



An optimization drone routing model for inspecting wind farms

Hyeoncheol Baik¹ · Jorge Valenzuela²

Published online: 22 September 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

The use of wind turbines to generate electricity is growing worldwide. They comprise an extended area of hundreds of square miles, making the inspection process difficult and time-consuming. Recently, there has been an increasing interest in using a drone, or also known as unmanned aircraft systems, for inspecting wind turbines. Motivated by leveraging drone technology, this paper provides a routing optimization model to reduce the total operation time for inspecting a wind farm. We assume that one drone and one ground vehicle which carries the drone and extra batteries and charging equipment are available. The optimization model is solved in two steps. The first step clusters the wind turbines and optimizes the drone routing in each cluster by solving the classical traveling salesman problem using an integer linear programming model. The second step optimizes the ground vehicle routing by solving the equality generalized traveling salesman problem using an integer linear programming model. We test our proposed model using three case studies created by using actual wind farm locations. We compare the results with two models. One model assumes no clustering of the wind turbines, and the other model uses a greedy approach for determining the ground vehicle route. The results show that the proposed model is more efficient at different flight speeds and endurances. Also, we confirm that the efficiency increases as the drone flies faster or it has longer flight endurance.

Keywords Unmanned aircraft system · Drone · Wind turbine inspection routing · Integer linear programming · Traveling salesman problem (TSP) · Equality generalized traveling salesman problem (E-GTSP)

1 Introduction

According to the U.S. Energy Information Administration (2018), on a percentage basis, renewables are the fastest growing energy source. Wind farms are being built in large numbers on land and offshore and usually comprise an extended area of hundreds of square miles. To identify potential problems to their proper operation, wind turbines are inspected on a regular basis. The most common inspection practice is a visual inspection by a technician using rope access. This technique is cost effective and

offers flexible access to the structure for performing the maintenance and repair tasks. However, this practice has a very low inspection rate, in the range of 2–5 turbines per day (Deign 2016). Recently, there has been increased interest in using an unmanned aircraft system (UAS), or a drone, for inspecting wind turbines. SkySpecs, a Michigan-based startup, can inspect a wind turbine in 15 min using a drone, covering 17 wind turbines per day (Capots 2017).

Motivated by the operational benefit of using a drone, we propose an approach to inspect several wind turbines using one drone and one ground vehicle which carries extra batteries and the battery-charging equipment. It is assumed that the ground vehicle carrying the drone departs from a depot. The ground vehicle stops at pre-defined locations. At each stop, the drone flies to inspect several wind turbines then returns to the ground vehicle. In the ground vehicle, the operator swaps the batteries and charges used batteries. The ground vehicle moves to another location, and the inspection process is repeated.

The remainder of this paper is organized as follows. In Sect. 2, we present related work to this research. In Sect. 3, we describe the proposed optimization model and solution

Communicated by V. Loia.

✉ Jorge Valenzuela
valenjo@auburn.edu

Hyeoncheol Baik
hyeoncheol.baik@stockton.edu

¹ School of Business, Stockton University, Galloway, NJ 08205, USA

² Department of Industrial and Systems Engineering, Auburn University, Auburn, AL 36849, USA

approach. In Sect. 4, we present the experimental results including three cases created by using real locations of wind farms. Finally, in Sect. 5, we state the conclusion.

2 Related work

Several recently published papers addressed the challenge of using a drone to inspect the structure of a wind turbine. Stokkeland et al. (2015) presented an algorithm that uses the Hough transform and Kalman filter for a camera-based autonomous navigation of a drone. In related research, Schäfer et al. (2016) used a light detection sensor and a global positioning system to ensure an automated collision-free flight for an inspection drone. Galleguillos et al. (2015) equipped a drone with an infrared thermography camera to visually inspect a wind turbine. For inspecting offshore wind farms, and Collins et al. (2017) used an unmanned surface vessel to shuttle an inspection drone and recharge the drone's batteries. However, these research works used one drone for inspecting only a single wind turbine.

When it comes to other application areas, the combination of a drone and ground vehicle has been successfully used. In parcel delivery, parcels are delivered directly to customers by either the drone or vehicle. The research problem has been in planning an optimal route for the drone and vehicle as well as assigning parcels to be delivered by either the drone or vehicle. Murray and Chu (2015) proposed two different optimization problems by considering two sets of assumptions. Each problem was formulated as a mixed-integer linear programming (MILP) problem with the objective to reduce the total delivery time. Because these optimization problems are NP-hard, the authors developed two heuristic solution approaches and tested them assuming 10 and 20 customers. These optimization problems were extended by several researchers (Ponza 2016; Ha et al. 2018; de Freitas and Penna 2018; Kim and Moon 2018; Marinelli et al. 2018; Jeong et al. 2019). In Ponza (2016), the optimization problem was solved using a SA metaheuristic. In Ha et al. (2018), the authors used an objective function to minimize operational costs. The proposed a more effective heuristic algorithm called greedy randomized adaptive search procedure (GRASP) and tested it assuming 10, 50, and 100 customers. In de Freitas and Penna (2018), the optimization problem was solved using the randomized variable neighborhood descent heuristic algorithm. Kim and Moon (2018) considered a drone station defined as a facility that stores drone and charging equipment. The problem was formulated using a mixed-integer programming (MIP) and solved by dividing it into a traveling salesman problem (TSP) and a parallel identical machine scheduling problem. Marinelli et al. (2018) assumed that the drone can depart

and return while the vehicle is moving. They solved the problem by using a heuristic algorithm which is a modified version of the GRASP method proposed in Ha et al. (2018). Jeong et al. (2019) considered two practical issues which are a parcel weight and no-fly zones and developed a two-phase heuristic algorithm to solve the problem. Also, Gambella et al. (2018) proposed a mixed-integer, second-order conic programming formulation and a ranking-based solution algorithm to provide an exact solution for the routings of the general carrier (e.g., truck)-vehicle (e.g., drone) systems.

Multiple variations of the parcel delivery problem using drones exist in the literature. In Agatz et al. (2018), the authors assumed that the drone is constrained to use the same road network as the vehicle when delivering the parcels. The authors developed an integer programming formulation and several heuristic algorithms based on local search and dynamic programming. Bouman et al. (2018) extended this research by finding an exact solution to the dynamic programming formulation and developing two dynamic programming-based heuristic algorithms for large instances. Using the same assumptions on the take-off and landing stop locations described in Agatz et al. (2018), Boysen et al. (2018) derived six subproblems to optimize the schedule of drones based on a given truck route. They analyzed the computational complexity of each subproblem and developed efficient MIP models. Hong et al. (2018) addressed the optimization problem on finding the locations of the drone charging stations to maximize the customer demand. They proposed a MIP formulation and a heuristic algorithm based on minimum spanning trees, greedy subtraction, spatial interchange, and simulated annealing (SA) algorithms.

Moreover, Ferrandez et al. (2016), Chang and Lee (2018), and Mathew et al. (2015) assumed that only the drone delivers the parcels, while the driver just drives the vehicle. Ferrandez et al. (2016) and Chang and Lee (2018) determined the stop locations of the vehicle by formulating the optimization problem as a TSP which is combined with the K-means clustering algorithm. To solve the TSP, a genetic algorithm (GA) was used in Ferrandez et al. (2016) and a nonlinear programming formulation with a commercial solver was used in Chang and Lee (2018). In Mathew et al. (2015), the aim was to reduce the delivery time or fuel consumption. The optimization problem is solved by using the Lin–Kernighan–Helsgaun (LKH) heuristic algorithm after transforming the problem into a generalized TSP which in turn is transformed into a standard TSP. Dorling et al. (2017) did not consider a ground vehicle. They formulated an optimization model for routing multiple drones with the two objectives of minimizing the overall delivery time of the parcels and total costs of drones. Their optimization model was a MILP formulation

that was solved by using a SA algorithm. Wang et al. (2017) found upper bounds for the time savings when using multiple drones and vehicles compared to using only one vehicle. In Poikonen et al. (2017), they extended this research by considering the operation cost and limited battery capacity and using other variants of the routing problem. Recently, Poikonen and Golden (2020) relaxed some simplifying assumptions such as homogeneous packages and carrying a single package at a time.

The combination of a drone and ground vehicle has also been explored in humanitarian logistics. Chowdhury et al. (2017) assumed that the drone is used to deliver emergency supplies when the truck roads are destroyed due to a natural disaster. They formulated a MIP for determining the locations of the distribution centers, their corresponding service areas and three ordering quantities of the emergency supplies. The objective was to minimize the total operation cost. The problem was solved using a continuous approximation technique. Rabta et al. (2018), that did not consider a ground vehicle, proposed a MILP formulation for routing a drone to minimize the total operation cost. They considered recharging stations and priority of outbreak locations.

Surveillance is another area where drones and vehicles are used together. The drone patrols certain targets, and the vehicle carries personnel and the drone. Luo et al. (2017) proposed an integer linear program and used two heuristic algorithms to solve the model based on the rules “drone first, ground vehicle second” and “ground vehicle first, drone second,” respectively. Sundar and Rathinam (2014) assumed that the drone is operated to visit targets with multiple fixed depots for refueling. The objective function aims to minimize the total fuel consumption. These authors formulated a MILP and solved it using two heuristic algorithms based on the LKH and k -opt heuristic algorithms, respectively. Sundar and Rathinam (2017) presented two MILP problems for either multiple ground vehicles or multiple drones and developed a branch-and-cut solution algorithm. Savuran and Karakaya (2016) solved the same optimization problem using a GA algorithm. In Arzamendia et al. (2019), the authors used a drone without a vehicle for environmental monitoring a lake. The drone optimization problem was solved using a tailored TSP model and a GA algorithm. For more applications on using the combination of a drone and vehicle, the interested reader is referred to the recent survey by Otto et al. (2018).

From the theoretical perspective, similar research has been conducted to solve the close-enough traveling salesman problem (CETSP) that is a variant of the TSP. Because the CETSP requires a salesman to pass through the neighborhood of each customer instead of visiting the exact location of each customer, it can be used to solve the

routing problem in which a drone flies within the neighborhood of each wind turbine. The CETSP has been solved by an exact algorithm such as a branch-and-bound algorithm (Coutinho et al. 2016). Also, several heuristic algorithms have been proposed to solve the CETSP. Carrabs et al. (2017a) presented a discretization scheme that provides both lower and upper bounds for the optimal solution. Also, this research work has been extended by combining the second-order cone programming algorithm (Carrabs et al. 2017b) and the Carousel Greedy algorithm (Carrabs et al. 2020). Wang et al. (2019) have developed a three-phase heuristic algorithm based on a Steiner zone method, and Faigl et al. (2019) have proposed two heuristic algorithms to quickly find a feasible solution.

The aim of this paper is to optimize the routing of a drone and a ground vehicle for inspecting wind turbines while minimizing the total operation time. This research has four primary differences compared to the existing literature:

- We introduce a drone routing optimization problem for inspecting multiple wind turbines.
- We allow the drone to visit multiple locations, limited by the battery power, after departing from the ground vehicle. In most of the parcel delivery problems, the drone is allowed visiting only one node and returning to the vehicle to retrieve another parcel.
- We combine the rule “drone first, ground vehicle second” with a clustering method. The rule has been used in the parcel delivery problem (Ferrandez et al. 2016; Chang and Lee 2018) and the partial accessibility constrained vehicle routing problem (Semet 1995). The clustering method has been used in the multiple depot vehicle routing problem (Salhi and Nagy 1999; Giosa et al. 2002).
- We test our proposed optimization model on actual wind farm locations while most of the previous research work uses artificially generated visiting locations.

3 Proposed optimization model and solution approach

The inspection system uses one drone and one ground vehicle. The drone performs the visual inspection of homogeneous wind turbines which requires the same inspection time of I minutes per wind turbine. It is assumed that the drone flies at a constant speed of FS mph and has a flight endurance of FE minutes (maximum flight time). The ground vehicle moves from cluster to cluster at a constant speed of GS mph and parks at only one of the designated locations near a wind turbine of a cluster to deploy and collect the drone. At each stop, the drone pilot conducts

pre- and post-flight procedures for P minutes. The optimization model is based on the rule “drone first, ground vehicle second” and is solved in two steps. The first step optimizes the drone routing in each cluster. The second step optimizes the ground vehicle routing using the clustering outcome from the first step.

3.1 Optimizing the drone route

We represent the depot and U wind turbines by the set $N = \{0, 1, 2, \dots, U\}$ where the value 0 refers to the depot. We first group the wind turbines into k clusters. The clustering approach is appropriate to our problem when wind farms have either a multiple/parallel string configuration or a cluster configuration (Denholm et al. 2009). We use the K-means clustering algorithm, also known as the Lloyd’s algorithm (Lloyd 1982). Given the number of centroids k as a parameter, the algorithm iteratively finds the best cluster assignments so that the sum of the distances between the wind turbines and their centroid is minimized. After the wind turbines have been grouped into k clusters, we obtain k mutually exclusive and exhaustive subsets of the set of the wind turbines. The subsets are denoted by N^a which includes all wind turbines in cluster $a = 1, 2, \dots, k$. Then, we find the shortest flight path of the drone to visit all wind turbines in N^a . Note that this drone routing optimization problem is the classical TSP (Dantzig et al. 1954). For cluster a , we solve the drone routing optimization problem using the integer linear programming model described in Eqs. 1–5.

$$\text{Min } t_a^D = \sum_{i \in N^a} \sum_{j \in N^a} F_{ij} z_{ij} \quad (1)$$

Subject to

$$\sum_{i \in N^a} z_{ij} = 1 \quad \forall j \in \{N^a : i \neq j\} \quad (2)$$

$$\sum_{j \in N^a} z_{ij} = 1 \quad \forall i \in \{N^a : i \neq j\} \quad (3)$$

$$\sum_{i \in S} \sum_{j \in S} z_{ij} \leq |S| - 1 \quad \forall S \subseteq N^a \setminus \{1\}, S \neq \emptyset \quad (4)$$

$$z_{ij} \in \{0, 1\} \quad \forall i, j \in \{N^a : i \neq j\} \quad (5)$$

The objective function (1) minimizes t_a^D , which is the total flight time of the drone between two wind turbines (i and j) in cluster a . Constraints (2) ensure that the drone enters to visit each wind turbine exactly once. Constraints (3) ensure that the drone leaves each wind turbine exactly once. Thus, constraints (2) and (3) force that each wind turbine is visited by the drone exactly once. Constraints (4) exclude subtours among wind turbines. Constraints (5) indicate that z_{ij} is a binary variable, which equals one if the drone flies from wind turbine i to wind turbine j , where

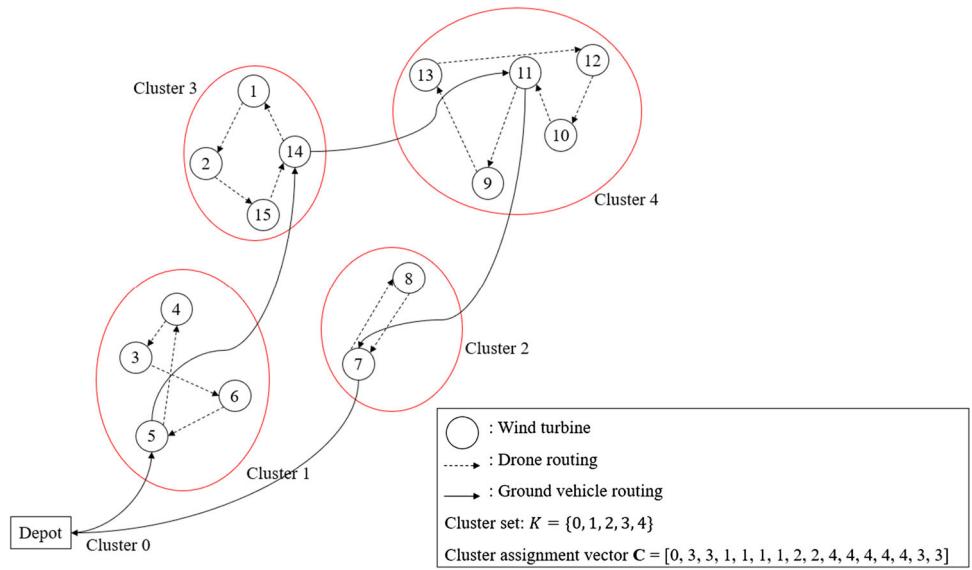
$i \neq j$. We solve this problem for each of the clusters. The nomenclature is provided in “Appendix 1.”

To find the optimal number of clusters, we start by considering only one cluster ($k = 1$). The shortest flight path of the drone is determined by solving the optimization problem described in Eqs. 1–5 with an integer linear programming. We ignore the constant time due to taking-off and landing. The total inspection time in cluster a is computed by multiplying the number of wind turbines in cluster a (m_a) by the inspection time of a wind turbine (I). The operation time of the drone in cluster a is obtained by adding the total flight time of the drone between wind turbines (t_a^D) and the total inspection time ($m_a \times I$). We compare the operation time of the drone with the flight endurance (FE), which is the maximum flight time due to the battery capacity. If the operation time of the drone in the cluster is greater than the flight endurance, we consider two clusters. The clusters are determined by using the K-means clustering algorithm. If again the operation time of the drone in each cluster is greater than the flight endurance, we increase k by 1 and repeat the procedure. Otherwise, we stop the procedure and store two clustering outcomes: the cluster set K and the cluster assignment vector \mathbf{C} (dimension $U + 1$). The value of the element in \mathbf{C} , C_i ($i = 0, 1, 2, \dots, U$), refers to the depot (0) or the cluster ($1, 2, \dots, k$) where the wind turbine i has been assigned. Figure 1 illustrates an example where 15 wind turbines have been grouped into 4 clusters. In this example, we begin by grouping 15 wind turbines into 1 cluster ($k = 1$). We solve the drone routing optimization problem and compute the operation time of the drone in the cluster. If the operation time of the drone exceeds the flight endurance, we group the wind turbines into 2 clusters ($k = 2$) and repeat the procedure until the operation time of the drone for each cluster is not greater than the flight endurance. At the end of the procedure, we store the cluster set K and the cluster assignment vector \mathbf{C} which are used to optimize the ground vehicle route.

3.2 Optimizing the ground vehicle route

The problem of routing the ground vehicle consists of determining the shortest route to visit all k clusters, only once, and return to the depot location. In addition, the park location at each cluster needs to be determined. We assume that the possible park locations are the same as the locations of the wind turbines. Note that this ground vehicle routing optimization problem is the equality generalized traveling salesman problem (E-GTSP) (Fischetti et al. 1997). Feeding two clustering outcomes generated by the previous step as inputs, we solve the ground vehicle routing

Fig. 1 15-wind turbine cluster assignment



optimization problem using the integer linear programming model described in Eqs. 6–18.

$$\text{Min } T^G = \sum_{i \in N} \sum_{j \in N} G_{ij} x_{ij} \quad (6)$$

Subject to

$$\sum_{\substack{i \in N \\ i \neq 0}} x_{0i} = 1 \quad (7)$$

$$\sum_{\substack{i \in N \\ i \neq 0}} x_{i0} = 1 \quad (8)$$

$$\sum_{i \in N} \sum_{j \in N} x_{ij} = 1 \forall a \in \{K : a \neq 0\} \quad (9)$$

$$\sum_{\substack{i \in N \\ C_i = a}} \sum_{\substack{j \in N \\ C_j = a}} x_{ij} = 1 \forall a \in \{K : a \neq 0\} \quad (10)$$

$$\sum_{\substack{i \in N \\ C_i = a}} \sum_{\substack{j \in N \\ C_j = b}} x_{ij} = y_{ab} \forall a, b \in \{K : a \neq b\} \quad (11)$$

$$\sum_{\substack{i \in N \\ C_i \neq C_j}} x_{ij} = \sum_{l \in N} x_{jl} \forall j \in \{N : j \neq 0\} \quad (12)$$

$$u_0 = 0 \quad (13)$$

$$1 \leq u_a \leq k \forall a \in \{K : a \neq 0\} \quad (14)$$

$$u_a - u_b + 1 \leq k(1 - y_{ab}) \quad \forall a, b \in \{K : a \neq b, a \neq 0, b \neq 0\} \quad (15)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in \{N : i \neq j\} \quad (16)$$

$$y_{ab} \in \{0, 1\} \quad \forall a, b \in \{K : a \neq b\} \quad (17)$$

$$u_a \in \{0, 1, 2, \dots, k\} \quad \forall a \in K \quad (18)$$

The objective function (6) minimizes the operation time by the ground vehicle, T^G , that is the total moving time departing from and returning to the depot. Constraints (7) and (8) ensure that the ground vehicle departs from and returns to the depot exactly once. Constraints (9) and (10) enforce that only one wind turbine in each cluster is selected as a stop location for the ground vehicle, ensuring that the ground vehicle visits each cluster exactly once. Constraints (11) force a relationship between the decision variables x_{ij} and y_{ab} . Suppose that wind turbines i and j were assigned to clusters a and b , respectively. If the ground vehicle moves from wind turbine i to j ($x_{ij} = 1$), the ground vehicle should also move from cluster a to b ($y_{ab} = 1$). Constraints (12) ensure that the ground vehicle moving to the wind turbine j from wind turbine i must also depart from the same wind turbine j to another wind turbine, say l . Constraints (13)–(15) exclude subtours among clusters and the depot. We apply the Miller–Tucker–Zemlin (MTZ) formulation given in Miller et al. (1960). Constraints (13) and (14) define another decision variable u to represent the sequence number cluster visits. Constraints (15) eliminate subtours with the bounds on u . Constraints (16) and (17) indicate that x_{ij} and y_{ab} are binary variables. Constraints (18) indicate that u is an integer variable.

As described in Eq. 19, the total operation time of the drone (T^D) is computed by summing the total flight time of

the drone between wind turbines (t_a^D) and the total inspection time ($m_a \times I$) over the k clusters.

$$T^D = \sum_{\substack{a \in K \\ a \neq 0}} (t_a^D + m_a \times I) \quad (19)$$

The total operation time for conducting pre- and post-flight procedures (T^P) is given by Eq. 20.

$$T^P = P \times k \quad (20)$$

The total operation time (T) given in Eq. 21 includes the total operation time of the drone (T^D), the total operation time of the ground vehicle (T^G), and the total operation time for conducting pre- and post-flight procedures (T^P).

$$T = T^D + T^G + T^P \quad (21)$$

The flowchart for optimizing the drone routing and the ground vehicle is shown in Fig. 2. In step 1, we start by clustering the wind turbines into one cluster ($k = 1$) and solving the drone routing optimization problem. We continue this procedure by increasing k by 1 until all the operation time of the drone ($t_a^D + m_a \times I$) for each cluster is not greater than the flight endurance. When the procedure stops, we store the cluster outcomes. In step 2, we solve the ground vehicle routing optimization problem to visit all the k clusters. Lastly, the solution of our optimization model becomes the shortest flight path of the drone for each cluster and the shortest moving path of the ground vehicle.

4 Experimental results

The drone routing is coded in MATLAB. We employ the K-means clustering function and solve the drone routing optimization problem using the “intlinprog” function of MATLAB. The ground vehicle routing is coded in AMPL and solved using the CPLEX 12.8 solver. Note that we cannot guarantee to solve our optimization problems in deterministic polynomial time because both TSP and E-GTSP are NP-hard problems. Thus, all the drone and ground vehicle routing optimization problems are solved with a 30-min time limit. All case studies are run on a computer with an Intel® Core™ i5-8250U (1.60 GHz) processor with 8 GB of RAM memory.

4.1 System parameters and case studies

We use the drone and ground vehicle parameters shown in Table 1.

To test the proposed model and solution approach, we create three case studies by examining the number of wind turbines per wind project in the state of Texas. According

to the American Wind Energy Association (2018), Texas has the largest wind energy capacity in the USA (more than 10,000 MW). As shown in Fig. 3, we plot a histogram of the number of wind turbines on 172 wind projects in Texas.

We use the first, second, and third quartiles of the size of the wind projects to determine the number of wind turbines for creating the three case studies. We select the Golden Spread Panhandle Wind Project near Wildorado, Texas, to represent a small wind farm (first quartile). This case comprises 34 wind turbines across a 6-mile square region. To represent a medium-sized wind farm (second quartile), we select the Hackberry Wind Project in Shackelford County, Texas. This wind project has 72 wind turbines across a 15-mile square region. For the third case representing a large-sized wind farm (third quartile), we select the Langford Wind Project located around 300 miles southwest of Dallas, Texas. This wind project has 100 wind turbines across a 55-mile square region. Figure 4 maps the actual locations of the wind turbines for these three cases. Note that the depots are located at the mass center of the wind turbines. “Appendix 2” provides the geographic locations of the wind turbines for each of the three wind farms.

4.2 Routing solutions

We solve the three case studies under the same system parameters. In Figs. 5, 6, and 7, we plot the routing solutions of the drone and ground vehicle. The x-axis and y-axis indicate the longitude and latitude, respectively. In the figures, the wind turbines are marked with circles, the optimal drone route represented by a dashed line, and the optimal ground vehicle route represented by a solid line. The clusters are marked with shaded color. For the small-sized wind farm shown in Fig. 5, the 34 wind turbines are grouped into 6 clusters having each cluster between 4 and 8 wind turbines. For the medium-sized wind farm shown in Fig. 6, the 72 wind turbines are grouped into 13 clusters having each cluster between 2 and 8 wind turbines. For the large-sized wind farm shown in Fig. 7, the 100 wind turbines are grouped into 21 clusters having each cluster between 3 and 7 wind turbines.

4.3 Effect of clustering and ground vehicle routing

To study the effects of the clustering of the wind turbines, we compare our proposed model to a model that assumes no clustering of the wind turbines, so the drone flies to inspect the closest wind turbine after taking-off from the ground vehicle. In this case, the ground vehicle needs to visit all wind turbines. The resulting ground vehicle routing optimization problem is the classical TSP which we solve

Fig. 2 Flowchart of the solution approach

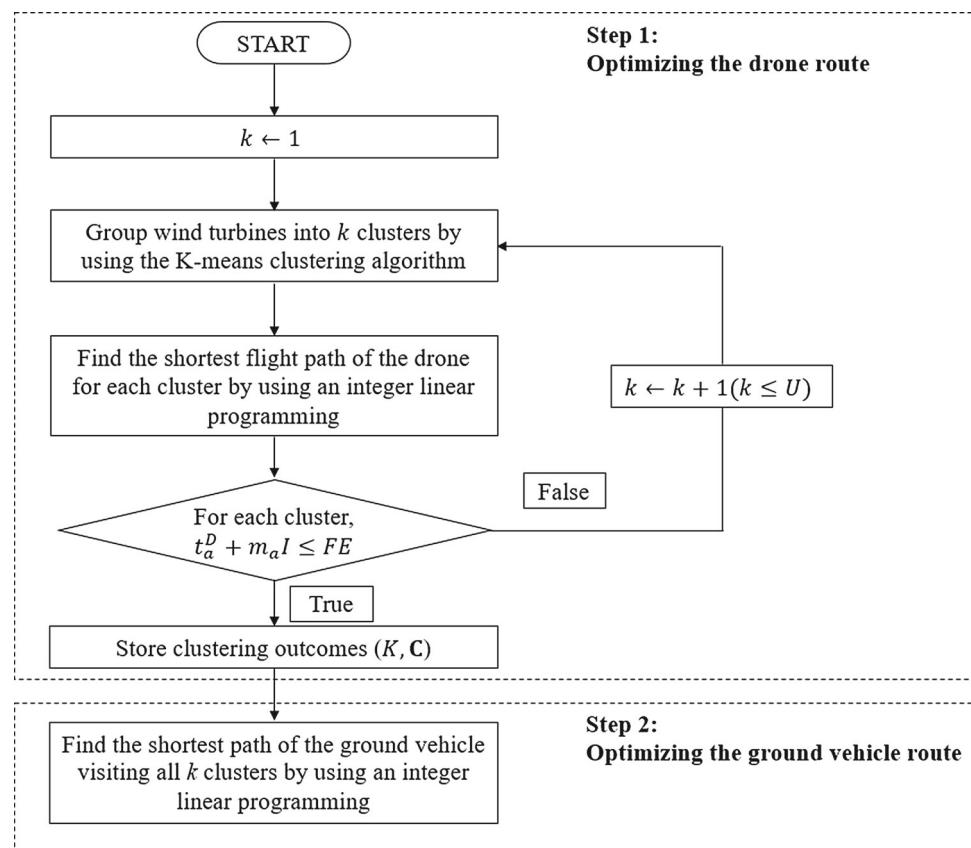


Table 1 System parameters

<i>Drone</i>	
Flying speed (FS)	40 mph
Flight endurance (FE)	50 min
Inspection time for a wind turbine (I)	5 min
<i>Ground vehicle</i>	
Moving speed (GS)	20 mph
Time for conducting pre- and post-flight procedures (P)	5 min

using the CPLEX solver. We named this model M1. Similarly, to study the effects of the ground vehicle routing optimization, we compare our proposed model to a model M2. The model clusters the wind turbines, and the “intlinprog” function of MATLAB is used to solve the routing of the drone. The routing of the ground vehicle is determined using a greedy approach. At each cluster, the wind turbine closest to the centroid is selected as a stop location of the ground vehicle. After inspection, the ground vehicle moves to the stop location closest to its current location. Our proposed approach is named M3.

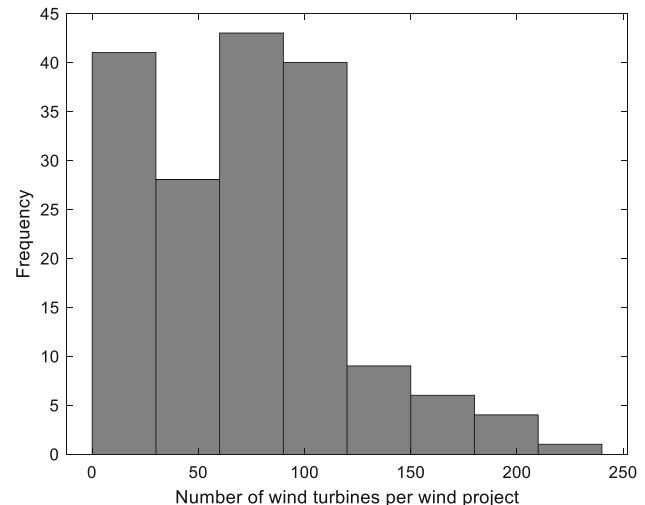


Fig. 3 Histogram of the number of wind turbines per wind project in Texas

4.4 Effect of total operation time

First, we study the effect of total operation time under the different models. We calculate the percentage of time reduction of M2 and M3 with respect to M1. The results are shown in Table 2. From the results, we observe that M3 outperforms M2 for the three wind farm sizes. The time

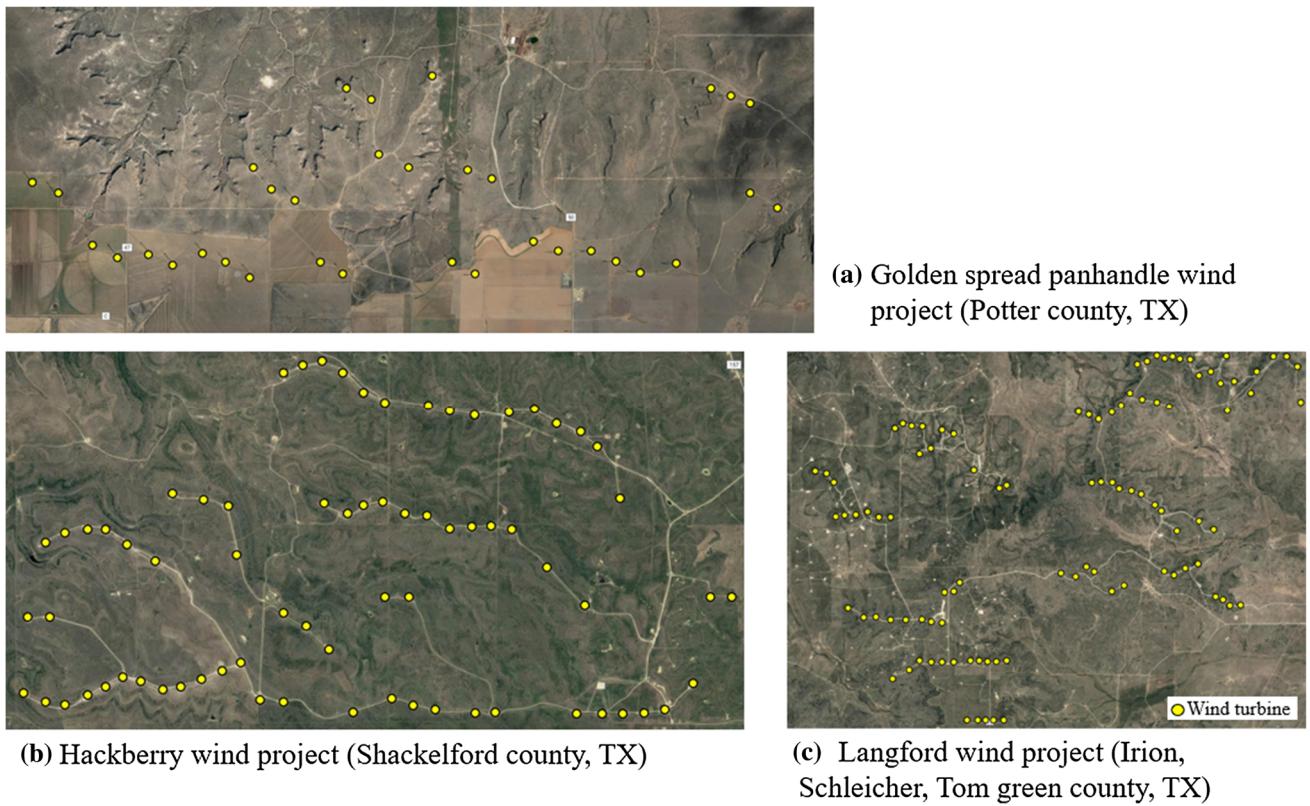


Fig. 4 Google satellite maps of the wind turbines for the three case studies

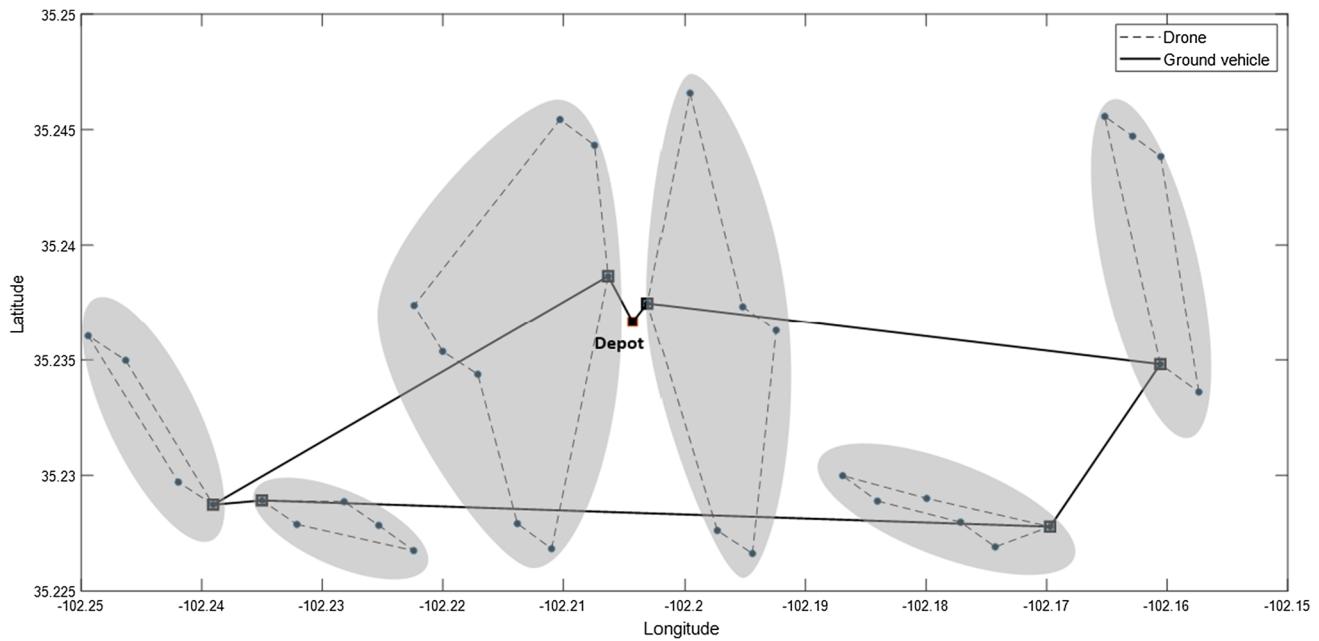


Fig. 5 Small-sized wind farm routing solution for the drone and ground vehicle

reduction in minutes increases approximate linearly as we study larger sized wind farms. The largest percentage of time reduction (36.8%) is obtained in the small-sized wind

farm (Case 1). Similarly, the lowest percentage of time reduction (34.5%) is obtained in the large-sized wind farm (Case 3).

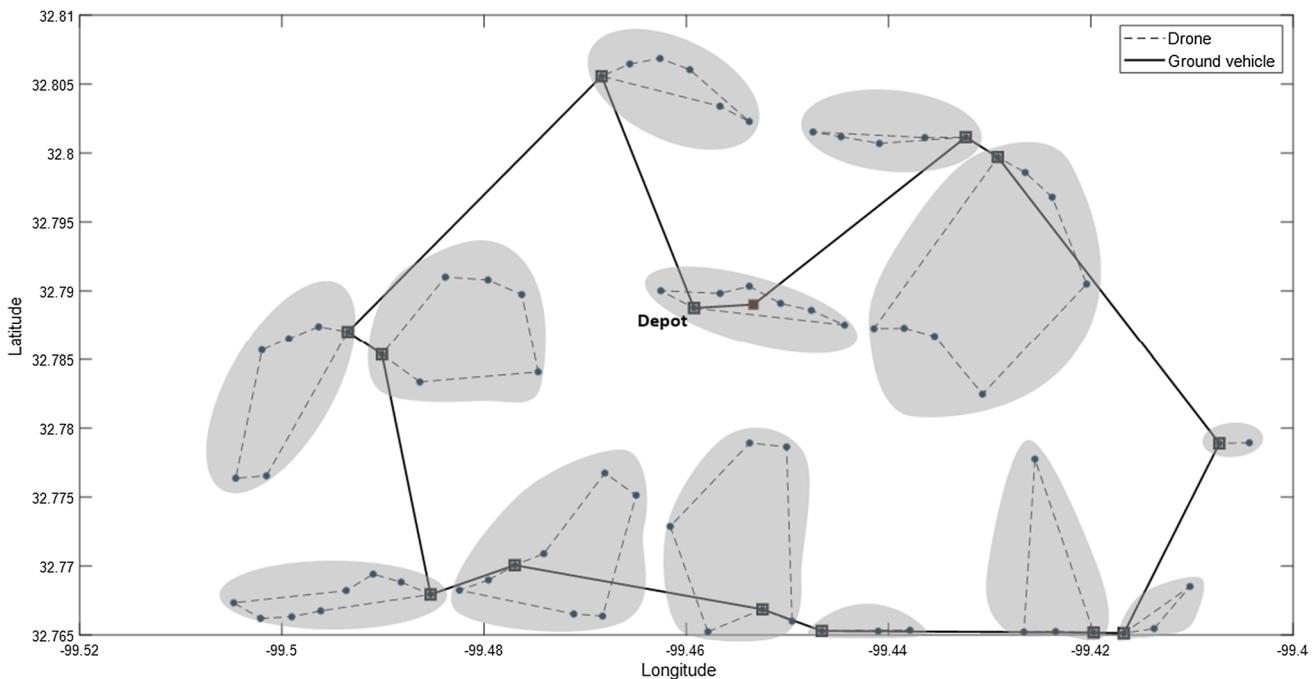


Fig. 6 Medium-sized wind farm routing solution of the drone and ground vehicle

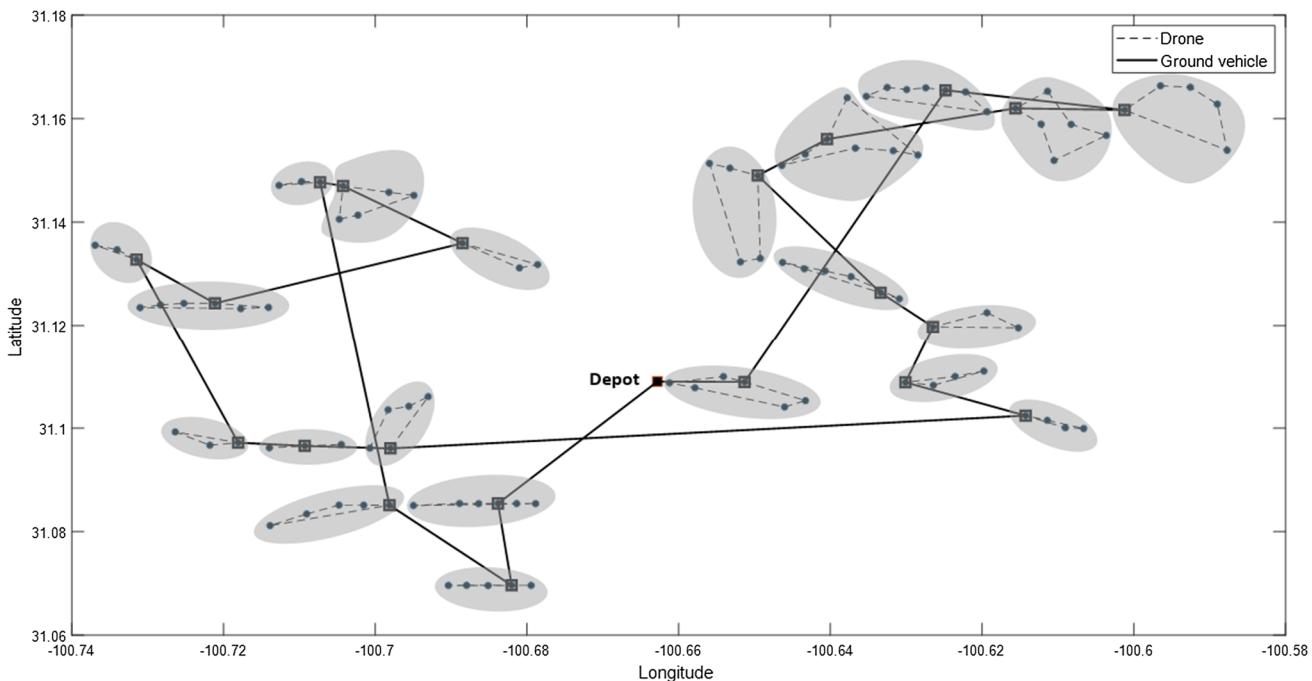


Fig. 7 Large-sized wind farm routing solution of the drone and ground vehicle

4.5 Effect of flight speed

To study the effect of flight speed, we solve all problems and using seven different flight speeds from 10 to 40 mph in steps of 5 mph. In Figs. 8, 9, and 10, we plot the percentage of time reduction of models M2 and M3 for each

wind farm cases. In the three wind farm sizes, we observe that the percentage of time reduction of M2 and M3 increases as the drone flies faster. In the small-sized wind farm, the slopes, which are time reduction rates, decrease significantly when the flight speed becomes larger than 20 mph which is the speed of the ground vehicle. In the

medium- and large-sized wind farms, the same change of slope occurs at 25 mph. Also, we observe that the difference in time reduction between M2 and M3 remains practically unchanged as the drone flies faster, except at the speed 20 mph in the small- and medium-sized wind farms, and at 10 mph in the large-sized wind farm. Overall, in all three wind farm sizes, the percentage of time reduction of our approach (M3) is always greater than of model M2 for all values of the flight speed.

4.6 Effect of flight endurance

To study the effect of flight endurance, we solve again all problems and use seven different flight endurances from 20 to 50 min in steps of 5 min. In Figs. 11, 12, and 13, we plot the percentage of time reduction for each wind farm case. In the three wind farm cases, we observe that the percentage of time reduction of M2 and M3 increases as the flight endurance increases. In the small-sized wind farm, no change in time reduction is observed when the flight endurance changes from 35 to 40 min and from 45 to 50 min. The same situation occurs for the medium-sized wind farm when the flight endurance changes from 25 to 30 min and for the large-sized wind farm when it changes from 40 to 50 min. Overall, in all three wind farm sizes, the percentage of time reduction for our approach (M3) is always greater than of model M2 for all values of the flight endurance. It is noted that the differences in time reduction between M2 and M3 for the small-sized farm are larger than those for the medium- and large-sized wind farms.

4.7 Practical guidance

The first step for the implementation of the proposed approach is the selection of an appropriate drone. The flight speed and endurance of the drone are the technical parameters necessary to run our algorithm. After the drone has been chosen, an inspection drone path for inspecting a

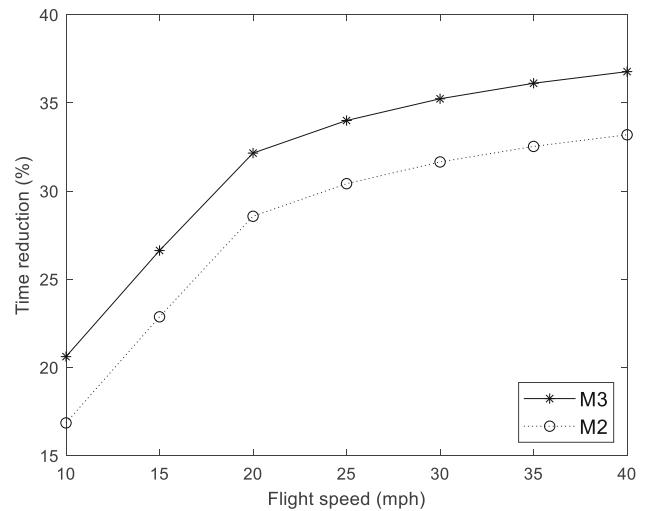


Fig. 8 Effect of changing the flight speed in the small-sized wind farm

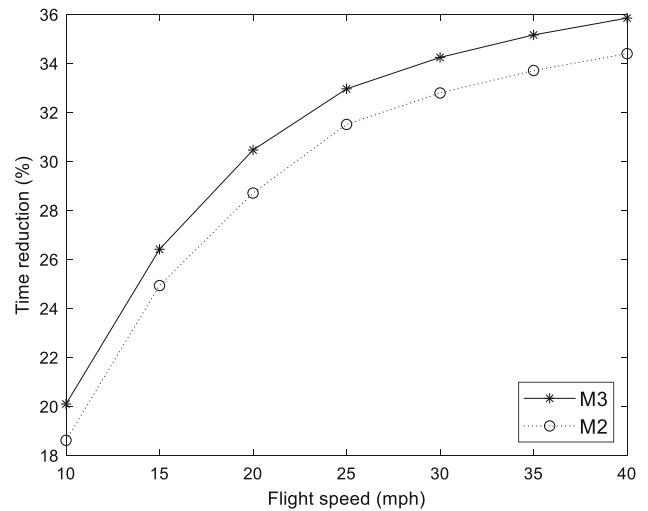


Fig. 9 Effect of changing the flight speed in the medium-sized wind farm

Table 2 Comparative results of the total operation time

Size	Model	T^D (min)	T^G (min)	T^P (min)	T (min)	Time reduction (min)	Time reduction (%)
Small	M1	170	68	170	408	—	—
	M2	189	54	30	272	135	33.2
	M3	189	39	30	258	150	36.8
Medium	M1	360	98	360	818	—	—
	M2	398	72	65	536	282	34.5
	M3	398	60	65	524	294	36.0
Large	M1	500	235	500	1235	—	—
	M2	556	180	105	841	394	31.9
	M3	556	148	105	808	426	34.5

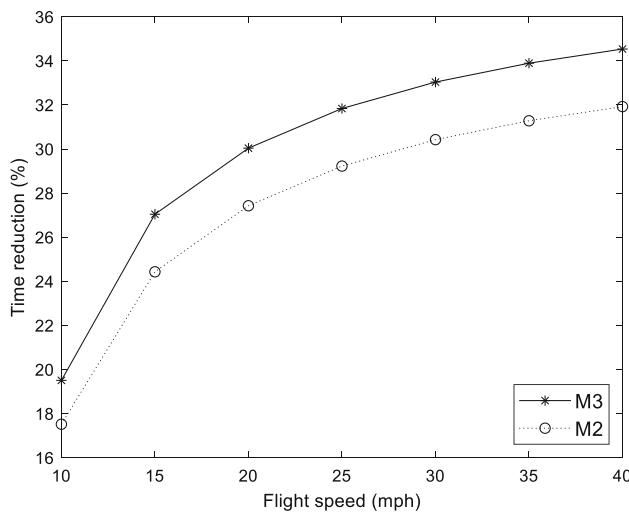


Fig. 10 Effect of changing the flight speed in the large-sized wind farm

wind turbine needs to be designed. The inspection drone path determines the inspection time of a wind turbine. Flight times between all pairs of wind turbines need to be calculated using their locations. Lastly, the speed of the ground vehicle and the time needed for conducting pre- and post-flight procedures are determined. All these parameters are input to the proposed algorithm. The solution provides the route of the ground vehicle and the routes of the drone at the stop locations.

5 Conclusions

We proposed a routing optimization model for a drone and ground vehicle to reduce the total operation time for inspecting a set of wind turbines. We solved the optimization model by using the K-means clustering algorithm and two integer linear programming formulations. Three wind farm size cases were created by using real locations of wind farms. We developed two other experimental models to evaluate the proposed model. We tested these models in the three cases. The experimental results showed that the proposed model reduces the total operation time of the inspection for the three wind farm sizes when compared to no clustering or the use of a greedy approach for determining the route of the ground vehicle. In addition, we studied the effects of the flight speed and endurance on the time reduction. The results showed that the total operation time can be reduced significantly as the flight speed and endurance are increased. Given a budget and technical constraints to operate the drone, these results can be of use to practitioners to determine the technical specifications of the drone. In all three wind farm sizes, our approach outperformed the model M2 for all tested values of the flight

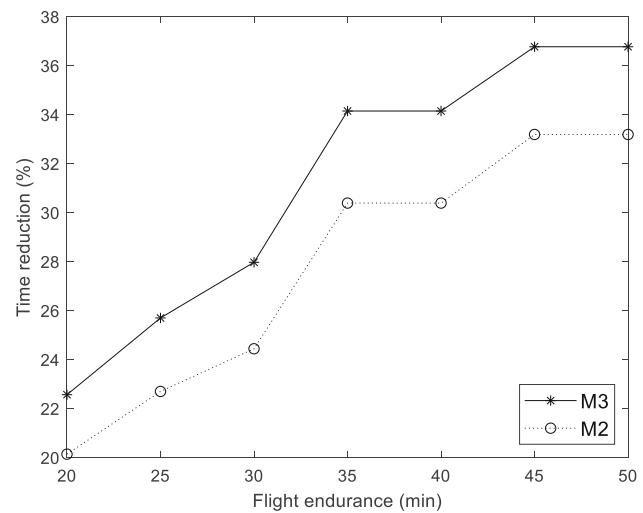


Fig. 11 Effect of changing the flight endurance in the small-sized wind farm

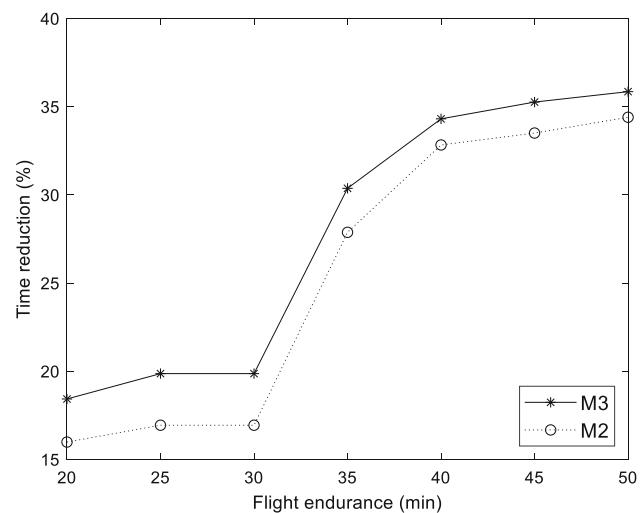


Fig. 12 Effect of changing the flight endurance in the medium-sized wind farm

speed and endurance. This result showed the robustness of our approach to changes to the system parameters.

Future work can be directed toward including multiple drones for covering larger wind farms and developing a battery power consumption model for reflecting the fact that a drone uses battery power differently depending on the flight modes such as cruising and hovering. Furthermore, other clustering algorithms (Kiran et al. 2020; Murugesan and Murugesan 2020) can be used and compared to the one used in this paper. In practice, the flight time between two wind turbines may be not be known with certainty due to varying wind conditions. To consider this factor, the flight time can include an uncertain amount as it has been done in TSP (Wang et al. 2015), supply chain management (Chen et al. 2017a; Yang et al. 2020), and

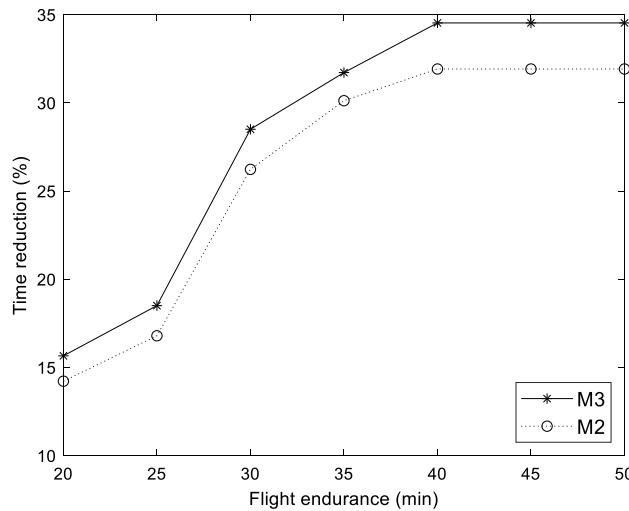


Fig. 13 Effect of changing the flight endurance in the large-sized wind farm

bicriteria solid transportation problem (Chen et al. 2017b). In addition, a wind speed prediction model (Natarajan and Nachimuthu 2020) can be used to model the flight time as a random variable.

Compliance with ethical standards

Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

Appendix 1: nomenclature

Indices

i, j, l Wind turbines

a, b Clusters

Sets

N Set of the wind turbines and depot

N^a Set of the wind turbines in cluster a

S Subset of N^a

K Set of the clusters

Parameters

FS	Flying speed of the drone (mph)
FE	Flight endurance of the drone (min)
GS	Moving speed of the ground vehicle (mph)
U	Number of wind turbines
I	Inspection time for a wind turbine (min)
P	Time for conducting pre- and post-flight procedures (min)
G_{ij}	Moving time between wind turbine i to wind turbine j by the ground vehicle (min)
F_{ij}	Flight time between wind turbine i to wind turbine j by the drone (min)
<i>Variables</i>	
C	Cluster assignment vector
T	Total operation time (min)
T^D	Total operation time of the drone (min)
T^G	Total operation time of the ground vehicle (min)
T^P	Total operation time for conducting pre- and post-flight procedures (min)
t_a^D	Total flight time of the drone between wind turbines in cluster a (min)
k	Number of clusters
m_a	Number of wind turbines in cluster a
x_{ij}	Binary decision variable that equals 1 if the ground vehicle moves from wind turbine i to wind turbine j , and 0 otherwise
y_{ab}	Binary decision variable that equals 1 if the ground vehicle moves from cluster a to cluster b , and 0 otherwise
z_{ij}	Binary decision variable that equals 1 if the drone flies from wind turbine i to wind turbine j , and 0 otherwise.
u	Integer variable representing the sequence number in which cluster is visited by the ground vehicle

Appendix 2: geographic locations of the wind turbines

Geographic locations of the wind turbines

No	Small-sized wind farm		Medium-sized wind farm		Large-sized wind farm	
	Latitude	Longitude	Latitude	Longitude	Latitude	Longitude
1	35.2360	–102.2494	32.7673	–99.5048	31.0812	–100.7139
2	35.2350	–102.2463	32.7662	–99.5021	31.0834	–100.7090
3	35.2297	–102.2419	32.7663	–99.4990	31.0851	–100.7048
4	35.2287	–102.2391	32.7668	–99.4961	31.0851	–100.7015
5	35.2289	–102.2350	32.7682	–99.4936	31.0851	–100.6981
6	35.2279	–102.2321	32.7694	–99.4910	31.0851	–100.6950
7	35.2289	–102.2282	32.7688	–99.4882	31.0855	–100.6889
8	35.2278	–102.2253	32.7679	–99.4853	31.0854	–100.6864
9	35.2267	–102.2224	32.7683	–99.4824	31.0855	–100.6838
10	35.2279	–102.2138	32.7690	–99.4796	31.0854	–100.6814
11	35.2268	–102.2110	32.7701	–99.4770	31.0854	–100.6789
12	35.2276	–102.1973	32.7709	–99.4741	31.0993	–100.7264
13	35.2266	–102.1944	32.7665	–99.4711	31.0967	–100.7218
14	35.2300	–102.1869	32.7664	–99.4683	31.0972	–100.7181
15	35.2289	–102.1840	32.7764	–99.5046	31.0962	–100.7140
16	35.2290	–102.1799	32.7765	–99.5015	31.0966	–100.7093
17	35.2280	–102.1771	32.7857	–99.5020	31.0969	–100.7045
18	35.2269	–102.1743	32.7865	–99.4993	31.0961	–100.7007
19	35.2278	–102.1697	32.7874	–99.4964	31.0961	–100.6980
20	35.2348	–102.1606	32.7870	–99.4935	31.1036	–100.6983
21	35.2336	–102.1574	32.7853	–99.4901	31.1042	–100.6956
22	35.2438	–102.1605	32.7834	–99.4864	31.1061	–100.6930
23	35.2447	–102.1629	32.7910	–99.4838	31.0969	–100.6904
24	35.2456	–102.1652	32.7908	–99.4796	31.0969	–100.6880
25	35.2363	–102.1924	32.7897	–99.4763	31.0969	–100.6851
26	35.2373	–102.1952	32.7841	–99.4746	31.0969	–100.6820
27	35.2375	–102.2030	32.7767	–99.4681	31.0969	–100.6795
28	35.2387	–102.2063	32.7751	–99.4650	31.1235	–100.7310
29	35.2466	–102.1996	32.7729	–99.4616	31.1240	–100.7283
30	35.2443	–102.2074	32.7652	–99.4579	31.1243	–100.7252
31	35.2454	–102.2103	32.7669	–99.4525	31.1243	–100.7211
32	35.2344	–102.2171	32.7660	–99.4495	31.1233	–100.7177
33	35.2354	–102.2200	32.7653	–99.4466	31.1235	–100.7141
34	35.2374	–102.2224	32.7653	–99.4410	31.1356	–100.7369
35			32.7654	–99.4379	31.1347	–100.7340
36			32.7652	–99.4266	31.1328	–100.7315
37			32.7653	–99.4235	31.1359	–100.6885
38			32.7652	–99.4197	31.1312	–100.6810
39			32.7651	–99.4167	31.1318	–100.6786
40			32.7655	–99.4137	31.1471	–100.7127
41			32.7685	–99.4102	31.1479	–100.7097
42			32.7789	–99.4537	31.1477	–100.7073
43			32.7786	–99.4501	31.1470	–100.7042
44			32.7900	–99.4625	31.1406	–100.7047
45			32.7888	–99.4592	31.1414	–100.7023
46			32.7898	–99.4567	31.1458	–100.6982
47			32.7903	–99.4538	31.1452	–100.6949

(continued)

No	Small-sized wind farm		Medium-sized wind farm		Large-sized wind farm	
	Latitude	Longitude	Latitude	Longitude	Latitude	Longitude
48		32.7891	–99.4507	31.1088	–100.6612	
49		32.7886	–99.4476	31.1078	–100.6579	
50		32.7875	–99.4444	31.1100	–100.6541	
51		32.7873	–99.4414	31.1089	–100.6513	
52		32.7873	–99.4385	31.1041	–100.6460	
53		32.7867	–99.4355	31.1053	–100.6433	
54		32.7825	–99.4307	31.1088	–100.6301	
55		32.7778	–99.4255	31.1083	–100.6264	
56		32.7789	–99.4073	31.1100	–100.6236	
57		32.7789	–99.4043	31.1110	–100.6198	
58		32.8056	–99.4684	31.1024	–100.6143	
59		32.8065	–99.4656	31.1015	–100.6114	
60		32.8069	–99.4626	31.1001	–100.6090	
61		32.8060	–99.4597	31.0999	–100.6066	
62		32.8034	–99.4567	31.1323	–100.6519	
63		32.8023	–99.4538	31.1330	–100.6492	
64		32.8015	–99.4475	31.1322	–100.6463	
65		32.8012	–99.4447	31.1310	–100.6434	
66		32.8007	–99.4409	31.1306	–100.6407	
67		32.8011	–99.4364	31.1295	–100.6373	
68		32.8012	–99.4324	31.1264	–100.6334	
69		32.7997	–99.4292	31.1252	–100.6309	
70		32.7986	–99.4265	31.1198	–100.6265	
71		32.7968	–99.4238	31.1225	–100.6194	
72		32.7905	–99.4204	31.1196	–100.6152	
73				31.1514	–100.6559	
74				31.1504	–100.6532	
75				31.1490	–100.6496	
76				31.1510	–100.6464	
77				31.1532	–100.6433	
78				31.1560	–100.6404	
79				31.1543	–100.6367	
80				31.1538	–100.6317	
81				31.1530	–100.6285	
82				31.1640	–100.6378	
83				31.1643	–100.6353	
84				31.1660	–100.6325	
85				31.1656	–100.6299	
86				31.1660	–100.6274	
87				31.1655	–100.6248	
88				31.1652	–100.6222	
89				31.1613	–100.6194	
90				31.1620	–100.6156	
91				31.1653	–100.6114	
92				31.1589	–100.6122	
93				31.1589	–100.6083	
94				31.1617	–100.6012	

(continued)

No	Small-sized wind farm		Medium-sized wind farm		Large-sized wind farm	
	Latitude	Longitude	Latitude	Longitude	Latitude	Longitude
95					31.1664	-100.5965
96					31.1519	-100.6105
97					31.1568	-100.6036
98					31.1661	-100.5925
99					31.1628	-100.5890
100					31.1539	-100.5877

References

- Agatz N, Bouman P, Schmidt M (2018) Optimization approaches for the traveling salesman problem with drone. *Transp Sci* 52(4):965–981
- American Wind Energy Association (2018) AWEA state wind energy facts. <https://www.awea.org/resources/fact-sheets/state-facts-sheets>. Accessed 28 Oct 2018
- Arzamendia M, Gregor D, Reina DG, Toral SL (2019) An evolutionary approach to constrained path planning of an autonomous surface vehicle for maximizing the covered area of Ypacarai Lake. *Soft Comput* 23(5):1723–1734
- Bouman P, Agatz N, Schmidt M (2018) Dynamic programming approaches for the traveling salesman problem with drone. *Networks* 72(4):528–542
- Boysen N, Briskorn D, Fedtke S, Schwerdfeger S (2018) Drone delivery from trucks: drone scheduling for given truck routes. *Networks* 72(4):506–527
- Capots M (2017) Self-flying drones and wind turbine blades: the new way of collecting data. Office of Energy Efficiency & Renewable Energy. <https://www.energy.gov/eere/articles/self-flying-drones-and-wind-turbine-blades-new-way-collecting-data>. Accessed 28 Oct 2018
- Carrabs F, Cerrone C, Cerulli R, Gaudioso M (2017a) A novel discretization scheme for the close enough traveling salesman problem. *Comput Oper Res* 78:163–171
- Carrabs F, Cerrone C, Cerulli R, D'Ambrosio C (2017b) Improved upper and lower bounds for the close enough traveling salesman problem. In: International conference on green, pervasive, and cloud computing, pp 165–177
- Carrabs F, Cerrone C, Cerulli R, Golden B (2020) An adaptive heuristic approach to compute upper and lower bounds for the close-enough traveling salesman problem. *INFORMS J Comput*. <https://doi.org/10.1287/ijoc.2020.0962>
- Chang YS, Lee HJ (2018) Optimal delivery routing with wider drone-delivery areas along a shorter truck-route. *Expert Syst Appl* 104:307–317
- Chen L, Peng J, Liu Z, Zhao R (2017a) Pricing and effort decisions for a supply chain with uncertain information. *Int J Prod Res* 55(1):264–284
- Chen L, Peng J, Zhang B (2017b) Uncertain goal programming models for bicriteria solid transportation problem. *Appl Soft Comput* 51:49–59
- Chowdhury S, Emeloglu A, Marufuzzaman M, Nur SG, Bian L (2017) Drones for disaster response and relief operations: a continuous approximation model. *Int J Prod Econ* 188:167–184
- Collins G, Clausse A, Twining D (2017) Enabling technologies for autonomous offshore inspections by heterogeneous unmanned teams. In: OCEANS 2017-Aberdeen, pp 1–5
- Coutinho WP, Nascimento RQ, Pessoa AA, Subramanian A (2016) A branch-and-bound algorithm for the close-enough traveling salesman problem. *INFORMS J Comput* 28(4):752–765
- Dantzig G, Fulkerson R, Johnson S (1954) Solution of a large-scale traveling-salesman problem. *J Oper Res Soc Am* 2(4):393–410
- de Freitas JC, Penna PHV (2018) A randomized variable neighborhood descent heuristic to solve the flying sidekick traveling salesman problem. *Electron Notes Discrete Math* 66:95–102
- Deign J (2016) Fully automated drones could double wind turbine inspection rates. New Energy Update. <http://newenergyupdate.com/wind-energy-update/fully-automated-drones-could-double-wind-turbine-inspection-rates>. Accessed 28 Oct 2018
- Denholm P, Hand M, Jackson M, Ong S (2009) Land-use requirements of modern wind power plants in the United States. <https://www.nrel.gov/docs/fy09osti/45834.pdf>. National Renewable Energy Laboratory. Accessed 28 Oct 2018
- Dorling K, Heinrichs J, Messier GG, Magierowski S (2017) Vehicle routing problems for drone delivery. *IEEE Trans Syst Man Cybern Syst* 47(1):70–85
- Faigl J, Váňa P, Deckerová J (2019) Fast heuristics for the 3-D multi-goal path planning based on the generalized traveling salesman problem with neighborhoods. *IEEE Robot Autom Lett* 4(3):2439–2446
- Ferrandez SM, Harbison T, Weber T, Sturges R, Rich R (2016) Optimization of a truck-drone in tandem delivery network using k-means and genetic algorithm. *J Ind Eng Manag* 9(2):374–388
- Fischetti M, Salazar González JJ, Toth P (1997) A branch-and-cut algorithm for the symmetric generalized traveling salesman problem. *Oper Res* 45(3):378–394
- Galleguillos C, Zorrilla A, Jimenez A, Diaz L, Montiano ÁL, Barroso M, Viguria A, Lasagni F (2015) Thermographic non-destructive inspection of wind turbine blades using unmanned aerial systems. *Plast Rubber Compos* 44(3):98–103
- Gambella C, Lodi A, Vigo D (2018) Exact solutions for the carrier–vehicle traveling salesman problem. *Transp Sci* 52(2):320–330
- Giosa ID, Tansini IL, Viera IO (2002) New assignment algorithms for the multi-depot vehicle routing problem. *J Oper Res Soc* 53(9):977–984
- Ha QM, Deville Y, Pham QD, Hà MH (2018) On the min-cost traveling salesman problem with drone. *Transp Res Part C Emerg Technol* 86:597–621
- Hong I, Kuby M, Murray AT (2018) A range-restricted recharging station coverage model for drone delivery service planning. *Transp Res Part C Emerg Technol* 90:198–212
- Jeong HY, Song BD, Lee S (2019) Truck-drone hybrid delivery routing: payload-energy dependency and no-fly zones. *Int J Prod Econ* 214:220–233
- Kim S, Moon I (2018) Traveling salesman problem with a drone station. *IEEE Trans Syst Man Cybern Syst* 49(1):42–52

- Kiran WS, Smys S, Bindhu V (2020) Enhancement of network lifetime using fuzzy clustering and multidirectional routing for wireless sensor networks. *Soft Comput* 24:11805–11818
- Lloyd S (1982) Least squares quantization in PCM. *IEEE Trans Inf Theory* 28(2):129–137
- Luo Z, Liu Z, Shi J (2017) A two-echelon cooperated routing problem for a ground vehicle and its carried unmanned aerial vehicle. *Sensors* 17(5):1144
- Marinelli M, Caggiani L, Ottomarelli M, Dell'Orco M (2018) En route truck-drone parcel delivery for optimal vehicle routing strategies. *IET Intel Transp Syst* 12(4):253–261
- Mathew N, Smith SL, Waslander SL (2015) Planning paths for package delivery in heterogeneous multirobot teams. *IEEE Trans Autom Sci Eng* 12(4):1298–1308
- Miller CE, Tucker AW, Zemlin RA (1960) Integer programming formulation of traveling salesman problems. *J ACM* 7(4):326–329
- Murray CC, Chu AG (2015) The flying sidekick traveling salesman problem: optimization of drone-assisted parcel delivery. *Transp Res Part C Emerg Technol* 54:86–109
- Murugesan VP, Murugesan P (2020) A new initialization and performance measure for the rough k-means clustering. *Soft Comput* 24:11605–11619
- Natarajan YJ, Nachimuthu DS (2020) New SVM kernel soft computing models for wind speed prediction in renewable energy applications. *Soft Comput* 24:11441–11458
- Otto A, Agatz N, Campbell J, Golden B, Pesch E (2018) Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: a survey. *Networks* 72(4):411–458
- Poikonen S, Golden B (2020) Multi-visit drone routing problem. *Comput Oper Res* 113:104802
- Poikonen S, Wang X, Golden B (2017) The vehicle routing problem with drones: extended models and connections. *Networks* 70(1):34–43
- Ponza A (2016) Optimization of drone-assisted parcel delivery. Master Degree Thesis, University Degli Studi Di Padova
- Rabta B, Wankmüller C, Reiner G (2018) A drone fleet model for last-mile distribution in disaster relief operations. *Int J Disaster Risk Reduct* 28:107–112
- Salhi S, Nagy G (1999) A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling. *J Oper Res Soc* 50(10):1034–1042
- Savuran H, Karakaya M (2016) Efficient route planning for an unmanned air vehicle deployed on a moving carrier. *Soft Comput* 20(7):2905–2920
- Schäfer BE, Picchi D, Engelhardt T, Abel D (2016) Multicopter unmanned aerial vehicle for automated inspection of wind turbines. In: 2016 24th mediterranean conference on control and automation, pp 244–249
- Semet F (1995) A two-phase algorithm for the partial accessibility constrained vehicle routing problem. *Ann Oper Res* 61(1):45–65
- Stokkeland M, Klausen K, Johansen TA (2015) Autonomous visual navigation of unmanned aerial vehicle for wind turbine inspection. In: 2015 International conference on unmanned aircraft systems, pp 998–1007
- Sundar K, Rathinam S (2014) Algorithms for routing an unmanned aerial vehicle in the presence of refueling depots. *IEEE Trans Autom Sci Eng* 11(1):287–294
- Sundar K, Rathinam S (2017) Algorithms for heterogeneous, multiple depot, multiple unmanned vehicle path planning problems. *J Intell Robot Syst* 88(2–4):513–526
- U.S. Energy Information Administration (2018) Annual energy outlook 2018 with projections to 2050
- Wang Z, Guo J, Zheng M, Wang Y (2015) Uncertain multiobjective traveling salesman problem. *Eur J Oper Res* 241(2):478–489
- Wang X, Poikonen S, Golden B (2017) The vehicle routing problem with drones: several worst-case results. *Optim Lett* 11(4):679–697
- Wang X, Golden B, Wasil E (2019) A steiner zone variable neighborhood search heuristic for the close-enough traveling salesman problem. *Comput Oper Res* 101:200–219
- Yang X, Jing F, Ma N, Nie F (2020) Supply chain pricing and effort decisions with the participants' belief under the uncertain demand. *Soft Comput* 24:6483–6497

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.