used drones are balanced in terms of their flight range and load carrying capacity.

(g) The impact of the LS rule is quantified allowing service providers to decide on the most suitable equipment configuration (vehicle-only system vs. integrated vehicle-drone system) for the service area under consideration based on the level of LS restrictions.

Chapter 7

CASE STUDY

7-1. Introduction

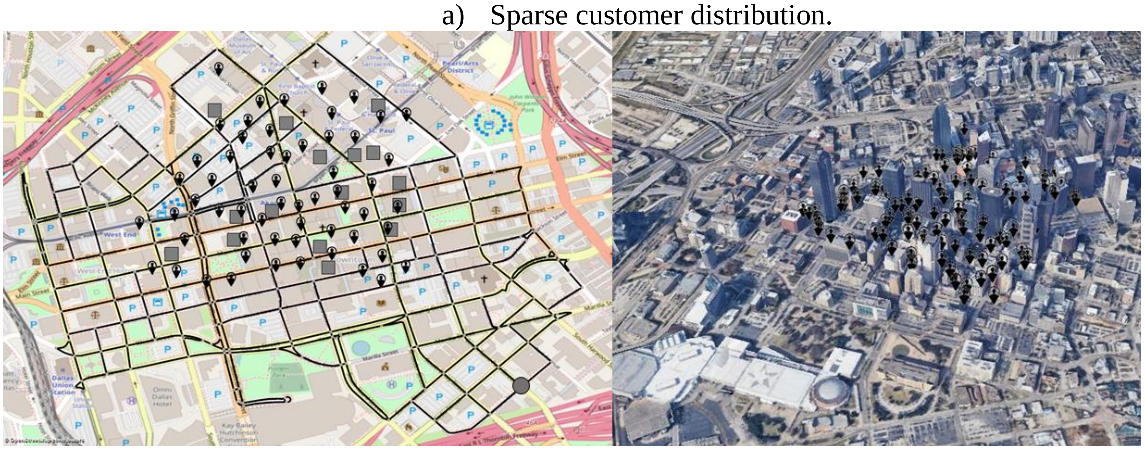
This Chapter provides a case study for the sake of quantifying the impact of the LS rule on overall system performance considering real-world urban settings. The MBH, explained in Chapter 5, is implement in Java to provide pick-up and delivery services in the downtown area of the City of Dallas, which spans an area of 1.123 square miles. This Chapter is organized as follows. Section 7-2 describes all the parameters considered by this case study. Sections 7-3 summarizes main operation statistics resulting from applying the MBH to serve customers in the studied area. Section 7-4 presents a sensitivity analysis to examine the effect of LS rule on several system parameters. Finally, Section 7-5 gives a summary of the chapter.

7-2. Description of the Case Study

In this case study, the MBH is applied to provide pick-up and delivery services in the downtown area of the City of Dallas, which spans an area of 1.123 square miles. As shown in Figure 7-1, the downtown area consists of two sections: the north section characterized by high-rise buildings (hotels, professional offices and apartments) and the south section characterized by short buildings (light industries and commercial services). Sixteen buildings in the study area have a height greater than 400 ft., which is the maximum flying altitude approved by FAA. Considering the proprietary nature of the demand data, two hypothetical customer distribution scenarios are assumed in which two hundred customers are randomly distributed in the service area. The first scenario represents a sparse customer distribution in which the two hundred customers are distributed over the entire service area with a density of about 180 customers per square mile. The second scenario considers a dense customer distribution in which the two hundred customers are concentrated in the north section of the service area with a density of 520 customers per square mile. Illustrations of the customer distributions for these two scenarios are given in Figures 7-1 (a) and 7-1 (b), respectively. Similar to the experiments above, each customer is associated with a pick-up weight and/or a delivery weight that are randomly generated following the uniform distribution $U(0.0 \text{ lbs}, 5.0 \text{ lbs})$.

All parking lots available in the downtown area are considered as candidate stations where the vehicle can stop to dispatch and collect the drones. The depot point is assumed to be the closest access point to the service area from an adjacent freeway that connects Amazon's distribution center to the downtown area. A mothership system of one vehicle and two identical drones is used to serve this demand. As mentioned in Chapter 6, both drones are assumed to have a maximum flight range of 7.0 miles and a load-carrying capacity of 10.0 lbs. The vehicle operation cost is assumed to be 25 times that of the drones. The vehicle is routed along the actual roads in the service area, while the drones are assumed to fly along the shortest Euclidian distance from their origins to destinations (i.e., dispatching station to a customer, a customer to customer, or a customer to a collection station). Buildings higher than 400 ft. are considered as obstacles that possibly obstruct the LS. The LS between every possible origin-destination pair of a drone flight is examined. As described above, a two-dimensional visibility graph is constructed. If the straight line connecting any origin-destination pair is obstructed by any obstacle, then the destination is assumed to be out of sight from the origin.

a) Sparse customer distribution.



b) Dense customer distribution.

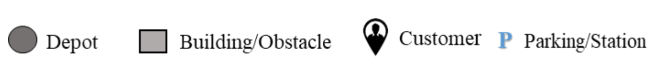


Figure 7-1: Customer distribution scenarios in the downtown area.

7-3. Operation statistics

This section summarizes the primary operation statistics resulting from applying the MBH for the two customer distribution scenarios mentioned above. As shown in the Table 7-1, the number of stops made by the vehicle increased from 17 in the case of sparse customer distribution to 21 for the dense customer distribution. The percentage of customers served by the vehicle is recorded at 3.5% for the sparse customer distribution scenario, compared to 6% for the dense customer distribution scenario. Although one might expect a solution that requires more vehicle stops in the sparse customer distribution case, the high-rise buildings cause more customers to be obstructed in the dense customer distribution scenario. Hence, the vehicle must be routed to keep the drone within the pilot's LS at all times and also to visit customers who cannot be served by the drones.

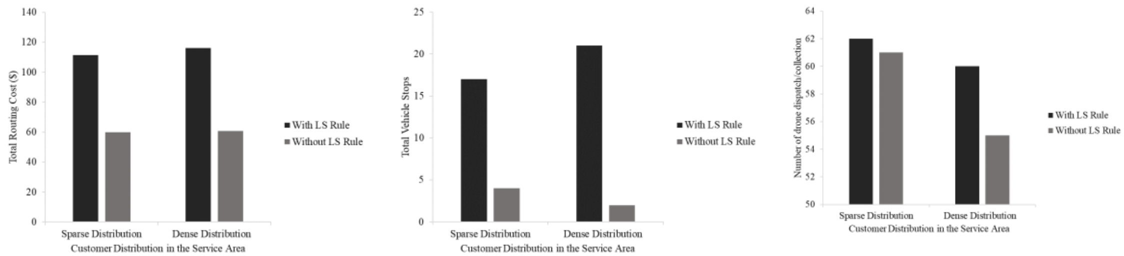
Table 7-1: Sparse versus dense customer distribution.

Performance Measure	Sparse Distribution	Dense Distribution
Number of stops made by the vehicle	17	21
Percentage of customers served by vehicle (%)	3.5	6.0
Number of drone dispatch/collection	62	60
Average number of customers served per drone route	3.11	3.13
Average flight distance per drone trip (miles)	0.45	0.46
Average load per drone trip (lbs.)	7.56	7.08

The table also gives the number of drone dispatches and the average number of customers per drone dispatch for both scenarios. A solution with a slightly smaller number of drone dispatches and more customers served per dispatch is recorded for the network with dense customer distribution, compared to these recorded for the network with sparse distribution scenario. Such results could be contributing to the proximity of the customers in the dense customer distribution scenario, which allows the combination of more customers in the drone route and hence reduces the number of drone dispatches. The average lengths of the drone routes per dispatch are almost equal for the sparse and the dense scenarios. Further investigating the solutions of these two scenarios shows that the drone routes are constrained mainly by the load-carrying capacity of the drones. As shown in the table, the average carried loads per drone tour are close to the maximum load-carrying capacity (10.00 lbs.). Average carried loads of 7.56 lbs. and 7.08 lbs. are recorded for the sparse and the dense customer distribution scenarios, respectively.

7-4. Results and analysis

This section presents the impact of the LS rule on several system parameters. Figure 7-2 compares the total routing cost, number of stops made by the vehicle, and number of drone dispatch/collection for the two customer distribution scenarios with and without mandating the LS rule. As shown in the figure, mandating the LS rule significantly increases the total routing cost. If the LS rule is ignored, routing costs of about \$59.81 and \$60.78 are recorded for the sparse and dense customer distribution scenarios, respectively. Mandating the LS rule increases these costs to about \$111.34 and \$116.13 for these two scenarios.

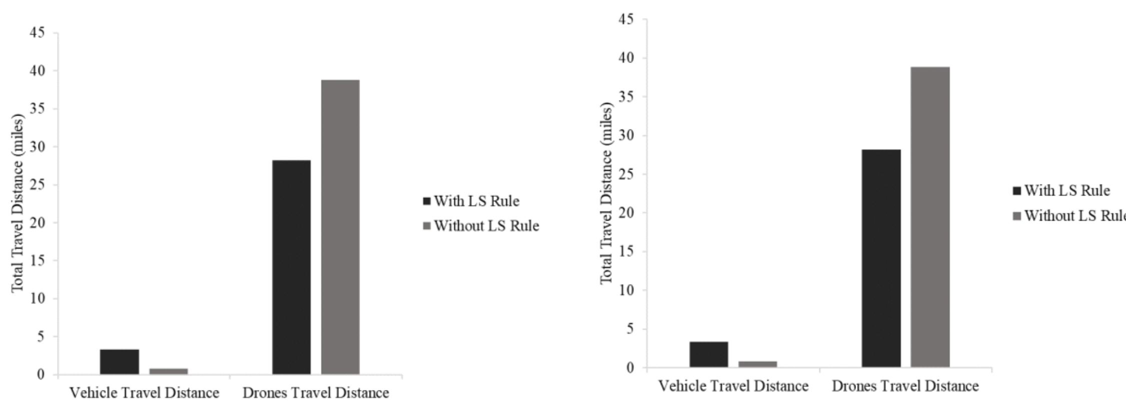


(a) Routing cost (b) No. of Vehicle stops (c) No. of Drone dispatch/collection

Figure 7-2: Impact of mandating the LS rule.

As for the number of stops made by the vehicle, different patterns are observed for the cases with and without mandating the LS rule. Without mandating the LS rule, the vehicle made four stops in the sparse customer distribution scenario and only two stops in the dense customer distribution scenario, respectively. One might expect such results as more stops are needed to cover the sparse demand. On the other hand, mandating the LS rule required the vehicle to make more stops in the dense scenario than those made in the sparse scenario, as the vehicle needs to position itself to allow serving customers with obstructed LS. Finally, the LS rule is shown to have more impact on the number of drone dispatch/collection for the dense scenario. While the number of drone dispatch/collection increased only from 61 to 62 in the sparse customer distribution scenario, it increased from 55 to 60 in the dense scenario. More LS obstruction characterizes the dense customer distribution scenario, which makes it difficult to construct drone routes that combine more customers as in the case in which the LS rule is ignored.

Figure 7-3 compares the distance travelled by the vehicle and the drones for the two customer distribution scenarios with and without mandating the LS rule. In both scenarios, mandating the LS rule increased the distance traveled by the vehicle and decreased the distance traveled by the drones. The vehicle travels more distances to serve customers that are not reachable by any of the drones, and to better position the pilot at stations with no LS obstruction. The traveled distance by the drones significantly decreased as the vehicle was expected to drive closer to the customers to ensure that the drones are within the pilot's LS. This result illustrates the effect of the LS rule on the integrated vehicle-drone system, which is envisioned to increase dependence on drones and reduce vehicle usage.



(a) Sparse customer distribution (b) Dense customer distribution

Figure 7-3: Impact of mandating the LS rule on the vehicle's and the drones' travel distance.

Finally, we examine the impact of relaxing the maximum flying altitude restriction on the routing cost. Figure 7-4 gives the routing cost for different flying altitudes (≥ 400 ft.) for both customer distribution scenarios. As shown in the figure, for both customer distribution scenarios, the cost decreases as the allowed maximum flying altitude increases. The cost continues to decrease until the flying limit reaches 900 ft. as all buildings in the downtown area are below this limit.

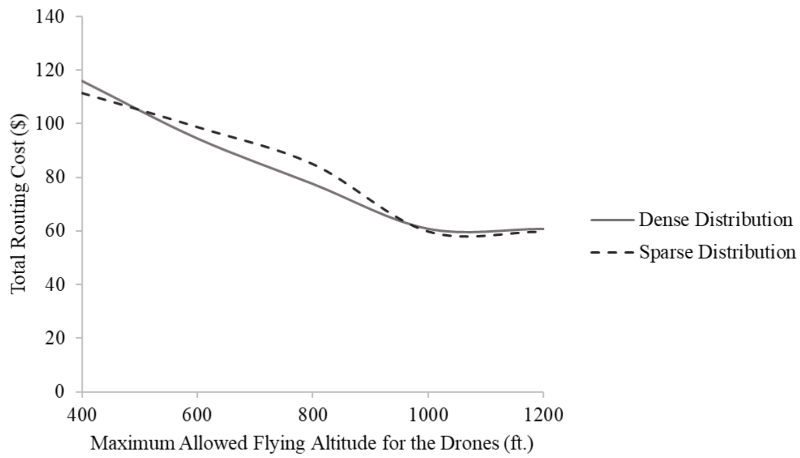


Figure 7-4: Impact of the flying altitude on the routing cost.

7-5. Summary

In this case study, the developed methodology is applied to provide pick-up and delivery services in the City of Dallas' downtown area. Mandating the LS rule is shown to double the total routing cost of the mothership system used to serve customers in the area. These results can be of great importance to the pick-up and delivery service providers to decide on the most suitable equipment configuration (vehicle-only system vs. mothership system) based on the level of LS restrictions in the service area under consideration. Based on the obtained results, it is generally recommended to use the mothership system in areas where the majority of the customers have clear LS (suburban and rural areas), and to adopt the traditional vehicle only delivery system in urban areas with high LS restrictions. Moreover, aviation authorities can use these results to study the tradeoff between the risk associated with using drones as part of the delivery system versus the total system cost associated with increasing the maximum flying altitude of the drones.

Chapter 8

SUMMARY AND FUTURE WORK

8-1. Summary

The vehicle-drone "mothership" system was recently conceptualized to provide efficient pick-up and delivery services. Drones could be mounted on the vehicles and dispatched from pre-specified stations to deliver and pick up products to/from a set of customers distributed in a given service area. To solve a basic mothership system, this research presents a model and efficient solution methodology for the hybrid vehicle-drone routing problem (HVDRP) for pick-up and delivery services. Aviation authorities in the US and abroad mandate several regulatory rules to ensure safe operations for drone-based delivery systems. These rules are expected to have a significant impact on the configuration and the cost performance of these systems. This paper presents a modeling framework for the integrated vehicle-drone routing problem for pick-up and delivery services considering the LS rule mandated by the FAA (IVDRP-LS).

Two mathematical formulations in the form of a mixed integer program are developed which solve for the optimal vehicle and drone routes to serve all customers such that the total cost of the pick-up and delivery operation is minimized. The first formulation captures the vehicle-drone routing interactions and considers the drone's operational constraints including flight range and load carrying capacity limitations. The second formulation includes constraints that represent the vehicle drone interactions, the LS rule, and constraints related to the drones maximum flight range and load-carrying capacity limitations.

A novel solution methodology that extends the classic Clarke and Wright algorithm is developed to solve the HVDRP, namely the hybrid Clarke and Wright heuristic (HCWH). The heuristic takes into consideration the cost savings resulting from connecting stops in the vehicle route and connecting customers in the drone routes that are dispatched and collected at these stops. The performance of the HCWH is benchmarked against a vehicle-driven heuristic (VDH) and a drone-driven heuristic (DDH). In the VDH, an efficient vehicle route is first obtained, and the drones are then routed considering the dispatching and collection stops specified in the vehicle route. The drone-driven routing heuristic determines the drone routes and specifies optimal locations for their dispatching and collection. The vehicle is then routed to visit these stops.

Also, the research presents a novel solution methodology to solve the IVDRP-LS. The heuristics adopt an updated version of the classic CW algorithm to consider the multimodality aspects of the integrated vehicle-drone routing problem and obey the LS rule. The solution methodology implements the MBH with randomization procedure to construct near optimal vehicle and LS-mandated drone routes. The performance of the MBH is benchmarked by comparing its performance against that of the SBH. The SBH is a lighter version of the MBH as it implements a vehicle-driven approach.

The developed heuristics are shown to produce high quality solutions that are comparable to the exact optimal solution for small problem instances. The heuristics are also able to solve large problem instances in shorter execution times. Regarding the heuristics developed to solve a basic mothership system, the HCWH is shown to outperform the VDH and the DDH in terms of minimizing the cost of the entire multimodal network. Concerning the heuristics developed to solve the mothership system that obeys the LS rule, the MBH outperforms SBH in terms of the operation cost yet the SBH is able to generate satisfactory solutions in less execution times.

The results also show the value of adopting the mothership system. Compared to a scenario in which the vehicle is used to visit all customers, the amount of cost reduction increases as the drone's operation cost decreases. Also, the network operation cost is shown to be minimal when the used drones are balanced in terms of their flight range and load carrying capacity. Generally, service areas characterized by high customer density require drones with large load carrying capacity. For service areas with sparse customers, drones with long flight range are more suitable.

In addition, a set of experiments are conducted to study the impact of the LS rule on the overall system performance. The results of these experiments allow service providers to decide on the most suitable equipment configuration (e.g., vehicle-only system vs. integrated vehicle-drone system) for the service area under consideration. As a case study, the developed methodology is applied to provide pick-up and delivery services in the City of Dallas' downtown area. Mandating the LS rule is shown to almost double the total routing cost of the mothership system used to serve all customers in the area. The experiments also show that relaxing this rule by increasing the maximum allowed flying altitude of the drones could significantly reduce the total system cost. The analysis could also assist the aviation authority to adjust the parameters of the LS rule to achieve the optimal balance between safety and operation cost. Based on the results obtained for Dallas' downtown area, slightly increasing the maximum flying altitude of the drones could have a significant impact on the overall routing cost.

8-2. Further Research Directions

While this dissertation provides a foundation to understand the mothership system, it is clear that due to the complexity of the problem there are still several research issues worthy of further investigation. Examples of research extensions for this work include:

a) Designing mothership system with different objectives

In this study we evaluate the efficiency of the mothership system and the impact of the LS rule on the mothership system from cost perspective. Both the HVDRP and the IVDRP-LS tend to minimize the total operational cost. Therefore, the framework could be extended to consider different objective functions. Examples of different objectives include: (1) minimizing the total travel time, which entails defining new variables and parameters that capture the vehicle's and drone's travel time as well as the waiting time of the vehicle at every station; (2) minimize the total carbon and different toxic gases caused by trucking, this extension can determine the environmental impact of the mothership system; and (3) maximizing safety, which requires determining the risks of using the traditional vehicle-only delivery system versus risks of integrating the drones.

b) Multi-vehicle mothership system

This study considers one vehicle with several drones on board. While this assumption is a good start for understanding the mothership system, the formulation and the solution methodology could be extended for a multi-vehicle mothership system in which multiple vehicles are used instead of one vehicle. The formulation could be modified to consider multiple vehicles and their maximum travel distance capacity. Furthermore, the formulation could be extended to allow drones to be shared between vehicles as long as the number of drones on board of each vehicle is conserved. Also, an efficient solution methodology that can solve the problem of the multi-vehicle mothership system can be developed.

c) Mothership system with time window

The formulation presented in Chapter 3 could be extended to solve the problem in which demand pick-up and delivery requests within a time window are considered. The model suggested in this research does not consider vehicle and drone waiting time. The focus was to minimize the distance (cost) traveled by both modes. However, extending this model to consider the customers' time windows will require including the waiting time for both modes as part of the objective function. An improved solution methodology for the above described formulation extension should also be developed. This step would entail extending one of the existing methodologies that solves classic vehicle routing problem with a time window (e.g. simulated annealing (SA), Tabu search (TS), and genetic algorithm (GA)) and modifying the algorithm to solve the mothership system with time window.

BIBLIOGRAPHY

Agatz, N., Bouman, P., & Marie, S. (2016). Optimization approaches for the traveling salesman problem with drone. *TRANSPORTATION SCIENCE*, 1-17.

Agatz, N., Bouman, P., & Schmidt, M. (2018). Optimization approaches for the traveling salesman problem with drone. *TRANSPORTATION SCIENCE*, 739-1034.

Altinkemer, K., & Gavish, B. (1991). Parallel Savings Based Heuristics for the Delivery Problem. *Operations Research*, 39(3), 352-525.

Applegate, D., Bixby, R., Chvatal, V., & Cook, W. (2008). Concorde TSP solver.

Arnold, L. L., & Zandbergen, P. A. (2011). Positional accuracy of the wide area augmentation system in consumer-grade GPS units. *Computers & Geosciences*, 37(7), 883-892.

Avellar, G., Pereira, G., Pimenta, L., & Iscold, P. (2015). Multi-uav routing for area coverage and remote sensing with minimum time. *Sensors*, 15(11), 27783-27803.

Baker, B. M., & Ayechew, M. A. (2003). A genetic algorithm for the vehicle routing problem. *Computers & Operations Research*, 30(5), 787-800.

Balasingam, M. (2017). Drones in medicine—The rise of the machines. *International Journal of Clinical Practice*, 71(9), 1-4.

Balinski, M., & Quandt, R. (1964). On an integer program for a delivery problem. *Operations Research*, 12, 300-304.

Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1), 238-252.

Berbeglia, G., J.-F. C., & Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1), 8-15.

Borcinova, Z. (2017). Two models of the capacitated vehicle routing problem. *Croatian Operational Research Review*, 8(2), 463-469.Bouman, P., Agatz, N., & Schmidt, M. (2017). Dynamic programming approaches for the traveling salesman problem with drone. *ERIM Research Paper Series*.

Bouman, P., Agatz, N., & Schmidt, M. (2018). Dynamic programming approaches for the traveling salesman problem with drone. *Networks*, 72(4), 528-542

Boysen, N., Briskorn, D., Fedtke, S., & Schwerdfeger, S. (2018). Drone delivery from trucks: Drone scheduling for given truck routes. *Networks*, 72(4), 506-527.

Braekers, K., Ramaekers, K., & Nieuwenhuysec, I. V. (2014). The vehicle routing problem: state of the art classification and review. *Computers & Industrial Engineering*, 99, 300-313.

Bräysy, D. M. (2007). Active-guided evolution strategies for large-scale capacitated vehicle routing problems. *Computers & Operations Research*, 34(10), 2964-2975.

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1997). Applying the Ant System to the Vehicle Routing Problem. *2nd International Conference on Metaheuristics.* Sophia-Antipolis, France.

Campbell, A., Clarke, L., Kleywegt, A., & Savelsbergh, M. (1998). The Inventory Routing Problem. *Fleet Management and Logistics* (pp. 95-113). Boston, MA: Springer.

Caramia, M., & Guerriero, F. (2009). A heuristic approach for the truck and trailer routing problem. *Journal of the Operational Research Society* , 61(7), 1168-1180.

Carlsson, J. G., & Song, S. (2017). Coordinated Logistics with a Truck and a Drone. *Management Science*, 64(9), 3971-4470.

Chang, Y., & Lee, H. (2018). Optimal delivery routing with wider drone-delivery areas along a shorter truck route. *Expert Systems with Applications*, 104, 307-317

Chao, M. (2002). A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29(1), 33-51.

Choi, Y., & Schonfeld, P. M. (2017). Optimization of Multi-package Drone Deliveries Considering Battery Capacity. *In Proceedings of the 96th Annual Meeting of the Transportation Research Board*, (pp. 8-12). Washington, DC, USA.

Chowdhury, S., Emelogua, A., Marufuzzamana, M., Nurre, S. G., & Bian, L. (2017). Drones for disaster response and relief operations: A continuous approximation model. *International Journal of Production Economics*, 188, 167-184.

Christofides, N. (1976). The vehicle routing problem. *Revue française d'automatique, informatique, recherche opérationnelle. Recherche opérationnelle,*, 10(V1), 55-70.

Christofides, N., & Eilon, S. (1969). An Algorithm for the Vehicle-dispatching Problem. *Journal of the Operational Research Society*, 20(3), 309-318.

Christofides, N., Mingozzi, A., & Toth, P. (1981(a)). Exact algorithms for the vehicle routing problem, based on spanning tree and. *Mathematical Programming*, 20, 255-282.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4), 568-581.

Clarke, R., & Moses, L. B. (2014). The regulation of civilian drones' impacts on public safety. *Computer Law & Security Review*, 30(3), 263-285.

Clarke, R., & Moses, L. B. (2014). The regulation of civilian drones' impacts on public safety. *Computer Law & Security Review*, 30(3), 263-285.

Contardo, C., Hemmelmayr, V., & Crainic, T. G. (2012). Lower and upper bounds for the two-echelon capacitated location-routing problem. *Computers & Operations Research*, 39(12), 3185-3199.

Corana, A., Marchesi, M., Martini, C., & Ridella, S. (1987). Minimizing multimodal functions of continuous variables with the "simulated annealing" algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 13(3). 262-280.

CPLEX, I. (2009). Ibm ilog cplex optimizer. *International Business Machines Corporation*, 46(53), 157-158

Cuda, R., Guastaroba, G., & Speranza, M. (2015). A survey on two-echelon routing problems. *Computers & Operations Research*, 55, 185-199.

Cullen, F. H., Jarvis, J. J., & Ratliff, H. D. (1981). Set partitioning based heuristics for interactive routing. *NETWORKS*, 11(2), 125-143.

D'Andrea, R. (2014). Guest editorial can drones deliver? *IEEE Transactions on Automation Science and Engineering*, 11(3), 647-648.

Dantzig, G., & Ramser, J. (1959). The truck dispatching problem. *Management science*, 80-91.

Dantzig, G., Fulkerson, R., & Johnson, S. (1954). Solution of a Large-Scale Traveling-Salesman Problem. *Operation Research*, 393-410.

de Freitas, J. C., & Penna, P. H. (2020). A variable neighborhood search for flying sidekick traveling salesman problem. *International Transactions in Operational Research*, 27(1), 267-290.

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

Dorling, K., Heinrichs, J., & Messier, G. G. (2016). Vehicle routing problems for drone delivery. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS*, 47(1), 70-85.

Dorr, L. (2018, 07 23). *Fact Sheet – Small Unmanned Aircraft Regulations (Part 107)*. Retrieved from Federal Aviation Administration : https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=22615.

Eilon, S., Watson-Gandy, C.D.T., & Christofides, N. (1971). Distribution Management . *Mathematical Modelling and Practical.* Griffin, London.

Eksioglu, B., Vuralb, A. V., & Reismanc, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, 57(4), 1472-1483.

EL Lawler, J. L. (1985). *The traveling salesman problem: a guided tour of combinatorial optimization* . Wiley, New York.

El-Adle, A., Ghoniem, A., & Haouari, M. (2019). Parcel delivery by vehicle and drone. *Journal of the Operational Research Society*, 1-19.

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

Fargeas, J. L., Kabamba, P., & Girard, A. (2015). Cooperative surveillance and pursuit using unmanned aerial vehicles and unattended ground sensors. *Sensors*, 15(1), 1365-1388.

Ferrandez Mourelo, S., Harbison, T., Webwer, T., Sturges, R., & Rich, R. (2016). Optimization of a truck-drone in tandem delivery network using K-means and genetic algorithm. *Journal of Industrial Engineering and Management*, 9(2), 374-388.

Fisher, M. L., & Jaikumar, R. (1978). An Algorithm for the Space-Shuttle Scheduling Problem. *Operations Research*, 26(1), 3-208.

Floreano, D., & Wood, R. J. (2015). Science, technology and the future of small autonomous drones. *Nature*, 521(7553), 460-466.

Flynt, J. (2017, 11 29). *How much weight can a drone carry?* Retrieved from 3DINSIDER: http://3dinsider.com/drone-payload/

Francis, P., Smilowitz, K., & Tzur, M. (2007). Flexibility and complexity in periodic distribution problems. *Naval Research Logistics*, 54, 136-150.

Frontera, G., Martín, D., Besada, J., & Gu, D. (2017). Approximate 3D Euclidean Shortest Paths for Unmanned Aircraft in Urban Environments. *J. Intell. Robot. Syst.*, 85, 353–368

Fukasawa, R. L., H., L., J., d. A., M. P., R., M., U. E., & Werneck, R. F. (2006). Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical programming*, 106(3), 491-511.

Gerdessen, J. C. (1996). Vehicle routing problem with trailers. *European Journal of Operational Research*, 135-147.

Getzin, S., Wiegand, K., & Schöning, I. (2012). Assessing biodiversity in forests using very high-resolution images and unmanned aerial vehicles. *Methods in Ecology and Evolution*, 3(2), 397-404.

GillettLel, B. E., & Miller, L. R. (1974). A Heuristic Algorithm For the Vehicle Dispatching Problem. *Operations research*, 22(2), 340-349.

Glover, F. (1986). Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research*, 13 (5): 533–549.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning.* New York: Inc.

Golden, B., Raghavan, S., & Wasil, E. (2008). *The vehicle routing problem: latest advances and new challenges.* Operations Research/Computer Science Interface Series, vol 43. Springer.

Govindan, K., Jafarian, A., Khodaverdi, R., & Devika, K. (2014). Two-echelon multiple-vehicle location–routing problem with time windows for optimization of sustainable supply chain network of perishable food. *International Journal of Production Economics*, 152, 9-28.

Grochlsky, B., Keller, J., Kumar, V., & G. P. (2006). Cooperative air and ground surveillance. *IEEE Robotics & Automation Magazine*, 13(3), 1070-9932.

Groër, C., Golden, B., & Wasil, E. (2008). The Consistent Vehicle Routing Problem. *Manufacturing & Service Operations Management*, 11(4), 630-643.

Ha, Q. M., Deville, Y., Pham, D. q., & Hà, M. H. (2015). *Heuristic methods for the Traveling Salesman Problem with Drone.* Comput.Sci.

Ha, Q. M., Deville, Y., Pham, Q. D., & Hà, M. H. (2018). On the min-cost Traveling Salesman Problem with Drone. *Transportation Research Part C: Emerging Technologies*, 86, 597-621.

Ham, A. M. (2018). Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming. *Transportation Research Part C: Emerging Technologies*, 91, 1-14.

Hiermann, G., Puchinger, J., Ropke, S., & Hart, R. F. (2016). The electric fleet size and mix vehicle routing problem with time windows and recharging stations. *European Journal of Operational Research*, 252(3), 995-1018.

Hsu, T. (2017, 1 4). *Mercedes-Benz to Show Drone Mothership Vision Van at CES*. Retrieved from Trucks: https://www.trucks.com/2017/01/04/mercedes-benz-drone-vision-van-ces/

Jacobsen, S., & Madsen, O. (1980). A comparative study of heuristics for a two-level routing-location problem. *European Journal of Operational Research*, 5(6), 378-387.

Jansen, B. (2017, march 8). *NYC firefighters use drone to help battle blaze for first time*. Retrieved from USA TODAY: https://www.usatoday.com/story/news/2017/03/08/drone-firefighters/98848038/.

Jeong, H. Y., Lee, S., & Song, B. D. (2019). Truck-drone hybrid delivery routing: payload-energy dependency and no-fly zones. *International Journal of Production Economics*, 214, 220-233

Jones, T. (2017). *OBJECTIVE ANALYSIS EFFECTIVE SOLUTIONS*. Retrieved January 22, 2020 from International Commercial Drone Regulation and Drone Delivery Services: https://www.rand.org/pubs/research_reports/RR1718z3.html

Karak, A., & Abdelghany, K. (2019). The hybrid vehicle-drone routing problem for pick-up and delivery services. *Transportation Research Part C: Emerging Technologies*, 102, 427-449.

Kim, S., & Moon, I. (2019). Traveling salesman problem with a drone station. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(1), 42-52

Kim, S. J., Lim, G. J., Cho, J., & Côté, M. J. (2017). Drone-aided healthcare services for patients with chronic diseases in rural areas. *Intelligent & Robotic Systems*, 88(1), 163-180.

Kitjacharoenchai, P., Ventresca, M., Moshref-Javadi, M., Lee, S., Tanchoco, J. M., & Brunese, P. A. (2019). Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. Computers & Industrial Engineering, 129, 14-30

Kunze, O. (2016). Replicators, ground drones and crowd logistics a vision of urban logistics in the year 2030. *Transportation Research Procedia*, 19, 286-299.

Laporte, G., Mercure, H., & and Nobert, Y. (1986). An exact algorithm for the asymmetrical capacitated vehicle routing problem. *Networks*, 16, 33-46.

Lei, S., Hua, Y., Xufeng, Y., Cong, M., & Jun, H. (2015). A study of instability in a miniature flying-wing aircraft in high-speed taxi. *Chinese Journal of Aeronautics*, 28(3), 749-756.

Lenstra, K. J., & Kan, R. (1981). Complexity of Vehicle Routing and Scheduling Problem. *Networks*, 11(2), 221-227.

Li, Y., Zhang, G., Pang, Z., & Li, L. (2018). Continuum approximation models for joint delivery systems using trucks and drones. *Enterprise Information Systems*, 1-30.

Lin, C. (2011). A vehicle routing problem with pickup and delivery time windows, and coordination of transportable resources. *Computers & Operations Research*, 1596-1609.

Lin, J.-R., & Lei, H.-C. (2009). Distribution systems design with two-level routing considerations. *Annals of Operations Research*, 172(1), 329-347.

Lin, S. W., Vincent F.Yu, & Chou, S.-Y. (2009). Solving the truck and trailer routing problem based on a simulated annealing heuristic. *Computers & Operations Research*, 36(5), 1683-1692.

Lin, S.-W., Yub, V. F., & Chou, S.-Y. (2010). A note on the truck and trailer routing problem. *Expert Systems with Applications*, 37(1), 899-903.

Lin, S.-W., Yub, V. F., & Luc, C.-C. (2011). A simulated annealing heuristic for the truck and trailer routing problem with time windows. *Expert Systems with Applications*, 38(12), 15244-15252.

Locklear, M. (2017, 10 25). *engadget*. Retrieved from Trump announces program to test drones beyond FAA regulations: https://www.engadget.com/2017/10/25/trump-program-test-drones-beyond-faa-regulations/

Lockridge, D. (2017, 04). *The Big and Small of Last Mile Delivery*. Retrieved from Fleet Management: http://www.truckinginfo.com/channel/fleet-management/article/story/2017/04/the-big-and-small-of-the-last-mile-delivery.aspx

Luo, Z., Liu, Z., & Shi, J. (2017). A Two-Echelon Cooperated Routing Problem for a Ground Vehicle and Its Carried Unmanned Aerial Vehicle. *Sensors*, 17(5), 1144.

Madsen, O. B. (1983). Methods for solving combined two level location-routing problems of realistic dimensions. *European Journal of Operational Research*, 12(3), 295-301.

Marinelli, M., Caggiani, L., Ottomanelli, M., & Dell'Orco, M. (2017). En route truck–drone parcel delivery for optimal vehicle routing strategies. *IET Intelligent Transportation System* , 12(4), 253-261.

Mathew, N., Smith, S. L., & Waslander, S. L. (2015). Planning paths for package delivery in heterogeneous multirobot teams. *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING*, 12(4), 1298-1308.

McFarland, M. (2016, 09 7). *This Mercedes-Benz van will carry a fleet of delivery robots*. Retrieved from CNN tech: http://money.cnn.com/2016/09/07/technology/starship-robot-mercedes-benz/index.html

McFarland, M. (2017, 02 21). *CNN Business*. Retrieved from UPS drivers may tag team deliveries with drones: https://money.cnn.com/2017/02/21/technology/ups-drone-delivery/index.html

McFarland, M. (2017, 02 21). *UPS drivers may tag team deliveries with drones*. Retrieved from CNN Business: https://money.cnn.com/2017/02/21/technology/ups-drone-delivery/index.html

Menouar, H., Guvenc, I., Akkaya, K., Uluagac, A. S., Kadri, A., & Tuncer, A. (2017). UAV-enabled intelligent transportation systems for the smart city: applications and challenges. *IEEE Communications Magazine*, 55(3), 22-28.

Merwaday, A., & Guvenc, I. (2015). UAV assisted heterogeneous networks for public safety communications. *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)* (pp. 329-334). New Orleans, LA, USA: IEEE.

Moshref-Javadi, M., & Lee, S. (2017). Using Drones to Minimize Latency in Distribution Systems . *IIE Annual Conference* (pp. 235-240). Institute of industrial and system engineers .

Murray, C., & Raj, R. (2020). The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. *Transportation Research Part C: Emerging Technologies*, 110, 368-398.

Murray, C., & Chu, A. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Part C: Emerging Technologies*, 54, 86-109.

Nguyen, V. P., Prins, C., & Prodhon, C. (2012). Solving the two-echelon location routing problem by a GRASP reinforced by a learning process and path relinking. *European Journal of Operational Research*, 216(1), 113-126.

Nguyen, V.-P., Prins, C., & Prodhon, C. (2012). A multi-start iterated local search with tabu list and path relinking for the two-echelon location-routing problem. *Engineering Applications of Artificial Intelligence*, 25(1), 56-71.

Paessens, H. (1988). The savings algorithm for the vehicle routing problem. *European Journal of Operational Research*, 34(3), 336-344.

Paust, J. J. (2010). Self-defense targetings of non-state actors and permissibility of U.S. use of drones in Pakistan. *Transnational Law & Policy*, 19(2), 237-282.

Petersen, R. (2016, 7 29). *Rethinking Last-Mile Logistics: Deploying Swarms of Drones with Self-Driving Trucks*. Retrieved from FLEXPORT: https://www.flexport.com/blog/deploying-drones-self-driving-trucks/

Poikonen, S., G. B., & Wasil, E. (2019). A branch-and-bound approach to the traveling salesman problem with a drone. *INFORMS Journal on Computing*, 31(2), 335-346.

Poikonen, S. &. (2020). Multi-visit drone routing problem. *Computers & Operations Research*, 113, 104802.

Poikonen, S., Wang, X., & Golden, B. (2017). The vehicle routing problem with drones: Extended models and connections. *Networks*, 70(1), 34-43.

Prins, C., Prodhon, C., & Calvo, R. W. (2006). Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *4OR - A Quarterly Journal of Operations Research*, 4(3), 221-238.

Prodhon, C., & Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1), 1-17.

Pugliese, L. D., & Guerriero, F. (2017). Last-Mile Deliveries by Using Drones and Classical Vehicles. *International Conference on Optimization and Decision Science* (pp. 557-565). Cham: Springer.

PYMNTS. (2016, 06 12). *MERCHANT INNOVATIONIs Drone Delivery Economically Feasible, Practical?* Retrieved from PYMNTS: https://www.pymnts.com/news/merchant-innovation/2016/economics-drone-delivery/

Rash, W. (2017, 2 22). *UPS Tests Delivery Trucks Equipped With Drones as Efficiency Booster*. Retrieved from eWeek: http://www.eweek.com/mobile/ups-tests-delivery-trucks-equipped-with-drones-as-efficiency-booster

Reimann, M., Doerner, K., & Hartl, R. F. (2004). D-Ants: Savings Based Ants divide and conquer the vehicle routing problem. *Computers & Operations Research*, 31(4), 563-591.

Ropke, S., & Pisinger, D. (2006). An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transportation Science*, 40(4), 393-546.

Rose, C. (2013, 12 01). *Amazon's Jeff Bezos looks to the future*. Retrieved from CBSN: https://www.cbsnews.com/news/amazons-jeff-bezos-looks-to-the-future/

San, K. T., Lee, E. Y., & Chang, Y. S. (2016). The delivery assignment solution for swarms of UAVs dealing with multi-dimensional chromosome representation of genetic algorithm. *IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)* (pp. 1-7). New York, NY, USA: IEEE.

Savuran, H., & Karakaya, M. (2016). Efficient route planning for an unmanned air vehicle deployed on a moving carrier. *Soft Computing*, 20(7), 2905-2920.

Scheuerer, S. (2006). A tabu search heuristic for the truck and trailer routing problem. *Computers & Operations Research*, 33(4), 894-909.

Schermer, D., Moeini, M., & Wendt, O. (2019). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies*, 106, 166-204.

Schneider, M., Stenger, A., & Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, 48(4), 500-520.

Schneiderman, R. (2012). Unmanned drones are flying high in the military/aerospace sector [special reports]. *IEEE Signal Processing Magazine*, 29(1), 8-11.

Shaw, P. (1998). Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. *Proceedings of the Fourth International Conference on Principles and Practice of Constraint Programming* (pp. 417-431). London, UK : Springer-Verlag.

Shetty, V., Sudit, M., & Nagi, R. (2008). Priority-based assignment and routing of a fleet of unmanned combat aerial. *Computers & Operations Research*, *35*, 1813–1828.

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations research*, 35(2), 254-265.

Sundar, K., & Depots, S. R. (2014). Algorithms for Routing an Unmanned Aerial Vehicle in the. *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING*, 11(1), 287- 294.

Thiels, C., Aho, J., Zietlow, S., & Jenkins, D. (2015). Use of unmanned aerial vehicles for medical product transport. *Air Medical Journal*, 34(2), 104-108.

Toth, P., & Vigo, D. (2002). Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics*, 123(1-3), 487-512.

Troudi, A., Addouche, S.-A., Dellagi, S., & Mhamed, A. E. (2017). Logistics support approach for drone delivery fleet. *International Conference on Smart Cities* (pp. 86-96). Malaga, Spain: Springer, Cham.

Tu, P., Dat, N., & Dung, P. (2018). Traveling salesman problem with multiple drones. *In Proceedings of the Ninth International Symposium on Information and Communication Technology*, 46-53. ACM.

Vanian, J. (2016, 07 22). 7-Eleven Just Used a Drone to Deliver a Chicken Sandwich and Slurpees. Retrieved January 24, 2020 from Tech.Drone: https://fortune.com/2016/07/22/7-eleven-drone-flirtey-slurpee/

Vattapparamban, E., Güvenç, İ., Yurekli, A. İ., Akkaya, K., & Uluağaç, S. (2016). Drones for smart cities: Issues in cybersecurity, privacy, and public safety. *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)* (pp. 216-221). Paphos, Cyprus: IEEE.

Villegas, J. G., Prins, C., Prodhon, C., L.Medaglia, A., & Velascod, N. (2013). A matheuristic for the truck and trailer routing problem. *European Journal of Operational Research*, 230(2), 231-244.

Villegas, J. G., Prins, C., Prodhon, C., Medaglia, A. L., & Velasco, N. (2010). GRASP/VND and multi-start evolutionary local search for the single truck and trailer routing problem with satellite depots. *Engineering Applications of Artificial Intelligence*, 23(5), 780-794.

Wang, X., Poikonen, S., & Golden, B. (2017). The vehicle routing problem with drones: several worst-case results. *Optimization Letters*, 11(4), 679-697.

Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation research part B: methodological*, 122, 350-364.

Wark, P., & Holt, J. (1994). A Repeated Matching Heuristic for the Vehicle Routeing Problem. *Journal of the Operational Research Society*, 45(10), 1156-1167.

Xiao, Y., Z. Q., Kaku, I., & Xu, Y. (2012). evelopment of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, 39(7), 1419-1431.

Yurek, E., & Ozmutlu, H. (2018). A decomposition-based iterative optimization algorithm for traveling salesman problem with drone. *Transportation Research Part C*, 91, 249-262.