The traveling salesman problem with drones: The benefits of retraversing the arcs

Morandi, Nicola; Leus, Roel; Matuschke, Jannik; Yaman, Hande

Published in:
Transportation Science

DOI:
10.1287/trsc.2022.0230

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Transportation Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

The Traveling Salesman Problem with Drones: The Benefits of Retraversing the Arcs

Nicola Morandi, Roel Leus, Jannik Matuschke, Hande Yaman

TRANSPORTATION SCIENCE
Articles in Advance, pp. 1–19
ISSN 0041-1655 (print), ISSN 1526-5447 (online)
https://pubsonline.informs.org/journal/trsc

The Traveling Salesman Problem with Drones: The Benefits of Retraversing the Arcs

Nicola Morandi,[a,*] Roel Leus,[a] Jannik Matuschke,[b] Hande Yaman[a]

[a] Research Centre for Operations Research and Statistics, KU Leuven, 3000 Leuven, Belgium; [b] Research Centre for Operations Management, KU Leuven, 3000 Leuven, Belgium

*Corresponding author

Contact: nicola.morandi@kuleuven.be, https://orcid.org/0000-0001-8947-9211 (NM); Roel.Leus@kuleuven.be, https://orcid.org/0000-0002-9215-3914 (RL); jannik.matuschke@kuleuven.be, https://orcid.org/0000-0002-7463-3279 (JM); hande.yaman@kuleuven.be, https://orcid.org/0000-0002-3392-1127 (HY)

Received: July 7, 2022
Revised: February 15, 2023; April 22, 2023
Accepted: April 27, 2023
Published Online in Articles in Advance: July 11, 2023

https://doi.org/10.1287/trsc.2022.0230

Abstract. In the traveling salesman problem with drones (TSP-mD), a truck and multiple drones cooperate to serve customers in the minimum amount of time. The drones are launched and retrieved by the truck at customer locations, and each of their flights must not consume more energy than allowed by their batteries. Most problem settings in the literature restrict the feasible truck routes to cycles (i.e., closed paths), which never visit a node more than once. *Revisiting* a node, however, may lower the time required to serve all the customers. Additionally, we observe that optimal solutions for the TSP-mD may *retraverse* arcs (i.e., optimal truck routes may contain the same arcs multiple times). We refer to such solutions as *arc retraversing* and include them in our solution space by modeling the truck route as a closed walk. We describe Euclidean instances where all the optimal solutions are arc retraversing. The necessity of arc retraversals does not seem to have been investigated in previous studies, and those that allow node revisits seem to assume that there always exists an optimal solution without arc retraversals. We prove that under certain conditions, which are commonly met in the literature, this assumption is correct. When these conditions are not met, however, excluding arc-retraversing solutions might result in an increase of the optimal value; we identify cases where a priori and a posteriori upper bounds hold on such increase. Finally, we prove that there is no polynomial-time heuristic that can approximate the metric TSP-mD within a constant factor, unless $P = NP$. We identify a (nonconstant) approximation factor explicitly when the truck can visit all the nodes.

Keywords: drone-aided routing • TSP with drone • arc retraversing • approximability • synchronization

1. Introduction

Since the pioneering introduction of the flying sidekick traveling salesman problem (FSTSP) by Murray and Chu (2015), the scientific literature about applications of drones to routing and parcel delivery has grown at a remarkable pace. At the time of writing, searching the words "truck drone routing" by Google Scholar produced 19,200 results, 20% of which from just 2021. An impressive number of surveys on the topic have already appeared (e.g., those by Otto et al. 2018 and Macrina, Pugliese, and Guerriero 2020).The interest in drones' applications also comes from public institutions and the private sector. The European Commission forecasts more than 1,00,000 people employed and an economic impact of over 10 billion euros per year in the European drone sector by 2035 (European Council 2019). At the same time, the e-commerce multinational Amazon obtained approval from the relevant U.S. authority for its Prime Air service "beyond visual line of sight" (CNBC News 2020). From the algorithmic point of view, the FSTSP and its generalizations can model a wide range of routing applications with cooperating vehicles, which are not necessarily drones; in principle, any vehicle with limited fuel or traveling person with limited payload capacity could play the role of the drone in these problems.

In the FSTSP,[1] a truck and a drone cooperate to visit all the customers in a given network in the minimum amount of time. The drone can only serve one customer per sortie (i.e., drone flight). A natural generalization of FSTSP is the traveling salesman problem with drones (TSP-mD), where multiple drones are allowed to serve multiple customers per sortie, with the length of each sortie bounded by a limited battery capacity (see, e.g., Luo et al. 2021). Most studies of FSTSP and TSP-mD impose the additional constraint that the truck cannot visit customers multiple times, with the notable exception of the work by Agatz, Bouman, and Schmidt (2018), Bouman, Agatz, and Schmidt (2018), and Tang, van Hoeve, and Shaw (2019). As pointed out by Roberti and

Ruthmair (2021), allowing such *revisits* poses additional computational challenges, which cannot be easily accommodated by many of the existing approaches in the literature. Indeed, by adding sufficiently many copies of the nodes to the underlying graph, one can always reduce revisiting truck routes to cycles; this approach, however, does not appear to be computationally viable in practice. Roberti and Ruthmair (2021) also state that an analysis of the cost savings by revisits had, at the time of their writing, not been conducted yet.

In this paper, we consider a problem setting where multiple drones can serve multiple customers per sortie. Indeed, the benefits of deploying a fleet of drones, as opposed to a single drone, were studied by, among others, Murray and Raj (2020). Also, the technology to enable the drones to deliver multiple packages during a single sortie has already been developed (*The Washington Post* 2021), allowing for a speedup in delivery compared with drones that can only visit one customer per sortie. We observe that optimal solutions for TSP-mD may require the truck to not only revisit nodes but also, *retraverse* arcs of the underlying directed graph (i.e., in the course of its tour, the truck might need to repeatedly travel directly from customer i to customer j for some fixed pair of customers i, j). As we show, excluding such *arc-retraversing* solutions can lead to a significant increase in the optimal value. The necessity of arc retraversals by the truck does not seem to have been investigated in previous studies, and those studies that allow node revisits seem to operate under the assumption that there always exists an optimal solution without retraversals. In fact, the integer programming (IP) formulation proposed in the seminal paper by Agatz, Bouman, and Schmidt (2018) for FSTSP with node revisits implicitly excludes certain arc-retraversing solutions. We prove that this implicit assumption is correct under specific conditions, which the FSTSP setting studied by Agatz, Bouman, and Schmidt (2018) meets. However, when these conditions are not satisfied (e.g., when allowing multiple customers to be visited by a single sortie), the optimal value might increase significantly when excluding arc-retraversing solutions. We provide asymptotically tight a priori (i.e., solution-independent) and a posteriori (i.e., solution-dependent) upper bounds on such an increase. In particular, the optimal value can increase by a factor of at most $1 + 2m$ (where m is the number of drones) in the worst case when excluding arc-retraversing solutions. The same worst-case increase holds when excluding node revisits, giving a partial answer to the question raised by Roberti and Ruthmair (2021). Finally, we provide an approximation algorithm whose approximation guarantee depends on the speed and the number of available drones and show that unless $P = NP$, no approximation algorithm can obtain a guarantee that does not depend on these two parameters.

The remainder of the paper is organized as follows. A review of relevant works is presented in Section 2, whereas the TSP-mD itself is formally defined in Section 3. In Section 4, we describe problem settings where it suffices to consider solutions that are not arc retraversing. We establish the aforementioned upper bounds on the increase of the optimal value in Section 5. Finally, Section 6 contains the proofs of our approximability results. We conclude with Section 7. To prove the results of Section 4, we solve a number of instances via a mixed-integer linear programming (MILP) formulation; we describe the relevant instances and provide the MILP formulation in Appendices A and B, respectively.

2. Related Literature

In this section, we review the drone routing literature that specifically addressed truck-drone(s) operation problems with exact methods. We further focus on settings with a single truck that visits or delivers to customers in parallel to the drone(s) and where the completion time is minimized.

For a fast entry point to the literature, not exclusively from the operations research perspective, we refer the reader to the surveys by Otto et al. (2018), Roca-Riu and Menendez (2019), Macrina, Pugliese, and Guerriero (2020), Merkert and Bushell (2020), Boysen, Fedtke, and Schwerdfeger (2021), Ding, Xin, and Chen (2021), Li et al. (2021), and Persson (2021). For surveys with a focus on drone applications to routing and parcel delivery, we refer the reader to Coutinho, Battarra, and Fliege (2018), Khoufi, Laouiti, and Adjih (2019), Chung, Sah, and Lee (2020), Macrina et al. (2020), Moshref-Javadi and Winkenbach (2021), and Rojas Viloria et al. (2021) in chronological order of publication from 2018 to 2021. Finally, we mention the instructive overview of the challenges ahead for drone-aided routing provided by Poikonen and Campbell (2021).

The FSTSP was introduced by Murray and Chu (2015). They provided an MILP model and solved the problem via heuristic methods. A two-stage decomposition for solving the FSTSP was developed by Yurek and Ozmutlu (2018), by which they were able to solve instances with 12 nodes to optimality within one hour of computations. Dell'Amico, Montemanni, and Novellani (2021b) provided two novel formulations for the FSTSP and further refined them in a follow-up paper (Dell'Amico, Montemanni, and Novellani 2022). In particular, the latter study tackled a number of variants of the problem, one of which includes sorties with the same launch and landing location; we hereafter refer to such sorties as *loops*. The same authors also proposed a branch-and-bound algorithm capable of solving instances with up to 19 vertices in one hour of CPU time (Dell'Amico, Montemanni, and Novellani 2021a). Recently, Freitas, Penna, and Toffolo (2021) proposed a novel MILP formulation for the FSTSP, by which they solved instances with up to 10 nodes in an average of less than two minutes.

Jeong, Song, and Lee (2019) proposed a mathematical model to solve a generalization of the FSTSP that incorporates circle-shaped no-fly zones for drones and parcel weights. They reported exact solutions for instances with up to 10 customers. Luo et al. (2019), González-R et al. (2020), and Ha et al. (2021) tackled variants of the FSTSP where sorties can contain multiple customers; Luo et al. (2019) tailored their variant to traffic patrolling applications. Boccia et al. (2021b) and Vásquez, Angulo, and Klapp (2021) considered a variant of the FSTSP that further allows loops. They both solved instances with up to 20 vertices in a reasonable amount of time, the former by a Benders decomposition and the latter via branch-and-cut. In a follow-up paper, Boccia et al. (2021a) also solved the FSTSP by combining a branch-and-cut procedure with a column generation procedure. Jeon et al. (2021) introduced a variant where both delivery and pickup demands are met by allowing every sortie to visit one delivery and one pickup location (in this order) before landing on the truck. They solved instances with up to nine customers via an MILP model within 30 minutes of CPU time on average.

Agatz, Bouman, and Schmidt (2018) introduced a variant of the FSTSP where the drone can perform loop sorties and the truck can revisit a customer. They named this variant the traveling salesman problem with a drone (TSP-D) and proposed a model that contains a large number of binary variables, one for each feasible truck-drone joint operation. The same authors devised a dynamic programming approach in Bouman, Agatz, and Schmidt (2018) and solved instances with up to 16 nodes within three hours of computations on average. Tang, van Hoeve, and Shaw (2019) proposed a constraint programming approach for the TSP-D and solved instances with up to 18 nodes in an average of less than 10 minutes of computations. To the best of our knowledge, these three papers are the only ones in the literature that solved a variant of the FSTSP with a single truck and node revisits. In particular, Agatz, Bouman, and Schmidt (2018) is the only study that proposed an IP formulation for the problem; in Section 3, we describe arc-retraversing solutions that are not allowed by this formulation.

There is no general consensus in the literature on whether the TSP-D should allow node revisits, as per the original definition in Agatz, Bouman, and Schmidt (2018). In the remainder of this section, we classify the references by their own definition of the TSP-D, even if their problem settings do not exactly coincide with the original definition of the FSTSP and the TSP-D by Murray and Chu (2015) and by Agatz, Bouman, and Schmidt (2018), respectively. For sake of clarity, we schematically summarize the problem settings considered in our selection of the literature in Table 1. In particular, we indicate whether the relevant papers considered node revisits and arc retraversals in the truck route, sorties that serve multiple customers (also known in the literature as *multivisit* sorties), and loop sorties. Furthermore, the column with the header "Metrics" indicates how the traveling times for the truck (first letter) and the drones (second letter) were determined, respectively. More precisely, the letters E and M stand for the Euclidean and Manhattan distances, respectively. The letter R indicates that the distance between two nodes of the input graph was given by the underlying road network. Finally, in the last tab of Table 1, we indicate with the letters M, E, and H whether the relevant studies proposed at least one mathematical formulation, one tailored exact algorithm, and one heuristic or metaheuristic method, respectively. When a paper tackled a number of variants, if at least one of them considered a certain setting (such as allowing drone loops), then we check the corresponding cell of the table.

Some studies (e.g., Schermer, Moeini, and Wendt 2020, El-Adle, Ghoniem, and Haouari 2021) opted for excluding node revisits in the TSP-D, contrary to the original definition by Agatz, Bouman, and Schmidt (2018). The former proposed MILP formulations capable of directly solving instances with up to 10 customers within one hour of CPU time and up to 20 customers when embedded in a branch-and-cut algorithm. The latter provided an MILP model that solved instances with up to 24 nodes. Zhu, Boyles, and Unnikrishnan (2022) tackled a variant of the TSP-D where the truck is also an electric vehicle and must visit recharge stations periodically. Their electric truck is only allowed to revisit the locations corresponding to recharging stations. They developed a branch-and-price algorithm by which they solved instances with up to 10 nodes in one hour. Finally, Roberti and Ruthmair (2021) proposed a branch-and-price algorithm to effectively solve the TSP-D without node revisits on instances with up to 39 nodes within one hour of CPU time. The main setting of their work considers drones with no battery limitations and prevents them from performing loops; they also discussed variants, however, that incorporate limited battery endurance and loop sorties.

The FSTSP was soon generalized to the multiple drones case. In 2019, Seifried (2019) proposed an MILP model based on vehicle flows. Murray and Raj (2020) solved instances with up to eight nodes via an MILP model within one hour of computations. In their setting, the launch and retrieval times for the drones are not negligible. Dell'Amico, Montemanni, and Novellani (2021c) tackled a further variant where the drones are allowed to wait for the truck by hovering, and their retrieval gives rise to a nested scheduling problem at any customer location. They provided four formulations and solved instances with 10 customers to optimality in one hour. Jeong and Lee (2019) and Luo et al. (2021) further allowed multiple customers in a single sortie and solved instances with up to 10 customers in a

Table 1. Overview of the Problem Settings in the Related Literature

	Revisits	Retraversing	Multiple drones	Multivisit sorties	Loops	Metrics	Method
Agatz, Bouman, and Schmidt (2018)	✓				✓	E/E	M, H
Boccia et al. (2021a)						M/E	M, E
Boccia et al. (2021b)					✓	E/E	M, E
Bouman, Agatz, and Schmidt (2018)	✓				✓	E/E	E
Cavani, Iori, and Roberti (2021)			✓		✓	M/E	M, E
Dell'Amico, Montemanni, and Novellani (2021a)						M/E	M, E, H
Dell'Amico, Montemanni, and Novellani (2021b)						M/E	M, E
Dell'Amico, Montemanni, and Novellani (2021c)			✓			M/E	M, E
Dell'Amico, Montemanni, and Novellani (2022)					✓	M/E	M, E
El-Adle, Ghoniem, and Haouari (2021)						E/E	M, E
Freitas, Penna, and Toffolo (2021)						R/E	M, H
González-R et al. (2020)				✓		E/E	M, H
Ha et al. (2021)				✓		E/E	M, H
Jeon et al. (2021)						M/E	M, H
Jeong and Lee (2019)			✓	✓		R/E	M
Jeong, Song, and Lee (2019)						M/E	M, H
Luo et al. (2019)						R/R	M, H
Luo et al. (2021)			✓	✓		E/E	M, H
Murray and Chu (2015)						R/E	M, H
Murray and Raj (2020)			✓			R/E	M, H
Roberti and Ruthmair (2021)					✓	M/E	M, E
Schermer, Moeini, and Wendt (2020)					✓	E/E	M, E
Seifried (2019)			✓			E/E	M
Tang, van Hoeve, and Shaw (2019)	✓				✓		E
Vásquez, Angulo, and Klapp (2021)					✓	E/E	M, E
Yurek and Ozmutlu (2018)						M/E	M, H
Zhu, Boyles, and Unnikrishnan (2022)					✓	M/E	M, E, H
This work	✓	✓	✓	✓	✓	E/E	M

Notes. The letters E and M in the "Metrics" column stand for the Euclidean and Manhattan distances, respectively. The letter R indicates that the distances were given by the underlying road network. In the "Method" column, we indicate with the letters M, E, and H whether the relevant studies proposed at least one mathematical formulation, one tailored exact algorithm, and one heuristic or metaheuristic method, respectively.

reasonable amount of time. The latter further took the drones' payload into account in the battery energy consumption. Cavani, Iori, and Roberti (2021) identified a number of symmetry-breaking and valid inequalities and solved instances with up to 25 nodes to optimality via branch-and-cut, with a time limit of two hours.

For sake of completeness, we also mention a number of studies on the further generalization to the multiple trucks case: among other ones, Kitjacharoenchai et al. (2019), Bakir and Tiniç (2020), Tamke and Buscher (2021), and Zhou et al. (2022). From the theoretical side, Wang, Poikonen, and Golden (2017) provided several worst-case bounds on the optimal value, involving the number of vehicles and their relative speed.

3. Problem Statement

The TSP-mD can be defined as follows. Let N be the set of nodes including the customer locations and the depot (denoted by zero) and A be the set of arcs (i, j) for any pair of distinct nodes $i, j \in N$. The resulting graph $G = (N, A)$ is complete and directed. A noncapacitated truck and m identical drones cooperate to serve all the customers in N. We denote by $N^{dr} \subseteq N$ and $N^{tr} \subseteq N$ the subsets of the locations that can be served by the drones and by the truck, respectively. The intersection of these two sets is not necessarily empty. The depot 0 belongs to N^{tr}, and $N^{dr} \cup N^{tr} = N$. Notice that if a node $i \in N \setminus N^{tr}$, then the truck cannot traverse any of the arcs that are incident to i. In particular, the truck cannot reach I, and thus, no drone can be launched or retrieved at i. This assumption is justified when the location represented by a node $i \in N \setminus N^{tr}$ is unfit for truck traffic. For example, i could represent a crossing that is impractical for the truck to traverse or a narrow passage. Traversing the arcs that are incident to node i would imply the physical presence of the truck at i, which under this interpretation of the nodes in $N \setminus N^{tr}$, should be avoided.

We associate two distinct metrics ℓ and ℓ' with the arcs in A, representing the time it takes for the truck and for the drones, respectively, to traverse the arcs. In particular, by defining ℓ and ℓ' as metrics, we implicitly require that they are symmetric (i.e., $\ell_{ij} = \ell_{ji}$ and $\ell'_{ij} = \ell'_{ji}$ for all $(i, j) \in A$). We believe that the symmetry of ℓ and ℓ' is a property that naturally holds in many practical applications, and it is commonly required in the related literature. By relaxing it, however, only our results in Section 4.1 and in Appendix B would still hold in the generic case.

We assume without loss of generality that the truck travels at unit speed; therefore, ℓ_{ij} also measures the length of arc (i, j), for every $(i, j) \in A$. We denote the maximum speedup of the drones compared with the truck by

$$\alpha = \max\left\{\ell_{ij}/\ell'_{ij} : (i,j) \in A, \ell'_{ij} > 0, \text{ and } i,j \in N^{dr} \cap N^{tr}\right\}. \tag{1}$$

When it further holds that $\ell_{ij} = \alpha \cdot \ell'_{ij}$ for all distinct $i,j \in N^{dr} \cap N^{tr}$, we say that the two metrics are *proportional*. This case accurately models applications to last-mile delivery within urban areas, where drones might be forced to follow the existing road network by local regulations for security, privacy, and noise pollution concerns.

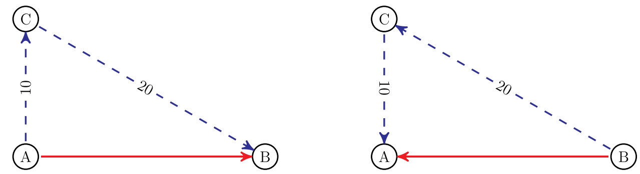
The truck route consists of a closed walk that starts and ends at zero and serves all the nodes contained therein. The drones can be independently launched onto airborne routes, hereafter referred to as *sorties*, and retrieved by the truck at the nodes on its route. Accordingly, we represent a sortie π by a tuple of nodes $(i_1, \ldots, i_r)$, with $r \geq 3$, where $i_1, i_r \in N^{tr}$, and $i_2, \ldots, i_{r-1} \in N^{dr}$. The nodes i_1 and i_r represent the starting and ending nodes of π, respectively. The set $\{i_2, \ldots, i_{r-1}\}$, which we denote by $N[\pi]$, represents the nodes that are served by the drone while flying along π. The drone can serve multiple customers in a single sortie. The amount of energy required to perform a sortie $\pi = (i_1, \ldots, i_r)$ is given by

$$w^{dr} \cdot \sum_{q=1}^{r-1} \ell'_{i_q i_{q+1}} + \sum_{p=2}^{r-1} \left(w_{i_p} \cdot \sum_{q=1}^{p-1} \ell'_{i_q i_{q+1}} \right), \tag{2}$$

where w^{dr} is the weight of a single drone and $w_i \geq 0$ is the payload to serve the customer at node i for every $i \in N[\pi]$. The energy consumed by any sortie must not exceed the maximum value $B > 0$ allowed by the battery, and every time a drone lands, its battery is swapped with a fully charged one in a negligible amount of time. It is good practice to choose a value B that underestimates the actual total energy contained in the battery of the drones. Indeed, one can significantly prolong the life of electrical batteries by not fully depleting them before recharging. Moreover, when a drone arrives at the destination node of a sortie before the truck, we allow it to land and to wait for the truck on the ground; this policy, however, might give rise to security concerns, which can be addressed by keeping a sufficient level of battery to operate the drone sensors, cameras, and wireless connection until the truck arrives. Alternatively, if serving a customer by a drone already implies that the relevant drone lands at the corresponding location (for instance, for package delivery), we can let the drone wait on the ground there before it flies to the ending location of the sortie without exposing it to further security risks.

We denote the set of the feasible sorties by P. By a slight abuse of notation, we denote by ℓ'_π the duration of a feasible sortie π (i.e., $\ell'_\pi = \sum_{q=1}^{r-1} \ell'_{i_q i_{q+1}}$). By setting $w_i = 0$ for every node $i \in N^{dr}$ in the energy consumption (2), we obtain that $\ell'_\pi \leq B/w^{dr}$ for every $\pi \in P$; we denote this upper bound on the duration of the feasible sorties by L.

Figure 1. (Color online) A Couple of Sorties, One Being the Inversion of the Other



Note. When $w^{dr} = 10$, $w_C = 5$, and $B = 350$, the sortie in the right panel is infeasible.

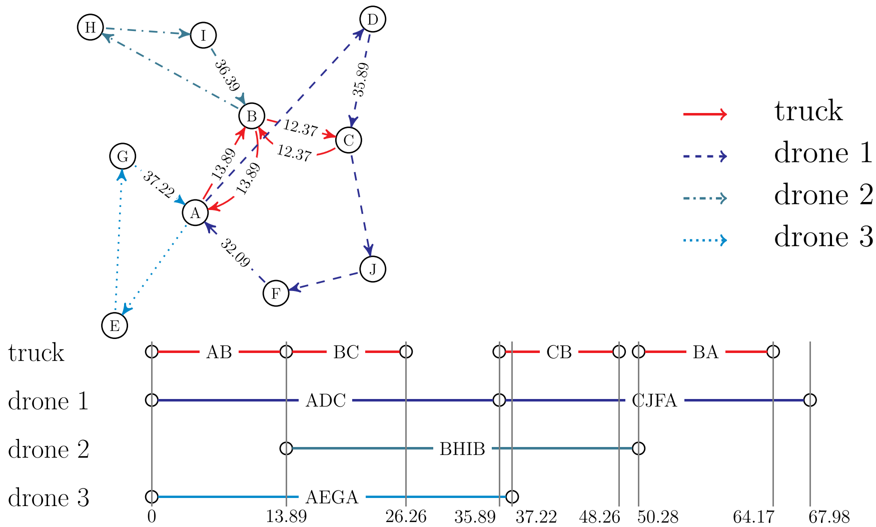
Notice that, because of the energy consumption (2), it might be infeasible to reverse the order of the nodes of a sortie, as illustrated in Figure 1; with $w^{dr} = 10$, $B = 350$, and a payload $w_C = 5$ at customer C, the energy consumption of the sortie on the left is $300 + 10 \cdot 5 = 350 \leq B$, whereas the one of its inverted counterpart is $300 + 20 \cdot 5 = 400 > B$. We refer to such sorties as *noninvertible*. The invertibility of the sorties is a crucial property in the results of Section 4; in fact, this property is satisfied by most of the settings in the related literature.

In the TSP-mD, both the truck and the drones are allowed to wait for each other at customer locations. The whole operation is complete when all the customers are served and both the truck and the drones have returned to the depot. We minimize the completion time. Figure 2 shows a feasible solution of an instance with 10 nodes (including the depot A) and three drones.

In line with the FSTSP variant of Agatz, Bouman, and Schmidt (2018), we allow the truck to visit customers multiple times. Note that this also opens the possibility for the truck to traverse the same arc multiple times (i.e., there may be pairs of customers i, j such that the truck travels directly from customer i to customer j on multiple occasions throughout its tour). We refer to solutions in which this happens as *arc retraversing*. Figure 3 shows an example of an (optimal) arc-retraversing solution for an instance with five nodes and two drones. Notice that the truck can traverse the same edge $\{i, j\}$ twice in the two opposite directions (i, j) and (j, i) without retraversing the same arc. In Section 4.1, we describe instances where all the optimal solutions are arc retraversing. Consequently, it is necessary to allow this type of solutions in our problem statement.

We observe that some arc-retraversing solutions are not feasible in the IP model of Agatz, Bouman, and Schmidt (2018). In particular, the solution space of the latter IP does not include any solutions in which the truck traverses the same arc twice when during both traversals, the drone is not airborne. Indeed, the two distinct traversals of the same arc constitute identical operations and thus, correspond to the same binary variable. In Section 4.2, we prove that in the FSTSP model

Figure 2. (Color online) A Feasible Solution for an Instance with 10 Nodes, Three Drones, and Proportional Metrics with $\alpha = 4/3$



Note. The routes of the truck and the drones (upper panel) and the corresponding Gantt chart (lower panel) are shown.

studied by Agatz, Bouman, and Schmidt (2018), at least one optimal solution is not arc retraversing, and hence, the formulation indeed finds an optimal solution. However, as soon as the FSTSP is generalized as to incorporate payloads, multiple customers per sortie or multiple drones, excluding arc-retraversing solutions, may lead to excluding all optimal solutions.

We distinguish between the TSP-mD and the two restrictions where arcs cannot be retraversed (m-CIRCUIT) and nodes cannot be revisited (m-CYCLE). Their optimal values are related by

$$\text{TSP-mD} \le \text{m-CIRCUIT} \le \text{m-CYCLE} \le \text{TSP}, \qquad (3)$$

where TSP refers to the traveling salesman problem.

The definitions of these restrictions of the TSP-mD are functional to the results of Section 5.

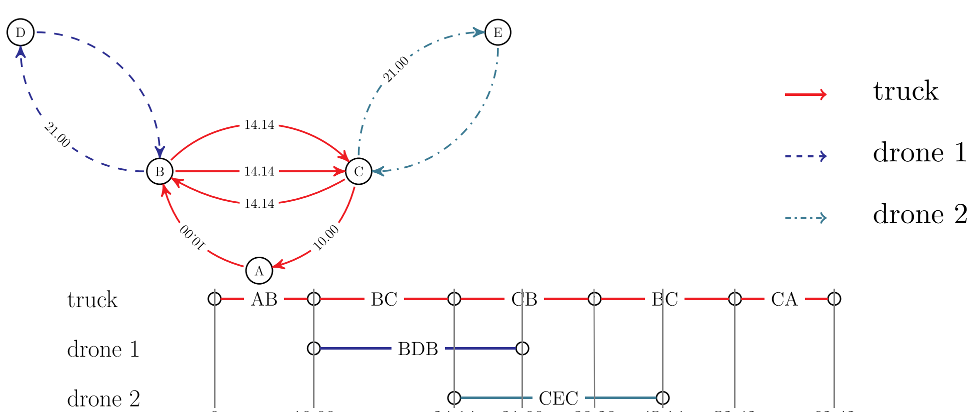
4. Arc-Retraversing Solutions

In this section, we describe instances for which all the optimal solutions are arc retraversing (Section 4.1) and provide conditions under which there always exists an optimal solution that is not arc retraversing (Section 4.2).

4.1. Necessity of Arc-Retraversing Solutions

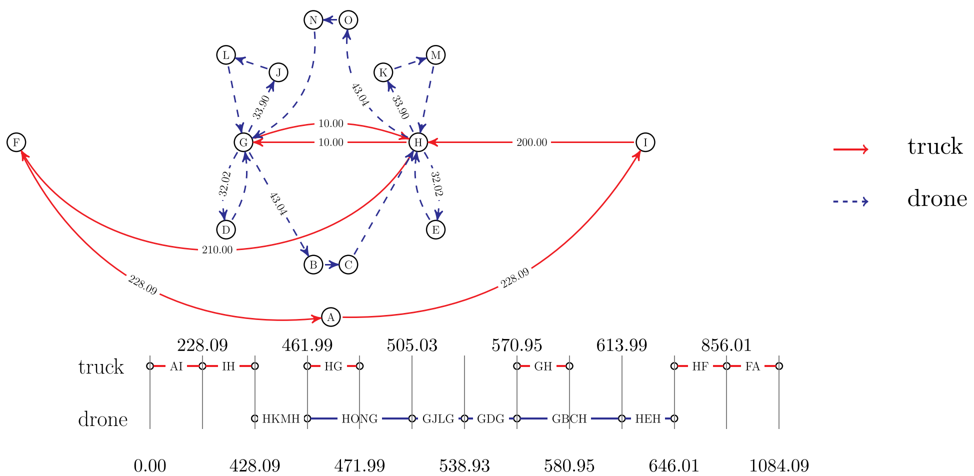
We prove that it is necessary to include arc-retraversing solutions in the solution space of the TSP-mD by describing instances whose optimal solutions are all arc retraversing. We show two instances that possess this property, with Euclidean metrics and all-zero payloads. The first instance has a single drone (Proposition 4.1), whereas the second one satisfies $N = N^{tr} = N^{dr}$ (Proposition 4.2). The proofs of the corresponding results require us to solve the TSP-mD a number of times. In Appendix B, we

Figure 3. (Color online) An Optimal Solution of an Instance with Five Nodes, $L = 28.0$, and Proportional Metrics with $\alpha = 4/3$



Note. The truck traverses arc (B, C) twice.

Figure 4. (Color online) An Optimal Solution of an Instance with 15 Nodes, Proportional Metrics with $\alpha = 1$, and $L = 43.04$ When the Truck Is Further Constrained to Traverse an Arc at Most Once



Note. The truck and the drone can visit the nodes in $N^{tr} = \{A, F, G, H, I\}$ and $N^{dr} = \{B, C, D, E, J, K, L, M, N, O\}$, respectively.

provide the MILP model by which we solve the relevant instances.

Proposition 4.1. *There exists an instance with a single drone, all-zero payloads, and Euclidean metrics such that all the optimal solutions are arc retraversing.*

Proof. The proof is complete if we can find an instance with Euclidean metrics, all-zero payloads, and $m = 1$ and the two feasible solutions $\mathcal{S}_1$ and $\mathcal{S}_2$ such that $\mathcal{S}_1$ is optimal under the further restriction that no arc can be retraversed; the respective objective function values OBJ_1 and OBJ_2 satisfy $\text{OBJ}_1 > \text{OBJ}_2$.

Such an instance is described in Appendix A.2 and shown in Figures 4 and 5; these figures represent solutions $\mathcal{S}_1$ and $\mathcal{S}_2$, and their objective function values OBJ_1 and OBJ_2 amount to 1,084.09 and 1,050.36, respectively. These solutions were obtained by solving the MILP model in Appendix B to optimality by a commercial solver. □

Notice that, in the instance shown in Figure 4, the truck and the drones are not allowed to visit all the nodes (i.e., $N^{tr} \neq N$ and $N^{dr} \neq N$). An analogous result to Proposition 4.1 also holds when $N = N^{dr} = N^{tr}$, even when the feasible sorties serve only one customer.

Proposition 4.2. *There exists an instance with Euclidean metrics and all-zero payloads such that $N = N^{dr} = N^{tr}$, the feasible sorties only serve one customer, and all the optimal solutions are arc retraversing.*

Proof. We follow an analogous argument to that of Proposition 4.1. Consider the instance described in Appendix A.3 and shown in Figures 3 and 6. On the

Figure 5. (Color online) A Feasible Solution of the Instance Shown in Figure 4 Without Any Constraints on the Number of Traversals of Any Arc

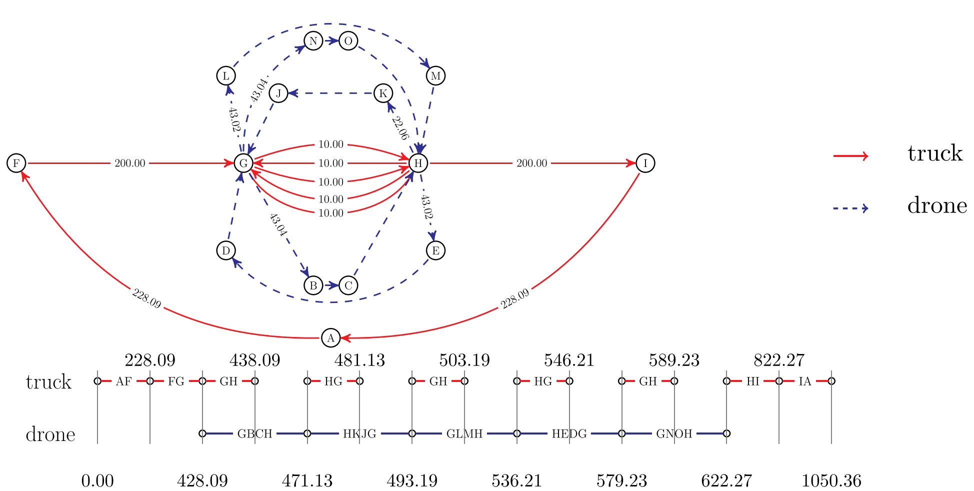
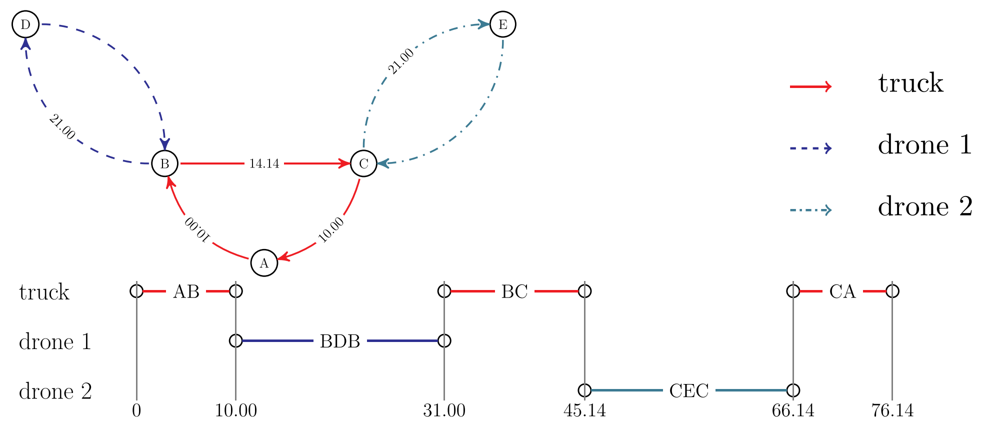


Figure 6. (Color online) An Optimal Solution of the Instance Shown in Figure 3 When the Truck Is Constrained Not to Traverse Any Arc More Than Once



one hand, the solution shown in Figure 6 is optimal under the condition that no arc-retraversing solutions are allowed and leads to a completion time of 76.14; on the other hand, that of Figure 3 (which is optimal and retraverses one arc) leads to a strictly lower completion time, namely 62.42. □

For the instance shown in Figures 3 and 6, not allowing arc-retraversing solutions leads to an increase of 18% of the optimal completion time.

4.2. Sufficiency of Nonarc-Retraversing Solutions

In this section, we provide conditions under which there exists an optimal solution that is not arc retraversing. The proofs of the corresponding propositions require a couple of intermediate results. We begin by considering the most general setting of the TSP-mD, namely, with multiple drones that can serve multiple customers per sortie and without conditions on the sets N^{dr} and N^{tr}. We then impose certain conditions on the problem input; in the corresponding settings, we prove the sufficiency of the solutions that are not arc retraversing.

Lemma 4.1. *There exists an optimal solution such that if a node $i \in N \setminus \{0\}$ is visited multiple times, then a drone is launched or retrieved every time the truck visits i.*

Proof. Suppose by contradiction that a node $i \in N \setminus \{0\}$ is visited multiple times, but during one of these times, no drone is launched or retrieved. Then, we can shortcut the truck route at i, without missing any drone operations, and the resulting route will not be strictly longer than the original one. □

An analogous result to Lemma 4.1 applies to the depot as well, starting from its third visit on (the first two visits being the start and the end of the whole operation). Given a feasible solution represented by a truck route π^{tr} and drone sorties, we define the *support* of a sortie π^{dr} as the subroute $\pi' \subseteq \pi^{tr}$ traversed by the truck when a drone flies along π^{dr}. For example, in Figure 7, the supports of the sorties (B, G, D), (D, H, K, D), (D, I, L, F), (F, J, B) are (B, C, D), D, (D, E, F), and the arc (F, B), respectively.

We now restrict to the setting where only a single drone is available (but can serve multiple customers per sortie).

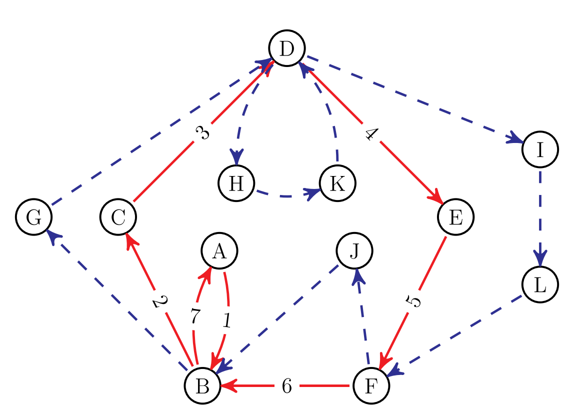
Lemma 4.2. *Let $m = 1$. There exists an optimal solution such that if an arc $(i, j) \in A$ is traversed multiple times, then the following conditions hold.*

i. *If the support of a sortie π' contains (i, j), then the support of π' is equal to arc (i, j).*

ii. *If the support of a sortie π' shares at least one arc with a truck subroute π_{ji} from j to i between two consecutive traversals of (i, j), then the support of π' is contained in π_{ji}.*

Proof. If the support of a sortie π' contains (i, j) and at least one more arc, then the truck visits either i or j

Figure 7. (Color online) Illustration of the Definition of Support of a Sortie



Note. The numbers indicate the positions of the corresponding arcs in the truck route.

without launching or retrieving the drone. Then, by Lemma 4.1, we can shortcut its route at the redundant node without losing optimality.

Suppose that a sortie π' shares at least one arc with a truck subroute π_{ji}, like in the hypothesis. Without loss of generality, we can assume that the arcs shared by both π' and π_{ji} are only traversed once. Indeed, if not, then the thesis holds by step (i). If the support of π' contains an arc that does not belong to π_{ji}, then the truck must visit either i or j while the drone is airborne. Again, by Lemma 4.1, we can shortcut the truck route at the redundant node without losing optimality. □

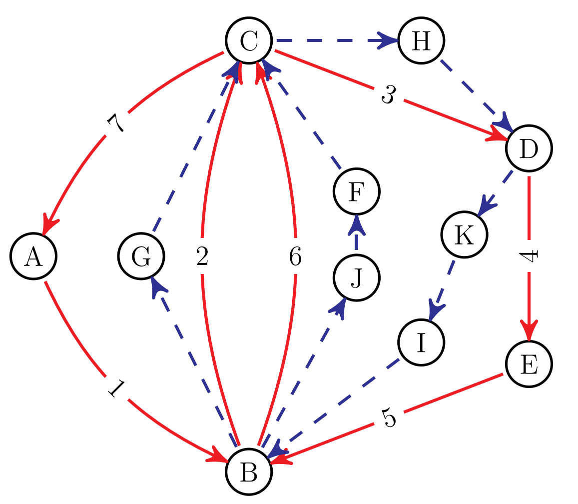
Figure 8 illustrates the thesis of Lemma 4.2. In this example, the support of sorties (B, G, C) and (B, J, F, C) is the arc (B, C), and the supports of sorties (C, H, D) and (D, K, I, B) are the truck subroutes (C, D) and (D, E, B), respectively.

We make use of Lemmas 4.1 and 4.2 to prove two sufficiency results for the solutions that are not arc retraversing. Both the results consider a single drone that can only perform invertible sorties.

Proposition 4.3. *At least one optimal solution is not arc retraversing if only a single drone is available, all sorties are invertible, and they serve at most one customer.*

Proof. Consider an optimal solution with the properties described in Lemma 4.2, and denote its truck route by π. Without loss of generality, suppose that at least one arc in π is traversed multiple times (if not, there is nothing to prove); we choose one and denote it by (i, j). The proof is complete if we show that there always exists another truck route π' that does not lead to a (strictly) longer completion time and that traverses (with multiplicity) two arcs less than π.

Figure 8. (Color online) Illustration of the Thesis of Lemma 4.2

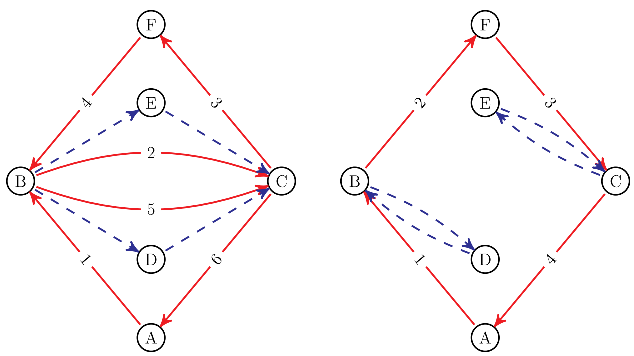


Because the truck travels from i to j at least twice, its route must contain a path from j to i, between two consecutive traversals of (i, j), that we denote by π_{ji}. Then, we call π_{0i} and π_{j0} the paths from the depot to i and from j to the depot, respectively, such that $\pi = \pi_{0i} \circ (i, j) \circ \pi_{ji} \circ (i, j) \circ \pi_{j0}$. Because we only have one drone, Lemma 4.2 implies that the supports of all the sorties performed while the truck travels along (i, j) or π_{ji} are contained in (i, j) and π_{ji}, respectively. Because sorties are invertible by hypothesis, we can invert both the truck route π_{ji} and the sorties whose support is contained in it; we call the resulting inverted route π_{ji}^{-1}. The new truck route defined by $\pi_{0i} \circ \pi_{ji}^{-1} \circ \pi_{j0}$ does not miss any drone operations whose supports were originally contained in π_{0i}, π_{ji}, or π_{j0}; however, by defining the arc (i, j) out of the new route twice, we might miss all those sorties whose supports were originally the arc (i, j) (at most two of them because we removed (i, j) twice, and only a single drone is available).

Because we are only allowed to visit one customer per sortie, such sorties must be of the form (i, k, j), for some intermediate customer $k \in N^{dr} \setminus \{i, j\}$. If $\ell'_{ik} \leq \ell'_{kj}$, then at node i, as soon as the truck has traveled along π_{0i} and would otherwise be ready to leave i, we can instead launch the drone back and forth from i to k and let the truck wait for it at i. Otherwise, we can launch the drone back and forth from j to k, immediately after the truck has traveled along π_{ji}^{-1}. These (possibly two) new sorties are by construction not longer than the original sorties of the form (i, k, j). Hence, they are feasible, and when incorporated into the new truck route $\pi' = \pi_{0i} \circ \pi_{ji}^{-1} \circ \pi_{j0}$, they lead to a completion time that is not greater than π's original one. In this solution, the truck traverses (with multiplicity) two arcs less than π. □

The idea of the transformation $\pi \longmapsto \pi'$ is depicted in Figure 9. The three conditions on Proposition 4.3 are met by most of the problem settings in the related literature; this fact justifies their implicit assumption that arc-retraversing solutions do not lead to a strictly lower completion time. Moreover, these conditions are minimal. Indeed, Proposition 4.3 does not hold when two drones are available (as implied by the solutions shown in Figures 3 and 6), sorties are noninvertible (in particular, the inverted path π_{ji}^{-1} might require the drone to fly along infeasible sorties), or sorties serve two customers (see Figure 5). For example, in the problem setting of Agatz, Bouman, and Schmidt (2018), Proposition 4.3 holds, and therefore, it is correct to solve the problem via an IP model that does not allow arc-retraversing solutions. However, if we wanted to generalize the setting so as to incorporate multiple drones or payloads in the energy consumption of a sortie, then excluding arc-retraversing solutions may lead to a strictly higher

Figure 9. (Color online) Illustration of the Argument of Proposition 4.3



completion time. An analogous result to Proposition 4.3 also holds in a slightly different setting, when the drone does not travel faster than the truck, and can serve multiple customers per sortie.

Proposition 4.4. *At least one optimal solution is not arc retraversing if only a single drone is available, all sorties are invertible, $N^{tr} = N$, and $\alpha = 1$.*

Proof. Consider an optimal solution with the properties described in Lemma 4.2, and suppose that it is arc retraversing. Analogously to Proposition 4.3, we denote one of the retraversed arcs by (i, j) and the truck route by π. Let π_{ji}, π_{0i}, and π_{j0} be the analogous paths to those defined in the proof of Proposition 4.3; accordingly, $\pi = \pi_{0i} \circ (i,j) \circ \pi_{ji} \circ (i,j) \circ \pi_{j0}$. The proof is complete if we show that there is always another solution π' with the same completion time, such that the total number of arcs (with multiplicity) that are traversed by the truck and the drone decreases by at least one unit.

Without loss of generality, we can assume that the drone is airborne during at least one truck traversal of (i, j), either immediately before the traversal of π_{ji} or immediately after. Indeed, if not, then we remove arc (i, j) twice from the route and invert π_{ji} (the relevant sorties are invertible by hypothesis, and their support is contained in π_{ji} by Lemma 4.2); the solution induced by the truck route $\pi' = \pi_{0i} \circ \pi_{ji}^{-1} \circ \pi_{j0}$ is not longer than π. Suppose that the drone flies along a path π_{ij}^{dr} during the truck traversal of arc (i, j) immediately before the traversal of π_{ji} (the proof is analogous with the traversal of (i, j) immediately after π_{ji}). Then, instead of launching the drone along π_{ij}^{dr}, we instead let the truck traverse it. This is feasible because $N^{tr} = N$, and it does not take (strictly) longer than the drone's sortie because $\alpha = 1$.

After routing the truck along π_{ij}^{dr}, we follow the original route π along $\pi_{ji} \circ (i,j)$. The new truck route given by $\pi' = \pi_{0i} \circ \pi_{ij}^{dr} \circ \pi_{ji} \circ (i,j) \circ \pi_{j0}$ is feasible and leads to a completion time that is not (strictly) larger than that of π. Additionally, in this solution, the total number of arcs (with multiplicity) that are traversed by the truck and the drone is strictly smaller than in the original solution. □

The idea of the proof of Proposition 4.4 is shown in Figure 10. The set of conditions required in Proposition 4.4 is minimal. Indeed, we would not be able to replace the truck route (i, j) with π_{ij}^{dr} if $N^{tr} \neq N$ or $\alpha > 1$. Sorties must be invertible to include cases where an arc (i, j) is traversed twice by the truck while the drone is not airborne. The route transformation $\pi \longmapsto \pi'$ of Proposition 4.4 is functional to the a posteriori upper bounds described in Section 5.2.

5. Excluding Arc-Retraversing Solutions

By excluding arc-retraversing solutions in the TSP-mD, the optimal value might increase. In Sections 5.1 and 5.2, we identify conditions under which a priori (i.e., solution-independent) and a posteriori (i.e., solution-dependent) upper bounds hold on such increase, respectively.

5.1. A Priori Upper Bounds on the Increase of the Completion Time

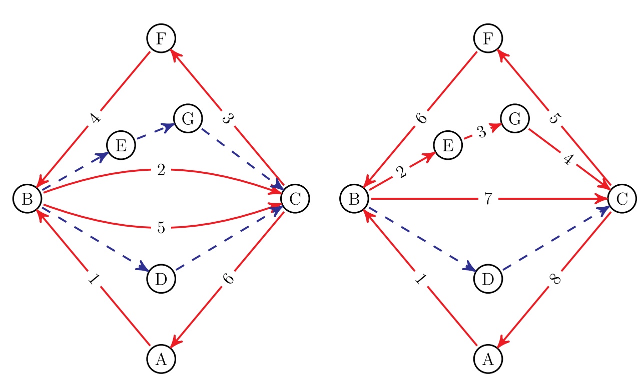
In Section 3, we have defined two restrictions of the TSP-mD, namely the m-CYCLE and the m-CIRCUIT; by Inequality (3), any upper bound on m-CYCLE/TSP-mD also holds on m-CIRCUIT/TSP-mD. We identify a priori upper bounds on the former ratio m-CYCLE/TSP-mD that depend on m and α.

Proposition 5.1. *If the truck can visit every node, then it holds that*

$$m\text{-}CYCLE \leq (1 + \alpha m)TSP\text{-}mD. \tag{4}$$

Inequality (4) follows from m-CYCLE ≤ TSP and from TSP ≤ $(1 + \alpha m)$TSP-mD; the latter inequality was shown by Wang, Poikonen, and Golden (2017) in an analogous setting. We now describe other a priori upper bounds that only depend on m and that dominate Inequality (4) for all $\alpha > 2$. For sake of clarity, we first prove a preliminary result by the following lemma.

Figure 10. (Color online) Illustration of the Argument of Proposition 4.4



Lemma 5.1. *Let $\pi = (i_1, \dots, i_r)$ be a feasible nonloop sortie. For every $t \in \{2, \dots, r-1\}$, at least one sortie out of $\pi_1 = (i_1, \dots, i_t, i_1)$ and $\pi_2 = (i_r, \dots, i_t, i_r)$ is feasible.*

Proof. For the sake of notation, we define the quantity $\overline{\ell}(p_1, p_2) = \sum_{q=p_1}^{p_2-1} \ell'_{i_q i_{q+1}}$ for indices p_1 and p_2 that satisfy $p_1 < p_2 \le r$. Because π is feasible, it holds that

$$w^{dr} \cdot \overline{\ell}(1,r) + \sum_{p=2}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)) \le B, \tag{5}$$

where the left-hand side quantifies the energy consumption of π. Let t and π_1 be an index and a sortie like in the hypothesis, respectively. Suppose that π_1 is not feasible. Because ℓ' is a metric, it holds that $\overline{\ell}(1,t) \ge \ell'_{1t}$. Then, the following quantity is not lower than the energy consumption of π_1:

$$\begin{aligned} &2w^{dr} \cdot \overline{\ell}(1,t) + \sum_{p=2}^{t} (w_{i_p} \cdot \overline{\ell}(1,p)) \\ &\quad \ge w^{dr} \cdot \overline{\ell}(1,t) + \sum_{p=2}^{t} (w_{i_p} \cdot \overline{\ell}(1,p)) + w^{dr} \cdot \ell'_{t,1} > B. \end{aligned} \tag{6}$$

We want to show that $\pi_2 = (i_r, \dots, i_t, i_r)$ is feasible. By subtracting the left-hand side of Inequality (5) from the first step of chain (6), we deduce that

$$w^{dr} \cdot \overline{\ell}(1,t) > w^{dr} \cdot \overline{\ell}(t,r) + \sum_{p=t+1}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)). \tag{7}$$

By dropping the last term on the right-hand side of Inequality (7), we obtain that, for every $p \ge t$,

$$\overline{\ell}(1,p) \ge \overline{\ell}(1,t) > \overline{\ell}(t,r) \ge \overline{\ell}(p,r). \tag{8}$$

The following chain of inequalities shows that the energy consumption of π_2 is not greater than that of π, which in turn, implies that π_2 is feasible:

$$\begin{aligned} &w^{dr} \cdot (\ell'_{tr} + \overline{\ell}(t,r)) + \sum_{p=t}^{r-1} (w_{i_p} \cdot \overline{\ell}(p,r)) \\ &\le 2w^{dr} \cdot \overline{\ell}(t,r) + \sum_{p=t}^{r-1} (w_{i_p} \cdot \overline{\ell}(p,r)) \\ &\le 2w^{dr} \cdot \overline{\ell}(t,r) + \sum_{p=t}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)) \\ &\le w^{dr} \cdot \overline{\ell}(t,r) + w^{dr} \cdot \overline{\ell}(1,t) + \sum_{p=t}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)) \\ &= w^{dr} \cdot \overline{\ell}(1,r) + \sum_{p=t}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)) \\ &\le w^{dr} \cdot \overline{\ell}(1,r) + \sum_{p=2}^{r-1} (w_{i_p} \cdot \overline{\ell}(1,p)). \end{aligned} \tag{9}$$

The first step of the chain follows from the triangle inequality, whereas the second and third steps follow from Inequalities (8). □

Notice that, with the notation of Lemma 5.1, the sorties π, π_1, and π_2 satisfy the inequality $\ell'_{\pi_1} + \ell'_{\pi_2} \le 2\ell'_{\pi}$ by construction. This fact is crucial to the proof of the following result.

Proposition 5.2. *It holds that*

$$m\text{-CYCLE} \le (1 + 2m)\text{TSP-}m\text{D}. \tag{10}$$

If the feasible sorties are either loops or contain only one customer, then it further holds that

$$m\text{-CYCLE} \le (1 + m)\text{TSP-}m\text{D}. \tag{11}$$

Proof. We first prove Inequalities (10) and (11) for the case $m = 1$, and we then adapt the argument to any $m \ge 2$. Suppose that $m = 1$; in this case, the proof of Inequality (10) is complete if, for every feasible solution, we can replace every nonloop sortie π by either a single sortie π_0 or by two sorties π_1 and π_2 that satisfy all the following conditions.

i. Sortie π_0 or sorties π_1 and π_2 are feasible loops.
ii. It holds that $N[\pi] = N[\pi_0]$, or $N[\pi] = N[\pi_1] \cup N[\pi_2]$.
iii. Either inequality $\ell'_{\pi_0} \le 2\ell'_{\pi}$ or inequality $\ell'_{\pi_1} + \ell'_{\pi_2} \le 2\ell'_{\pi}$ holds.

Indeed, if we can always replace every nonloop sortie π either by the corresponding π_0 or by π_1 and π_2, then we let the truck wait at the starting node of every sortie, and we shortcut its route to eliminate every node revisit. The resulting solution is feasible to the 1-CYCLE restriction, and its objective value satisfies 1-CYCLE $\le 3 \cdot$ TSP-1D $= (1 + 2m)$TSP-1D because the travel times of both the truck and the drone are not longer than the TSP-1D objective value and because of condition (iii).

Then, let $\pi = (i_1, \dots, i_r)$ be a feasible nonloop sortie. We choose the relevant sorties π_0 or π_1 and π_2 as follows. If at least one sortie out of $(i_1, \dots, i_{r-1}, i_1)$ and $(i_r, \dots, i_2, i_r)$ is feasible, we choose one and denote it as π_0. Otherwise, by Lemma 5.1, sorties (i_1, i_2, i_1) and (i_r, i_{r-1}, i_r) are both feasible. Therefore, there exists a unique $t \in \{2, \dots, r-2\}$ such that the sortie $(i_1, \dots, i_t, i_1)$ is feasible but $(i_1, \dots, i_{t+1}, i_1)$ is not. We can then choose $\pi_1 = (i_1, \dots, i_t, i_1)$, and by Lemma 5.1, $\pi_2 = (i_r, \dots, i_{t+1}, i_r)$. The sorties π_0 or π_1 and π_2 satisfy the aforementioned conditions (i)–(iii).

If a nonloop sortie contains only a single customer (i.e., it is of the form (i_1, i_2, i_3)), then we can choose the shortest arc between (i_1, i_2) and (i_2, i_3) and set $\pi_0 = (i_1, i_2, i_1)$ or $\pi_0 = (i_3, i_2, i_3)$ accordingly. Notice that in this case, it holds that $\ell'_{\pi_0} \le \ell'_{\pi}$. If the feasible sorties are either loops or satisfy $r = 3$, an argument analogous to that for the $(1 + 2m)$ bound leads to the inequality 1-CYCLE $\le 2 \cdot$ TSP-1D $= (1 + m)$TSP-1D.

If $m \ge 2$, we can repeat the same construction for the sorties of every drone, but this time, the truck has to wait up to m times longer for all the drones to perform their sorties. □

We complement the results of the previous proposition by showing that Inequalities (10) and (11) are asymptotically tight with a single drone and that the ratio m-CYCLE/TSP-mD cannot be upper bounded by any constant. In particular, we cannot replace m by any constant in the right-hand sides of Inequalities (10) and (11).

Proposition 5.3. *For $m = 1$, the bounds (10) and (11) are asymptotically tight. Moreover, for every m, there exists an instance for which (m-CYCLE/TSP-mD) $\geq m$.*

Proof. We first show that bound (10) is asymptotically tight for $m = 1$. Consider the set of instances shown in Figure 11; the truck must visit nodes $0, 1, v_1, v_2, \ldots, v_k, w_1, w_2, \ldots, w_k$, whereas the drone must serve nodes $i_1, i_2, \ldots, i_{2k}, j_1, j_2, \ldots, j_{2k}$ for any $k \geq 1$; the node 0 is the depot. The lengths of the arcs are either those shown in the figure or the shortest path between the relevant end points. The drone travels equally as fast as the truck (i.e., $\ell_{ij} = \ell'_{ij}$ for every arc $(i, j) \in A$). We consider all-zero payloads and set $L = 2 + \epsilon$. An optimal TSP-1D solution can be described as follows; the drone flies along sorties of the form $(0, i_q, j_q, 1)$ and $(1, j_q, i_q, 0)$ for $q \leq 2k$, whereas the truck traverses the edge $\{0, 1\}$ for a total of $2k$ times and subroutes of the form $(0, v_q, 0)$ and $(0, w_q, 0)$ for $q \leq k$ every time the drone is airborne. This solution leads to an objective value of $2k(2 + \epsilon)$. An optimal solution to the 1-CYCLE can only serve two customers in a single sortie only twice, and the truck can visit one node v_q and w_q while the drone is airborne only once; hence, the optimal value for the 1-CYCLE amounts to $2(2 + \epsilon) + 4(k - 1) + 4(2k - 2)$. By choosing $\epsilon = \frac{1}{k}$, we get that $\lim_{k \to \infty}$(1-CYCLE/TSP-1D) = 3.

Analogously, bound (11) is asymptotically tight for $m = 1$. Consider the instance shown in Figure 11 and remove nodes $i_1, \ldots, i_{2k}, v_1, \ldots, v_k, j_{k+1}, j_{k+2}, \ldots, j_{2k}$. Again, the drone travels as fast as the truck; we consider all-zero payloads and set $L = 2$. The only allowed sorties are then loops of the form $(1, j_q, 1)$ for $q \leq k$. An optimal TSP-1D solution can visit nodes w_q for $q \leq k$ while the drone is airborne, and this leads to an optimal value equal to $2\epsilon + 2k$. At the same time, the optimal value for the 1-CYCLE amounts to $2\epsilon + 4k$, which leads to an asymptotic bound of two.

Finally, we need to show that the ratio m-CYCLE/TSP-mD cannot be upper bounded by any constant. Let $m > 1$, and consider m nodes at the vertices of a regular m-tope with edge length ϵ; for each of these nodes $i_1, \ldots, i_m$, we consider nodes $j_1, \ldots, j_m$ such that the length $\ell_{i_q j_q} = \frac{1}{2}$ for all $q \leq m$. The length of all the other arcs is set to the shortest path between their end points. We choose i_1 as the depot and set proportional metrics with $\alpha = 2$ and $L = 1$. The only sorties that can be selected in an optimal solution are loops of the form (i_q, j_q, i_q) for $q \leq m$. The optimal m-CYCLE objective amounts to $m(1 + \epsilon)$, whereas the optimal TSP-mD amounts to $1 + m\epsilon$. □

5.2. A Posteriori Upper Bounds on the Increase of the Completion Time

In this section, we describe a method to provide a solution-dependent upper bound on the increase of the completion time because of the exclusion of arc-retraversing solutions from the solution space of the TSP-mD. Consider an optimal solution $\mathcal{S}$ to the TSP-mD, and let $h = (h_1, h_2, \ldots, h_{|A|})$ be a vector in $\mathbb{Z}_+^{|A|}$, whose components h_a represent the number of times the truck traverses arc a in $\mathcal{S}$, for any $a \in A$. We call h the *retraversing vector* of $\mathcal{S}$.

Proposition 5.4. *Suppose that $N^{tr} = N$, $\alpha \geq 1$ and only a single drone is available. Then, for any optimal solution whose retraversing vector is $h \in \mathbb{Z}_+^{|A|}$, it holds that*

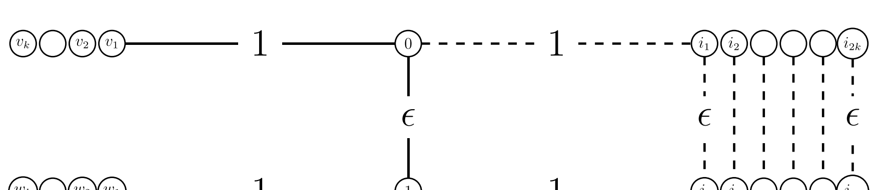
$$\textit{TSP-mD} + \left(\sum_{a \in A} (h_a - 1)^+ \right) \cdot (\alpha - 1) \cdot L \geq \textit{m-CIRCUIT}. \tag{12}$$

Proof. The proof follows from transforming the truck route analogously to the proof of Proposition 4.4 for every arc a such that $h_a > 1$. Any such route transformation removes one arc retraversal and instead, routes the truck along a sortie of the initial solution; the corresponding increases in the objective value are at most $(\alpha - 1) \cdot L$ because no sortie can be strictly longer than L. □

When the conditions of Proposition 5.4 are met and given an optimal solution to the TSP-mD with a retraversing vector h, we can conclude that the percentage increase in the completion time because of not allowing arc-retraversing solutions satisfies the following inequality:

$$\frac{\text{m-CIRCUIT} - \text{TSP-mD}}{\text{TSP-mD}} \leq \frac{\left(\sum_{a \in A} (h_a - 1)^+\right) \cdot (\alpha - 1) \cdot L}{\text{LB}}, \tag{13}$$

Figure 11. Illustration of the Set of Instances in the Proof of Proposition 5.3 Parametrized by $k \geq 1$



where LB is a lower bound for the TSP-mD, which can be easily precomputed as follows.

Proposition 5.5. *With a single drone and $N^{tr} = N$, let $f \in N$ be the farthest node from the depot. Then, the quantity*

$$\mathrm{LB} = \min\{\ell_{0i} + \ell_{0j} + \max\{\ell_{ij}, \ell'_{if} + \ell'_{fj}\} \ : \ i,j \in N \text{ and } (i,f,j) \in P\} \tag{14}$$

is a lower bound on the objective value for the TSP-mD.

Proof. The node f must be visited by either the truck or the drone; hence, the completion time must be greater than or equal to the minimum time it takes to visit f. If f is visited by the truck in an optimal solution, then such a time is $2\ell_{0f}$, which is included in (14) by choosing $i = j = f$. If f is instead visited by the drone, without loss of generality, we only have to consider the feasible sorties of the form (i,f,j) (with possibly $i = j = 0$ that leads to the drone visiting f directly from the depot). Indeed, sorties serving strictly more nodes cannot lead to a shorter time to visit f. Once a sortie of the form (i,f,j) is used, the truck needs to traverse the arcs $(0,i)$, (i, j), and $(j,0)$; the drone is airborne while the truck travels along (i, j), hence the quantity $\max\{\ell_{ij}, \ell'_{if} + \ell'_{fj}\}$ in (14). □

Propositions 5.4 and 5.5 provide an upper bound on the percentage increase of the completion time because of not allowing arc-retraversing solutions, with a single drone and $N^{tr} = N$. Consider, for example, the instance corresponding to Figures 4 and 5, which satisfies $N^{tr} \neq N$ and $\alpha = 1$. In Section 4.1, we have shown that, in this instance, the percentage increase of the completion time is at least $(1084.09 - 1050.36)/1050.36 = 3.21\%$. If we instead set $N^{tr} = N$ and $\alpha = \frac{4}{3}$, like for Figures 3 and 6, we can show that, by Inequality (13), the percentage increase is at most $(3 \cdot (1/3) \cdot 43.04)/(2 \cdot 228.09 + 410) = 4.97\%$. Indeed, because of the metric ℓ' and the parameter L, the truck must visit nodes F and I in any feasible solution. Consequently, the minimum amount of time it takes to visit F and I (namely, $2 \cdot 228.09 + 410$) is a lower bound on the objective value of the TSP-mD.

6. Approximability Results for the TSP-mD

Contrary to the well-known results for the metric TSP, we prove that it is not possible to approximate the TSP-mD within any constant factor (unless $P = NP$). For the special case where the truck is allowed to visit all the nodes, we identify a (nonconstant) approximation factor explicitly via a result that relies on the Christofides heuristic for the metric TSP.

6.1. Inapproximability Within Any Constant Factor

In this section, we prove that there is no constant-factor approximation for the TSP-mD (unless $P = NP$), even when either α or m is given and when the truck and the drone metrics are proportional (i.e., $\ell_a = \alpha \cdot \ell'_a$ for any arc $a \in A$ and $\alpha \geq 1$). The result is based on a reduction to the TSP-mD of the minimum set cover (MSC) problem, which is briefly defined.

Definition 6.1. Let X be a finite set and $\mathcal{S}$ be a finite collection of subsets of X such that $\cup_{U \in \mathcal{S}} U = X$. The MSC problem is the problem of finding

$$\min\{|\mathcal{S}'| \ : \ \mathcal{S}' \subseteq \mathcal{S} \text{ and } \cup_{U \in \mathcal{S}'} U = X\}. \tag{15}$$

Unless $P = NP$, there is no polynomial-time heuristic algorithm that approximates the MSC within a factor $c \cdot \ln(|X|)$, for any $c \in (0,1)$. In particular, the MSC cannot be approximated within any constant factor (Dinur and Steurer 2014).

Proposition 6.1. *Unless $P = NP$, there is no constant-factor approximation algorithm for TSP-mD, even when restricting to instances where $\ell_a = \alpha \cdot \ell'_a$ for all $a \in A$ and either α or m is bounded by a constant.*

Proof. The proof consists of five steps, which we enumerate as follows.

i. For every MSC instance $(X, \mathcal{S})$, $\alpha \geq 1$ and m, we can construct an instance $I^{X,\mathcal{S}}$ for the TSP-mD in polynomial time.

ii. For every m and every $\alpha \geq 1$, if there exists a polynomial-time heuristic for the TSP-mD, then there exists a polynomial-time heuristic for the MSC such that for any MSC instance $(X, \mathcal{S})$, it holds that

$$\mathrm{HEUR}_{\mathrm{MSC}}(X, \mathcal{S}) \leq \mathrm{HEUR}_{\text{TSP-mD}}(I^{X,\mathcal{S}}). \tag{16}$$

iii. For every MSC instance $(X, \mathcal{S})$ and for every $\alpha \geq 1$, there exists an m such that

$$\text{TSP-mD}(I^{X,\mathcal{S}}) \leq 2 \cdot \mathrm{MSC}(X, \mathcal{S}). \tag{17}$$

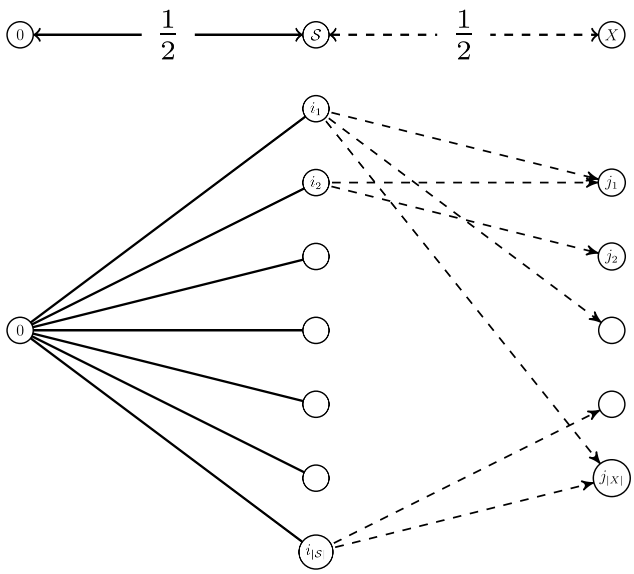
iv. For every MSC instance $(X, \mathcal{S})$, every $\epsilon > 0$, and every $m \geq 1$, there exists an $\alpha \geq 1$ such that

$$\text{TSP-mD}(I^{X,\mathcal{S}}) \leq (1 + \epsilon) \cdot \mathrm{MSC}(X, \mathcal{S}). \tag{18}$$

v. By contradiction, for both the cases where either $\alpha \geq 1$ or m is given, if there existed a polynomial-time heuristic that approximates the TSP-mD within a constant factor, then the same would apply to the MSC, contradicting its inapproximability properties.

Proof of i. Let $(X, \mathcal{S})$ be an instance for the MSC. We construct an instance $I^{X,\mathcal{S}}$ for the TSP-mD as follows. The set N of nodes consists of the depot 0, a node i_S for every $S \in \mathcal{S}$, and a node j_x for every $x \in X$. We set the length of the arc $\ell_{0i_S} = 1/2$ for every $S \in \mathcal{S}$ and $\ell_{i_S j_x} = 1/2$ for every $S \in \mathcal{S}$ and every $x \in S$. All the other arcs have a length equal to the shortest path between their end points. We set $N^{dr} = N^{tr} = N$, $w_i = 0$ for every $i \in N \setminus \{0\}$, and $L = 1/\alpha$. Figure 12 illustrates the construction of $I^{X,\mathcal{S}}$.

Figure 12. Illustration of the TSP-mD Instance $I^{(X,\mathcal{S})}$ in Step (i) of the Proof of Proposition 6.1



Proof of ii. Let $\text{HEUR}_{\text{TSP-mD}}$ be a polynomial-time heuristic for the TSP-mD. Given an MSC instance $(X,\mathcal{S})$, we construct a feasible MSC solution $\mathcal{S}'$ as follows. We compute $\text{HEUR}_{\text{TSP-mD}}(I^{X,\mathcal{S}})$. Then, for every $S\in\mathcal{S}$, we include S in $\mathcal{S}'$ if the corresponding node i_S is visited by the truck. For all $x\in X$, if j_x is visited by the truck, we arbitrarily choose a set $S\ni x$; if S was not yet included in $\mathcal{S}'$, then we include it in $\mathcal{S}'$. By construction, for all $x\in X$, we have included at least one set $S\in\mathcal{S}$ such that $x\in S$. Moreover, because the length of the arcs satisfies the triangle inequality, it holds that $|\mathcal{S}'|\le\text{HEUR}_{\text{TSP-mD}}(I^{X,\mathcal{S}})$. This construction defines a polynomial-time heuristic for the MSC with the required property.

Proof of iii. Let $\alpha\ge 1$ be given, and let $(X,\mathcal{S})$ be an instance for the MSC. We choose $m=|X|+|\mathcal{S}|$. Moreover, let $\mathcal{S}'\subseteq\mathcal{S}$ be an optimal subcollection for the MSC, and let $I^{X,\mathcal{S}}$ be the TSP-mD instance from step (i). We construct a feasible solution for the TSP-mD as follows. The truck travels $|\mathcal{S}'|$ times back and forth from the depot to the nodes i_S for all $S\in\mathcal{S}'$. Every time the truck reaches a node i_S, a drone is launched onto a sortie (i_S,j_x,i_S) for every $x\in S$ while the truck stops and waits. At the same time, for every set $S\in\mathcal{S}\setminus\mathcal{S}'$, we launch one drone onto sortie $(0,i_S,0)$. This solution is feasible and leads to an objective value of $(1+(1/\alpha))(\text{MSC}(X,\mathcal{S}))\le 2\cdot\text{MSC}(X,\mathcal{S})$.

Proof of iv. Let $m\ge1$ and $\epsilon>0$ be given, and let $(X,\mathcal{S})$ be an instance for the MSC. We now choose $\alpha=(|X|+|\mathcal{S}|)/\epsilon$. With a construction similar to that of step (iii), we get a feasible solution for the TSP-mD that uses only one drone and leads to an objective value $(|\mathcal{S}|-\text{MSC}(X,\mathcal{S}))/\alpha+(\text{MSC}(X,\mathcal{S})+|X|)/\alpha\le\epsilon+\text{MSC}(X,\mathcal{S})\le(1+\epsilon)\cdot\text{MSC}(X,\mathcal{S})$.

Proof of v. This step is analogous for both the cases where either $\alpha\ge1$ or m is given. If a polynomial-time heuristic for the TSP-mD and a constant $k>1$ existed such that $\text{HEUR}_{\text{TSP-mD}}\le k\text{TSP-mD}$ for all the instances, then the following chain of inequalities would hold by all the previous steps for any MSC instance $(X,\mathcal{S})$:

$$\begin{aligned}\text{HEUR}_{\text{MSC}}(X,\mathcal{S})&\le\text{HEUR}_{\text{TSP-mD}}(I^{X,\mathcal{S}})\le k\text{TSP-mD}(I^{X,\mathcal{S}})\\&\le 2k\text{MSC}(X,\mathcal{S}).\quad\square\end{aligned}\tag{19}$$

Notice that, because of the construction of the TSP-mD instance described in step (i) of the proof, Proposition 6.1 even holds when restricting the problem to instances with $N=N^{tr}=N^{dr}$. The same reduction of the MSC can be used to derive a further double-logarithmic factor inapproximability for the TSP-mD.

Corollary 6.1. *Unless P = NP, the TSP-mD is not approximable within a factor of $c\cdot\ln\ln(|N|)$ for any instance with N as a set of nodes and any $c\in\left(0,\frac{1}{2}\right)$.*

Proof. For every MSC instance $(X,\mathcal{S})$ with $|X|\ge2$, it holds that

$$\begin{aligned}&c\cdot\ln\ln(1+|X|+|\mathcal{S}|)\le c\cdot\ln\ln(2^{|X|^2})=c\cdot\ln(|X|^2\cdot\ln 2)\\&=c\cdot\ln(|X|^2)+c\cdot\ln\ln 2=2c\cdot\ln(|X|)\\&\quad+c\cdot\ln\ln 2\le 2c\cdot\ln(|X|).\end{aligned}\tag{20}$$

The first inequality sign follows from $|\mathcal{S}|\le2^{|X|}$ and $|X|+1\le2^{|X|}$ if $|X|\ge2$. Indeed, when $|X|\ge2$, it holds that $1+|X|+|\mathcal{S}|\le2^{1+|X|}\le2^{|X|^2}$. All the other signs of the chain (20) follow from standard logarithm properties and from $\ln\ln 2<0$.

Suppose that a polynomial-time heuristic for the TSP-mD existed that approximates it within a factor $c\cdot\ln\ln(|N|)$ for $c\in\left(0,\frac{1}{2}\right)$. Then, analogously to step (iv) in the proof of Proposition 6.1, it holds that

$$\begin{aligned}\text{HEUR}_{\text{MSC}}(X,\mathcal{S})&\le\text{HEUR}_{\text{TSP-mD}}(I^{X,\mathcal{S}})\\&\le c\cdot\ln\ln(1+|X|+|\mathcal{S}|)\text{TSP-mD}(I^{X,\mathcal{S}})\\&\le 2c\cdot\ln(|X|)\text{TSP-mD}(I^{X,\mathcal{S}})\\&\le 2c(1+\epsilon)\cdot\ln(|X|)\text{MSC}(X,\mathcal{S}),\end{aligned}\tag{21}$$

for every $\epsilon>0$, which contradicts the approximability properties of the MSC. $\square$

6.2. An Approximation Within a Nonconstant Factor

If the truck can visit all the nodes (i.e., $N^{\text{tr}}=N$), then it holds that $\text{TSP}\le(1+\alpha m)\text{TSP-mD}$. The proof can be found in Wang, Poikonen, and Golden (2017). This bound is tight (i.e., there exist instances for which the bound is satisfied to equality). Notice that when $N^{tr} = N$, a Hamiltonian cycle induces a feasible solution for the TSP-mD; this fact leads us to establish the following (nonconstant) approximation factor.

Proposition 6.2. *The TSP-mD is approximable within a* $\frac{3}{2}(1+\alpha m)$ *factor in polynomial time if the truck can visit all the nodes.*

The proof follows from the fact that the Christofides' heuristic is well known to produce a $\frac{3}{2}$ approximation cycle for the TSP. Notice that, by Proposition 6.1, we cannot replace α or m by a constant in this approximation factor.

7. Conclusion

In this work, we have introduced arc-retraversing solutions for the TSP-mD and showed via two Euclidean instances that ignoring them may increase the completion time. On the other hand, we have identified conditions, which are commonly met in the drone routing literature, under which it suffices to consider solutions that are not arc retraversing. By excluding arc retraversals when these conditions are not satisfied, the optimal value may increase; we have described cases where a priori and a posteriori upper bounds on such increases hold. Furthermore, we have shown that no polynomial-time heuristic algorithm exists that approximates the metric TSP-mD within a constant factor, unless $P = NP$. We have explicitly found a nonconstant approximation factor for the special case without restrictions on the truck route.

Interesting avenues for further work are the potential extension of the results of Section 4.2 to the multiple drones case and the possibility for the truck to visit a node $i \in N \setminus N^{tr}$ for launching or retrieving a drone. These extensions, however, come with their own sets of challenges. Indeed, the former does not appear to immediately follow from any results or ideas of this paper. The latter, albeit interesting for practical applications, leads to a fundamentally different problem than those that we here generalize because it introduces the possibility for a node not to be visited by the truck and the drones in a feasible solution. Potential future work might also include the development of a branch-and-price algorithm where the columns correspond to feasible drone operations and a classification of the instances in the literature where all the optimal solutions are arc retraversing.

Acknowledgments

The authors thank Bernard Fortz (Université Libre de Bruxelles), Ben Hermans (KU Leuven), and Ivana Ljubić (ESSEC Business School, Paris) for their valuable suggestions on a draft of this work.

Appendix A. Description of the Instances

An instance of the TSP-mD is defined by the distance matrices representing the metrics ℓ and ℓ', by the drone weight w^{dr} and the payload of the nodes, by the parameters m and B, and by the subsets of nodes N^{tr} and N^{dr}. If the metrics are proportional, ℓ' is completely determined by ℓ and the parameter α; if the payloads are null, it suffices to specify L instead of the parameters w^{dr} and B.

A.1. Instance of Figure 2

In Table A.1, we show the distance matrix corresponding to the metric ℓ in the instance of Figure 2. The instance is further defined by $m = 3$, proportional metrics with $\alpha = \frac{4}{3}$, $L = 40.00$, $N = N^{dr} = N^{tr}$, and all-zero payloads.

A.2. Instance of Figures 4 and 5

In Table A.2, we show the distance matrix corresponding to the metric ℓ of the instance shown in Figures 4 and 5. The instance is further defined by $m = 1$, proportional metrics with $\alpha = 1$, $L = 43.04$, $N^{tr} = \{A, F, G, H, I\}$, $N^{dr} = \{B, C, D, E, J, K, L, M, N, O\}$, and all-zero payloads.

A.3. Instance of Figures 3 and 6

In Table A.3, we show the distance matrix corresponding to the metric ℓ of the instance shown in Figures 3 and 6. The instance is further defined by $m = 2$, proportional metrics with $\alpha = \frac{4}{3}$, $L = 21.00$, $N = N^{dr} = N^{tr}$, and all-zero payloads.

Appendix B. MILP Model

The proofs of Propositions 4.3 and 4.4 are complete if we provide an MILP model for the TSP-mD; we propose one based on the formulation for the time-dependent traveling

Table A.1. Matrix Representing the Metric ℓ of the Instance Shown in Figure 2

ℓ	A	B	C	D	E	F	G	H	I	J
A	0	13.89	21.02	32.56	17.2	14.14	11.4	26.42	22.02	23.09
B	13.89	0	12.37	19.21	31.06	22.2	16.76	22.83	11.66	24.21
C	21.02	12.37	0	15.3	37.01	21.02	28.07	34.93	22.2	16.28
D	32.56	19.21	15.3	0	49.68	36.06	35.36	35.01	21.1	31
E	17.2	31.06	37.01	49.68	0	20.4	21.02	37.12	37.64	32.76
F	14.14	22.2	21.02	36.06	20.4	0	25.5	40.22	33.24	12.37
G	11.4	16.76	28.07	35.36	21.02	25.5	0	16.49	18.03	34.01
H	26.42	22.83	34.93	35.01	37.12	40.22	16.49	0	14.04	46.1
I	22.02	11.66	22.2	21.1	37.64	33.24	18.03	14.04	0	35.81
J	23.09	24.21	16.28	31	32.76	12.37	34.01	46.1	35.81	0

Table A.2. Matrix Representing the Metric ℓ of the Instance Shown in Figures 4 and 5

ℓ	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	0	79.15	79.15	84.18	84.18	228.09	100.13	100.13	228.09	109.01	109.01	116.13	116.13	120.85	120.85
B	79.15	0	0.2	7.26	7.41	205.96	21.42	21.46	206.16	29.87	29.88	37.24	37.27	41.7	41.7
C	79.15	0.2	0	7.41	7.26	206.16	21.46	21.42	205.96	29.88	29.87	37.27	37.24	41.7	41.7
D	84.18	7.26	7.41	0	11	200.14	16.01	19.14	211.11	25.35	25.91	32	33.84	37.24	37.27
E	84.18	7.41	7.26	11	0	211.11	19.14	16.01	200.14	25.91	25.35	33.84	32	37.27	37.24
F	228.09	205.96	206.16	200.14	211.11	0	200	210	410	203.9	206.5	200.14	211.11	205.96	206.16
G	100.13	21.42	21.46	16.01	19.14	200	0	10	210	9.73	10.99	16.01	19.14	21.42	21.46
H	100.13	21.46	21.42	19.14	16.01	210	10	0	200	10.99	9.73	19.14	16.01	21.46	21.42
I	228.09	206.16	205.96	211.11	200.14	410	210	200	0	206.5	203.9	211.11	200.14	206.16	205.96
J	109.01	29.87	29.88	25.35	25.91	203.9	9.73	10.99	206.5	0	2.6	8.16	9.76	11.91	11.93
K	109.01	29.88	29.87	25.91	25.35	206.5	10.99	9.73	203.9	2.6	0	9.76	8.16	11.93	11.91
L	116.13	37.24	37.27	32	33.84	200.14	16.01	19.14	211.11	8.16	9.76	0	11	7.26	7.41
M	116.13	37.27	37.24	33.84	32	211.11	19.14	16.01	200.14	9.76	8.16	11	0	7.41	7.26
N	120.85	41.7	41.7	37.24	37.27	205.96	21.42	21.46	206.16	11.91	11.93	7.26	7.41	0	0.2
O	120.85	41.7	41.7	37.27	37.24	206.16	21.46	21.42	205.96	11.93	11.91	7.41	7.26	0.2	0

salesman problem (TDTSP) by Picard and Queyranne (1978). In the TDTSP, the cost of using an arc depends on its position in the route. Similarly, we describe the truck route π of a TSP-mD solution as a sequence of (possibly repeated) arcs in $N \times N = A \cup \{(i,i) : i \in N\}$, where the arcs of the form $\{(i,i) : i \in N\}$ represent the truck waiting at a node $i \in N$ for a drone. Furthermore, we describe a sortie as a sequence of nodes and a couple of positive numbers $t_1 \le t_2$; the nodes of the sortie represent (in the order) the starting, served, and ending locations, whereas t_1 and t_2 correspond to the positions in the truck route of the arcs that are traversed immediately after launching and right before retrieving the relevant drone, respectively. We refer to t_1 and t_2 as starting and ending positions of the sortie, respectively. For example, in the solution shown in Figure 2, the sortie of drone 2 starts in position 2 and ends in position 3 because the truck traverses the second arc of its route immediately after launching and the third arc right before retrieving the drone. Analogously, the sortie of drone 3 starts in position 1 and ends in position 4.

We prove that we can solve the TSP-mD by only considering truck routes that contain at most $2 \cdot |N|$ arcs.

Proposition B.1. *There always exists an optimal solution whose truck route contains at most* $2 \cdot |N|$ *arcs in* $N \times N$ *and such that no two solution sorties have the same ending position.*

Proof. Consider an optimal solution, and call its truck route π. We want to construct an optimal solution (possibly the same) such that no two solution sorties have the same ending position. This can be achieved, for any visited node $i \in N$ where multiple sorties end, by inserting arcs of the form $(i,i) \in N \times N \setminus A$ as many times as needed so as to let the relevant drones land at i with distinct ending positions. Call the latter optimal truck route π' (which is possibly π itself if no two solution sorties ended at the same positions in the first place).

If π' contains at most $2 \cdot |N|$ arcs, then there is nothing to prove. We suppose then that π' contains strictly more than $2 \cdot |N|$ arcs. We claim that π' can always be shortcut at revisited nodes in such a way that the resulting truck route π'' contains at most $2 \cdot |N|$ arcs and no two solution sorties have the same ending position. Indeed, suppose by contradiction that such a shortcut route π'' does not exist. For every arc $(i,j) \in N \times N$ in π', if node j was not previously visited, then the truck traversal of (i, j) contributes by visiting at least one new node. If instead, node j was already visited, then by Lemma 4.1, we can assume that there is at least one drone that either lands or takes off at j immediately after the truck has traversed (i, j). Thus, the truck traversal of (i, j) contributes half a new node to the visits on average because every sortie has two end points and serves at least one node. However, π' contains strictly more than $2 \cdot |N|$ arcs, which would imply that its arcs in total contribute by visiting strictly more than $|N|$ nodes. □

Requiring that no two solution sorties have the same ending position is crucial to simplify our MILP formulation. We denote the position of an arc in a truck route by $t \in \{1, \dots, T\}$, with $T = 2 \cdot |N|$. We define a *drone operation* h as a four-tuple (d, π, t^1, t^2) made by a drone $d \in \{1, \dots, m\}$, a feasible sortie $\pi \in P$, a starting position $t^1 \in \{1, \dots, T\}$, and an ending position $t^2 \in \{t^1, \dots, T\}$. The set of all the drone operations is denoted by H.

Our decision variables can be described as follows. For every arc $a \in N \times N$ and for every position $t \in \{1, \dots, T\}$, the binary variable x_{at} is one if and only if arc a is in position t in the truck route. For every position $t \in \{1, \dots, T\}$, the continuous variable $w_t \ge 0$ represents the amount of time the truck waits for a drone at the destination node of the arc in position t. For every drone operation $h \in H$, the binary variable z_h is one if and only if h is performed in the solution. We

Table A.3. Matrix Representing the Metric ℓ of the Instance Shown in Figures 3 and 6

ℓ	A	B	C	D	E
A	0	10.00	10.00	24.00	24.00
B	10.00	0	14.14	14.00	26.00
C	10.00	14.14	0	26.00	14.00
D	24.00	14.00	26.00	0	33.94
E	24.00	26.00	14.00	33.94	0

assume that $N = N^{dr} = N^{tr}$; if not, it is easy to impose the following constraints:

$$\sum_{t\in\{1,\ldots,T\}}\sum_{j\in N\setminus\{i\}} x_{ijt} \le 0 \qquad \forall i \in N \setminus N^{tr}, \tag{B.1}$$

$$\sum_{t\in\{1,\ldots,T\}}\sum_{i\in N\setminus\{j\}} x_{ijt} \ge 1 \qquad \forall j \in N \setminus N^{dr}. \tag{B.2}$$

For the sake of conciseness, we slightly abuse the notation of a number of subsets of H; this never gives rise to ambiguities because we always denote drones and starting and ending positions by the symbols d, t_1, and t_2, respectively. Table B.1 provides a description of every such subset of H:

$$\min \sum_{t\in\{1,\ldots,T\}}\left(w_t + \sum_{a\in A}\ell_a x_{at}\right) \tag{B.3}$$

s.t.

$$\sum_{a\in N\times N} x_{at} \le 1 \qquad \forall t \in \{1,\ldots,T\} \tag{B.4}$$

$$\sum_{j\in N} x_{ij1} = \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in N \tag{B.5}$$

$$\sum_{j\in N} x_{ijt} \le \sum_{j\in N} x_{ji,t-1} \qquad \forall i \in N, \quad \forall t \in \{2,\ldots,T\} \tag{B.6}$$

$$\sum_{i\in N}\sum_{j\in N\setminus\{0\}} x_{ijt} \le \begin{cases} \sum_{a\in N\times N} x_{a,t+1} & \text{if } t \in \{1,\ldots,T-1\} \\ 0 & \text{otherwise} \end{cases} \qquad \forall t \in \{1,\ldots,T\} \tag{B.7}$$

$$\sum_{t^1\in\{1,\ldots,t\}}\sum_{t^2\in\{t,\ldots,T\}}\sum_{h\in H_{d,t^1t^2}} z_h \le \sum_{a\in N\times N} x_{at} \qquad \forall d\in D,\ \forall t\in\{1,\ldots,T\} \tag{B.8}$$

$$\sum_{h\in H_{t^1\cdot,i\cdot}} z_h \le m\cdot\sum_{j\in N} x_{ijt^1} \qquad \forall i\in N,\ \forall t^1\in\{1,\ldots,T\} \tag{B.9}$$

$$\sum_{h\in H_{\cdot t^2,\cdot j}} z_h \le \sum_{i\in N} x_{ijt^2} \qquad \forall j\in N,\ \forall t^2\in\{1,\ldots,T\} \tag{B.10}$$

$$\sum_{t\in\{1,\ldots,T\}}\sum_{j\in N} x_{kjt} + \sum_{h\in H^k} z_h \ge 1 \qquad \forall k\in N \tag{B.11}$$

$$\sum_{t\in\{t^1,\ldots,t^2\}}\left(w_t + \sum_{a\in A}\ell_a x_{at}\right) \ge \sum_{h\in H_{t^1t^2}} \ell'_h z_h \qquad \forall t^1\in\{1,\ldots,T\},\ \forall t^2\in\{t^1,\ldots,T\} \tag{B.12}$$

$$x_{at}\in\{0,1\} \qquad \forall a\in A,\ \forall\{1,\ldots,T\} \tag{B.13}$$

$$z_h\in\{0,1\} \qquad \forall h\in H \tag{B.14}$$

$$w_t\ge 0 \qquad \forall t\in\{1,\ldots,T\}. \tag{B.15}$$

The objective function (B.3) quantifies the completion time. Constraints (B.4) impose that at most one arc in $N\times N$ can be in position $t\in\{1,\ldots,T\}$ in the truck route. Constraints (B.5) force that, in position 1 of the truck route, the depot has one outgoing arc, whereas all the other nodes $i\in N\setminus\{0\}$ do not have any. For all the positions $t\in\{2,\ldots,T\}$, by Constraints (B.6), the total outgoing flow of the x variables from a given node $i\in N$ in t is smaller than or equal to the total inflow into i in position $t-1$. For any position $t\in\{1,\ldots,T-1\}$, if the truck route does not have any arc in position $t+1$, then by Constraints (B.7), in position t either the truck does not move, or it travels directly to the depot. For every position $t\in\{1,\ldots,T\}$ and for every drone $d\in D$, the left-hand side of Constraints (B.8) quantifies the number of drone operations by drone d that are being performed, whereas the truck traverses the tth arc of its route (i.e., with a starting position up to t and an ending position greater than or equal to t). If the truck route contains strictly less than t arcs, then Constraints (B.8) correctly set the number of these drone operations to zero. Otherwise, the number of these drone operations is correctly upper bounded to one. Constraints (B.9) and (B.10) force a sortie to start and end where and when the truck is located. Indeed, their left-hand sides quantify the number of drone operations that start at a given node i and position t^1 and end at a given node j and position t^2, respectively. These quantities are nonzero only if an arc outgoing from i or going into j, respectively, are in positions t^1 and t^2 in the truck route. Up to m drones can take off from a given node, and Constraints (B.9) incorporate this property via the factor m in their right-hand side. We can, however, drop this multiplicative factor in Constraints (B.10) because by Proposition B.1, there is always an optimal solution such that no two drone operations end in the same position. Furthermore, dropping this multiplicative factor simplifies Constraints (B.12), whose right-hand side can now be expressed as a summation over the drone operation index h. Constraints (B.11) make sure that all the nodes are visited by either the truck or a drone. Finally, Constraints (B.12) synchronize the truck and the drones by giving to the variables w their meaning as per definition, with ℓ'_h denoting the length of the sortie in operation $h\in H$.

This model was implemented in Gurobi 9 exclusively for the sake of proving the results in Section 4.1. The task

Table B.1. Description of the Subsets of H That Are Relevant to Our Formulation of the TSP-mD

Subsets of H	Definition
H^k	Set of drone operations in H whose sorties are in P^k with $k\in N^{dr}$
$H_{t^1\cdot}$	Set of drone operations in H that start at period t^1
$H_{\cdot t^2}$	Set of drone operations in H that end at period t^2
$H_{t^1t^2}$	Intersection of sets $H_{t^1\cdot}$ and $H_{\cdot t^2}$
H_{d,t^1t^2}	Set of drone operations in $H_{t^1t^2}$ performed by drone d
$H_{t^1\cdot,i\cdot}$	Set of drone operations in $H_{t^1\cdot}$ whose sorties start at node $i\in N^{tr}$
$H_{\cdot t^2,\cdot j}$	Set of drone operations in $H_{\cdot t^2}$ whose sorties end at node $j\in N^{tr}$

required the solution of a small number of instances (see Section 4.1), which we were able to solve in at most three hours of CPU time for each instance on a regular laptop computer. We did not investigate the computational effectiveness of this solution method further. We believe, however, that branch-and-price methods for this problem could prove very effective and deserve to be the subject of future investigation.

Endnote

[1] There are numerous similar problem variants studied in literature under the name FSTSP or traveling salesman problem with a drone. The generic setting we outlined here under the name FSTSP is equivalent to the one studied by Agatz, Bouman, and Schmidt (2018) and Roberti and Ruthmair (2021).

References

Agatz N, Bouman P, Schmidt M (2018) Optimization approaches for the traveling salesman problem with drone. *Transportation Sci.* 52(4):965–981.

Bakir I, Tiniç GÖ (2020) Optimizing drone-assisted last-mile deliveries: The vehicle routing problem with flexible drones. *Optimization Online*. Accessed February 5, 2023, http://www.optimization-online.org/DB_FILE/2020/04/7737.pdf.

Boccia M, Masone A, Sforza A, Sterle C (2021a) A column-and-row generation approach for the flying sidekick travelling salesman problem. *Transportation Res. Part C Emerging Tech.* 124:102913.

Boccia M, Masone A, Sforza A, Sterle C (2021b) An exact approach for a variant of the FS-TSP. *Transportation Res. Procedia* 52:51–58.

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

Boysen N, Fedtke S, Schwerdfeger S (2021) Last-mile delivery concepts: A survey from an operational research perspective. *OR Spectrum* 43(1):1–58.

Cavani S, Iori M, Roberti R (2021) Exact methods for the traveling salesman problem with multiple drones. *Transportation Res. Part C Emerging Tech.* 130:103280.

Chung SH, Sah B, Lee J (2020) Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions. *Comput. Oper. Res.* 123:105004.

CNBC News (2020) Amazon wins FAA approval for Prime Air drone delivery fleet. (August 31), https://www.cnbc.com/2020/08/31/amazon-prime-now-drone-delivery-fleet-gets-faa-approval.html.

Coutinho WP, Battarra M, Fliege J (2018) The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review. *Comput. Indust. Engrg.* 120:116–128.

Dell'Amico M, Montemanni R, Novellani S (2021a) Algorithms based on branch and bound for the flying sidekick traveling salesman problem. *Omega* 104:102493.

Dell'Amico M, Montemanni R, Novellani S (2021b) Drone-assisted deliveries: New formulations for the flying sidekick traveling salesman problem. *Optim. Lett.* 15(5):1617–1648.

Dell'Amico M, Montemanni R, Novellani S (2021c) Modeling the flying sidekick traveling salesman problem with multiple drones. *Networks* 78(3):303–327.

Dell'Amico M, Montemanni R, Novellani S (2022) Exact models for the flying sidekick traveling salesman problem. *Internat. Trans. Oper. Res.* 29(3):1360–1393.

Ding Y, Xin B, Chen J (2021) A review of recent advances in coordination between unmanned aerial and ground vehicles. *Unmanned Systems* 9(02):97–117.

Dinur I, Steurer D (2014) Analytical approach to parallel repetition. *Proc. 46th Annual ACM Sympos. Theory Comput. (STOC '14)* (Association for Computing Machinery, New York), 624–633.

El-Adle AM, Ghoniem A, Haouari M (2021) Parcel delivery by vehicle and drone. *J. Oper. Res. Soc.* 72(2):398–416.

European Council (2019) Drones: Reform of EU aviation safety. Accessed February 5, 2023, https://www.consilium.europa.eu/en/policies/drones/.

Freitas JC, Penna PHV, Toffolo TA (2021) Exact and heuristic approaches to drone delivery problems. Preprint, submitted July 29, https://arxiv.org/abs/2108.01996.

González-R PL, Canca D, Andrade-Pineda JL, Calle M, León-Blanco JM (2020) Truck-drone team logistics: A heuristic approach to multi-drop route planning. *Transportation Res. Part C Emerging Tech.* 114:657–680.

Ha QM, Vu DM, Le XT, Hoang MH (2021) The traveling salesman problem with multi-visit drone. *J. Comput. Sci. Cybernetics* 37(4):465–493.

Jeon A, Kang J, Choi B, Kim N, Eun J, Cheong T (2021) Unmanned aerial vehicle last-mile delivery considering backhauls. *IEEE Access* 9:85017–85033.

Jeong HY, Lee S (2019) Optimization of vehicle-carrier routing: Mathematical model and comparison with related routing models. *Procedia Manufacturing* 39:307–313.

Jeong HY, Song BD, Lee S (2019) Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones. *Internat. J. Production Econom.* 214:220–233.

Khoufi I, Laouiti A, Adjih C (2019) A survey of recent extended variants of the traveling salesman and vehicle routing problems for unmanned aerial vehicles. *Drones* 3(3):66.

Kitjacharoenchai P, Ventresca M, Moshref-Javadi M, Lee S, Tanchoco JM, Brunese PA (2019) Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Comput. Indust. Engrg.* 129:14–30.

Li H, Chen J, Wang F, Bai M (2021) Ground-vehicle and unmanned-aerial-vehicle routing problems from two-echelon scheme perspective: A review. *Eur. J. Oper. Res.* 294(3):1078–1095.

Luo H, Zhang P, Wang J, Wang G, Meng F (2019) Traffic patrolling routing problem with drones in an urban road system. *Sensors* 19(23):5164.

Luo Z, Poon M, Zhang Z, Liu Z, Lim A (2021) The multi-visit traveling salesman problem with multi-drones. *Transportation Res. Part C Emerging Tech.* 128:103172.

Macrina G, Pugliese LDP, Guerriero F (2020) The green-vehicle routing problem: A survey. Derbel H, Jarboui B, Siarry P, eds. *Modeling and Optimization in Green Logistics* (Springer International Publishing, Cham, Switzerland), 1–26.

Macrina G, Pugliese LDP, Guerriero F, Laporte G (2020) Drone-aided routing: A literature review. *Transportation Res. Part C Emerging Tech.* 120:102762.

Merkert R, Bushell J (2020) Managing the drone revolution: A systematic literature review into the current use of airborne drones and future strategic directions for their effective control. *J. Air Transportation Management* 89:101929.

Moshref-Javadi M, Winkenbach M (2021) Applications and research avenues for drone-based models in logistics: A classification and review. *Expert Systems Appl.* 177:114854.

Murray CC, Chu AG (2015) The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Res. Part C Emerging Tech.* 54:86–109.

Murray CC, Raj R (2020) The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. *Transportation Res. Part C Emerging Tech.* 110:368–398.

Otto A, Agatz N, Campbell J, Golden B, Pesch E (2018) Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks* 72(4):411–458.

Persson E (2021) A systematic literature review on drones' application in last-mile delivery. Accessed February 5, 2023, http://urn.kb.se/resolve?urn=urn:nbn:se:hig:diva-36733.

Picard JC, Queyranne M (1978) The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling. *Oper. Res.* 26(1):86–110.

Poikonen S, Campbell JF (2021) Future directions in drone routing research. *Networks* 77(1):116–126.

Roberti R, Ruthmair M (2021) Exact methods for the traveling salesman problem with drone. *Transportation Sci.* 55(2):315–335.

Roca-Riu M, Menendez M (2019) Logistic deliveries with drones: State of the art of practice and research. *19th Swiss Transport Res. Conf.* (STRC, Ascona, Switzerland). https://www.research-collection.ethz.ch/handle/20.500.11850/342823.

Rojas Viloria D, Solano-Charris EL, Muñoz-Villamizar A, Montoya-Torres JR (2021) Unmanned aerial vehicles/drones in vehicle routing problems: A literature review. *Internat. Trans. Oper. Res.* 28(4):1626–1657.

Schermer D, Moeini M, Wendt O (2020) A branch-and-cut approach and alternative formulations for the traveling salesman problem with drone. *Networks* 76(2):164–186.

Seifried K (2019) The traveling salesman problem with one truck and multiple drones. Preprint, submitted May 1, http://dx.doi.org/10.2139/ssrn.3389306.

Tamke F, Buscher U (2021) A branch-and-cut algorithm for the vehicle routing problem with drones. *Transportation Res. Part B Methodological* 144:174–203.

Tang Z, van Hoeve WJ, Shaw P (2019) A study on the traveling salesman problem with a drone. Rousseau LM, Stergiou K, eds. *Integration of Constraint Programming, Artificial Intelligence, and Operations Research* (Springer International Publishing, Cham, Switzerland), 557–564.

The Washington Post (2021) German start-up creates a delivery drone capable of toting three separate packages. (April 28), https://www.washingtonpost.com/technology/2021/04/28/drone-delivery-triple-drop/.

Vásquez SA, Angulo G, Klapp MA (2021) An exact solution method for the TSP with drone based on decomposition. *Comput. Oper. Res.* 127:105127.

Wang X, Poikonen S, Golden B (2017) The vehicle routing problem with drones: Several worst-case results. *Optim. Lett.* 11(4):679–697.

Yurek EE, Ozmutlu HC (2018) A decomposition-based iterative optimization algorithm for traveling salesman problem with drone. *Transportation Res. Part C Emerging Tech.* 91:249–262.

Zhou H, Qin H, Cheng C, Rousseau LM (2022) A branch-and-price algorithm for the vehicle routing problem with drones. *Optimization Online*. Accessed July 6, 2022, http://www.optimization-Online.org/DB_FILE/2022/02/8801.pdf.

Zhu T, Boyles SD, Unnikrishnan A (2022) Electric vehicle traveling salesman problem with drone. Preprint, submitted April 29, https://arxiv.org/abs/2201.07992.