Routing of Carrier-Vehicle Systems with Dedicated Last-Stretch Delivery Vehicle *

Mohd Shahrizan bin Othman[1,2] Aleksandar Shurbevski[1]
Hiroshi Nagamochi[1]

[1]Department of Applied Mathematics and Physics, Kyoto University
[2]Technical University of Malaysia Malacca
{shah, shurbevski, nag}@amp.i.kyoto-u.ac.jp

Abstract

We examine a routing problem arising when an unmanned aerial vehicle (UAV), or *drone*, is used in the last-stretch of parcel delivery to an end customer. In the scenario that we study, a delivery truck is dispatched carrying a shipment of parcels to be delivered to customers, and it is required to end its route at a predetermined location, which is not necessarily the same as the starting location. A drone is charged with making the last-stretch delivery of a parcel from the truck to a customer's doorstep. Given a set of customers to be served, and a set of rendezvous points where the drone can meet with the truck to pick up a parcel, we ask to determine a route for the truck and an assignment for the drone to deliver parcels between rendezvous points and customers, such that all parcels are delivered to end customers in the minimum amount of time. We model this problem as a problem of finding a special type of a path in a graph. We introduce two problem models: the No-Wait Transit Last-Stretch Delivery Problem (NW-TLSDP), and the Transit Last-Stretch Delivery Problem (TLSDP). Both of these graph problems are NP-hard, and we propose polynomial time approximation algorithms for each of the problem settings. We show that in metric graphs, there is a 2.6-approximation algorithm for the NW-TLSDP, and a 2.5-approximation algorithm for the special case when the given terminating location for the truck is the same as the starting one. Further, we show a 1.6-approximation algorithm for the TLSDP in a metric graph, and a 1.5-approximation algorithm for the special case of identical starting and ending locations of the truck.

1 Introduction

For over a decade, Unmanned Aerial Vehicles (UAV-s), or widely known as *drones*, have been utilized for many purposes, such as in military applications, command, control and communications (3C) [8], remote sensing and scientific research [19], and in precision agriculture [20]. Parcel delivery [3, 4, 18] is yet another application of drones that attracts a great deal of attention with announcements for using drones. With the expectation that drones could improve the competency of delivery operations, major

*Technical Report 2017-001, April 26, 2017.

delivery and logistics companies have unveiled investigations as they test the feasibility and profitability of unmanned delivery drone services.

In this study, we undertake a scenario in which a drone and a delivery truck operate in tandem to perform the *last-stretch* delivery of parcels to customers' doorsteps. The truck is required to start from a distribution center carrying a drone and parcels for a set C of customers, and may either be required to end its route at the same, or some other distribution center. The last-stretch deliveries of parcels from the truck to a customer's doorstep are performed exclusively by the drone. The drone has a payload capacity of at most one parcel, and hence must return to the truck after each delivery. The drone must return to the truck at the end of the truck's route. Moreover, the drone can only rendezvous with the truck at a set R of points along the truck's route. The set R is distinct from the set C. While the drone is making a delivery of a parcel, the truck may either wait for it to return at the last rendezvous point, or proceeds along its route and will intercept the drone at some future rendezvous point in the set R. Our objective is to determine a routing policy for both the truck and the drone, such that all parcels are delivered in the least amount of time.

With a motivation of cooperatively routing heterogeneous vehicles with different capabilities, a similar routing problem was studied by Garone et al. [6], who named the scenario *carrier-vehicle systems*.

A closely related model to our problem appears in the work of Mathew et al. [12]. In their study, they investigate a combination of a drone and a truck to perform parcel deliveries. The drone is used to perform last-stretch deliveries, and deliveries are performed solely by the drone, which delivers parcels between the truck and a customer's doorstep. The truck in turn is routed along a street network. In addition, they also examined a special case of their problem where the drone's route alternately visits customers and points of a fixed set of depots. However, for this special case, the authors do not give a comment on the computational complexity, or the approximability of the problem.

Othman et al. [14] investigated a similar scenario as Mathew et al. [12], where a drone is used in tandem with a delivery truck for the last-stretch delivery. In their problem setting, it is assumed that a route for the truck is predetermined. The authors introduced four problem models as problems of finding a minimum-cost path of a special structure in a graph, and showed that all of those problems are NP-hard. They also identified a special type of instance that can be solved in polynomial time. They proposed a polynomial-time approximation algorithm for the graph problem, and proved that the approximation ratio is bounded above by 2 in restricted metric graphs.

Apart from the application of the drone as a dedicated delivery vehicle, there are several works in the literature that consider a setting where the last-stretch delivery can be performed by either the truck or the drone, see, e.g., Agatz et al. [1], Murray and Chu [13], Ha et al. [9], and Ponza [15].

We examine two particular cases of the problem arising in the scenario outlined above. In the first setting, while the drone is making a delivery of a parcel, the truck is not allowed to wait for the drone to return at the last rendezvous point. In particular, the truck can only intercept the drone at any rendezvous point at most once. In contrast, for the second setting, the truck is allowed to wait for the drone to return at the last rendezvous point, or re-visit some rendezvous point multiple times. We term the former the No-Wait Transit Last-Stretch Delivery Problem, or NW-TLSDP for short. We call the latter problem the Transit Last-Stretch Delivery Problem, or TLSDP.

With this work, we propose graph problem models for the NW-TLSDP and the TLSDP. Both of the graph problems are NP-hard. Next, we propose polynomial time approximation algorithms for each of the problem settings. We show that in metric graphs, there is a 2.6-approximation algorithm for the NW-TLSDP, and a 2.5-approximation algorithm for a special instance of this problem where the truck has to return to the same distribution center. Finally, we show that in metric graphs, there is a 1.6-approximation algorithm for the TLSDP, and when the truck has to return to the same distribution center, we show a 1.5-approximation algorithm.

The rest of this paper is organized as follows. Section 2 outlines the basic notation, the problem models for this paper, and the NP-hardness of our problem models. Section 3 describes the approximation algorithm for our problem models, and finally, Sect. 4 concludes the paper.

2 Preliminaries

2.1 Notation

The set of reals (resp., nonnegative reals) is denoted by $\mathbb{R}$ (resp., $\mathbb{R}_+$).

The vertex set and the edge set of a graph G are denoted by $V(G)$ and $E(G)$, respectively. For vertices $u, v \in V(G)$, we use uv to refer to an edge $e \in E(G)$ such that e is incident to u and v. Note that graphs in this paper are undirected, i.e., $uv = vu$. We call u and v the *end vertices* of the edge uv. For a set $E' \subseteq E(G)$ of edges, we write $V(E')$ for the set of all end vertices of edges in E'. A graph G is *complete* if every two vertices $u, v \in V(G)$ are adjacent. The *degree* of a vertex $u \in V(G)$ in a graph G is the number of edges in $E(G)$ incident to u.

A *subgraph* G' of a graph G is a graph such that $V(G') \subseteq V(G)$, and $E(G') \subseteq E(G)$, and we write $G' \subseteq G$. A graph $G' \subseteq G$ is an *induced* subgraph of G if it holds that $E(G') = \binom{V(G')}{2} \cap E(G)$, and we also say that G' is induced by $V(G')$. Given a graph G and a set $V' \subseteq V(G)$, we write $G[V']$ for the subgraph of G induced by V'. In addition, for a subset $V'' \subseteq V(G)$, we write $G - V''$ for the graph $G[V(G) - V'']$.

Given a graph G, a *matching* $M \subseteq E(G)$ is a subset of edges such that each vertex in $V(G)$ is incident to at most one edge in M. A matching $M \subseteq E(G)$ is *perfect* if it holds that $V(M) = V(G)$. Given a graph G and a matching $M \subseteq E(G)$, a vertex $v \in V(G)$ such that $v \notin V(M)$ is called *exposed*. Given two sets $A, B \subseteq V(G)$, we call a matching $M \subseteq E(G)$ an *A, B-matching* if all edges $e \in M$ are of the form $e = ab$, $a \in A$ and $b \in B$.

A *path* $P = (v_1, v_2, \ldots, v_p)$ is a graph such that $V(P) = \{v_1, v_2, \ldots, v_p\}$ and $E(P) = \{v_i v_i + 1 \mid i = 1, 2, \ldots, p-1\}$. Such a graph P is also called a v_1, v_p-path. A path $P = (v_1, v_2, \ldots, v_p)$ such that $v_i \neq v_j$ for $1 \leq i \neq j \leq p$ is called a *simple path*. Given two sets $A, B \subseteq V(G)$, we call a path $P \subseteq G$ an *A, B-alternating path* if P is of the form $P = (a_1, b_1, a_2, b_2, \ldots)$, $a_i \in A$, $b_i \in B$.

A *cycle* $C = (v_1, v_2, \ldots, v_p)$ is defined to be a path $(v_1, v_2, \ldots, v_p)$ such that $p \geq 3$ and $v_1 = v_p$. If $v_i \neq v_j$ for all $1 \leq i \neq j < p$, then the cycle C is called a *simple cycle*.

Given a graph G and an *edge weight* function $w : E(G) \to \mathbb{R}_+$, we say that the graph G is *weighted* by w, and write (G, w). For convenience, for any vertex $v \in V(G)$, we define that $w(vv) = 0$. For a subset $E' \subseteq E(G)$, let $w(E')$ denote $\sum_{e \in E'} w(e)$. For brevity, let $w(G)$ denote $w(E(G))$.

A weighted graph (G, w) is called *metric* if the edge weight function w satisfies the *triangle inequality*, that is, for all $u, v, q \in V(G)$ it holds that

$$w(uv) \leq w(uq) + w(qv). \tag{1}$$

2.2 Problem Models

Let C be the set of customers to which parcels need to be delivered, and let R be the set of points at which the delivery drone can rendezvous with the truck. In general, the truck is required to start from a point $s \in R$, and end its route at $t \in R$. We call the points s and t *depots*. If $s = t$, then the truck starts and returns to the same point s, and we have a single depot. Let $|R| = m$ and $|C| = n$. The drone has unit payload capacity, and it never delivers two parcels consecutively without rendezvousing with the truck. With this observation, we introduce the following distance functions:

- $d(u, v)$: the time it takes for the drone to travel between a point $u \in R$ and a customer $v \in C$; and
- $t(u, v)$: the time it takes for the truck to move from a point $u \in R$ to a point $v \in R$ along its route.

We assume that for all $u, v \in R$ and any $q \in C$, it holds that

$$t(u, v) \leq d(u, q) + d(q, v), \tag{2}$$

which is a natural assumption that the drone cannot take a shortcut between rendezvous points by visiting a customer.

An illustration of the problem scenario when the truck is required to return to the same depot, and it can wait for the drone to deliver a parcel at the last rendezvous point is depicted in Fig. 1(a), while an illustration of a feasible solution of the given problem scenario is depicted in Fig. 1(b).

Let (G, w) be a weighted graph such that $R, C \subseteq V(G)$ and for the weight function w restricted to the edge set $\binom{R \cup C}{2} - \binom{C}{2}$ it holds that

$$w(uv) = \begin{cases} d(u, v), & \text{for } u \in C \text{ and } v \in R, \\ t(u, v), & \text{for } u, v \in R. \end{cases} \tag{3}$$

With the observation of Eq. (2), given depots $s, t \in R$, in order to find a delivery route for both the truck and the drone, we need to find an s, t-path P in the graph G such that it holds that $C \subseteq V(P)$ and no two vertices in C appear consecutively in the path P. The total time needed for the drone and the truck to deliver all parcels and arrive to the depot t will be given by $w(P)$. Then, by the request that the truck does not rendezvous with the drone more than once at any point in R, such a path P in the graph G must be a simple path, whereas it need not be simple if repeated rendezvouses at a single point in R are allowed.

Formally, we get the following problems.

The No-Wait Transit Last-Stretch Delivery Problem - NW-TLSDP

Instance: A weighted graph (G, w) with subsets $R, C \subseteq V(G)$ and two vertices $s, t \in R$.

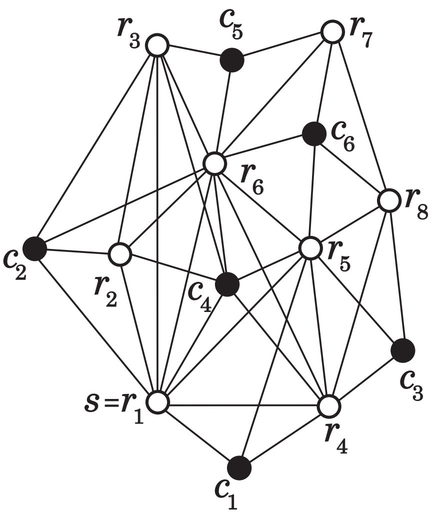
Feasible Solution: A simple s, t-path $P \subseteq G$ such that $C \subseteq V(P)$, and no two vertices in C are visited consecutively.

Objective: Minimize $w(P)$.

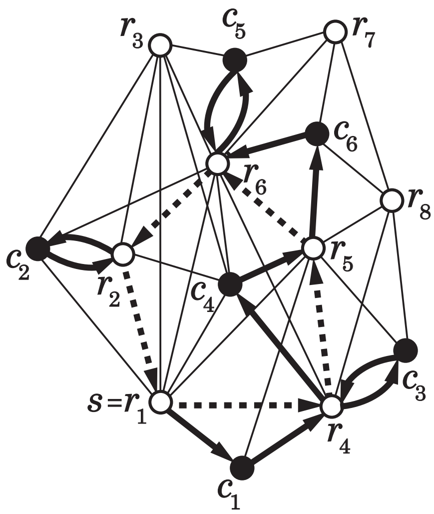
Notice that such a path does not exist if $|R| \leq |C|$.

The Transit Last-Stretch Delivery Problem - TLSDP

Instance: A weighted graph (G, w) with subsets $R, C \subseteq V(G)$ and two vertices $s, t \in R$.



(a)



(b)

Figure 1: (a) An illustration of the problem scenario. The points at which the delivery drone can rendezvous with the truck are illustrated as white circles. The customers to which parcels need to be delivered are represented by black circles. (b) An illustration of a feasible routing for the given problem scenario in (a). The truck's route is shown by dashed arrows, and the drone's route is shown by solid arrows.

Feasible Solution: A (not necessarily simple) s, t-path $P \subseteq G$ such that $C \subseteq V(P)$, and no two vertices in C are visited consecutively.

Objective: Minimize $w(P)$.

Notice that the only difference between the NW-TLSDP and the TLSDP is in the requirement that a feasible solution to the former must be a simple path, whereas this is not required in the latter where some vertices in R may be visited more than once, see Figs. 2(a) and 2(b). As will be seen in Section 3, this seemingly minor difference results in approximation algorithms with different approximation ratios between the two problems.

Observation 1 *Let (G, w) be a metric graph, and $R, C \subseteq V(G)$. Then, for both the NW-TLSDP and the TLSDP there exists an R, C-alternating optimal path in G.*

Proof. Due to the triangle inequality, without loss of generality we can assume that $V(G) = R \cup C$. Let P^* be an optimal path in the graph G. If there exist three vertices u, v, q, such that $u, v \in R$, $q \in C$ and they appear consecutively in the path P^* in the order u, v, q or q, v, u, then we can obtain a path P' from P^* by replacing the edges uv and vq by a single edge uq. For the cost $w(P')$ of the path P' we get

$$w(P') = w(P^*) - w(uv) - w(vq) + w(uq).$$

Then, from the triangle inequality we know that it holds that

$$w(uq) \leq w(uv) + w(vq),$$

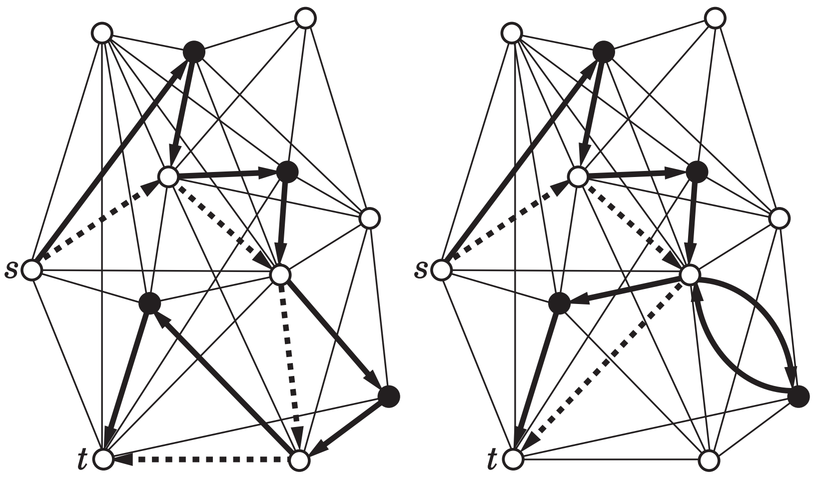


Figure 2: Feasible solutions to the NW-TLSDP and the TLSDP. The set R of rendezvous points is illustrated as white circles. The set C of customers points is presented by black circles. (a) A simple s, t-path that visits all vertices in C, as a feasible path for the NW-TLSDP. (b) An s, t-path that visits all vertices in C, as a feasible path for the TLSDP. Some vertices in R may be visited more than once.

and therefore

$$w(P') \leq w(P^*).$$

We can repeat the above shortcutting procedure until no more such triplets u, v, q of vertices exist. The resulting path from these shortcuts is alternating. $\square$

Throughout this paper, we assume that the edge weight function w satisfies the triangle inequality.

2.3 NP-hardness

Othman et al. [14] examined a similar problem with the one from this paper, where in addition, for a given total order $\prec$ over the set R, a feasible path P needs to visit the vertices in the set R obeying this total order. They have shown that their problem is NP-hard even if the edge weight function takes only values in $\{1, 2\}$, and that the total order over the set R can be taken without loss of generality. Therefore, we have the following claim.

Theorem 1 *The NW-TLSDP and the TLSDP are NP-hard.*

3 Approximation Algorithms

First, we propose a $(\rho + 1)$-approximation algorithm for the NW-TLSDP with metric weight w, where ρ is an approximation ratio for the metric TSP-path problem with fixed terminals [10, 16]. Currently the lowest bound on ρ of 1.6 was given by [16].

For the special instance of the NW-TLSDP where $s = t$, our algorithm has an approximation ratio of $(\delta + 1)$, where δ is an approximation ratio for the metric TSP. In this case, we can use the algorithm due to Christofides [2], which gives as a value for δ of 1.5, the lowest one known so far for the general metric.

We will use the algorithm of Sebő in a similar fashion, to propose a ρ-approximation algorithm for the TLSDP, and Christofides' algorithm to obtain a δ-approximation algorithm in the special case when $s = t$ in the TLSDP.

3.1 Approximation algorithm for the NW-TLSDP

Before proceeding with the description of an approximation algorithm and the analysis of its approximation ratio, we provide two technical lemmas which are useful for the analysis.

Lemma 1 *For an instance $I = ((G, w); R, C \subseteq V(G); s, t \in R)$ of the NW-TLSDP such that w satisfies the triangle inequality, let $P^* = (s = r_1, c_1, r_2, \ldots, r_n, r_{n+1} = t)$ be an R, C-alternating optimal solution for I. Let M be a minimum weight R, C-matching such that $s, t \notin V(M)$, and M leaves exposed exactly one $c \in C$. Then, it holds that*

$$2w(M) + w(r_1c_1) + w(c_nr_{n+1}) \leq w(P^*). \tag{4}$$

Proof. Notice that $E(P^*)$ contains two disjoint matchings

$$M' = \{c_1r_2, c_2r_3, \ldots, c_{n-1}r_n\}$$

and

$$M'' = \{r_2c_2, r_3c_3, \ldots, r_nc_n\},$$

such that it holds that $C - \{c_n\} \subseteq V(M')$ and $C - \{c_1\} \subseteq V(M'')$, and

$$w(P^*) = w(M') + w(M'') + w(r_1c_1) + w(c_nr_{n+1}).$$

Moreover, notice that it holds that $sc_1 \notin M'$ and $c_nt \notin M''$. By this observation, for the minimum cost R, C-matching M it holds that

$$w(M) \leq w(M')$$

and

$$w(M) \leq w(M'').$$

Based on these observations, it must hold that

$$\begin{aligned} w(P^*) &= w(M') + w(M'') + w(r_1c_1) + w(c_nr_{n+1}) \\ &\geq 2w(M) + w(r_1c_1) + w(c_nr_{n+1}), \end{aligned}$$

as required. □

Lemma 2 *For an instance $I = ((G, w); R, C \subseteq V(G); s, t \in R)$ of the NW-TLSDP such that w satisfies the triangle inequality, let $P = (s = r_1, c_1, r_2, \ldots, r_n, c_n, r_{n+1} = t)$ be a feasible R, C-alternating solution for I. Let $p, q \in C$ be such that $\{sp, qt\} \subseteq E(P)$, i.e., p and q are the vertices in C visited by P immediately after s and before t, respectively.*
(i) *If $s = t$, then for any δ-approximate Hamiltonian cycle $H_C = (p = c_1, c_2, \ldots, c_n, p)$ in $(G[C], w)$, $\delta \geq 1$, it holds that*

$$w(H_C) \leq \delta \cdot w(H). \tag{5}$$

(ii) *If $s \neq t$, then for any ρ-approximate Hamiltonian p, q-path $P_C = (p = c_1, c_2, \ldots, c_n = q)$ in $(G[C], w)$, $\rho \geq 1$, it holds that*

$$w(P_C) \leq \rho \cdot w(P). \tag{6}$$

Proof. Note that it holds that $p = c_1$ and $q = c_n$.
(i) If $s = t$, then for $w(H)$ we get

$$\begin{aligned} w(H) &= w(sc_1) + w(c_n s) + \sum_{k=1}^{n-1} (w(c_k r_k) + w(r_k c_{k+1})) \\ &\geq w(c_1 c_n) + \sum_{k=1}^{n-1} w(c_k c_{k+1}), \end{aligned} \tag{7}$$

which follows by the triangle inequality of Eq. (1). Notice that the right-hand-side of Eq. (7) defines the length $w(H'_C)$ of the Hamiltonian cycle H'_C in $(G[C], w)$. It certainly holds that

$$w(H_C) \leq \delta \cdot w(H'_C),$$

from where the claim follows.
(ii) If $s \neq t$, then writing the expression for $w(P)$ we get

$$\begin{aligned} w(P) &= w(sp) + w(qt) + \sum_{k=1}^{n-1} (w(c_k r_k) + w(r_k c_{k+1})) \\ &\geq \sum_{k=1}^{n-1} w(c_k c_{k+1}), \end{aligned} \tag{8}$$

which follows by the triangle inequality of Eq. (1). Notice that the right-hand-side of Eq. (8) defines the length $w(P'_C)$ of the Hamiltonian p, q-path $P'_C = (c_1, c_2, \ldots, c_n)$ in $(G[C], w)$. It certainly holds that

$$w(P_C) \leq \rho \cdot w(P'_C),$$

from where the claim follows. □

Next, we propose an approximation algorithm for the NW-TLSDP, summarized as Procedure Alternating Route I.

Procedure Alternating Route I

Input: A metric graph (G, w) with two subsets $R, C \subseteq V(G)$, $|R| > |C|$, vertices $s, t \in R$, and real numbers $\rho \geq 1$ and $\delta \geq 1$.
Output: If $s = t$, a simple $(\delta + 1)$-approximate R, C-alternating cycle H such that $C \subseteq V(H)$, otherwise a simple $(\rho + 1)$-approximate R, C-alternating s, t-path $P \subseteq G$ such that $C \subseteq V(P)$.

1: Let $n := |C|$;
2: **if** $s = t$ **then**
3: Find a δ-approximate Hamiltonian cycle H_C in $(G[C], w)$;
4: **for each** $p \in C$ **do**
5: Compute a minimum cost R, C-matching M_p in $(G - \{p, s\}, w)$ such that

$C - \{p\} \subseteq V(M_p)$;
Re-index the vertices $c_1, c_2, \ldots, c_n \in C$ such that $H_C = (p = c_1, c_2, \ldots, c_n, p)$;
Let $R' := R \cap V(M_p)$;
Let $\mu : C \to R'$ be a bijection such that $u = \mu(v)$ holds iff $uv \in M_p$;
Let H_p be the cycle $(s, c_1, \mu(c_2), \ldots, \mu(c_n), c_n, s)$
end for;
Let $H := \mathrm{argmin}\{w(H_p) \mid p \in C\}$;
return H
else
for all distinct $p, q \in C$ **do**
Find a ρ-approximate Hamiltonian p, q-path $P_C = (p = c_1, c_2, \ldots, c_n = q)$ in $(G[C], w)$;
Compute a minimum cost R, C-matching M_p in $(G - \{p, s, t\}, w)$ such that $C - \{p\} \subseteq V(M_p)$;
Compute a minimum cost R, C-matching M_q in $(G - \{q, s, t\}, w)$ such that $C - \{q\} \subseteq V(M_q)$;
if $w(M_p) \leq w(M_q)$ **then**
Let $R' := R \cap V(M_p)$;
/* The vertices in R that are matched by M_p */
Let $\mu : C \to R'$ be a bijection such that $u = \mu(v)$ holds iff $uv \in M_p$;
Let P_{pq} be the path $(s, c_1, \mu(c_2), \ldots, \mu(c_n), c_n, t)$
else /* $w(M_q) < w(M_p)$ */
Let $R' := R \cap V(M_q)$;
/* The vertices in R that are matched by M_q */
Let $\mu : C \to R'$ be a bijection such that $u = \mu(v)$ holds iff $uv \in M_q$;
Let P_{pq} be the path $(s, c_1, \mu(c_1), \ldots, \mu(c_{n-1}), c_n, t)$
end if
end for;
Let $P := \mathrm{argmin}\{w(P_{pq}) \mid p, q \in C, p \neq q\}$;
return P
end if.

Concerning the running time of the Procedure Alternating Route I, a minimum cost R, C-matching M such that $C \subseteq V(M)$ in Procedure Alternating Route I can be computed by standard combinatorial optimization techniques (see, e.g., Korte and Vygen [11]), whereas Christofides [2] gave a δ-approximation algorithm for the metric TSP with $\delta = 1.5$, and this algorithm can be used to find a δ-approximate Hamiltonian cycle in $(G[C], w)$ in polynomial time. Furthermore, Sebő [16] gave a polynomial time algorithm which can be used as a black-box to get a ρ-approximate Hamiltonian p, q-path in $(G[C], w)$ when $\rho = 1.6$.

Therefore, we have the following claim.

Lemma 3 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the NW-TLSDP where w satisfies the triangle inequality, and for values $\delta = 1.5$ and $\rho = 1.6$, Procedure Alternating Route I can be implemented to run in polynomial time.*

Theorem 2 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the NW-TLSDP such that w satisfies the triangle inequality, let $P^* = (s = r_1, c_1, r_2, \ldots, r_n, c_n, r_{n+1} = t)$ be an R, C-alternating optimal solution for I.*

(i) *If* $s = t$*, then for a simple cycle* $H = (s, c_1, \mu(c_2), \ldots, \mu(c_n), c_n, s)$ *returned by Procedure Alternating Route I given the instance* I *as an input, it holds that*

$$w(H) \leq (\delta + 1) \cdot w(H^*). \tag{9}$$

(ii) *If* $s \neq t$*, then for a simple path* $P = (s, c_1, \mu(c_2), \ldots, \mu(c_n), c_n, t)$ *returned by Procedure Alternating Route I given the instance* I *as an input, it holds that*

$$w(P) \leq (\rho + 1) \cdot w(P^*). \tag{10}$$

Proof.
(i) First, we examine the case when $s = t$. Let us observe a single iteration of the for-loop of Lines 4 to 10 of Procedure Alternating Route I. Let H_C be the δ-approximate Hamiltonian cycle in $(G[C], w)$ of Line 3 of the procedure. Then, for the cycle H_p of Line 9 of Procedure Alternating Route I it holds that

$$\begin{aligned}
w(H_p) &= w(sc_1) + \sum_{i=1}^{n-1} (w(c_i\mu(c_{i+1})) + w(\mu(c_{i+1})c_{i+1})) + w(c_n s) \\
&\leq w(sc_1) + w(c_n s) + \sum_{i=1}^{n-1} w(c_i c_{i+1}) + 2\sum_{i=2}^{n} w(c_i\mu(c_i)) \quad \text{(by Eq. (1))} \\
&\leq w(sc_1) + w(c_n s) + w(H_C) + 2w(M).
\end{aligned}$$

The algorithm tries all possible choices of p, and returns a cycle $H = H_p$ that minimizes $w(H_p)$. Then, by Eqs. (4) and (5) for the cycle H it holds that

$$w(H) \leq \delta \cdot w(H^*) + w(H^*),$$

as required.
(ii) Next, if $s \neq t$, then we observe a single iteration of the for-loop of Lines 14 to 27 of Procedure Alternating Route I. Let P_C be the ρ-approximate path in $(G[C], w)$ as in Line 15 of the procedure. Then, for the path P_{pq} in Line 21 of Procedure Alternating Route I it holds that

$$\begin{aligned}
w(P_{pq}) &= w(sp) + \sum_{i=1}^{n-1} (w(c_i\mu(c_{i+1})) + w(\mu(c_{i+1})c_{i+1})) + w(qt) \\
&\leq w(sp) + w(qt) + \sum_{i=1}^{n-1} w(c_i c_{i+1}) + 2\sum_{i=2}^{n} w(c_i\mu(c_i)) \quad \text{(by Eq. (1))} \\
&\leq w(sp) + w(qt) + w(P_C) + 2w(M).
\end{aligned}$$

The algorithm tries all possible choices of p and q, and returns a path $P = P_{pq}$ that minimizes $w(P_{pq})$. Then, by Eqs. (4) and (6) for the path P it holds that

$$w(P) \leq \rho \cdot w(P^*) + w(P^*),$$

as required. □

3.2 Approximation algorithm for the TLSDP

Again, before proceeding with the description of an approximation algorithm and the analysis of its approximation ratio, we state two technical lemmas to be used in the proof.

First, we define a function $\widetilde{w} : \binom{C\cup\{s,t\}}{2} \to \mathbb{R}_+$ to be

$$\widetilde{w}(uv) \triangleq \begin{cases} w(uv), & \text{for } u \in \{s,t\}, v \in C \cup \{s,t\}, \\ \min_{r\in R}\{w(ur) + w(rv)\}, & \text{for } u, v \in C. \end{cases} \tag{11}$$

Lemma 4 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the TLSDP , if the weight function w satisfies the triangle inequality, then so does the weight function $\widetilde{w}$ in Eq.* (11).

Proof. We set to show that for all $u, v, q \in C \cup \{s, t\}$ it holds that

$$\widetilde{w}(uv) \le \widetilde{w}(uq) + \widetilde{w}(qv).$$

First, observe that if any of u and v is in $\{s, t\}$, then by the definition of $\widetilde{w}$ it holds that $\widetilde{w}(uv) = w(uv)$, $\widetilde{w}(uq) = w(uq)$, and $\widetilde{w}(qv) = w(qv)$, and the claim follows from the assumption that w satisfies the triangle inequality.

Second, if it holds that $\{u, v\} \cap \{s, t\} = \emptyset$, but $q \in \{s, t\}$, then we get that it holds that $\widetilde{w}(uq) = w(uq)$ and $\widetilde{w}(qv) = w(qv)$, and the claim follows from the definition of $\widetilde{w}$, i.e.,

$$\begin{aligned} \widetilde{w}(uv) &= \min_{r\in R}\{w(ur) + w(vr)\} \\ &\le w(uq) + w(vq) \\ &= \widetilde{w}(uq) + \widetilde{w}(qv). \end{aligned}$$

Finally, let $\{u, v, q\} \cap \{s, t\} = \emptyset$. Let $r_{vq}, r_{uq} \in R$ be the minimizers of $\widetilde{w}(vq)$ and $\widetilde{w}(uq)$, respectively. By the definition of $\widetilde{w}$, it holds that

$$\widetilde{w}(uv) \le w(ur_{uq}) + w(vr_{uq}).$$

From the triangle inequality we know that it holds that

$$w(vr_{uq}) \le w(vr_{vq}) + w(r_{vq}r_{uq}),$$

and also,

$$w(r_{vq}r_{uq}) \le w(qr_{vq}) + w(qr_{uq}).$$

Then, we get

$$\begin{aligned} \widetilde{w}(uv) &\le w(ur_{uq}) + w(vr_{vq}) + w(r_{vq}r_{uq}) \\ &\le w(ur_{uq}) + w(vr_{vq}) + w(qr_{vq}) + w(qr_{uq}) \\ &= \widetilde{w}(uq) + \widetilde{w}(vq), \end{aligned}$$

as required. □

Lemma 5 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the TLSDP such that w satisfies the triangle inequality, let $P = (s = r_1, c_1, r_2, \ldots, r_n, c_n, r_{n+1} = t)$ be a feasible R, C-alternating solution for I. Let $\widetilde{w}$ be as in Eq.* (11).

(i) *If $s = t$, then for any δ-approximate Hamiltonian cycle $H'_C = (s, c'_1, c'_2, \ldots, c'_n, s)$ in $(G[C \cup \{s\}], \widetilde{w})$, $\delta \geq 1$, it holds that*

$$\widetilde{w}(H'_C) \leq \delta \cdot w(H). \tag{12}$$

(ii) *If $s \neq t$, then for any ρ-approximate Hamiltonian s,t-path $P'_C = (s, c'_1, c'_2, \ldots, c'_n, t)$ in $(G[C \cup \{s, t\}], \widetilde{w})$, $\rho \geq 1$, it holds that*

$$\widetilde{w}(P'_C) \leq \rho \cdot w(P). \tag{13}$$

Proof.

(i) Writing the expression for $w(H)$, we get

$$\begin{aligned} w(H) &= w(sc_1) + w(c_n s) + \sum_{k=1}^{n-1}(w(c_k r_k) + w(r_k c_{k+1})) \\ &\geq \widetilde{w}(sc_1) + \widetilde{w}(c_n s) + \sum_{k=1}^{n-1} \widetilde{w}(c_k c_{k+1}), \end{aligned} \tag{14}$$

which follows from the definition of $\widetilde{w}$ of Eq. (11). The right-hand-side of Eq. (14) gives the cost $\widetilde{w}(H_C)$ of a Hamiltonian cycle H_C in $(G[C \cup \{s\}], \widetilde{w})$, and it certainly holds that

$$\widetilde{w}(H'_C) \leq \delta \cdot \widetilde{w}(H_C),$$

as required.
(ii) If $s \neq t$, then writing the expression for $w(P)$, we get

$$\begin{aligned} w(P) &= w(sc_1) + w(c_n t) + \sum_{k=1}^{n-1}(w(c_k r_k) + w(r_k c_{k+1})) \\ &\geq \widetilde{w}(sc_1) + \widetilde{w}(c_n t) + \sum_{k=1}^{n-1} \widetilde{w}(c_k c_{k+1}), \end{aligned} \tag{15}$$

which follows from the definition of $\widetilde{w}$ of Eq. (11). The right-hand-side of Eq. (15) gives the cost $\widetilde{w}(P_C)$ of an s,t-Hamiltonian path P_C in $(G[C \cup \{s, t\}], \widetilde{w})$, and it certainly holds that

$$\widetilde{w}(P'_C) \leq \rho \cdot \widetilde{w}(P_C),$$

as required. □

Next, we propose an approximation algorithm for the TLSDP, summarized as Procedure Alternating Route II.

Procedure Alternating Route II

Input: A metric graph (G, w) with two subsets $R, C \subseteq V(G)$, vertices $s, t \in R$, and real numbers $\rho \geq 1$ and $\delta \geq 1$.
Output: If $s = t$, a ρ-approximate R, C-alternating (but not necessarily simple) cycle H such that $C \subseteq V(H)$, otherwise a δ-approximate R, C-alternating (but not necessarily simple) s,t-path $P \subseteq G$ such that $C \subseteq V(P)$.

1: Calculate $\widetilde{w}$ according to Eq. (11);

2: **if** $s = t$ **then**
3: Find a δ-approximate Hamiltonian cycle $H_C = (s, c_1, c_2, \ldots, c_n, s)$ in $(G[C] \cup \{s\}, \widetilde{w})$;
4: **for** $i = 1, 2, \ldots, n-1$ **do**
5: Let $r'_i := \mathrm{argmin}\{w(c_i r) + w(r c_{i+1}) \mid r \in R\}$
6: **end for**;
7: Let $R' := \{r'_1, r'_2, \ldots, r'_{n-1}\}$;
8: Let H be the cycle $(s, c_1, r'_1, c_2, \ldots, r'_{n-1}, c_n, s)$;
9: **return** H
10: **else**
11: Find a ρ-approximate Hamiltonian s,t-path $P_C = (s, c_1, c_2, \ldots, c_n, t)$ in $(G[C \cup \{s,t\}], \widetilde{w})$;
12: **for** $i = 1, 2, \ldots, n-1$ **do**
13: Let $r'_i := \mathrm{argmin}\{w(c_i r) + w(r c_{i+1}) \mid r \in R\}$
14: **end for**;
15: Let $R' := \{r'_1, r'_2, \ldots, r'_{n-1}\}$;
16: Let P be the path $(s, c_1, r'_1, c_2, \ldots, r'_{n-1}, c_n, t)$;
17: **return** P
18: **end if**.

By the virtue of Lemma 4, we can be assured that the Christofides algrithm with $\delta = 1.5$, and the ρ-approximation algorithm with $\rho = 1.6$ due to Sebő [16], can indeed be used, and that Procedure Alternating Route II can be implemented to run in polynomial time.

Lemma 6 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the TLSDP such that w satisfies the triangle inequality, and for values $\delta = 1.5$ and $\rho = 1.6$, Procedure Alternating Route II can be implemented to run in polynomial time.*

Theorem 3 *For an instance $I = ((G, w);\ R, C \subseteq V(G);\ s, t \in R)$ of the TLSDP such that w satisfies the triangle inequality, let $P^* = (s = r_1, c_1, r_2, \ldots, r_n, c_n, r_{n+1} = t)$ be an R, C-alternating optimal solution for I.*
(i) *If $s = t$, then for a cycle $H = (s, c_1, r'_1, c_2, \ldots, r'_{n-1}, c_n, s)$ returned by Procedure Alternating Route II given the instance I as an input, it holds that*

$$w(H) \leq \delta \cdot w(H^*). \tag{16}$$

(ii) *If $s \neq t$, then for a path $P = (s, c_1, r'_1, c_2, \ldots, r'_{n-1}, c_n, t)$ returned by Procedure Alternating Route II given the instance I as an input, it holds that*

$$w(P) \leq \rho \cdot w(P^*). \tag{17}$$

Proof.
(i) First, we examine the case when $s = t$. Let us observe a single iteration of the for-loop of Lines 4 to 6 of Procedure Alternating Route II. Let H_C be a δ-approximate Hamiltonian cycle in $(G[C \cup \{s\}], \widetilde{w})$ as in Line 3 of the procedure. Then, for the cycle

H in Line 8 Procedure Alternating Route II it holds that

$$\begin{aligned} w(H) &= w(sc_1) + w(c_n s) + \sum_{i=1}^{n-1} w(c_i r'_i) + \sum_{i=1}^{n-1} w(r'_i c_{i+1}) \\ &= \widetilde{w}(sc_1) + \widetilde{w}(c_n s) + \sum_{i=1}^{n-1} \widetilde{w}(c_i c_{i+1}) \\ &= \widetilde{w}(H_C). \end{aligned} \tag{18}$$

(ii) Next, if $s \neq t$, then we observe a single iteration of the for-loop of Lines 12 to 14 of Procedure Alternating Route II. Let P_C be a ρ-approximate Hamiltonian s,t-path in $(G[C \cup \{s,t\}], \widetilde{w})$ as in Line 11 of the procedure. Then, for the path P in Line 16 Procedure Alternating Route II it holds that

$$\begin{aligned} w(P) &= w(sc_1) + w(c_n t) + \sum_{i=1}^{n-1} w(c_i r'_i) + \sum_{i=1}^{n-1} w(r'_i c_{i+1}) \\ &= \widetilde{w}(sc_1) + \widetilde{w}(c_n t) + \sum_{i=1}^{n-1} \widetilde{w}(c_i c_{i+1}) \\ &= \widetilde{w}(P_C). \end{aligned} \tag{19}$$

The claim readily follows from Eqs. (12) and (13) of Lemma 5. □

4 Conclusion

In this work, we have investigated a scenario in which a drone is used in tandem with a delivery truck for the last-stretch delivery of parcels to customers' doorsteps.

In the problem that we investigated, only the drone is allowed to perform deliveries. As future work, it would be interesting to analyze some extensions of this routing problem, especially to examine a combinatorial optimization based model for a routing problem where both the delivery truck and the drone can make deliveries [1, 9].

Another possible extension would be to investigate metric settings, but with edge weight bias to account for additional transportation effort exerted by the drone when delivering a parcel [17].

References

[1] Agatz, N., Bouman, P., Schmidt, M.: Optimization approaches for the traveling salesman problem with drone. ERIM Report Series Reference No. ERS-2015-011-LIS (2015)

[2] Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, 1976.

[3] D'Andrea, R.: Guest editorial, Can drones deliver? IEEE Transactions on Automation Science and Engineering 11(3), pp. 647–648 (2014)

[4] Gallacher, D.: Drones to manage the urban environment: Risks, rewards, alternatives. Journal of Unmanned Vehicle Systems 4(2), pp. 115–124 (2016)

[5] Garey, M. R., Johnson, D. S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company (1979)

[6] Garone, E., Naldi, B., Casavola, A.: Traveling-Salesman Problem for a Class of Carrier-Vehicle Systems. Journal of Guidance, Control, and Dynamics, 34(4), pp. 1272–1275 (2011)

[7] Grötschel, M., Lovász, L., Schrijver, A.: The ellipsoid method and its consequences in combinatorial optimization. Combinatorica, 1(2), pp. 169–197, (1981)

[8] Gupta, S. G., Ghonge, M. M., Jawandhiya, P.: Review of unmanned aircraft system (UAS). International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 2(4), pp. 1646–1658 (2013)

[9] Ha, Q. M., Deville, Y., Pham, Q. D., Hà, M. H.: On the min-cost traveling salesman problem with drone. arXiv preprint, arXiv:1512.01503 (2015)

[10] Hoogeveen, J.: Analysis of Christofides' heuristic: Some paths are more difficult than cycles. Operations Research Letters, 10(0), pp. 291–295 (1991)

[11] Korte, B., Vygen, J.: Combinatorial Optimization: Theory and Algorithms. Springer Science & Business Media (2006)

[12] Mathew, N., Smith, S. L., Waslander, S. L.: Planning paths for package delivery in heterogeneous multirobot teams. IEEE Transactions on Automation Science and Engineering, 12(4), pp. 1298–1308 (2015)

[13] Murray, C. C., Chu, A. G.: The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transportation Research Part C: Emerging Technologies 54, pp. 86–109 (2015)

[14] Othman, M. S., Shurbevski, A., Nagamochi, H., Karuno, Y.: Routing Problems with Last-Stretch Delivery. The 19th Japan Conference on Discrete and Computational Geometry, Graphs, and Games (JCDCGGG 2016), pp. 70–71 (2016)

[15] Ponza, A.: Optimization of drone-assisted parcel delivery. Master's thesis, University of Padova (2016)

[16] Sebő, A.: Eight-fifth approximation for the path TSP. In: Proceedings of International Conference on Integer Programming and Combinatorial Optimization IPCO 2013, pp. 362–374, Springer (2013)

[17] Shurbevski, A., Nagamochi, H., Karuno, Y.: Approximating the bipartite TSP and its biased generalization. In: Proceedings of International Workshop on Algorithms and Computation WALCOM 2014, pp. 56–67, Springer (2014)

[18] Van Woensel, L., Archer, G., Panades-Estrutch, L., Vrscaj, D.: Ten technologies which could change our lives: Potential impacts and policy implications. European Parliamentary Research Service In-Depth Analysis, PE 527.417 (2015)

[19] Watts, A. C., Ambrosia, V. G., Hinkley, E. A.: Unmanned aircraft systems in remote sensing and scientific research: Classification and considerations of use. Remote Sensing 4(6), pp. 1671–1692 (2012)

[20] Zhang, C., Kovacs, J. M.: The application of small unmanned aerial systems for precision agriculture: a review. Precision Agriculture 13(6), pp. 693–712 (2012)