

# A heterogeneous vehicle routing problem with drones and multiple depots

Panagiotis Zachos

June 2024

## 1 INTRODUCTION

## 2 Related Work

*Introductory statement on why routing problems (TSP, VRP) are fundamental in the field of operations research and logistics, why they receive much attention from the academic community, in what practical applications are they used in*

### 2.1 The Traveling Salesman Problem with UAVs

The Traveling Salesman Problem (TSP) is a classic, widely researched optimization problem where the objective is to determine the shortest possible route that allows a salesman, starting from an origin city, to visit a set of cities exactly once and then end the tour by returning to the origin city. The TSP is known for its computational complexity, being an NP-hard problem, which means that the time required to solve it increases exponentially with the number of cities. **(Make the point that a truck with unlimited capacity can take on the role of the salesman (refer to Dell’Amico 2021))** Murray and Chu were the first researchers who formalized and extensively studied the concept of UAVs working alongside trucks in routing problems in 2015. In their seminal paper {ADD REFERENCE} the authors introduced the Flying Sidekick Traveling Salesman Problem (FSTSP) and the Parallel Drone Scheduling Traveling Salesman Problem (PDSTSP). The two problems are similar in their use of a single truck alongside one or more UAVs but there are key differences between the two.

#### 2.1.1 The Flying Sidekick Traveling Salesman Problem

The FSTSP involves a truck which takes on the role of the salesman, and is equipped with a single UAV (drone) which the driver can utilize by dispatching it to serve drone-eligible customers. The objective is to minimize the total time in which all customers are serviced. **{This is general and concerns every problem so it could be placed in the introduction (2.1)}** The underlying principle is that by making use of a lower-cost vehicle which is also not bound by road conditions (drone), the total time of the operation can be reduced substantially, leading to increased efficiency and thus profitability for the distributor but also increased customer satisfaction and reduced environmental footprint. By adding a drone to the operation, the distance that the truck needs to travel is reduced, leading to overall cost reduction of the operation (fuel consumption, working hours) while also limiting CO2 emissions.}} In the FSTSP, a truck and a drone are called to begin from a depot, although not necessarily in tandem, visit a set of customers exactly once, and then end their

tour by returning to the depot. The driver is considered to load the UAV with the parcel of the intended customer inbetween traveling i.e when the truck has stopped at a customer location to deliver a parcel. Then, the driver dispatches the UAV and immediately continues with the truck onto the next customer(s), thus creating a time window in which both the truck and UAV are traveling simultaneously. After the UAV has delivered its parcel, it must return to the truck to be retrieved by the driver, or if flight endurance limitations allow, it can return to the depot, thus ending its tour. The retrieval of the UAV by the truck must be performed at a customer location assigned to the truck, which must be different from the launch location i.e the truck cannot remain stationary after launching the drone. The vehicle which arrives first at the determined retrieval node has to wait for the other vehicle, therefore synchronization between the truck and drone is required. Furthermore, the drone is associated with a launch setup and a retrieval time cost which must be considered in the synchronization and a maximum flight endurance value (battery capacity) which must be respected when assigning customers to the drone.

The authors used MILP to model the problem and noted that, deriving from the NP-hard nature of the problem, MILP solvers in certain cases required several hours even for instances with as low as 10 customers. Consequently, they proposed a *route and re-assign* heuristic. The heuristic begins by solving a TSP that assigns the truck to visit all customers. This TSP subproblem was solved using various techniques, namely *Integer Programming (IP)*, and the *Clarke-Wright Savings*, *Nearest Neighbour*, *Sweep* heuristics. Once the truck route is established, the heuristic attempts to improve the solution by re-assigning customers to the drone. The algorithm iterates through each pair of consecutive stops in the truck route, and for each pair, it considers the possibility of sending the drone from the first stop to the second stop. It calculates the savings in travel time that would result from making this change, while considering the drone’s and truck’s travel time as well as the time required to retrieve the drone. If the resulting savings are greater than zero, the customer at the second stop is reassigned to the drone and the truck route is updated accordingly. The heuristic continues to iterate through all pairs of stops in the truck route, re-assigning customers to the drone whenever possible, until no further improvements can be made.

The authors’ findings show that this *route and re-assign* heuristic provides a framework for solving the FSTSP in a reasonable amount of time, while still achieving good solutions. Especially in the version where the initial truck route is solved with Integer Programming (IP), the proposed heuristic, on average, outperformed the MILP solver. Based on these results, the authors highlight the impact of the initial TSP tour towards the final solution and how effective TSP solution approaches show promise in positively impacting the performance of their algorithm. Regarding the

remaining approaches, the use of the *Clarke-Wright Savings* heuristic yielded good results with minimal computation time whereas the use of the *Nearest Neighbour* and *Sweep* heuristics did not provide competitive results.

Ha et. al (2015) researched the FSTSP, although under a different name; the "Traveling Salesman Problem with Drone". They proposed two heuristics using two opposite concepts; (i) *cluster first - route second* and (ii) *route first - cluster second*. The clustering step consists of determining the set of drone routes that should be in the final solution. To solve this, the authors propose a Mixed Integer Linear Programming (MILP) model where the objective function is to maximize the profit generated by selecting specific drone routes. In the first approach, the MILP model is initially solved, to identify the optimal set of drone routes. Once those are determined, the heuristic then constructs the truck's route, using the Concorde TSP solver, by integrating the drone's routes with the remaining customer locations, producing the final solution. In the second *route first - cluster second* approach, the truck route is first solved (Concorde), visiting all customers. The truck's route is then used as the basis to identify potential drone routes. The MILP model used in the clustering step is adapted to include constraints that ensure that any drone route selected is part of the truck's predetermined tour and that the nodes that form the drone *sortie* are arranged in the same order as they appear in the truck's route. **Results** Experiments compared the two heuristics to the FSTSP model and the standard TSP without drones, using 2 newly generated datasets with 10 and 100 customers. The results show that the proposed heuristics are competitive compared to the FSTSP model, especially as customer size increases and indicate that drones servicing more distant customers from the TSP route is more beneficial.

Freitas & Penna (2018) {Add reference} extended the FSTSP research and employed a hybrid heuristic termed *Randomized Variable Neighborhood Descent Framework* (RVNDF) to solve the problem. This heuristic makes use of a three-step approach. Similar to the previous heuristic by Murray & Chu, the RVNDF begins by generating the optimal solution for the TSP where all of the customers are serviced by the truck only. To achieve this, they use the widely known in the field Concorde TSP solver {ADD REFERENCE}. The second phase consists of creating an initial solution which makes use of the drone, by assigning it to random drone-eligible customers while respecting the drone's flight endurance. Finally, in the third step, the *Randomized Variable Neighborhoods Descent* (RVND) algorithm is called, which is the core of the heuristic and as the name implies, a randomized variant of the VND {ADD REFERENCE} algorithm, where the order of the neighborhood visits is not predetermined. RVND aims to improve the initial solution by systematically exploring different neighboring solutions. It does this by applying five different "neighborhoods" or local search operations, each designed to modify the current solution in a specific way. These neighborhoods consist of the following 5 operations: (i) *Reinsertion*: Moving a truck customer to a different position on its route, (ii) *Exchange*: Swap the positions of two truck customers, (iii) *Exchange(2,1)*: Swap the positions of two consecutive customers and an other one, (iv) *2-Opt*: Remove two non-adjacent arcs from the truck's route and insert two new arcs such as to create a new route, (v) *RelocateCustomer*: Transfer a customer from the truck's route to the drone's route. The RVNDF evaluates all possible moves within each neighborhood, ensuring

that the drone's flight range and customer accessibility constraints are respected. After each operation, the resulting solution is either accepted and used to update the current solution if the overall delivery time improves, or discarded otherwise. **Report results or not?** {To evaluate their algorithm's quality, they solved 11 instances ranging from 50 to 99 customers, running the RVNDF ten times for each one. The authors then compared the FSTSP solutions to the TSP solution where only a truck is used. Results varied, improving by as high as 19.28% on a certain instance, while not being able to improve in the case where the distances between customers exceed the drone's flight endurance.}

Dell'Amico et al. (2019) {ADD REF} proposed a modified formulation of Murray & Chu's FSTSP, aiming to improve it and achieve a more realistic interpretation of the problem. They considered two different versions, one where the drone is allowed to land and wait at customers to conserve battery and another where waiting is only allowed in flight. The authors also introduce a three-indexed formulation termed *DMN* and a two-indexed formulation *DMN2*. The objective function aims to minimize the overall completion time, accounting for the truck's travel time, drone launch and return times, and waiting time at customers (depending on the version).

The *DMN* formulation leverages three-indexed boolean variables to represent *sorties*; flights undertaken by the drone. This formulation includes various improvements which are focused on eliminating infeasible drone sorties. To achieve this, a new objective function and a set of inequalities are presented. The two-indexed formulation, *DMN2*, simplifies the representation of sorties by using two-indexed arc variables. These variables represent the drone's entry and exit points from customer nodes. This approach significantly reduces the number of variables needed, leading to a more efficient and faster solution process. *DMN2* is built upon the *DMN* formulation, adapting all constraints involving the three-indexed variables to use the new two-indexed variables. The objective function in *DMN2* is also modified to reflect the use of two-indexed variables.

The authors conducted comprehensive computational experiments to evaluate the performance of their proposed formulations -  $\bar{MC}$ , *DMN* and *DMN2*. While the modified Murray & Chu ( $\bar{MC}$ ) model could be solved directly by CPLEX, the *DMN*, *DMN2* models required a branch-and-cut implementation due to their exponentially large number of constraints. The authors tested their methods on the benchmark instances provided by Murray & Chu, for both the "wait" and "no-wait" versions of the problem, and with varying characteristics, including different drone endurance levels, depot locations, and speeds and compared their results to those of existing methods. The results showed that *DMN2*, the two-indexed formulation, significantly outperformed previous methods, solving 59 out of 72 benchmark instances to optimality, with an average percentage gap between the best lower and upper bound of 1.70% and an average computing time of less than 20 minutes, demonstrating the effectiveness of their newly proposed formulations in solving the FSTSP.

Dell'Amico et al. {ADD REFERENCE} also then proposed two algorithms to solve the FSTSP; a Branch and Bound (BB) algorithm for small instances and a heuristic based on the former for larger instances. The BB algorithm utilizes the concept of "missions," which can be either truck movements or phases where

the truck and drone operate in parallel. It works by incrementally expanding partial solutions by adding new missions at each level of the search tree until the final depot is reached. Customers are assigned to drones only in the final stage by solving an Assignment Problem, a strategy aimed at limiting the search tree size. The heuristic, heavily based on BB, starts with an initial feasible TSP solution and repeatedly applies the BB method as a subroutine to improve the solution. It is designed to iteratively refine the solution by adding drone services to optimize the overall delivery time. The authors report that the BB algorithm performs well on small instances, showing competitive computation times compared to other approaches in the literature. The heuristic algorithm also demonstrates promising results on medium and large instances, providing state-of-the-art solutions.

Boccia et al. (2021) proposed new integer linear programming formulations for both the FSTSP and PDSTSP and a column-and-row generation approach to handle synchronization issues between the truck and drone solving to optimality and improving upon instances with up to 20 customers. Kuroswiski et al. (2023) presented a hybrid Genetic Algorithm combined with Mixed Integer Linear Programming (MILP) to efficiently solve the FSTSP for up to ten customers. Pilcher (2023) proposed a self-adaptive genetic algorithm and introduced a novel two-stage mutation process designed to address the complexities of the FSTSP.

### 2.1.2 The Parallel Drone Scheduling Traveling Salesman Problem

The PDSTSP, extends the concept of the FSTSP by involving multiple drones and optimizing the delivery process through parallel drone operations. In contrast to the FSTSP, the drones act independently from the truck and launch from and return to the depot. It was designed for scenarios where a significant portion of the customers are located within the UAV's flight range from the depot. The authors proposed a MILP formulation and employed heuristic algorithms to solve larger instances of the PDSTSP. Mbiadou Saleu et al. (2018) proposed an iterative two-step heuristic along with a MILP formulation. Nguyen et al. (2022) proposed an efficient branch-and-cut algorithm to solve the problem. The authors broke new ground by achieving optimal solutions for instances up to 783 customers for the first time.

Nguyen et al. (2023) introduced a new variant of the parallel drone scheduling traveling salesman problem that aims to increase the utilization of drones, particularly for heavy item deliveries, which they termed the PDSTSP with collective drones (PDSTSP-c). The authors developed a two-index MILP formulation to solve small-size instances to optimality and proposed a ruin-and-recreate metaheuristic to efficiently handle larger problem instances. Montemanni et al. (2024) expanded on the research for the PDSTSP-c and introduced a MILP and constraint programming approach.

Constraint programming models have been effectively utilized to handle the scheduling complexities of drone-assisted TSP variants. Ham (2018) extended the PDSTSP by considering drone drops and pickups and time-window constraints for which the author proposed a constraint programming approach to handle the problem's complexity. Montemanni and Dell'Amico (2023) presented a

constraint programming model for the PDSTSP, achieving optimal solutions for various instances.

Hybrid optimization techniques have also been explored to solve drone-assisted TSP variants. Dinh et al. (2021) proposed a hybrid ant colony optimization metaheuristic for the PDSTSP, achieving several new best-known solutions.

### 2.1.3 TSP-D

Agatz et al. (2018) introduced the Travelling Salesman Problem with Drones (TSP-D). While the TSP-D shares mostly the same characteristics as the FSTSP, the key distinction between the TSP-D and the FSTSP is that the latter requires the distinctive location between launch point and recovery point of the drone, while the former allows the drone to be launched and recovered by the truck in the same location. The authors model the problem and propose an IP model which solves instances with up to 12 customers to optimality, along with several fast heuristics to solve larger instances of the problem. Yurek et al. (2018) contributed to the TSP-D research by proposing a decomposition-based iterative optimization algorithm and verified its efficiency. Ha et al. (2018) introduced and solved the min-cost TSP-D, whose objective is to minimize the total operational costs, which include transportation costs and waiting penalties. The authors proposed a MILP formulation and two heuristic methods to solve the problem.

### 2.1.4 Other

Schermer et al. (2019) introduced the Traveling Salesman Drone Station Location Problem (TSDSLP), combining TSP, facility location, and parallel machine scheduling problems to minimize operational costs. Kim & Moon (2019) introduced a mixed integer programming model for the TSP with a drone station (TSP-DS), which extends the operational range of drones.

Numerical results indicate that optimizing battery weight and reusing drones are important considerations for drone delivery (Dorling 2016).

**We thank Dr. Eddie Cheng for asking about faster drones versus more drones.**

At the same time, it is important that we address flight range limitations. Such limitations have important roles in much of the existing drone routing research. However, even the first drones developed by Amazon and JD.com had a flight range of 15 to 20 miles [33, 56]. Such a range is suitable to allow out and back travel in the medium-sized cities in which the authors live, Braunschweig, Germany, and Iowa City, United States. In fact, it is suitable for many larger cities such as Hamburg, Munich and Paris. Thus, in contrast to other drone applications, in this work, we do not consider flight range as a limiting factor. (Sacramento, 2019)

## 2.2 The Vehicle Routing Problem

While the TSP is concerned with finding a single optimal tour for one vehicle, the Vehicle Routing Problem (VRP), generalizes this concept by involving multiple vehicles and additional constraints, making it more applicable to real-world logistics scenarios.

Therefore, the VRP extends the TSP by considering multiple vehicles, each with a specific capacity, to service a set of customers with known demands. The goal is to determine the optimal set of routes for a fleet of vehicles such as to deliver goods to or to collect goods from customers while minimizing the total cost, which is often determined as the vehicles' travel distance, time, fuel consumption, pollution or a combination of those.

### **2.2.1 VRP Variants**

The VRP can also include various constraints like vehicle capacities, delivery time windows, and the requirement for all vehicles to start and end their routes at a central depot.

### **2.2.2 Drone-related VRP Research**

Wang et al. [12] introduced the Vehicle Routing Problem with Drones (VRPD), where a fleet of trucks, each truck equipped with a given number of drones, delivers packages to customers. According to the classification of Toth and Vigo, the VRPD can be classified as a variant of the Distance-Constrained Capacitated Vehicle Routing Problem (DCVRP) with a set of heterogeneous vehicles.

Among the several features considered, the drone endurance is the one that has the stronger influence on the convergence of the algorithm, a smaller endurance allows the algorithms to have better performances, while reducing the number of feasible sorties.

A problem very similar to the FS-TSP is the TSP with drones (TSP-D) defined for the first time in Agatz et al. (2018). The only difference between the TSP-D and the FS-TSP consists in the fact that in the first the truck can visit the same customer more than once in order to recover the drone, which, in turn, may be launched and return to the same location (Boccia et al. 2021)

## **3 The Novel Problem**