Scheduling Diagnostic Testing Kit Deliveries with the Mothership and Drone Routing Problem

Hyung Jin Park[*], Reza Mirjalili[*], Murray J. Côté[†], and Gino J. Lim[*]

April 8, 2022

Abstract

A critical component in the public health response to pandemics is the ability to determine the spread of diseases via diagnostic testing kits. Currently, diagnostic testing kits, treatments, and vaccines for the COVID-19 pandemic have been developed and are being distributed to communities worldwide, but the spread of the disease persists. In conjunction, a strong level of social distancing has been established as one of the most basic and reliable ways to mitigate disease spread. If home testing kits are safely and quickly delivered to a patient, this has the potential to significantly reduce human contact and reduce disease spread before, during, and after diagnosis. This paper proposes a diagnostic testing kit delivery scheduling approach using the Mothership and Drone Routing Problem (MDRP) with one truck and multiple drones. Due to the complexity of solving the MDRP, the problem is decomposed into 1) truck scheduling to carry the drones and 2) drone scheduling for actual delivery. The truck schedule (TS) is optimized first to minimize the total travel distance to cover patients. Then, the drone flight schedule is optimized to minimize the total delivery time. These two steps are repeated until it reaches a solution minimizing the total delivery time for all patients. Heuristic algorithms are developed to further improve the computational time of the proposed model. Experiments are made to show the benefits of the proposed approach compared to the commonly performed face-to-face diagnosis via the drive-through testing sites. The proposed solution method significantly reduced the computation time for solving the optimization model (less than 50 minutes) compared to the exact solution method that took more than 10 hours to reach a 20% optimality gap. A modified basic reproduction rate (i.e., mR_0) is used to compare the performance of the drone-based testing kit delivery method to the face-to-face diagnostic method in reducing disease spread. The results show that our proposed method (mR_0=0.002) outperformed the face-to-face diagnostic method (mR_0=0.0153) by reducing mR_0 by 7.5 times.

Keywords: Pandemic response, COVID-19, Testing kit delivery, Social distancing, Drone-truck scheduling, Mothership and Drone Routing Problem

1 Introduction

Currently, while the world's largest pharmaceutical companies are actively developing and distributing COVID-19 vaccines and treatments, the spread of the pandemic still persists and is entering its fourth wave [46]. Countries are continuing to establish and maintain various policies to slow the spread of infection. Health authorities in many countries have set up drive-through testing sites to cover the enormous demand for diagnostics. The principal diagnostic method to confirm COVID-19 cases at drive-through testing sites is by a molecular diagnostic kit such as Polymerase Chain Reaction (PCR). However, there are drawbacks to the approach. The first is the slow process to get the diagnostic result back to the patient. In many cases, it can take over 6 hours to determine the status of the collected sample [27]. Additionally, it typically takes 2 to 5 days for the patient to receive the result. If a patient with COVID-19 symptoms is not quarantined and contacts other people during this waiting period, the spread of the virus will not be contained. Secondly, the PCR testing kit requires a sterile plastic swab to be inserted deep into the nose to collect sputum as a sample. Since the sample must be handled by someone else, it may also lead to

[*]Department of Industrial Engineering, University of Houston, Corresponding author: Gino Lim, ginolim@uh.edu
[†]Department of Health Policy and Management, School of Public Health, Texas A&M University

personal contact with the possibility of infecting the testing personnel. Accordingly, more advanced diagnostic methods are desired to help contain infectious diseases.

In November 2020, the Food and Drug Administration (FDA) in the United States approved a diagnostic testing kit that allows patients to self-test for the COVID-19 at home. In particular, this testing kit allows patients to take samples by themselves and check the results directly within 30 minutes. However, patients who become aware of suspected COVID-19 symptoms must go to a hospital or pharmacy to obtain these testing kits, and there is still an opportunity to contact another non-infected person in the process.

In this study, we propose a strategy to deliver COVID-19 self-testing kits to patients at home using the Mothership and Drone Routing Problem (MDRP) with one truck and multiple drones. Hence, potential patients can use at-home testing kits to determine whether they are infected with COVID-19 while social distancing is assured as much as possible. The primary focus of this study is to lower the infection rate by avoiding unnecessary personal contacts while testing for COVID-19. The main contribution of this study is as follows:

- To propose an approach to test COVID-19 using the Mothership and Drone Routing Problem (MDRP) that can help mitigate the possibility of human contact that may occur during the diagnosis process. This has the potential to suppress the spread of COVID-19.
- To provide decision-makers with a performance metric for using a truck and drones to deliver testing kits to patients during a pandemic: We show that the proposed method has the possibility of significantly reducing the risk of infection compared to the current testing methods.

The rest of this paper is organized as follows. Section 2 reviews literature related to this area. Section 3 describes the problem in detail and presents the mathematical models applied to find a solution. In Section 4, solution techniques to solve the MDRP (Mothership and Drones Routing Problem). Section 5 describes an analysis tool for numerically measuring the risk of infection in the process of each diagnostic method and provides a numerical experiment to illustrate the effectiveness of the proposed model and numerical results for infection risk. Finally, our research is summarized in Section 6 including potential extensions of the research.

2 Literature Review

Patients with suspected COVID-19 symptoms need to be tested and can crowds diagnostic locations such as hospitals and drive-through sites. Despite the elevated disinfection efforts, the risk of infection can still remain. In the case of MERS (Middle East Respiratory Syndrome) and SARS-CoV-22, which are viruses of the same family as COVID-19, many cases of infection were reported in hospitals [42]. For example, 99% (i.e., 185 cases) of the total confirmed cases of MERS in South Korea in 2015 were infected in hospitals [23]. The data from the International Council of Nurses (ICN) released in October 2020 estimated that more than four million healthcare workers worldwide are infected with COVID-19 [2]. Social distancing has been shown to be most effective in reducing the infection risk. This underscores an urgent need for a new technology for patients to be tested without joining the crowd of numerous suspected patients. This underscores an urgent need for a new technology for patients to be tested without joining the crowd of numerous suspected patients.

There have been recent studies on mathematical models for using drones to deliver medical kits and packages to customers. The drone delivery routing problems can be classified into two categories (see Figure 1): (1) delivery by both trucks and drones and (2) delivery by drones only.

Delivery by both trucks and drones: The main idea of this problem is to utilize both a truck and drones to deliver medical testing kits to where the patients live. Optimization models are often a variant of travelling salesman problem (TSP) or vehicle routing problem (VRP) depending on the number of trucks to be used in the problem. In TSP with drones (TSP-D), the delivery schedule is optimized for a combination of one truck and multiple drones[6, 43, 49]. Similarly, the VRP

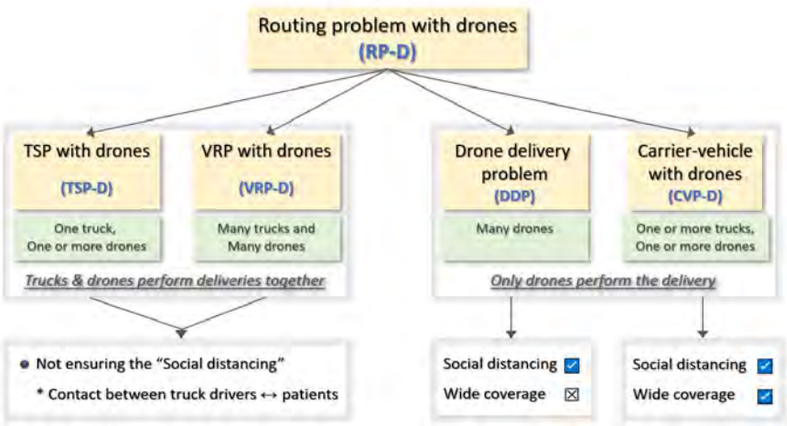


Figure 1: The categorization of drone routing problem by [30].

with drones (VRP-D) is to optimize the delivery schedule utilizing multiple trucks and multiple drones [29, 31, 36, 52]. Common objectives in the literature include minimizing the duration of operation (i.e., makespan) routing and operation cost, and customers waiting time [26, 35, 51]. Because this class of problem is identified as *NP-hard*, exact solution techniques are only used to solve small problems. But, most solution approaches are largely heuristic methods to handle larger scale problems providing good solutions in a timely manner [30, 34, 36, 38].

Delivery by drones only: This category of problems uses drones only for parcel delivery, and there are two versions of models in the literature. The first version schedules drone flights to-and-from a depot [24, 47]. Kim et al. (2017) propose a model in which drones are utilized to deliver medical kits from fixed locations of ground control centers (GCCs) for patients with chronic diseases in rural areas. Due to the limited battery capacity of drones, their model suggests using a larger number of drones from multiple GCCs. Alternatively, the short flight range limitation from a GCC can be addressed by operating drones from a mobile control center such as a truck. It can also help ensure social distancing during a pandemic because people can stay at home to get the testing done. The main idea of the Carrier-Vehicle Problem with Drones (CVP-D) is that the mobile base (e.g., truck) serves as a mothership that dispatches and collects multiple drones to perform delivery. Hence, it requires a fleet of cooperating vehicles with complementary functions that perform autonomous delivery [8, 16, 28, 30]. Table 1 presents the main features of CVP-D currently found in the literature. In the majority of the works the fleet is composed of one truck and one or several drones.

Table 1: Summary of the CVP-D contributions in the scientific literature

References	Trucks	Drones	Objcetive function (minimize)	Time-Windows	Drop & Pickup	Multiple visits
Savuran and Karakaya (2016)	1	1	number of targets	No	No	Yes
Mathew et al. (2015)	1	1	total distance	No	No	No
Bin Othman et al. (2017)	1	1	total distance	No	No	No
Gambella et al. (2018)	1	1	completion time	No	No	No
Boysen et al. (2018)	1	multiple	make span	No	No	No
Dukkanci et al. (2019)	multiple	multiple	operating cost	Yes	No	No
Karak and Abdelghany (2019)	1	multiple	operating cost	No	Yes	Yes
Wikarek et al. (2019)	multiple	multiple	total distance	No	Yes	Yes
Bai et al. (2019)	1	1	completion time	No	No	No
Poikonen and Golden (2020a)	1	1	completion time	No	No	No
Poikonen and Golden (2020b)	1	multiple	completion time	No	No	Yes
Moeini and Salewski (2019)	1	multiple	total distance	No	No	No
Han et al. (2020)	multiple	multiple	truck number	Yes	No	No

Variants of the CVP-D have been proposed for pickup and delivery. For example, Karak and Abdelghany (2019) proposed a model in which a drone is mounted on a single vehicle to visit more than one customer to deliver and pick up parcels. Drones can be dispatched and collected multiple times following the truck path to visit each customer only once. The truck returns to the depot

after collecting all drones which completed the delivery missions. In another CVP-D variant by Wikarek et al. (2019), drones can only be launched or retrieved from trucks at predefined mobile distribution centers where the launch and retrieve locations are allowed to be different. Their model focused on optimizing drone routing and the location of mobile distribution centers. But, truck routing was not discussed. Mathew et al. (2015) presented a delivery system in which trucks carry drones and the delivery is carried entirely by the onboard drones. A drone performs a single delivery mission to a specific point.

Some researchers proposed drone routing approaches based on a predetermined route of the carrying truck (Savuran and Karakaya (2016), Bin Othman et al. (2017) and Boysen et al. (2018)). Typically, a large vehicle is used as a mobile depot for a single drone onboard. Unlike Mathew et al. (2015), drones are allowed to visit multiple locations. With the goal of visiting as many locations as possible, Savuran and Karakaya (2016) proposed a genetic algorithm to solve the model. Bin Othman et al. (2017) allowed drones to take off from an intermediate node and get picked up by a truck at another location. The problem was modeled as a graph problem and a polynomial-time approximation algorithm was developed as a solution method. Boysen et al. (2018) focus on optimizing the schedule of drones for a given truck path. The authors examined three truck-drone operation policies: same take-off and pick-up location, different locations for takeoff and pickup of drones, and no restriction on the locations.

Poikonen et al. (2019) consider the problem of optimizing both the truck routes and the drone paths. They proposed a branch-and-bound algorithm to solve small-size instances, and heuristic methods such as a greedy algorithm and local search strategies for larger instances. As an extension to the previous work, Poikonen and Golden (2020b) allowed drones to visit multiple customers in one flight path, and the energy consumption rate was also considered to account for the limited capacity of a drone battery. A heuristic approach "path, transform, shortest path" was proposed to solve the problem. In another extension, Poikonen and Golden (2020a) introduced the "Mothership and Drone Routing Problem (MDRP)" which considers the routing of a mothership and a drone to visit several designated locations. Unlike the previous model, the drone is allowed to visit multiple targets consecutively before returning to the mothership for refueling In the infinite-capacity drone routing problem (MDRP-IC) setting. They developed an exact branch-and-bound solution approach and proposed two greedy heuristic approaches to find near-optimal solutions.

Although the existing approaches work well for their respective problems, they do not consider the important public health concept of "social distancing" as we discuss in this paper. Hence, this paper builds on the existing literature to solve the problem of delivering medical kits to patients to minimize human contact within the context of an existing pandemic (i.e., COVID-19).

3 Problem Description and Model Formulation

We propose a medical kit delivery approach, in which a truck carries drones that perform the actual delivery. The drones are assumed to follow the pre-determined routes within the line of sight of the pilot. Due to the limited battery capacity of drones, the drone flight range is limited. Hence, a truck acts as a mobile drone control center and carries the drones onboard to launch and retrieve them while visiting patients in the designated region. Although it is limited, the truck is capable of carrying multiple drones so that parallel flights can be performed to cover multiple patients to speed up the delivery mission. Therefore, our goal is to find a coordinated schedule between the truck and drones so that the total delivery completion time is minimized. Such a delivery system is referred to "Mothership and Drones Routing Problem" (MDRP) [39].

Before describing the model formulation, we state few problem-specific assumptions used in this paper. The maximum drone flight time is not affected by the amount of load to be carried by a drone. Since most testing kits are lightweight, the payload has minimal impact on the flight time. The truck driving range is also not affected by the number of drones or testing kits loaded. The truck can stop and collect a drone at intermediate nodes (I) only.

Using sets, parameters, and variables defined in Table 2, the proposed MDRP model is formu-

Table 2: Sets, parameters, and variables that are used in MDRP.

	Notation	Description
Sets	I	Set of the intermediate nodes on the truck network.
	A_t	Set of arcs representing road segments
	N	Set of patients
	A_d	Set of all possible flight arcs.
Parameters		
	$G_t(I, A_t)$	The directed network graph of truck
	$G_d(\{N \cup I\}, A_d)$	The directed network graph of flights
	$wt_{i,j}(\min)$	Travel time of truck traversing arc $(i,j) \in A_t$
	$wd_{i,j}(\min)$	Flight time of drone traversing link $(i,j) \in A_d$
	$s_i(\min)$	Service time at patient location i
	n	Total number of drones
	m	Total number of operators performing launch and retrieve
	$W_{\max}(\min)$	Maximum waiting time.
	$H_{\max}(\min)$	Maximum hovering time.
	$\delta(\min)$	Setup time for a drone
	$\theta(\min)$	Drone retrieval time
	M	A very big number
	$R_{\max}(\min)$	Maximum flight time of drone.
Variables		
	$y_{i,j} \in \{0,1\}$	If truck chooses link $(i,j) \in A_t$ to traverse it gets 1 otherwise 0.
	$x_{i,j} \in \{0,1\}$	If drone chooses link $(i,j) \in A_d$ to flight it gets 1 otherwise 0.
	$v_i \in \mathbb{Z}_+^0$	Order of visiting node $i \in I$ in truck path.
	$u_i \in \mathbb{Z}_+^0$	Order of visiting node $i \in I \cup N$ in flight path.
	$w_i \in \mathbb{R}_+^0$	Waiting time of truck at node $i \in I$.
	$h_i \in \mathbb{R}_+^0$	Waiting time of drone (hovering) over node $i \in N$.
	$T_i \in \mathbb{R}_+^0$	Time of entrance of truck at node $i \in I$.
	$t_i \in \mathbb{R}_+^0$	Time of entrance of drone at node $i \in I \cup N$.
	$b_i \in \mathbb{R}_+^0$	Remaining flight range, at the time of reaching node $i \in I \cup N$.
	$D_i \in \mathbb{Z}_+^0$	The number of drones mounted in truck after leaving node $i \in I$.

lated as follows:

$$\min_{x,y,w,h,t,T,b,u,v,D} \quad Z = \left(\sum_{i,j \in A_t} wt_{i,j} y_{i,j} + \sum_{i \in I} w_i \right) + \left(\sum_{i,j \in A_d} wd_{i,j} x_{i,j} + \sum_{i \in N} (h_i + s_i) \right) \tag{1}$$

s.t.

$$\sum_{(i,C) \in A_t} y_{i,C} = \sum_{(C,j) \in A_t} y_{C,j} = 1 \tag{2}$$

$$\sum_{(i,j) \in A_t} y_{i,j} = \sum_{(j,i) \in A_t} y_{j,i}, \quad \forall j \in I \backslash \{C\} \tag{3}$$

$$v_i - |I|(1 - y_{i,j}) + 1 \leq v_j, \quad \forall (i,j) \in A_t, j \neq C \tag{4}$$

$$T_i + w_i + wt_{i,j} \leq T_j + M(1 - y_{i,j}), \quad \forall (i,j) \in A_t \tag{5}$$

$$T_i + w_i + wt_{i,j} \geq T_j - M(1 - y_{i,j}), \quad \forall (i,j) \in A_t \tag{6}$$

$$w_i \leq W_{\max} \quad \forall i \in I \tag{7}$$

$$h_i \leq H_{\max} \quad \forall i \in N \tag{8}$$

$$\sum_{(i,j) \in A_d} x_{i,j} = \sum_{(j,i) \in A_d} x_{j,i} = 1, \quad \forall j \in N \tag{9}$$

$$1 - |N|(1 - x_{i,j}) \leq u_j, \quad \forall (i,j) \in A_d, \ i \in I \tag{10}$$

$$1 + u_i - |N|(1 - x_{i,j}) \leq u_j, \quad \forall (i,j) \in A_d, \ j \notin I \tag{11}$$

$$b_j + s_i + h_i + wd_{i,j} - M(1 - x_{i,j}) \leq b_i, \quad \forall (i,j) \in A_d, \ i \notin I \tag{12}$$

$$b_j + s_i + h_i + wd_{i,j} - M(1 - x_{i,j}) \leq R_{\max}, \quad \forall (i,j) \in A_d, \ i \in I \tag{13}$$

$$t_i + s_i + h_i + wd_{i,j} \leq M(1 - x_{i,j}) + t_j, \quad \forall (i,j) \in A_d, \ i \notin I \tag{14}$$

$$t_i + s_i + h_i + wd_{i,j} \geq t_j - M(1 - x_{i,j}), \quad \forall (i,j) \in A_d, \ i \notin I \tag{15}$$

$$T_i - M(1 - x_{i,j}) \leq t_i \leq T_i + w_i + M(1 - x_{i,j}), \quad \forall (i,j) \in A_d,\ i \in I \tag{16}$$

$$T_j - M(1 - x_{i,j}) \leq t_j \leq T_j + w_j + M(1 - x_{i,j}), \quad \forall (i,j) \in A_d,\ j \in I \backslash \{C\} \tag{17}$$

$$D_i - D_j - M(1 - y_{i,j}) \leq \sum_{(j,ii) \in A_d} x_{j,ii} - \sum_{(ii,j) \in A_d} x_{ii,j}, \forall ((i,j) \neq C) \in A_t \tag{18}$$

$$D_i - D_j + M(1 - y_{i,j}) \geq \sum_{(j,ii) \in A_d} x_{j,ii} - \sum_{(ii,j) \in A_d} x_{ii,j}, \forall ((i,j) \neq C) \in A_t \tag{19}$$

$$D_i + \sum_{(ii,C) \in A_d} x_{ii,C} \leq n + M(1 - y_{i,C}), \quad \forall (i,C) \in A_t \tag{20}$$

$$D_i + \sum_{(ii,C) \in A_d} x_{ii,C} \geq n - M(1 - y_{i,C}), \quad \forall (i,C) \in A_t \tag{21}$$

$$D_C + \sum_{(C,jj) \in A_d} x_{C,jj} \geq n - M(1 - y_{C,j}), \quad \forall (C,j) \in A_t \tag{22}$$

$$D_C + \sum_{(C,jj) \in A_d} x_{C,jj} \leq n + M(1 - y_{C,j}), \quad \forall (C,j) \in A_t \tag{23}$$

$$\sum_{(i,k) \in A_d} x_{i,k} \leq n \sum_{(i,j) \in A_t} y_{i,j}, \quad \forall i \in I \tag{24}$$

$$\sum_{(k,j) \in A_d} x_{k,j} \leq n \sum_{(i,j) \in A_t} y_{i,j}, \quad \forall j \in I \tag{25}$$

$$\frac{\delta}{m} \sum_{(i,j) \in A_d} x_{i,j} + \frac{\theta}{m} \sum_{(j,i) \in A_d} x_{j,i} \leq w_j, \quad \forall j \in I \tag{26}$$

The objective of model (1) consists of two parts to minimize the total truck operation time and the drone flight time. The first part is to minimize both the operation time and the wait time of the truck at all stop points. The second part is to minimize the total time of drone operation including the flight time, service time, and idle hovering to synchronize with the truck schedule. The truck starts and ends its journey from and to the depot (2), while maintaining the flow conservation (3). Constraints (4) and (11) are to eliminate sub-tours for the truck and drone path. Constraints (5) and (6) are used to keep track of the order of visits that the truck makes to form a path. Similarly, constraints (14) and (15) are used to keep track of the order of visits to nodes by drones. To synchronize the movements between the truck and drones, the truck must be present at the launch and return locations of a drone: constraints (16) and (17). A drone will hover at a patient's location during the delivery service, and the hovering time must not exceed the allowed maximum time (8). Similarly, the truck's waiting time at any stopping point must be less than the maximum waiting time (7). Adding the hovering time for drones and waiting time for the truck also adds flexibility in the schedule so that the truck and drones can rejoin at return locations. Each patient needs to be served once. Therefore, only one drone must be assigned to serve the patient (9). Constraints (12) and (13) limit drones to fly within the allowed maximum flight range ($R_{\max}$). The next set of constraints (18)-(23) are for the flight assignment based on the availability of drones; the constraints conserve the flow of the drones in the G_t (#drones in-going = #drones out-going in each intermediate node). Constraints (24) and (25) are to synchronize the launch and retrieval location by the truck and drones. Constraint (26) assures that the setup time for departure flights and the retrieval time for arrivals do not exceed the allotted waiting time in any intermediate node. Note that, based on the number of operators (m) and waiting time (w), the available man-hour will be $m \times w$.

4 Solution Technique

The MDRP is an *NP*-hard problem because a simpler version of the problem, a modified Location Routing Problem, is known to be *NP*-hard [21]. Therefore, most exiting methods for solving an MDRP model often resort to developing computationally efficient algorithms to find good feasible solutions. In this paper, we propose a decomposition method coupled with fast heuristic algorithms to expedite the solution process. In the decomposition scheme, the original problem is decomposed

into a truck scheduling problem and a flight scheduling problem so that each problem can be solved much faster than solving the complex model altogether.

4.1 Decomposition scheme

The MDRP constraints can be categorized into three groups: "Truck related", "Drone related", and "Mutual". "Truck related" refers to constraints (2) to (7) and they are associated with designing the truck schedule only. Constraints (8) to (15) belong to "Drone related", which apply to the drone flight scheduling alone. Note that these two groups of constraints are mutually exclusive. "Mutual" is the group of constraints (16–26) that enable the collaboration between the truck and drones. Based on these categories, we can transform the model into a bi-level model structure:

TS: The upper-level problem is to optimize the truck schedule (TS). Based on the sequence of serving patients, which is determined at the lower level, the goal is to find an optimal schedule for the truck and assign launch-return locations for drones. Thus, the optimization model (see Section 4.1.1) contains "Truck related" constraints as well as "Mutual" constraints of MDRP. Note that the "Drone related" constraints are already satisfied at this stage.

FS: For a given truck schedule, the lower-level problem is optimized the drone flight schedule (FS). The model (see Section 4.1.2) contains "Drone related" and "Mutual" constraints only.

Our proposed solution approach recursively solves TS and FS, in which a solution from one helps improve the solution for the other. This iterative process continues until the solution can no longer be improved.

4.1.1 Optimize truck schedule for a given sequence of flight sequence

Sets, parameters, and variables are presented in Table 3. For a given sequence of customers to visit, the objective of TS is to find the shortest truck path with complete drone flights along the path. A complete drone flight schedule includes the sequence of customers and the launch ($l \in I$) and return ($r \in I$) locations to join the truck along the truck path. For all given sequences of customers, we need to build the flight. For example, SP^i constitutes of b customers and its launch and return nodes are not determined.

$$SP^i = c_1^i \rightarrow c_2^i \rightarrow ... \rightarrow c_b^i \tag{27a}$$

By adding $l, r \in I$ to the SP^i, it turns into a complete flight $F_{l,r}^i$;

$$F_{l,r}^i = l \rightarrow c_1^i \rightarrow c_2^i \rightarrow ... \rightarrow c_b^i \rightarrow r \tag{27b}$$

And a new binary decision variable ($f_{l,r}^i$) will be introduced whether to choose the $F_{l,r}^i$. The flight duration of $F_{l,r}^i$, ($t_{l,r}^i$) is calculated as follows:

$$t_{l,r}^i = wd_{l,c_1^i} + \sum_{j=1}^{b-1} wd_{c^i j, c^i j+1} + wd_{c_b^i, r} + \sum_{j \in SP^i} (s_j + h_j) \tag{27c}$$

This reformulation helps reduce a considerable number of variables and constraints in MDRP and, therefore, TS is much easier to solve than the original MDRP model.

$$\min_{y,\, f,\, w,\, T,\, D} \quad TS = \sum_{i,j \in A_t} wt_{i,j} y_{i,j} + \sum_{(l,i,r) \in LFR} t_{l,r}^i f_{l,r}^i + \sum_{i \in I} w_i \tag{28}$$

s.t.

$$\text{(2)-(8)} \tag{29}$$

Table 3: Sets, parameters, and variables that are used in truck schedule optimization based on given patients sequences.

Sets:		
	Notation	Description
	SP	Set of visiting sequences for the patients. $SP = \{SP^i \mid SP^i \subset N,\ \forall i, j\ SP^i \cap SP^j = \phi,\ \bigcup_{i=1}^{\lvert SP\rvert} SP^i = N\ \ i = 1, ..., \lvert SP\rvert\}$
	LFR	Set of possible flights within the maximum flight range. $LFR = \{(l, i, r) \mid F^i_{l,r} = \{l\} \cup SP^i \cup \{r\},\ l, r \in I,\ t^i_{l,r} \leq R_{\max}\}$
Parameters		
	Notation	Description
	$t^i_{l,r}(\min)$	The flight duration of $F^i_{l,r} \in LFR$ including service, flight, and hovering time.
Variables		
	Notation	Description
	$f^i_{l,r} \in \{0, 1\}$	If a drone assigned to serve SP^i, chooses node l,$r \in I$ as its launch and return nodes, then it gets 1 otherwise 0.

$$T_r - T_l - w_l \leq t^i_{l,r} + M(1 - f^i_{l,r}) \quad \forall (l, i, r \neq C) \in FLR \tag{30}$$

$$t^i_{l,r} - M(1 - f^i_{l,r}) \leq T_r + w_r - T_l \quad \forall (l, i, r \neq C) \in FLR \tag{31}$$

$$\sum_{\substack{(l,ii,r)\in FLR,\\ ii=i}} f^i_{l,r} = 1,\ \ i = 1, ..., |SP| \tag{32}$$

$$D_l - D_r \leq \sum_{(r,i,j)\in LFR} f^i_{r,j} - \sum_{(k,i,r)\in LFR} f^i_{k,r} + M(1 - y_{l,r})\ , \forall (l, r) \neq C \in A_t \tag{33}$$

$$D_l - D_r \geq \sum_{(r,i,j)\in LFR} f^i_{r,j} - \sum_{(k,i,r)\in LFR} f^i_{k,r} - M(1 - y_{l,r})\ , \forall (l, r) \neq C \in A_t \tag{34}$$

$$D_l + \sum_{(l,i,C)\in FLR} f^i_{l,C} \leq n + M(1 - y_{l,C}), \quad \forall (l, C) \in A_t \tag{35}$$

$$D_l + \sum_{(l,i,C)\in FLR} f^i_{l,C} \geq n - M(1 - y_{l,C}), \quad \forall (l, C) \in A_t \tag{36}$$

$$D_C + \sum_{(C,i,r)\in FLR} f^i_{C,r} = n \tag{37}$$

$$\sum_{(l,i,r)\in FLR} f^i_{l,r} \leq n \sum_{(l,j)\in A_t} y_{l,j} \quad \forall l \in I \tag{38}$$

$$\sum_{(l,i,r)\in FLR} f^i_{l,r} \leq n \sum_{(k,r)\in A_t} y_{k,r} \quad \forall r \in I \tag{39}$$

$$\frac{\delta}{m} \sum_{(l,i,j)\in FLR} f^i_{l,j} + \frac{\theta}{m} \sum_{(j,i,r)\in FLR} f^i_{j,r} \leq w_j \quad \forall j \in I \tag{40}$$

The goal of TS is to build the truck path and drone flight schedule to minimize the total flight time for drones and travel time for trucks (28). Constraints (29) are adopted from Section 3 to create feasible flight paths. The constraints (30) and (31) are to synchronize the drone departure and return to join the truck based on the truck schedule. If the return location is the depot, the delivery mission is complete, and there is no need to coordinate with the truck schedule anymore. Constraint (32) to ensure that there will be one launch node (location) and one return node for the sequence. (33)-(40) are the updated version of (18)-(26) due to variable f.

Table 4: New or modified sets, parameters, and variables that are used in Flight schedule optimization based on a given truck path.

Sets		
	Notation	Description
	P	Itinerary of truck. $P : C \rightarrow\ p_2 \rightarrow\ ...\ \rightarrow\ p_{\lvert P\rvert-1}\ \rightarrow\ C$
	A'_d	$\{(i,j)\mid\ i \neq j \in P \cup N,\ i\&j \notin P\}$
Parameters:		
	Notation	Description
	$G'_d(\{P \cup N\}, A'_d)$	The new network graph of flying drones based on new truck path (P). G'_d is a subgraph of G_d.
	$w_{p_i}(\min)$	The allotted waiting for truck in node $p_i \in P$.
	$T_{p_i}(\min)$	Entrance time of truck at node $p_i \in P$. This value includes all the travel and wait times till the node p_i. $T_{p_{i-1}} + w_{p_{i-1}} + wt_{p_{i-1},p_i} = T_{p_i}, i = 2, ..., \lvert P\rvert - 1$

4.1.2 Optimize Flight Schedule based on a given truck schedule

For a given truck schedule, the lower level problem is to find a drone flight schedule to visit customers following the tuck path. Using sets, parameters, and variables of this level defined in Table 4, the optimization model is formulated as follows:

$$\min_{x,h,t,v,D,b} \quad FS = \sum_{i,j \in A'_d} (wd_{i,j} + s_i)x_{i,j} + \sum_{i \in N} h_i \tag{41}$$

s.t.

$$(8) - (15), (26) \tag{42}$$

$$T_{p_i} - M(1 - x_{p_i,j}) \leq t_{p_i} \leq T_{p_i} + w_{p_i} + M(1 - x_{p_i,j}) \quad \forall p_i \in P, \forall (p_i, j) \in A'_d \tag{43}$$

$$T_{p_j} - M(1 - x_{i,p_j}) \leq t_{p_j} \leq T_{p_j} + w_{p_j} + M(1 - x_{i,p_j}) \quad \forall p_j \in P, \forall (i, p_j \neq C) \in A'_d \tag{44}$$

$$D_{p_{i-1}} - D_{p_i} = \sum_{(j,p_i) \in A_d} x_{j,p_i} - \sum_{(p_i,j) \in A'_d} x_{p_i,j}\ , i = 2, ..., \lvert P\rvert \tag{45}$$

$$D_C + \sum_{(C,i) \in A'_d} x_{C,i} = D_{p_{\lvert P\rvert-1}} + \sum_{(j,C) \in A'_d} x_{j,C} = n \tag{46}$$

The objective is to minimize the total flight time of drones to complete all visits along the truck path. The first term of FS is to minimize the total flight time and the second term is to minimize the total hovering time of drones. Constraints (43) and (44) synchronize the truck schedule and the drone flight schedule to coordinate the efforts on launching and retrieving drones. Constraints (45) and (46) are based on available drones (idle ones in the truck + returned ones) to determine the number of drones to be launched for the next flight. They are equivalent to (18) and (23) when the truck path is already determined.

4.2 The proposed solution approach

The general procedure to solve MDRP using the bi-level framework discussed in Section 4.1 is shown in Algorithm 1. The algorithm consists of three stages: Initialization, Heuristic Loop, and

Final Loop. Our goal is to solve the two optimization models (TS and FS) corresponding to the truck schedule and the drone flight schedule, respectively. However, there are two challenges. First, finding an initial feasible solution is not easy. To address this, a heuristic algorithm TSP_{BFS} is developed. Second, solving those models solely based on the initial solution still can take a very long time. Hence, a heuristic algorithm ("Heuristic Loop") has been developed to find a good feasible solution so that the optimization models can converge fast using the heuristic solution as a warm-up solution.

Initialization: The main purpose of this stage is to provide good and fast warm-up solutions for both TS and FS to solve the problem faster. As an initial solution, a set of single flights to visit each of the patients is constructed. For instance, the flight that serves the i^{th} patient is $(C \to i \to C)$. Some of the flights may be infeasible due to the violation of flight range and availability of drones. The TSP_{BFS} (see Section 4.2.1) is to design a truck path (P_0) that guarantees the feasibility of single-visit flights by choosing an appropriate location for launching and retrieving the drones. The next step in "Exchange" is to search for a better solution by allowing multiple visits by a drone with the aim to reduce the total flight time to serve all patients for a given truck path (P_0). The stopping criterion for solving the TS and FS is initialized here including solver run-time (tm).

Heuristic Loop: Given the initial solution from the previous step (i.e., P_{i-1} and F_{i-1}), the purpose of this stage is to find a near optimal solution for both TS and FS in a fast manner. This loop consists of solving $TS$and the $Exchange$ heuristic algorithm. The TS will be solved until one of the termination criteria is met. In the case that TS fails to improve the truck path, its heuristic equivalent (TSP_{BFS}) is executed to improve the current truck path if possible. If a new improved truck path (P_i) is found, its corresponding drone flight schedule is optimized using $Exchange$.

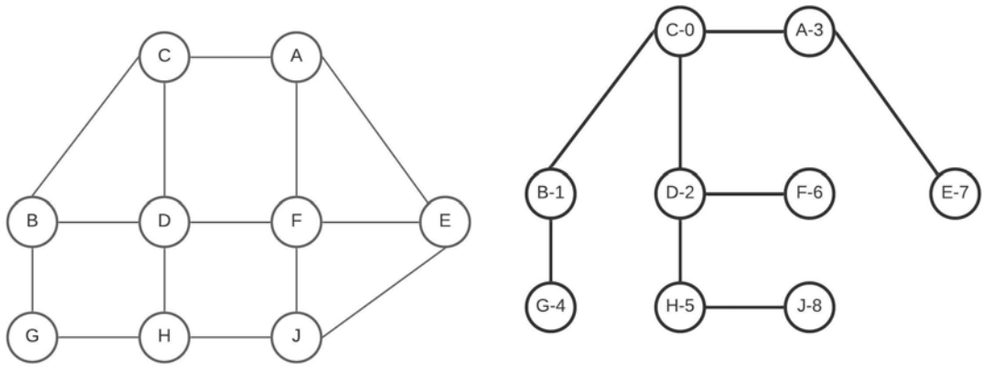
Final Loop: Based on the warm-start from the previous step, this is the final attempt to reach optimality, the TS and FS are solved recursively until no further improvement is made. First, TS is solved to improve the truck schedule, followed by solving FS for the corresponding drone schedule.

Data: $G_t(I, A_t), G_d(I \cup N, A_d), wd, wt, s, N, I, n, M$
Result: Optimal Truck path (P^*), and Flight paths (F^*)
\# **Initialization**;
$F_{-1} = \{(C \to i \to C) | i \in N\}$;
$P_0 = TSP_{BFS}(F_{-1})$;
$F_0 = Exchange(P_0, F_{-1})$;
$tm = tm_{max}$;
\# **Heuristic Loop**;
while *TS or Exchange finds better solution* **do**
  $P_i = TS(P_{i-1},\ F_{i-1}, tm)$;
  if *TS does not improve the truck path in tm unit of time* **then**
    $P_i = TSP_{BFS}(F_{i-1})$
  end
  $F_i = Exchange(P_i, F_{i-1})$;
  $i+ = 1$;
end
#**Final Loop**;
while *TS or FS provides new solution* **do**
  $P_i = TS(P_{i-1},\ F_{i-1}, tm)$;
  $F_i = FS(P_i,\ F_{i-1}, tm)$;
  $i+ = 1$;
end

Algorithm 1: The procedure of solving the MDRP

4.2.1 Heuristic Algorithms

Algorithm 1 contains two heuristic algorithms: TSP_{BFS} and $Exchange$. This section describes details on the algorithms.



(a) A sample truck network, G_t, with 9 nodes

(b) The BFS tree and the ranking of the nodes based on their closeness to the root. The rank of each node is added to its node label.

Figure 2: The node rankings based on closeness of the nodes to depot by the BFS

TSP with Breadth First approach (TSP_{BFS}): This is an iterative algorithm that solves a modified travelling salesman problem (mTSP) at each iteration. For a given sequence of patients (SP), the objective of $mTSP$ is to find a feasible and short distance truck path (P) to cover those patients. Unlike TSP that requires visiting all locations, mTSP constructs a tour covering a subset (MV) of the intermediate nodes, $MV \subseteq I$. With the aim to find a short distance truck path, we use the search tree structure (G_t) consisting of a root node, which is the drone launch location including the depot (C). The BFS ranks the nodes based on their closeness [1] to the root node. For example, consider a sample network, Figure 2a, with 9 nodes. By running the BFS on the graph having C as the root node, Figure 3c shows the ranked order of the nodes sorted by two principles: (1) the proximity to the depot and (2) the left-node first. Hence, node B has the highest ranked order (B-1) because it is one unit away from the root node, and it is located far left among other nodes (D-2 and A-3) having the same unit distance from C. The rest of the rankings follow the same principle to construct the spanning tree shown in Figure 3c. The result is *Ranking List* (R) containing the nodes in ascending order, i.e., $R = [C, B, D, A, G, H, F, E, J]$.

The next step is to construct MV following Algorithm 2. This algorithm contains a loop to search for a feasible truck path ($\delta = 1$). As the first step, MV is initialized as an empty set (i.e., $MV_0 = MV$) and $R_0 = R$; hence, no truck path is available and the feasibility indicator δ is zero. At the i^{th} iteration, we select and remove the first element in R_{i-1} and add it to MV_{i-1} to build MV_i. Then, $mTSP(MV_i)$ is solved to obtain a new truck path (P_i). The next step is to solve a MIP model **Launch-Return Optimizer (LRO)** (see Appendix A). The goal of LRO is to find the best launch and return locations on P_i for each sequence of patients in SP so as to minimize the total flight time. The procedure for building a flight path for a sequence of patients in LRO is similar to TS (see Equation (27)). If the LRO returns a feasible set of flights (F), then the P_i and SP are considered compatible and it will output $\delta = 1$; otherwise, it will output $\delta = 0$ and loop repeats.

[1] The closeness is measured by the number of nodes between the current node to the root node. A lower number represents a closer distance.

Data: $G_t(I, A_t), SP, wd, wt, s, N, I, n, M$

Result: feasible truck path (P), and feasible flight paths (F)

initialization;

$R_0 = BFS(G_t, C)$# Run BFS and R will the sorted set of nodes based on their closeness to depot(node C);

$MV_0 = [\ ]$;

$\delta = 0$;

$i = 1$;

Main Loop ;

while $\delta = 0$ **do**

  $MV_i = MV_{i-1}$.append($R_{i-1}[0]$);

  $R_i = R_{i-1}$.remove($R_{i-1}[0]$);

  $P_i = mTSP(MV_i)$;

  $[F, \delta] = LRO(P_i, SP)$;

  $i+ = 1$;

end

Algorithm 2: The procedure of TSP_{BFS} to build feasible truck path with a given set of sequences of patients (SP).

Figure 3 shows the steps of the TSP_{BFS} to achieve the feasible truck path for G_t with 9 nodes and 3 patients.

Exchange: This iterative algorithm is designed to improve the solution resulting from the previous step to minimize the total flight time. For a given set of flights (F) and truck path (P) at each iteration, a pair of flights (e.g. $F^k_{old}, F^o_{old} \in F$) is selected as input to a modified version of FS (mFS) to find a better solution (e.g., F^k_{new} and F^o_{new}). The selection of a pair of two flights is based on a metric representing the Euclidean distance between two centroids. Suppose a flight path begins from a drone launching location followed by a sequence of locations to visit until it is retrieved by the truck at the collection location. We can view this in a graph where each location is associated with two-dimensional (2D) coordinates. Hence, a centroid for each flight can be calculated based on the set of x and y coordinates of the locations. Once the centroid calculations are completed for all flights, a distance matrix between any two flight paths is constructed as the Euclidean distance between two centroids corresponding to two flight paths. We then select a pair of two flights whose distance is the smallest. If a tie occurs in selecting two paths, a pair is selected randomly among the candidates. The loop continues until no further improvement can be made.

Data: $P, F_{\text{old}}, Sq, wd, s, N, I, n, M$

Result: F_{new}

while *($\exists F^k_{old}, F^o_{old} \in F_{old}$) and ($mFS(F^k_{old}, F^o_{old})$ lowers FS objective value)* **do**

  $[F^k_{new}, F^o_{new}] = mFS(F^k_{old}, F^o_{old})$;

  update F;

end

Algorithm 3: The Exchange algorithm to improve flight time

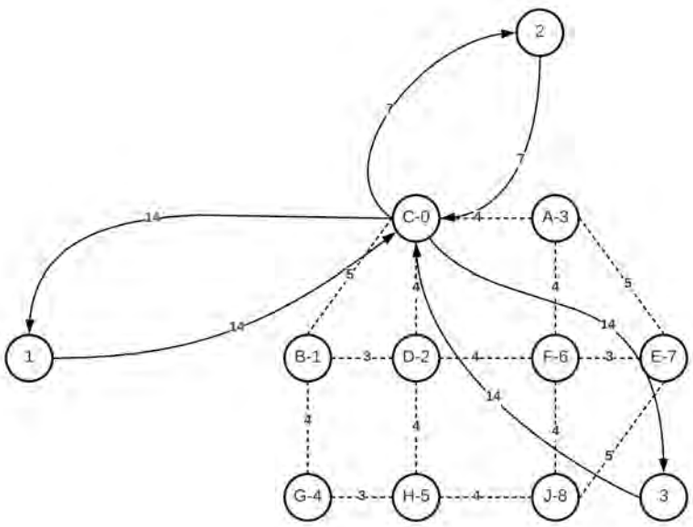
The algorithm includes solving mFS, which is an MIP model shown as (47-49). Compared to FS, the mFS model has an additional constraint (49) that keeps the itinerary of the rest of the flights (F') fixed, where $F' = F \backslash \{F^k, F^o\}$.

$$\min_{x,h,t,v,D,b} \quad mFS(f^k, f^o) = \sum_{i,j \in A'_d} (wd_{i,j} + s_i)x_{i,j} + \sum_{i \in N} h_i \tag{47}$$

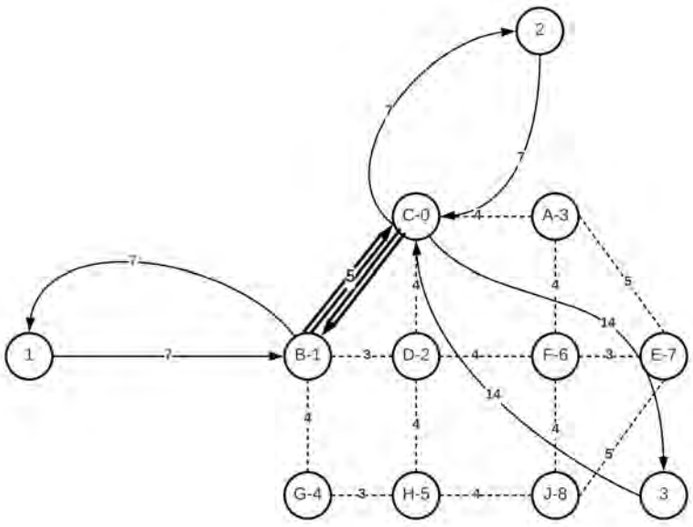
$$\text{s.t.} \quad (42) - (46), \tag{48}$$

$$x_{F^q_i, F^q_{i+1}} = 1, \quad i = 1, ..., |F^q| - 1, \forall F^q \in F' \tag{49}$$

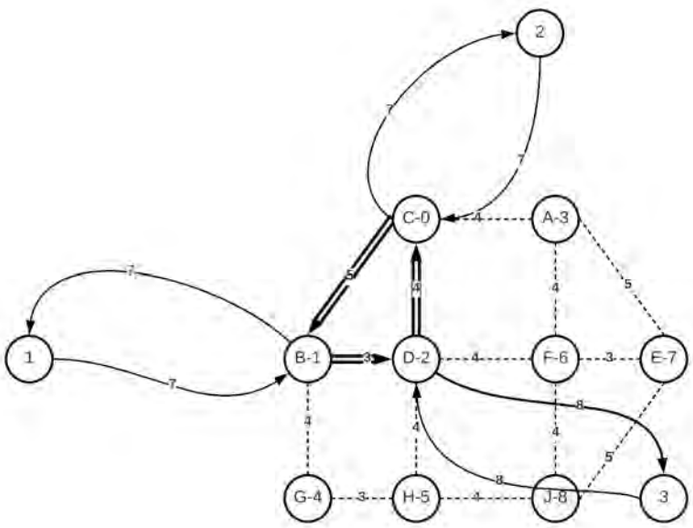
Note that constraint (49) contributes to a huge reduction in solution space of FS and a substantial reduction can be achieved in computational time for solving mFS.



(a) The initialization step of TSP_{BFS}. Choosing the depot (C) as launch and return nodes results of infeasiblity of two flights out of three due to violation of maximum flight range and therefore $\delta = 0$.



(b) Expand the truck path by adding B as the first element in R_1. The infeasiblity of one of flights has been resolved but there exists one forbidden flight and therefore $\delta = 0$.



(c) Expand the truck path by adding the D. All the flights are feasible and therefore $\delta = 1$.

Figure 3: An illustrative example of TSP_{BFS} algorithm of a sample G_t with 9 nodes and 3 patients.

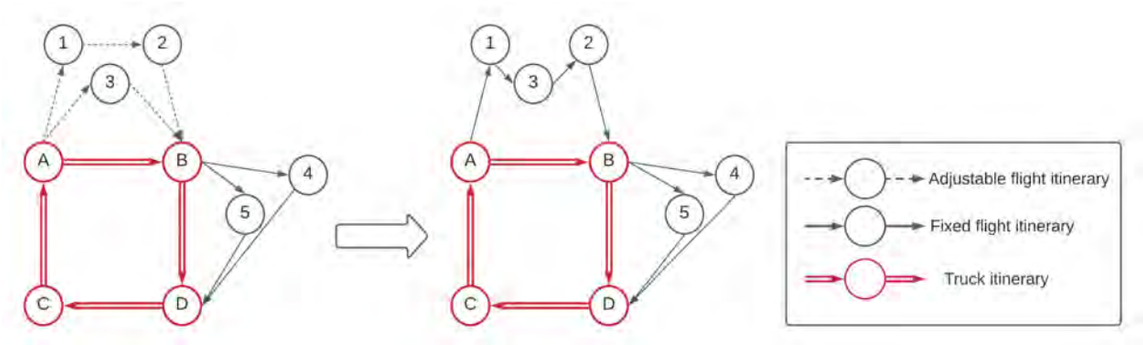


Figure 4: An illustrative example of *Exchange*

Table 5: An example of one iteration of *Exchange* algorithm to improve fights

	Old	New
Adjustable flight itinerary ($F \backslash F'$)	$A \to 1 \to 2 \to B$ $A \to 3 \to B$	$A \to 1 \to 3 \to 2 \to B$
Fixed flight itinerary (F')	$B \to 4 \to D$ $B \to 5 \to D$	$B \to 4 \to D$ $B \to 5 \to D$

4.2.2 An illustrative example of *Exchange*

Figure 4 shows one iteration of the *Exchange* algorithm to improve flight paths for a problem instance with five patients. The set of flight paths are:

$$F = \{F^1 :\ A \to 1 \to 2 \to B,\ F^2 :\ A \to 3 \to B,\ F^3 :\ B \to 4 \to D,\ F^4 :\ B \to 5 \to D\} \tag{50}$$

Among all possible six pairs of flights, flights F^1 and F^2 are the closest two flights and they are selected as input to mFS. The remaining flights (i.e., F^3 and F^4) remain unchanged.

$$F' = F \backslash \{F^1, F^2\} = \{F^3 :\ B \to 4 \to D,\ F^4 :\ B \to 5 \to D\} \tag{51}$$

The following set of constraints (52) in mFS keep those flights in F' unchanged while attempting to find an improved solution:

$$x_{i,j} = 1, \qquad \forall (i,j) \in \{(B,4),(4,D),(B,5),(5,D)\} \tag{52}$$

Because the paths for serving patients 4 and 5 are predetermined, the mFS can focus on covering patients 1, 2, and 3 only. This reduction in the number of patients to serve reduces the search space and results in reduced computational time. The output of $mFS(F^1, F^2)$ shows that F^1 and F^2 must merge into a new flight to be able to lower the total flight time (see Table 5).

5 Numerical Experiments

In this section, we apply the MDRP model to illustrate our solution technique and to demonstrate the applicability of our model to help reduce the spread of a pandemic. The model and corresponding algorithms were implemented in Python 3.7 [48] and Gurobi 9.1.0 (Python API)[19]. All experiments were performed on a server running RedHat Linux 64-bit with 24 core Intel Xeon processors and 384GB RAM.

5.1 Drone and Truck Scheduling

Our application is based on the city of Sioux Falls, South Dakota, U.S. (see Figure 5)[4]. This is a medium-sized city with a population of 150,000, covering the characteristics of large cities and rural areas. On average, the city had around 37.5 COVID-19 confirmed cases per day[9]. We selected

this area for the experiments because the area is a reasonable size in which the patients can be covered by a single truck and multiple drones. This network segment of Sioux Falls has 40 patients (i.e., yellow nodes within a 4-*mile* radius, and they are labeled from 1 to 40 in Figure 5 to be served by one truck with three drones. The routes that the truck can use are based on the existing road network (i.e., bidirectional arcs in Figure 5. The truck must stop to release drones with testing kits or to pick up the drones once their services are completed. The road conditions limit the number of points that the truck can stop to 24 (i.e., red nodes labeled from 'A' to 'X' in Figure 5. Each drone can fly for a maximum of 26 *min* at an average flight speed of 20 *mph* and that the atmospheric conditions are stable during delivery. The truck can run for 200 *min* at an average speed of 20 *mph*. Note that the speeds of drones and the truck were set conservatively (i.e., slow) taking into account unforeseen obstacles and traffic congestion during the delivery. Therefore, the results discussed in this section can be viewed as worst-case scenarios. In practice, the truck can travel faster than 20 mph, which will result in a shorter total delivery time. Furthermore, we assume that the service time per patient is expected to last 3 *min* including delivery of the test kit, and the waiting time for drones landing at each truck stop is expected to be 5 *min*.

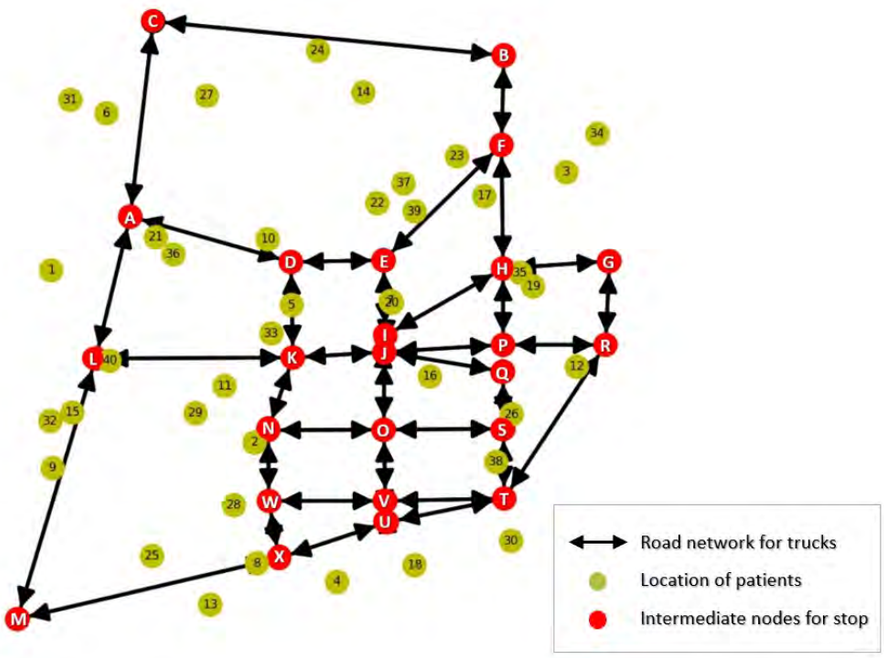


Figure 5: Numerical Example with 40 patients and the road network(G_t)

The optimal routes for the truck and three drones for serving 40 patients are shown in Figure 6, where black arcs indicate the truck path while green arcs indicate the drone paths. The three drones take off from the truck at an intermediate node and return to the truck at a designated (intermediate) node. The number of patients being served on a given path varies truck specification and the drones on-board (i.e., maximum drone flight range and the truck's stopping points). The three drones return to the depot (i.e., the red node labeled 'C' in Figure 6) rather than the truck after performing their final delivery mission from intermediate nodes 'A' and 'L', respectively.

the Gantt chart in Figure 7 illustrates specific sequences of events (i.e., providing service or waiting to provide service) for the truck and the three drones. The truck stops at 13 locations following the road network. At these stopping points, the truck either picks up a drone that has completed delivery or releases a drone for the next service assignment. For our example, it took a total of 98 minutes to deliver a testing kit to each of the 40 patients.

5.2 Discussion on the Solution Approach

The performance of a heuristic method is measured by the speed for solving the problem and the quality of the solution that it provides. We demonstrate the performance of the proposed solution approach as discussed in Section 4.2 using a problem instance with 40 patients and 3 drones. Figure 8 shows the progression of Algorithm 1 to solve the problem. At time zero ($t = 0$),

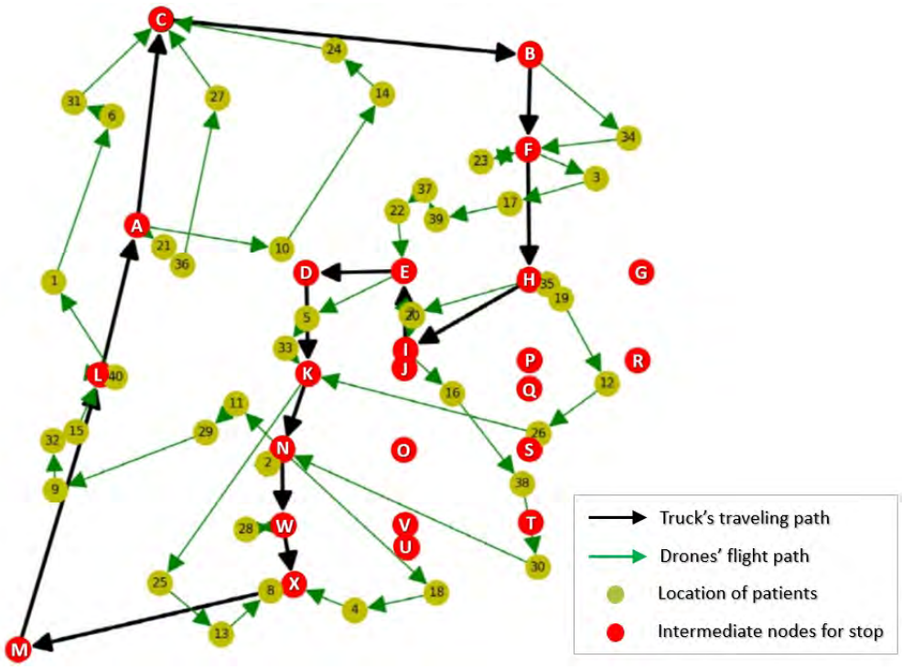


Figure 6: Optimal Flight Schedule and Drone Assignment to Patients

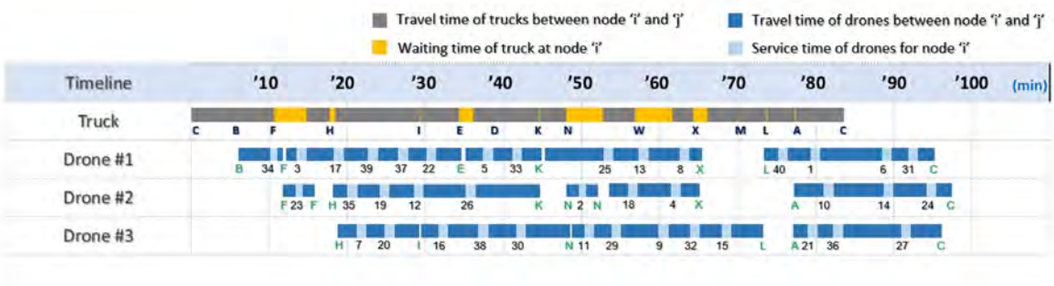


Figure 7: Optimal Schedule for One Truck and Three Drones

the **Initialization** stage starts and TSP_{BFS} returns an initial solution (a set of flights) based on visiting one patient at each flight. In the following step ($t \in [5, 1900]$), *Exchange* takes the initial solution and attempts to minimize the total flight duration. As a result, the number of flights required to visit all patients is reduced by 75%.

In the next step, **Heuristic Loop** started and the loop repeated twice until no further improvement was made, $t = 2600$. As seen in Figure 8, the reduced total flight time achieved in the previous stage helped reduce TS as well as the value of TS^1_{UB}, and it reached the optimality gap of 10% in less than 200 seconds. **Final Loop** started at $t = 2600$ with an attempt to improve the lower-bound of the FS model (FS^4_{LB}). Since it could not find any improvement in FS_{UB} after 600 seconds, the algorithm was terminated.

This example demonstrates the advantage of using the decomposition and the recursive optimization proposed in this paper, in which an improvement made in one level directly helps improve the solution in the next step. As a comparison to the exact method without the decomposition approach, the MDRP took over 1 hour just to find an initial feasible solution. Then, the Branch-and-Bound algorithm took over 10 hours to reach 20% optimality gap before it was terminated.

5.3 The effect of vehicle speed on total completion time

Vehicle speed can affect the delivery completion time. The network as shown in Figure 5 with 40 customers is used to investigate the effect of the total completion time (TCT) as we vary the vehicle speed. For a fixed drone speed of 20 *mph*, we vary the truck speed at 15, 20, 25, 30, 35, 40 *mph* and the results are plotted in Figure 9a. The total completion time (i.e., $\sum_{(i,j)\in A_t} wt_{i,j} y^*_{i,j} + \sum_{i\in I} w^*_i$) appears to decrease as the truck speed was increased until 35 *mph*. But the trend sharply reversed after 35 *mph*. This is because at beyond 35 mph, the waiting time of the truck to collect the drone

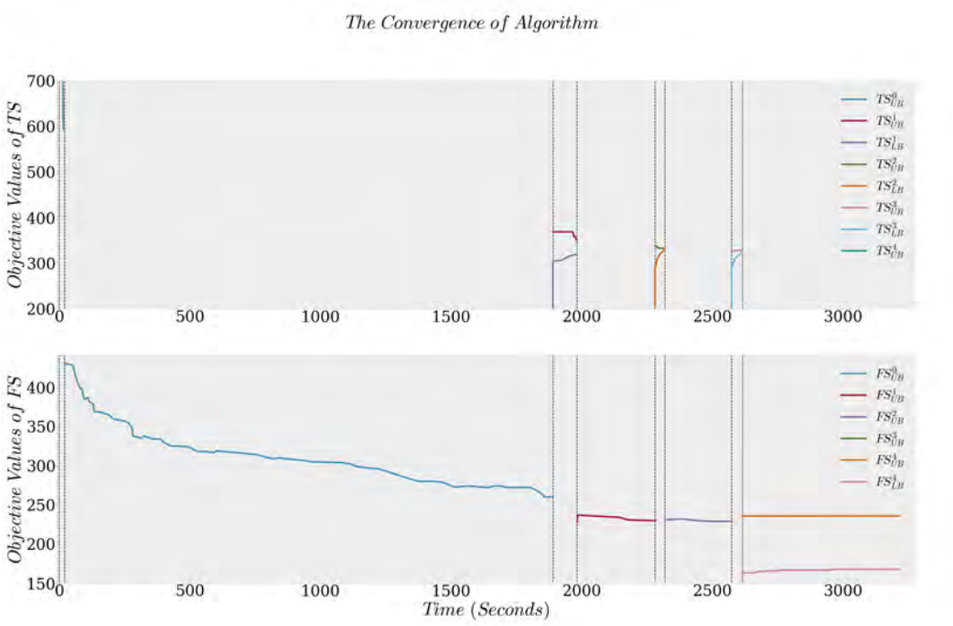
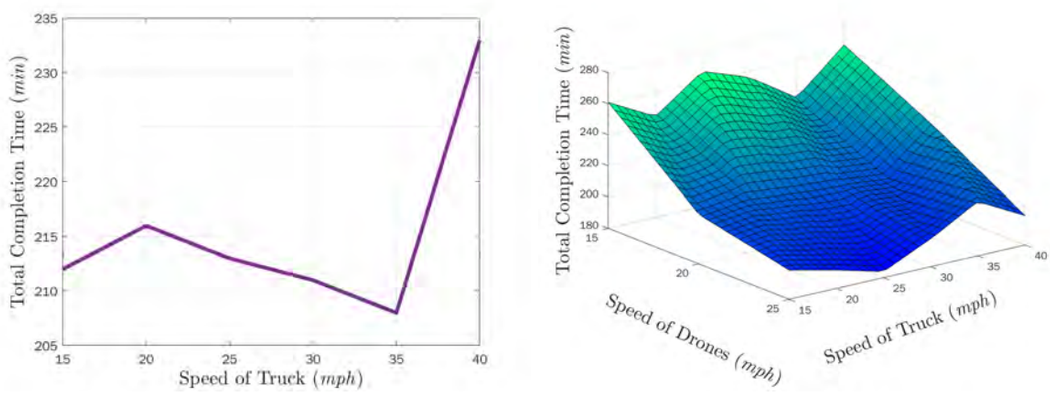


Figure 8: The convergence of TS and FS to optimal solution

increases. Hence, we further investigate the effect of the TCT under 18 different vehicle scenarios: six different speeds for the truck at 15, 20, 25, 30, 35, 40 *mph* and three different speeds at 15, 20, 25 *mph* for the truck. Note that the drone speed was capped at 25 *mph* because most small drones fly within this range.



(a) Total completion time .vs. truck speed.

(b) Total completion time .vs. vehicle speed

Figure 9: Total completion time .vs. vehicle speed

Figure 9b shows the results of the experiments. We made following observations. First, the TCT decreased as both vehicle speeds were increased up to a certain point. Second, when the speed of one vehicle was fixed, increasing the speed of the other vehicle did not result in a monotonic decrease in the TCT. Third, when the truck traveled much faster than the drone, the truck ended up waiting longer for the drone, which resulted in an increase in TCT.

5.4 The effect of the number of drones on total completion time

One can expect that adding more drones can potentially reduce the TCT. It is because having multiple drones allows multiple deliveries to be carried out in parallel. Hence, we investigate the effect of TCT on the number of drones carried on the truck. Figure 10 shows the results of this experiment. As we increase the number of drones, the TCT monotonically decreased until 4 drones and converged at 160. This indicates that adding more drones can help reduce the TCT up to a certain point. Adding more drones beyond this point does will not improve the performance due to other factors affecting the TCT, but add unnecessary additional capital cost and computational

burden for solving the optimization model.

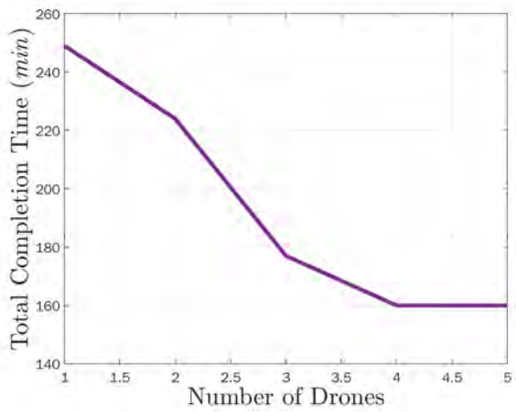


Figure 10: The total completion time behaviour by increment in number of drones.

5.5 Public Health Implications with the MDRP Model

To determine the public health implications with the MDRP, we propose a modified metric (i.e., mR_0) of the basic reproduction number (R_0) to estimate how many secondary cases might arise from a single infected case [13]. In general, R_0 value greater than 1 indicates that the disease is spreading and the outbreak is continuing, while R_0 value below 1 indicates that the disease is waning and the outbreak is ending. The basic reproduction number is generally determined by a disease's infection period (e.g., the longer the infection period, the more contagious an infected individual will be with the disease) and contact rate (i.e., the number of people an infected individual comes into contact with). The contact rate can be reduced through well-established public health interventions that require individuals to socially distance, quarantine, and/or stay at home [22]. Mathematically, the basic reproduction number is calculated as $R_0 = \beta S/\gamma$, where β is the transmission rate of the disease, S is the number of susceptible individuals in a given population, and γ is the disease's infection period. However, the basic reproduction number is limited when used to calculate the risk of infection in a specific region or at a specific point in time[11][15]. As such, the R_0 can be transformed to accommodate geographic, population, and temporal modifications [45][14]. For the MDRP model, we modify the disease's infection period to scale it to the total scheduled time as:

$$\bar{\gamma} = \frac{\text{Infection period (in days)}}{\text{Total schedule time from MDRP (in days)}} \tag{53}$$

Accordingly, the modified reproduction rate is calculated as:

$$mR_0 = \beta S/\bar{\gamma} \tag{54}$$

Similar to R_0, mR_0 is defined so that a smaller value corresponds to a better outcome (i.e., fewer infections). The R_0 is designed to measure the number of people who might be infected during a complete incubation period (e.g., approximately 14 days for COVID-19 [10]). Therefore, it is not suitable to predict the number of possible infections during a specific time period Hence, we revised the metric by introducing $\bar{\gamma}$ to predict the number of possible infections over a specific period. Considering the characteristics of face-to-face diagnosis and testing kit delivery, it is possible to predict more accurately the number of infections that may occur during a specific period by assigning an appropriate transmission rate. Table 6 shows an example to show how this new measure works and a comparison between a face-to-face testing approach and the drone-based approach. In the case of face-to-face diagnosis, it takes 3.3 hours to diagnose 40 patients, during which about 0.0153 additional infections could occur. For the same problem, the drone-based testing kit delivery would take 1.63 hours to diagnose those patients. It is estimated that approximately 0.002 additional infections could occur during this time. Recall that if patients have to travel to a

	β **(Transmission rate)**	**S (no of person)**	$\bar{\gamma}$ **(Time period)**	mR_0
Face-to-face test (Drive-thru site)	3.1 %	52	101.82	**0.0153**
Testing kit delivery using MDRP	1.0 %	42	205.73	**0.002 (86.9% reduction)**

Table 6: Estimated modified reproduction rate (mR_0)

testing site, there is a possibility of virus transmission through an interaction between the patient and the medical personnel or among the patients themselves. To reduce that likelihood, patients are generally advised to wear an appropriate face mask and follow the social distancing guidelines. Therefore, we expect the MDRP to help significantly reduce the spread of a pandemic.

Consider the example provided in Section 5.1; namely 40 patients and two delivery personnel (i.e., a truck driver and a drone operator). An equivalent drive-through testing site would have the same number of patients but 10 medical personnel (i.e., five per testing to perform all of the activities associated with the testing process). To calculate comparable mR_0 for these scenarios, we note the following. According to Chu *et al.* (2020) if two people wearing masks talk to each other within 6 feet, the transmission rate, β was 3.1 %, and when both people did not wear masks, the transmission rate (β) increased to 17.4 % [12]. During the test for COVID-19 with a face-to-face method, patients need to take off their masks for collecting samples, but the sampling time is very short. Therefore, we assume that the transmission rate of the face-to-face test is 0.031. If the surface of a drone or testing kit is contaminated with the virus, it is possible to spread the virus by contact with it, but we expect it to be extremely low [1]. Hence, we conservatively assume the transmission rate at 0.01. The number of people at risk of infection (S) is 42 for MDRP and 50 for the drive-thru testing site. The average recovery period (γ) (in days) is divided by the total testing (or delivery) time (in days). As in current research [37], we assume that the average recovery period is 14 days. Then, for a face-to-face test, it takes about 10 minutes to collect a sample per patient. Having two test booths on site, the total time required for 40 patients would be approximately 3.3 hours (0.137 of a day). Then, dividing this value by 14 days, the average recovery period of COVID-19 (γ) is 101.82. On the other hand, the minimum required time using the MDRP was 1.63 hours (0.0681 of a day). Accordingly, γ for this method will be 205.73, and mR_0 for the drive-through testing site is 0.0153 while MDRP's mR_0 is 0.002. This means that during the face-to-face test of 40 patients, about 0.0153 new infection cases could occur (approximately one secondary infection per 2,614 patients). With MDRP, it would be one secondary infection per 20,000 patients. Thus, the MDRP has the potential to significantly reduce the transmission of a pandemic.

6 Conclusion

A delivery system consisting of a single truck and multiple drones was studied in this paper, in which a truck was considered as a mothership of drones and the group of drones deliver testing kits for potential patients. The ultimate goal of this study was to show that this hybrid delivery model can be a great alternative to any in-person diagnostic methods for the purpose of infection prevention. Because the optimization model for solving this complex problem is not scalable, a decomposition algorithm has been proposed to develop a coordinated schedule of the truck and the drones in a timely manner. The proposed solution method consists of three stages: Initialization, Heuristic, and Exact algorithm. In the initialization stage, a heuristic algorithm, TSP_{BFS}, is developed to find an initial solution (i.e., feasible truck and flight paths). In the Heuristic stage, *Exchange* heuristic is developed to improve the MDRP solution starting from the initial solution. Using the test problem, we showed that the *Exchange* algorithm works fast in finding an improved solution by utilizing the modified version of FS model and its embedded heuristic. We have demonstrated that the proposed method can find good quality solutions in a reasonable amount of time (less than 44 minutes) as compared to the exact method that failed to find an optimal solution in 10 hours for the test problem instance. Furthermore, a modified reproduction metric (i.e., mR_0) was proposed

as a performance measure to show the effectiveness of the proposed method. It considered the total time to complete the delivery mission, the number of people involved in the delivery process, and the transmission rate of the virus. For comparison, we also derived the mR_0 of the face-to-face diagnosis. The results showed that mR_0 of the testing kit delivery system (i.e., mR_0=0.002) was significantly lower than that of the face-to-face method (i.e., mR_0=0.0153), hence, it achieved a 7.5 reduction compared to a face-to-face diagnosis used in a 'drive-thru' testing site.

COVID-19 has greatly increased the awareness of infectious diseases. We believe that the proposed testing kit delivery method using the MDRP model can easily be applied to other applications, especially, the diagnosis of infectious diseases such as the seasonal flu, MERS, and SARS. As an extension to this paper, one can develop a scalable optimization model and solution algorithms for a system of multiple trucks and multiple drones to support larger densely populated cities or sparsely populated rural areas. Another direction for future work is to investigate the influence of traffic congestion, speed limit, traffic light and other road driving restrictions in an optimization model.

References

[1] Science brief: SARS-CoV-2 and surface (fomite) transmission for indoor community environments. 2020. URL `https://www.cdc.gov/coronavirus/2019-ncov/more/science-and-research/surface-transmission.html`.

[2] ICN confirms 1,500 nurses have died from covid-19 in 44 countries and estimates that healthcare worker covid-19 fatalities worldwide could be more than 20,000. 2020. URL `https://www.icn.ch/news`.

[3] Xiaoshan Bai, Ming Cao, Weisheng Yan, and Shuzhi Sam Ge. Efficient routing for precedence-constrained package delivery for heterogeneous vehicles. *IEEE Transactions on Automation Science and Engineering*, 17(1):248–260, 2019.

[4] Hillel Bar-Gera. Transportation networks for research core team. `https://github.com/bstabler/TransportationNetworks`, 2020.

[5] Mohd Shahrizan Bin Othman, Aleksandar Shurbevski, Yoshiyuki Karuno, and Hiroshi Nagamochi. Routing of carrier-vehicle systems with dedicated last-stretch delivery vehicle and fixed carrier route. *Journal of Information Processing*, 25:655–666, 2017.

[6] Paul Bouman, Niels Agatz, and Marie Schmidt. Dynamic programming approaches for the traveling salesman problem with drone. *Networks*, 72(4):528–542, 2018.

[7] Nils Boysen, Dirk Briskorn, Stefan Fedtke, and Stefan Schwerdfeger. Drone delivery from trucks: Drone scheduling for given truck routes. *Networks*, 72(4):506–527, 2018.

[8] John Gunnar Carlsson and Siyuan Song. Coordinated logistics with a truck and a drone. *Management Science*, 64(9):4052–4069, 2018.

[9] CDC. City of Sioux Falls coronavirus resources, 2021. URL `https://coronavirus-cityofsfgis.hub.arcgis.com/`.

[10] CDC. Centers for disease control and prevention, 2021. URL `https://time.com/6081407/covid-19-fourth-wave/`.

[11] Gerardo Chowell, Hiroshi Nishiura, and Luis MA Bettencourt. Comparative estimation of the reproduction number for pandemic influenza from daily case notification data. *Journal of the Royal Society Interface*, 4(12):155–166, 2007.

[12] Derek K Chu, Elie A Akl, Stephanie Duda, Karla Solo, Sally Yaacoub, Holger J Schünemann, Amena El-harakeh, Antonio Bognanni, Tamara Lotfi, Mark Loeb, et al. Physical distancing,

face masks, and eye protection to prevent person-to-person transmission of sars-cov-2 and covid-19: a systematic review and meta-analysis. *The Lancet*, 2020.

[13] Paul L Delamater, Erica J Street, Timothy F Leslie, Y Tony Yang, and Kathryn H Jacobsen. Complexity of the basic reproduction number (R0). *Emerging infectious diseases*, 25(1):1, 2019.

[14] Paul Diaz, Paul Constantine, Kelsey Kalmbach, Eric Jones, and Stephen Pankavich. A modified seir model for the spread of ebola in western africa and metrics for resource allocation. *Applied Mathematics and Computation*, 324:141–155, 2018.

[15] Klaus Dietz. The estimation of the basic reproduction number for infectious diseases. *Statistical methods in medical research*, 2(1):23–41, 1993.

[16] Leandro do C. Martins, Patrick Hirsch, and Angel A Juan. Agile optimization of a two-echelon vehicle routing problem with pickup and delivery. *International Transactions in Operational Research*, 28(1):201–221, 2021.

[17] Okan Dukkanci, Bahar Y Kara, and Tolga Bektas. The drone delivery problem. *Tolga, The Drone Delivery Problem (January 10, 2019)*, 2019.

[18] Claudio Gambella, Andrea Lodi, and Daniele Vigo. Exact solutions for the carrier–vehicle traveling salesman problem. *Transportation Science*, 52(2):320–330, 2018.

[19] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2021. URL `http://www.gurobi.com`.

[20] Yun-qi Han, Jun-qing Li, Zhengmin Liu, Chuang Liu, and Jie Tian. Metaheuristic algorithm for solving the multi-objective vehicle routing problem with time window and drones. *International Journal of Advanced Robotic Systems*, 17(2):1729881420920031, 2020.

[21] Aline Karak and Khaled Abdelghany. The hybrid vehicle-drone routing problem for pick-up and delivery services. *Transportation Research Part C: Emerging Technologies*, 102:427–449, 2019.

[22] Hamid Khataee, Istvan Scheuring, Andras Czirok, and Zoltan Neufeld. Effects of social distancing on the spreading of covid-19 inferred from mobile phone data. *Scientific reports*, 11 (1):1–9, 2021.

[23] Moran Ki. 2015 mers outbreak in korea: hospital-to-hospital transmission. *Epidemiology and health*, 37, 2015.

[24] Seon Jin Kim and Gino J. Lim. A real-time rerouting method for drone flights under uncertain flight time. *Journal of Intelligent & Robotic Systems*, 100(1):1355–1368, 2021.

[25] Seon Jin Kim, Gino J Lim, Jaeyoung Cho, and Murray J Côté. Drone-aided healthcare services for patients with chronic diseases in rural areas. *Journal of Intelligent & Robotic Systems*, 88 (1):163–180, 2017.

[26] RJ Kuo, Shih-Hao Lu, Pei-Yu Lai, and Setyo Tri Windras Mara. Vehicle routing problem with drones considering time windows. *Expert Systems with Applications*, 191:116264, 2022.

[27] Lan Lan, Dan Xu, Guangming Ye, Chen Xia, Shaokang Wang, Yirong Li, and Haibo Xu. Positive RT-PCR test results in patients recovered from covid-19. *JAMA*, 323(15):1502–1503, 2020.

[28] Min Lin, Yuming Chen, Rui Han, and Yao Chen. Discrete optimization on truck-drone collaborative transportation system for delivering medical resources. *Discrete Dynamics in Nature and Society*, 2022, 2022.

[29] Zhihao Luo, Mark Poon, Zhenzhen Zhang, Zhong Liu, and Andrew Lim. The multi-visit traveling salesman problem with multi-drones. *Transportation Research Part C: Emerging Technologies*, 128:103172, 2021.

[30] Giusy Macrina, Luigi Di Puglia Pugliese, Francesca Guerriero, and Gilbert Laporte. Drone-aided routing: A literature review. *Transportation Research Part C: Emerging Technologies*, 120:102762, 2020.

[31] Adriano Masone, Stefan Poikonen, and Bruce L Golden. The multivisit drone routing problem with edge launches: An iterative approach with discrete and continuous improvements. *Networks*, 2022.

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

[33] Mahdi Moeini and Hagen Salewski. A genetic algorithm for solving the truck-drone-atv routing problem. In *World Congress on Global Optimization*, pages 1023–1032. Springer, 2019.

[34] Mohammad Moshref-Javadi and Matthias Winkenbach. Applications and research avenues for drone-based models in logistics: A classification and review. *Expert Systems with Applications*, 177:114854, 2021.

[35] Minh Anh Nguyen, Giang Thi-Huong Dang, Minh Hoàng Hà, and Minh-Trien Pham. The min-cost parallel drone scheduling vehicle routing problem. *European Journal of Operational Research*, 2021.

[36] Murugaiyan Pachayappan and Vijayakumar Sudhakar. A solution to drone routing problems using docking stations for pickup and delivery services. *Transportation Research Record*, 2675 (12):1056–1074, 2021.

[37] Feng Pan, Tianhe Ye, Peng Sun, Shan Gui, Bo Liang, Lingli Li, Dandan Zheng, Jiazheng Wang, Richard L Hesketh, Lian Yang, et al. Time course of lung changes on chest ct during recovery from 2019 novel coronavirus (covid-19) pneumonia. *Radiology*, 2020.

[38] Stefan Poikonen and James F Campbell. Future directions in drone routing research. *Networks*, 77(1):116–126, 2021.

[39] Stefan Poikonen and Bruce Golden. The mothership and drone routing problem. *INFORMS Journal on Computing*, 32(2):249–262, 2020a.

[40] Stefan Poikonen and Bruce Golden. Multi-visit drone routing problem. *Computers & Operations Research*, 113:104802, 2020b.

[41] Stefan Poikonen, Bruce Golden, and Edward A Wasil. A branch-and-bound approach to the traveling salesman problem with a drone. *INFORMS Journal on Computing*, 31(2):335–346, 2019.

[42] Mario Rivera-Izquierdo, María Del Carmen Valero-Ubierna, Silvia Martínez-Diz, Miguel Ángel Fernández-García, Divina Tatiana Martín-Romero, Francisco Maldonado-Rodríguez, María Rosa Sánchez-Pérez, Luis Miguel Martín-delosReyes, Virginia Martínez-Ruiz, Pablo Lardelli-Claret, et al. Clinical factors, preventive behaviours and temporal outcomes associated with covid-19 infection in health professionals at a spanish hospital. *International Journal of Environmental Research and Public Health*, 17(12):4305, 2020.

[43] Roberto Roberti and Mario Ruthmair. Exact methods for the traveling salesman problem with drone. *Transportation Science*, 55(2):315–335, 2021.

[44] Halil Savuran and Murat Karakaya. Efficient route planning for an unmanned air vehicle deployed on a moving carrier. *Soft Computing*, 20(7):2905–2920, 2016.

[45] Zhou Tang, Xianbin Li, and Houqiang Li. Prediction of new coronavirus infection based on a modified seir model. *medRxiv*, 2020.

[46] TIME. A fourth wave of covid-19 is brewing in the u.s. is there enough time to stop it?, 2021. URL `https://time.com/6081407/covid-19-fourth-wave/`.

[47] Maryam Torabbeigi, Gino J. Lim, Navid Ahmadian, and Seon Jin Kim. An optimization approach to minimize the expected loss of demand considering drone failures in drone delivery scheduling. *Journal of Intelligent & Robotic Systems*, 102(22), 2021.

[48] Guido Van Rossum and Fred L. Drake. Python 3 reference manual, 2021. URL `https://www.python.org/`.

[49] Sebastián A Vásquez, Gustavo Angulo, and Mathias A Klapp. An exact solution method for the tsp with drone based on decomposition. *Computers & Operations Research*, 127:105127, 2021.

[50] Jarosław Wikarek, Paweł Sitek, and Łukasz Zawarczyński. An integer programming model for the capacitated vehicle routing problem with drones. In *International Conference on Computational Collective Intelligence*, pages 511–520. Springer, 2019.

[51] Wen Yi and Monty Sutrisna. Drone scheduling for construction site surveillance. *Computer-Aided Civil and Infrastructure Engineering*, 36(1):3–13, 2021.

[52] Fanruiqi Zeng, Zaiwei Chen, John-Paul Clarke, and David Goldsman. Nested vehicle routing problem: Optimizing drone-truck surveillance operations. *arXiv preprint arXiv:2103.01528*, 2021.

A Launch-Return Location Optimization Model

The *LRO* chooses an optimal set of launch and return nodes for a current sets of patients to minimize the MDRP objective function. Unlike MDRP, P and SP are pre-determined in LRO. The primary purpose of this model is to check the feasibility of an initial solution provided by TSP_{BFS}.

$$\min_{f,w,h} \quad \text{LRO} = \sum_{(l,i,r)\in LFR} t^i_{l,r} f^i_{l,r} + \sum_{i\in P} w_i + \sum_{i\in N} h_i \tag{55}$$

s.t.

$$(7), (8), (32) \tag{56}$$

$$T_{p_r} - T_{p_l} + \sum_{k=l+1}^{r-1} w_{p_k} - M(1 - f^j_{p_l,p_r}) \leq t^j_{p_l,p_r} + \sum_{i\in S_j} h_i \quad \forall (p_l, j, p_r) \in FLR \tag{57}$$

$$t^j_{p_l,p_r} + \sum_{i\in S_j} h_i - M(1 - f^j_{p_l,p_r}) \leq T_{p_r} - T_{p_l} + \sum_{k=l}^{r} w_{p_k} \quad \forall (p_l, j, p_r) \in FLR \tag{58}$$

$$D_{p_{i-1}} - D_{p_i} = \sum_{(l,j,p_i)\in LFR} f^j_{l,p_i} - \sum_{(p_i,j,r)\in LFR} f^j_{p_i,r} \ , i = 2, ..., |P| \tag{59}$$

$$D_C + \sum_{(C,j,r)\in LFR} f^j_{C,r} = n \tag{60}$$