Cleaner Logistics and Supply Chain 4 (2022) 100041

Contents lists available at ScienceDirect

Cleaner Logistics and Supply Chain

journal homepage: www.elsevier.com/locate/clscn

The electric vehicle routing problem with drones: An energy minimization approach for aerial deliveries

Nikolaos A. Kyriakakis, Themistoklis Stamadianos, Magdalene Marinaki, Yannis Marinakis *

Technical University of Crete, School of Production Engineering and Management, University Campus, 73100 Chania, Greece

ARTICLE INFO

Keywords:
Drones
Electric vehicle
Unmanned aerial vehicle routing
Ant colony optimization

ABSTRACT

This paper introduces the Electric Vehicle Routing Problem with Drones (EVRPD), the first VRP combining electric ground vehicles (EVs) with unmanned aerial vehicles (UAVs), also known as drones, in order to deliver packages to customers. The problem's objective is to minimize the total energy consumption, focusing on the main non-constant and controllable factor of energy consumption on a delivery vehicle, the payload weight. The problem considers same-sized packages, belonging to different weight classes. EVs serve as motherships, from which drones are deployed to deliver the packages. Drones can carry multiple packages, up to a certain weight limit and their range is depended on their payload weight. For solving the EVRPD, four algorithms of the Ant Colony Optimization framework are implemented, two versions of the Ant Colony System and the Min–Max Ant System. A Variable Neighborhood Descent algorithm is utilized in all variants as a local search procedure. Instances for the EVRPD are created based on the two-echelon VRP literature and are used to test the proposed algorithms. Their computational results are compared and discussed. Practical, real-life scenarios of the EVRPD application are also presented and solved.

1. Introduction

There is currently a major force driving the developments in the logistics industry, that is, the necessity for green transportation, minimizing green house gas emissions. The proposed research addresses this aspect of the current and forthcoming routing problems, combining two novel means of transportation, namely Electric Vehicles (EVs) and Unmanned Aerial Vehicles (UAVs), with the objective of minimizing energy consumption.

According to the European Environment Agency, more than 70% of greenhouse gas emissions come from road vehicles, obscuring the climate goals set by the European Union. The phase-out of fossil fuel vehicles is more imminent than ever. Sooner or later, logistics companies will be required to shift from traditional vehicles to electric ones. New laws are constantly being put in place that prohibit either the sale or even the use of some kinds of traditional vehicles. For companies to stay competitive, the proper use of the EVs which will replace traditional vehicles is crucial.

Such circumstances necessitate the research and development in the field of electric vehicle routing, to make the adoption of EVs as cost efficient as possible. Great progressions on EV related technologies in the last two decades, mainly battery technologies, make them more appealing. Range remains their biggest shortcoming, and requires proper handling. Prices have come decreased enough for transportation companies to deem buying EVs a cost-effective decision, especially when governments subsidize part of the cost or provide tax reliefs. EVs operate quietly, without any local emissions, making them perfect for urban transportation. Light-weight vans, electric versions of which are commonplace in today's market, can be utilized to transition into an Eco-friendly transportation scheme.

Another type of electric vehicles are the UAVs, also known as drones, which share many of their technical aspects with the larger, ground EVs. Drones have additional limitations, as the weight of batteries today restrict their flight time to a fraction of their ground counterparts. Another limitation related to weight is the maximum payload they can transport. Despite their drawbacks drones can offer benefits, leveraging on their small size and agile nature. They are much more energy efficient for carrying lightweight packages, as they don't require moving a vehicle weighing multiple tons, only to deliver a package weighing just a few kilos. Furthermore, as they are airborne,

* Corresponding author.
E-mail addresses: nkyriakakis@isc.tuc.gr (N.A. Kyriakakis), tstamadianos@isc.tuc.gr (T. Stamadianos), magda@dssl.tuc.gr (M. Marinaki), marinakis@ergasya.tuc.gr (Y. Marinakis).

https://doi.org/10.1016/j.clscn.2022.100041
Received 30 November 2021; Revised 10 March 2022; Accepted 12 March 2022

they are not affected by traffic congestion and they do not contribute to its creation. This allows for faster transportation overall, not only directly by drone usage, but also indirectly by lowering traffic for road vehicles. Taking in consideration the benefits and limitations of drones, an urban area is a perfect environment for incorporating them in transportation operations. The combined utilization of these novel vehicle types offers advantages beyond the capabilities of traditional means of transportation. By utilizing a fleet of EVs, from which the drones are deployed and retrieved, the effective range of their operation can be significantly increased.

The proposed Electric Vehicle Routing Problem with Drones (EVRPD), combines EVs and drones to leverage on the strengths of each vehicle type and to overcome their limitations. This novel problem considers joint EVs and drone operations, where EVs are utilized to carry drones in pre-designated launch and retrieval locations, in order to perform the deliveries. The objective of the problem is to minimize the total energy consumption, while taking in consideration the travel range limitations, quantity capacities and maximum payload limits.

The EVRPD considers delivery packages belonging to different weight classes, which is often the case in practical delivery operations. Depending on the weight classes of the packages the drones carry, their energy consumption rate can change significantly, affecting their maximum travel distance and the total energy consumption. To model energy consumption, the energy cost model based on load and distance of Kara et al. (2007) is utilized, originally used for the Energy Minimizing Routing Problem.

For solving this novel and complex vehicle routing problem, four Ant Colony Optimization (ACO) algorithms are implemented and tested. Algorithms of the ACO family of algorithms, are well-studied and have been able to obtain quality solutions in many combinatorial optimization problems, such as the VRP.

Two of the algorithms used in this paper, are the Ant Colony System (ACS) and the Min–Max Ant Colony System (MMAS). The two other algorithms are the Hybrid Ant Colony System (HACS) and the Hybrid Min Max Ant System (HMMAS). These hybrid variants are based on successful implementations for the Cumulative Capacitated Vehicle Routing problem (CCVRP) found in Kyriakakis et al. (2021), combining the exploration abilities of swarm intelligence algorithms with the exploitation capabilities of Variable Neighborhood Descent (VND). Unlike the classical ACO algorithms, which generate a population based on the ACO rules, these hybrid algorithms utilize neighborhood operators in order to generate the ant population. Furthermore, a Variable Neighborhood Descent algorithm is used as a local search procedure to reinforce the exploitation properties of the hybrid algorithms. This paper introduces eight neighborhood operators for the EVRPD, which are used for generating the ant population and for the VND procedure.

This paper introduces the first vehicle routing problem integrating electric ground vehicles (EVs) and drones. Furthermore, the EVRPD utilizes a cost formulation which considers the minimization of total energy consumption. This objective is one of the main reasons for adopting these new vehicle types in modern supply chains. The different weight class packages considered, along with the energy, capacity and weight limitations taken in account, make the EVRPD formulation capable of closely modeling practical EV and drone delivery applications. The first results for the EVRPD are presented for abstract instances created based on the two-echelon VRP. Additionally, a practical case study is conducted on instances based on real-life data for an EVRPD application to a delivery operation in Chania, Greece. The results, highlight the importance of optimizing both EV and drone routes in conjunction, in order to minimize total energy consumption of operations. The EVRPD model minimizes total energy consumption in joint EV and drone operations.

The main goals of the research can be summarized as follows:

- The introduction of a novel VRP which focuses on minimizing the total energy consumption of delivery and combines state-of-the-art means of transportation, both in literature and practice.
- The theoretical contribution to the VRP literature of the EVRPD mathematical formulation.
- The development of four hybrid ACO algorithms in order to address the complexity of this novel problem.
- The suggestion of an effective approach for solving the EVRPD, based on the results of the computational experiments.
- The application of the EVRPD to practical scenarios as a case study in the city of Chania.

The rest of the paper is structured as follows. Section 2 includes a thorough literature review of the most recent and important research in the field of drone integration, VRP with drones (VRPD), and the Electric VRP (EVRP). In Section 3, the EVRPD is described in detail and its mathematical formulation is presented. Section 4 describes the proposed Hybrid ACO approach for solving the EVRPD. Section 5 presents and discusses the experimental results. Lastly, Section 6 presents the conclusions of the research.

2. Related literature

2.1. Electric ground vehicle routing problems

Conrad and Figliozzi (2011) introduce a problem resembling EVRP. It can be considered a predecessor for both EVRP and the Green VRP (GVRP). The novelty of their research is in the use of a vehicle with a limited range that requires refueling at customer locations but not specifically EVs. The GVRP, introduced in Erdoğan and Miller-Hooks (2012), employs Alternative Fuel Vehicles in place of the traditional vehicles. EVRP may be considered a variant of GVRP using EVs. Schneider et al. (2014) present first a variant of EVRP with Time Windows and Recharging Stations. Zhang et al. (2018) present an ant colony metaheuristic to solve the EVRP with recharging stations, aiming to minimize energy expenditure. Mao et al. (2020) augment the proposed ACO algorithm with local search methods and solve an EVRP with time windows and multiple charging options. To the best of the authors' knowledge, this is the only implementation providing the option to either swap or recharge the EV. Zang et al. (2021) introduce a Column Generation algorithm and focus on better battery handling. Keskin et al. (2021) examine the effects that the a priori unknown waiting time for charging may have when solving an EVRP with Time Windows. Basso et al. (2021) present a machine learning approach to determine the state of charge in an EVRP variant. They conduct two types of experiments to show the quality of their model and the potential benefits. Lin et al. (2021) consider the grid in their EVRP. Electricity prices fluctuate, and there is also the option for Vehicle to Grid energy transfer. Napoli et al. (2021) propose the construction of a Distribution Center equipped with renewable energy generation infrastructure. They conduct a case study to evaluate the feasibility. Schiffer et al. (2021) tackle a niche problem, mid-haul logistics with EVs. They study the feasibility and costs in the long term and carry out a case study to verify their claims. Chakraborty et al. (2021) aim to lower energy consumption which is shown to be an effective goal. Furthermore, the characteristics of the charging stations affect the solution quality. Another interesting scenario is the use of mixed fleets. Al-dalain and Celebi (2021) first determine the routes for urban deliv-

eries and then allocate them to an electric or internal combustion engine vehicle. Erdelić and Carić (2019) review all EV relative literature and highlight the underdeveloped infrastructure, fleet composition derived problems, and the economic impact to be among the challenges faced when looking into freight EVs. Some of the biggest impediments to an EV's range are reported to be poor weather conditions and range anxiety. Xiao et al. (2021) publish a recent review on EVRP, present a realistic model and provide valuable insights related to costs. They discuss charging strategies and energy consumption, two critical aspects of the EVRP.

2.2. Routing problems with drones

In the context of supply chains, the VRP seeks to determine the best customer visiting order for a vehicle fleet while minimizing one or more objectives. The Traveling Salesman Problem (TSP) is a case of VRP with only one vehicle. Murray and Chu (2015) first present a Traveling Salesman Problem with a Flying Sidekick (FSTSP). It portrays the most simple possible drone integration scheme. Kitjacharoenchai et al. (2019) extend the FSTSP formulation to solve the multiple TSP with Drones. They also allow drones to return to any vehicle. Jeong et al. (2019), apart from the payload, also consider possible detours needed to avoid places where drones are not allowed to operate. Murray and Raj (2020) solve the multiple FSTSP and determine that serving all customers with drones may not always be the best case. Raj and Murray (2020) solve again the multiple FSTSP with varying drone velocity, but conclude that dense delivery locations diminish the positive effect. Nguyen et al. (2020) solve the TSPD using a Monte-Carlo Tree Search algorithm. Gonzalez-R et al. (2020) introduce a broader model and do not determine a priori the meeting points of the truck and drones. Pina-Pardo et al. (2021) employ drones to resupply their vehicle while on the road and lower the total time by up to a fifth and argue this implementation is more realistic for now. Luo et al. (2021) present a model in which a drone may visit more than one node per trip.

Wang and Sheu (2019) are the first to address the VRPD and present a mixed-integer problem formulation. They assume identical travel speeds for both types of vehicles, which undermines the drone's abilities. Coindreau et al. (2019) present a generalized approach, considering VRP with transportable resources, including drones. Schermer et al. (2019) first route the trucks and then use a meta-heuristic for the drone operation, aiming to minimize the makespan. Sacramento et al. (2019) present an Adaptive Large Neighborhood Search method for the VRPD, and their purpose is to minimize the cost. They highlight the strong relationship between the range of the drones and the potential savings. Chiang et al. (2019) equip the vehicles with a single drone and use a GA to solve the VRP. Emissions and cost are their two concerns. Hu et al. (2019) address separately the truck and drone path planning and then proceed to jointly optimize them. Karak and Abdelghany (2019) present a realistic formulation too. Drones transported by trucks (serving as depots and battery swapping stations) can make multiple deliveries and are allowed to return to any station, making many trips if necessary. Moshref-Javadi et al. (2020) set to lower the customer waiting time. They allocate multiple drones per truck and conduct a case study. Deng et al. (2020) go beyond just drones and tackle a movement synchronization VRP, applicable to other means of transport too. In the study of Rossello and Garone (2020), only drones make deliveries, while trucks transport them to points dictated by the city's governing body. Pugliese et al. (2020), also, conclude that drone and ground vehicle integration in logistics has monetary, environmental, and other benefits. Kitjacharoenchai et al. (2020) and Li et al., 2020 take a two-echelon approach in drone integration. Both use the trucks as mobile depots, with the latter allowing for direct drone deliveries from the depot, addressed separately. Tamke and Buscher (2021) introduce a branch and cut algorithm, Euchi and Sadok (2021) introduce a hybrid sweep algorithm for the VRPD. Liu et al. (2021) focus on drone scheduling and strive to minimize their number. They develop a Genetic Algorithm and suggest that with high customer density or a large delivery radius, more drones are required. Shahzaad et al. (2021) introduce a drone delivery system closer to reality. They consider no-fly zones and wind conditions and assume that rooftops of city buildings may be charging points of delivery coordinates. Nguyen et al. (2021) route drones and trucks independently. Drones make only a single delivery each time, and their main handicap is determined to be battery capacity. They also improve upon 26 benchmark instances.

There are some other notable papers employing drones. Thibbotuwawa et al. (2019) look into the effects weather may have on drone deliveries and include collision avoidance. Liu (2019) solve a dynamic VRP for food delivery application. Lemardelé et al. (2021) research the effectiveness of autonomous vehicles for deliveries. They conclude that drones are more suitable for low density or large radius delivery areas, while ground vehicles would be a better option for dense, city application, following experiments in European capitals. Chauhan et al. (2019) tackle a maximum coverage problem with drones. Gu et al. (2020) present a set-covering problem for instant deliveries and look for the best drone take-off locations. They minimize both the number of vehicles and the makespan. Macias et al. (2020) intend to select the optimal hub for delivering essentials in disaster relief scenarios to best serve those in need and present a theoretical example. Ghelichi et al. (2021) introduce a new formulation to serve medical supply demand with drones. Rashid et al. (2020) set a goal of minimizing the cost of surveillance with drones. Zhen et al. (2019) do not only consider the visiting sequence, but also the height to save energy when possible. Cheng et al. (2020) give emphasis on the energy consumption. They compare their non-linear function to a linear one and observe differences of almost 10% on average. The transported weight is, also, taken into account.

Numerous reviews on drone routing and related problems are found in the literature. The most recent ones are presented below. Vidal et al. (2020) refer to drone integrated routing problems as an emerging branch of research. Macrina et al. (2020) focus their review on routing problems with drones, reviewing the TSP and VRP, as well as a drone-only and a combination of the previous three. Cheikhrouhou and Khoufi (2021) include in their review a comprehensive list of the recent literature concerning drone integration in multiple TSPs. A valuable review is presented by Moshref-Javadi and Winkenbach (2021), along with a taxonomy proposal. They discuss practical applications and conduct an extensive review. Li et al. (2021) review the drone integration problem from a two-echelon perspective. They look into related issues and concentrate on modeling perspectives. Besides the problem variants and solutions methods, Chung et al. (2020) discuss the factors that impede the real-world applications and provide a glimpse into research gaps.

The majority of VRPs found in the literature assume certainty in elements such as demand and fuel consumption. In practice, these values often are within an expected range and therefore they are uncertain (Kondratenko et al., 2006). An effective way to deal with uncertainty is through fuzzy approaches such as those suggested by Werners and Kondratenko (2018) for the VRP with bunkering tankers and Solesvik et al. (2021) which integrates a fuzzy decision support system for marine practices. The research of Kondratenko et al. (2021) combines fuzzy approaches and evolutionary algorithms to address transportation problems under uncertainty.

The proposed EVRPD, does not consider uncertainty in any of its elements and therefore, a "worst-case scenario" should be considered in its practical application. Extending the state-of-the-art in EV and drone routing literature, this paper aims to address the weight contribution

to the energy consumption, and minimize the latter, by proposing, formulating and solving the EVRPD. It combines several aspects of the above cited literature as it can be considered a two-echelon VRP, in which each drone's routing is a multi-trip VRP. Furthermore, the energy consumption element, which is a common issue in EV and drone routing applications, is considered both as objective and constraint.

3. The electric vehicle routing problem with drones

The EVRPD introduces a joint routing problem of EVs and drones, with the objective of minimizing the overall energy consumption of operations. Unlike the other approaches in the literature, it considers an energy consumption function based on payload weights for both types of vehicles. Moreover, it considers different types of packages which introduce additional constraints related to both capacity and energy limitations.

The problem considers the following scenario: A fleet of EVs, each capable of carrying up to a certain number of drones, are utilized in order to perform deliveries of packages to customers. The EVs begin their routes from the depot carrying all the items to be delivered by their assigned drones. The EVs visit pre-designated launch/retrieval locations, from which the drones begin their routes in order to make the final deliveries to customers. The EVs wait at each launch/retrieval position for their drones to complete all their routes starting from that particular position. Once all of the drones have returned to an EV, the EV continues its route to the other launch/retrieval positions. The routing operation ends when all customers have been served and all EVs have returned to the depot.

The packages to be delivered are of the same size and belong to three different weight classes. Each vehicle type has a maximum number of packages and a maximum payload weight it can carry. Furthermore, each vehicle type has a maximum energy limitation which imposes a constraint on the maximum distance it can travel. The remaining distance (range) at each step of the route depends on the payload weight the vehicle currently carries.

The goal of the EVRPD is to create EV and drone routes which minimize the energy consumption, while taking into consideration the different package types and the vehicles' range limitations. In the following subsections the package payload, energy and cost functions of the EVRPD routes are explained in detail.

Table 1
Weight classes of packages.

Package Weight Class	Weight Range (Weight Units)	Weight Accounted (Weight Units)
1	(0.0,1.0]	1.0
2	(1.0,2.0]	2.0
3	(2.0,3.0]	3.0

Fig. 1. EVRPD route example.

3.1. Payload weight

The problem considers same sized packages belonging to three weight classes. Table 1 presents an example of these weigh classes with their respective weight range, as well as, the weight considered in the payload, energy and cost functions.

Each drone can carry combinations of these packages in containers of standard dimensions. The number of containers used in this paper is three, as it is realistic number of small packages a drone can carry in real delivery applications. A single container corresponds to a single customer. Although drones can carry up to three packages, depending on the weight class of the item and the given payload weight limit of the drone, this might not always be the case. For this application the payload weight limit used is 4.0, which is a realistic value if the weight classes presented previously consider weight ranges in kilograms. The possible combinations of packages carried by a drone at any time are described in Table 2.

Therefore, at each arc between two customers at positions i, j in a drone route, the payload weight W_{ij} of the drone depends on the remaining customers to be visited after the customer at position j. For example we consider the drone route in Fig. 1, with its corresponding arc distances, in which all customers require one package of weight class 1:

The arc payload weights W_{ij} are calculated as follows:

$$W_{01} = 1.0 + 1.0 + 1.0 = 3$$
$$W_{12} = 1.0 + 1.0 + 0.0 = 2$$
$$W_{23} = 1.0 + 0.0 + 0.0 = 1$$
$$W_{30} = 0.0 + 0.0 + 0.0 = 0$$

3.2. Energy cost

Many elements are likely to influence the energy consumption. For the ground vehicles, traffic conditions, driving characteristics and even traffic lights alter the energy consumption rate. Weather conditions such as temperature, wind and humidity, also play an important role for the EVs, and particularly for drones, they influence their energy consumption rate significantly. However, the most important energy consumption factor for the drone, is the payload weight. Unlike weather conditions which cannot be controlled, the payload weight, and thus, the energy consumption of a drone can be controlled through routing optimization.

Table 2
Drone payload combinations.

Case	Containers			Payload Quantity (max: 3)	Payload Weight (max: 4.0 Weight Units)
1	PWC 1	–	–	1	1.0
2	PWC 1	PWC 1	–	2	2.0
3	PWC 1	PWC 1	PWC 1	3	3.0
4	PWC 1	PWC 2	–	2	3.0
5	PWC 1	PWC 1	PWC 2	3	4.0
6	PWC 2	–	–	1	2.0
7	PWC 2	PWC 2	–	2	4.0
8	PWC 3	–	–	1	3.0
9	PWC 3	PWC 1	–	2	4.0

The energy cost function used for the EVRPD is based on the route cost function for the Energy Minimizing Routing Problem introduced by Kara et al. (2007), which is an extension of the physical law of Mechanical Work found in classical mechanics:

$$Work = Force \times Displacement \tag{1}$$

It is intuitive that the more payload weight a vehicle carries, the more energy it consumes in order to traverse a fixed distance. Likewise, the energy consumption in order to transport a fixed payload weight increases relatively to the distance traveled. Based on the work formula in Eq. 1, the energy required to traverse an arc (i,j) for the EVRPD is given by the formulation in Eq. 2:

$$ArcEnergy = ArcDistance \times (1 + ArcPayload) \tag{2}$$

With the addition of an extra unit of payload weight, the formula calculates the energy consumption when the arc payload is zero, such as in the case where a drone is returning to the EV after completing all deliveries.

Thus, for the EVRPD route example, illustrated in Fig. 1 and presented in the previous subsection, the energy e_{ij} required at each arc (i,j) with distance d_{ij} is calculated below:

$$\begin{aligned} e_{01} &= d_{01} \times (1 + W_{01}) = 8 \times 4 = 32 \\ e_{12} &= d_{12} \times (1 + W_{12}) = 14 \times 3 = 42 \\ e_{23} &= d_{23} \times (1 + W_{23}) = 6 \times 2 = 12 \\ e_{30} &= d_{30} \times (1 + W_{31}) = 9 \times 1 = 9 \end{aligned}$$

Since the objective of the EVRPD is to minimize energy consumption, the route cost is given by the sum of arc energy consumption. Therefore, the cost for the example route is calculated as:

$$C = \sum_{(i,j) \in route} e_{ij} = 95$$

An interesting observation on the cost structure of the EVRPD is that if the order of the customers is reversed in the route, the cost changes. For the example used, it would be calculated as:

$$C_{reversed} = 9 \times 4 + 6 \times 3 + 14 \times 2 + 8 \times 1 = 90$$

3.3. Assumptions

As with all VRPs certain assumptions are made for the EVRPD and are presented below:

- Deliveries are performed by drones, only.
- One package is demanded by each customer.
- The EVs may not revisit previous nodes.
- The deployment and retrieval location is the same.
- EVs wait at the deployment/retrieval locations.
- Drones may be re-deployed if the range is sufficient.
- Multiple drones may be deployed from the same EV, simultaneously.
- Drones fly at a high enough altitude for which the area of interest can be considered flat. Drones achieve that altitude instantaneously after launch.
- The provided EVs and Drones can fulfill the demand.
- The drone deployment and retrieval process is considered to be instantaneous.
- No external forces affecting the vehicles are considered (i.e. weather conditions).

Most of the assumptions made, contribute to the overall goal of minimizing energy expenditure, such as the allowing only to drones to perform the deliveries to customers and keeping the EVs at the launch/retrieval locations waiting.

Since one package is the expected number of packages to be required by a customer in most practical drone delivery applications, the single package assumption is considered. Nevertheless, more than one packages can be taken in consideration, without changes in the formulation, by having the same customer as a duplicate (same location), demand another package.

3.4. Mathematical formulation of the EVRPD

In this subsection, the mathematical formulation for the EVRPD is presented. This problem is similar to a two-echelon formulation since the EVs drive to specific locations, launch the drones, and wait for them to return. When they return, they may move to the next stopping point. Stopping points may be considered the equivalent of satellite locations from the classic two-echelon. Therefore, the first echelon considers the EV routes between the depot and satellite positions, while the second echelon considers the drone routes from the satellite positions to the final customers.

The presented formulation borrows concepts from the works of Perboli et al. (2011) that introduced two-echelon problems in VRP, of Karak and Abdelghany (2019) that solve a VRP combining drones and trucks, of Lin et al. (2016) and Kancharla and Ramadurai (2020), in both of which the transported freight affects the energy consumption, and lastly of Jie et al. (2019), that proposed the two-echelon EVRP with battery swapping. Following the commodity flow type of formulation, dummy variables are used to denote the return to the depot and satellite locations.

V_D and $\acute{V_D}$ denote the set of the depot node and its dummy, respectively. The dummy depot set is used in the definition of set A_1. V_S is the set of satellites, and $\acute{V_S}$ is again a dummy. Set V_C includes the customer nodes. To separate the echelons, A_1 is the set of all the first echelon elements, while A_2 is the set of the second echelon ones. The number of customers and satellites are defined in n_c and n_s, respectively. K^{EV} and K^D are the sets of EVs and drones for their respective echelon, while their numbers are k_{EV} and k_d, respectively. The maximum payload capacity is Q^{EV} and Q^D, and their maximum energy is E^{EV} and E^D. d_{ij} represents the distance between nodes i and j and q_i represents the demand of vertex i.

As for the decision variables, x_{ijk} describes whether or not the EV k traversed the arc(i,j). z_{ijsk} is the equivalent variable for drones with the addition of satellite notation. t_i connects the two echelons together, by specifying the demand value of each satellite, while w_{ik} is responsible for the payload delivered to satellite i by the EV k. f^1_{ijk} and f^2_{ijsk} are variables that determine the payload of each vehicle k that arrives from i to j. Lastly, TD_{ijs+}, TD_{ijs-} help connect the two types of vehicles. TD_{ijs+} is equal to one if EV i transported drone j from the depot to satellite s and TD_{ijs-} is equal to one if EV i transported drone j from satellite s back to the depot.

The mathematical formulation is the following:

$$\begin{aligned} \min f = & \sum_{i \in A_1} \sum_{j \in A_1} \sum_{k \in K^{EV}} \left(d_{ij} \times \left(1 + f^1_{ijk}\right) \times x_{ijk} \right) \\ & + \sum_{i \in A_2} \sum_{j \in A_2} \sum_{k \in K^D} \sum_{s \in V_S} \left(d_{ij} \times \left(1 + f^2_{ijsk}\right) \times z_{ijsk} \right) \end{aligned} \tag{3}$$

Subject to:

$$\sum_{(i,j) \in A_1} x_{ijk} = \sum_{(i,j) \in A_1} x_{jik}, \forall i \in V_S, k \in K^{EV} \tag{4}$$

$$\sum_{(i,j)\in A_2} z_{ijsk} = \sum_{(i,j)\in A_2} z_{jisk}, \forall i \in V_C, s \in V_S, k \in K^D \quad (5)$$

$$\sum_{(i,j)\in A_1} x_{ijk} \leqslant 1, \forall i \in V_S, k \in K^{EV} \quad (6)$$

$$\sum_{k\in K^D} \sum_{s\in V_S} \sum_{(i,j)\in A_2} z_{ijsk} = 1, \forall i \in V_C \quad (7)$$

$$w_{ik} = \sum_{(i,j)\in A_1} f^1_{jik} - \sum_{(i,j)\in A_1} f^1_{ijk}, \forall i \in V_S, k \in K^{EV} \quad (8)$$

$$\sum_{i\in V_S} w_{ik} \leqslant Q^{EV}, \forall k \in K^{EV} \quad (9)$$

$$f^1_{ijk} \leqslant Q^{EV} \times x_{ijk}, \forall (i,j) \in A_1, k \in K^{EV} \quad (10)$$

$$\sum_{(i,j)\in A_2} f^2_{ijsk} = \sum_{(i,j)\in A_2} f^2_{jisk} - q_i, \forall i \in V_C, s \in V_S, k \in K^D \quad (11)$$

$$f^2_{ijsk} \leqslant Q^D \times z_{ijsk}, \forall (i,j) \in A_2, s \in V_S, k \in K^D \quad (12)$$

$$\sum_{k\in K^D} \sum_{(i,j)\in A_2} q_i \times z_{ijsk} = \sum_{l\in K^{EV}} w_{sl}, \forall s \in V_S \quad (13)$$

$$\sum_{s\in V_S} \sum_{k\in K^D} \sum_{(s,j)\in A_2} z_{sjsk} \leqslant k_d, \quad (14)$$

$$\sum_{i\in A_1} \sum_{j\in A_2} \left(1 + f^1_{ijk}\right) \times d_{ij} \times x_{ijk} \leqslant E^{EV}, \forall k \in K^{EV} \quad (15)$$

$$\sum_{i\in A_2} \sum_{j\in A_2} \sum_{s\in V_S} \left(1 + f^2_{ijsk}\right) \times d_{ij} \times z_{ijsk} \leqslant E^D, \forall k \in K^D \quad (16)$$

$$\sum_{\acute{s}\in\{V_S:\acute{s}\neq s\}} \left(\sum_{(s,j)\in A_2} z_{sj\acute{s}k} + \sum_{(i,\acute{s})\in A_2} z_{i\acute{s}sk} \right) = 0, \forall s \in V_s, \acute{s} \in \acute{V_S}, k \in K^D \quad (17)$$

$$\sum_{k\in K^D} \sum_{s\in V_S} z_{s\acute{s}k} = 0 \quad (18)$$

$$TD_{ijs+} = TD_{ijs-}, \forall i \in K^{EV}, j \in K^D, s \in V_S \quad (19)$$

$$x_{ijk} \in \{0,1\}, \forall (i,j) \in A_1, k \in K^{EV} \quad (20)$$

$$TD_{ijs+}, TD_{ijs-} \in \{0,1\}, \forall i \in K^{EV}, j \in K^D, s \in V_S \quad (21)$$

$$z_{ijsk} \in \{0,1\}, \forall (i,j) \in A_2, s \in V_S, k \in K^D \quad (22)$$

$$w_{ik} \geqslant 0, \forall i \in V_S, k \in K^{EV} \quad (23)$$

$$t_i \geqslant 0, \forall i \in V_S \quad (24)$$

$$f^1_{ijk} \geqslant 0, \forall (i,j) \in A_1, k \in K^{EV} \quad (25)$$

$$f^2_{ijsk} \geqslant 0, \forall (i,j) \in A_2, s \in V_S, k \in K^D \quad (26)$$

Constraints 4 and 5 are responsible for controlling the amount of inbound and outbound node traffic, ensuring each node has one incoming and one outgoing arc for the EVs and drones, respectively. Next, constraints 6 make sure that no node is visited more than once by the same EV. Constraints 7 require all customers to be visited exactly once by a drone. Constraints 8 help determine the amount of load transported by each EV to each satellite location, while constraints 9 make sure no violations of EV payload capacity take place. Constraints 10 state that the payload at each node may not exceed the maximum EV capacity. If arc (i,j) is not traversed, then the payload is 0.

As for the drones, constraints 11 update the transported payload at each stop (delivery point). Constraints 12 prohibit the remaining payload from exceeding the maximum drone capacity. If arc (i,j) is not traversed, then the payload is 0. Constraints 13 make sure the payload drones deliver from each satellite location is equal to the payload transported there by the EVs. Constraint 14 is responsible for limiting the number of available drones. Constraints 15 and 16 make sure there are no energy violations concerning the vehicles. Constraints 19 help identify and assign drones to EVs, to make sure that each drone is transported by the same EV during the whole journey. Constraints 17 and 18 aim to deter unwanted connections made between satellite locations or between satellites and their dummy counterparts. Lastly, constraints 21 to 26 limit all variables to their designated range.

4. The proposed hybrid ant colony optimization approach

4.1. Ant colony optimization

Ant Colony Optimization algorithms are part of the swarm intelligence optimization family with successful implementations in various routing problems. They search the solution space mimicking the way ants search for food in nature. Biological ants explore the surrounding areas of the nest for food and lay pheromone on the ground, based on the distance and quality of the food source. Paths with more pheromone are more likely to be chosen by other ants of the colony, while pheromone trails with lower amounts, tend to progressively disappear as a result of pheromone evaporation. After sufficient time, all ants converge to the path with the most pheromone. This mechanism of indirect communication by altering the environment is called stigmergy.

Based on the biological ants, Dorigo et al. (1996) introduced the fist ACO algorithm, the Ant System (AS). Since then, many ACO algorithms have been proposed, two of the most successful of which are the Ant Colony System (ACS) presented in Dorigo and Gambardella (1997) and the Max–Min Ant System (MMAS), introduced in Stützle and Hoos (2000). A great survey of different variants and applications of the ACO framework can be found in Mohan and Baskaran (2012).

For solving the EVRPD, this paper implements and tests four ACO algorithms. Two are adaptations of hybrid ACO-VND variants used in the Kyriakakis et al. (2021), originally designed to solve the Cumulative Capacitated Routing Problem (CCVRP). In these hybrid ACO-VND variants, only one ant solution is generated based on the ACO transition rules and the rest of the population is generated utilizing neighborhood structures. This neighborhood-generated population method has shown to provide better quality solutions for the CCVRP, requiring less computational time as it limits the search inside the feasible solution space. In order to exploit good solutions, a Variable Neighborhood Descent algorithm is included as a local search procedure. Based on this framework two algorithms, a Hybrid Ant Colony System (HACS) and a Hybrid Min–Max Ant System (HMMAS) are implemented to address the EVRPD.

The other two ACO algorithms tested are the classical ACS and MMAS, where the population is generated using the ACO transition rules. To increase exploitation, these algorithms also incorporate a VND procedure as local search. The VND utilized in the ACS and MMAS is the same used for the hybrid versions HACS and HMMAS.

The outlines of the HACS and HMMAS are presented in Algorithm 1 and Algorithm 2, respectively. Their individual steps and components are described in detail in the following subsections.

Algorithm 1: Hybrid Ant Colony System

Data: Intance,
$ACOiters, q_0, \rho, \beta, VNDiters, NBS = \{NB_1, NB_2, ..\}$
Result: S^{BSF}
Initialize τ, η ;
for $iter \leftarrow 1$ *to* $ACOiters$ **do**
  $S \leftarrow \text{createSolution}(\tau, \eta, q_0, \beta)$;
  for $ant \leftarrow 1$ *to* $|NBS|$ **do**
    $S[ant] = NB_{ant}(S)$;
    $\tau \leftarrow \text{locPherUpdate}(\tau, \rho, S[ant])$;
    if $cost(S[ant]) < cost(S^{IB})$ **then**
      $S^{IB} \leftarrow S[ant]$;
  $S^{IB} \leftarrow VND(S^{IB}, VNDiters)$;
  if $cost(S^{IB}) < cost(S^{IB})$ **then**
    $S^{BSF} \leftarrow S^{IB}$;
  $\tau \leftarrow \text{globalPheromoneUpdate}(\tau, \rho, S^{BSF})$;
return S^{BSF};

Algorithm 2: Hybrid Min–Max Ant System

Data: Intance,$ACOiters, Q, \rho, \alpha, \beta, VNDiters, NBS =$
$\{NB_1, NB_2, ..\}$
Result: S^{BSF}
Initialize $\tau, \eta, \tau_{max}, \tau_{min}$;
for $iter \leftarrow 1$ *to* $ACOiters$ **do**
  $S \leftarrow \text{createSolution}(\tau, \eta, \alpha, \beta)$;
  for $ant \leftarrow 1$ *to* $|NBS|$ **do**
    $S[ant] = NB_{ant}(S)$;
    if $cost(S[ant]) < cost(S^{IB})$ **then**
      $S^{IB} \leftarrow S[ant]$;
  $S^{IB} \leftarrow VND(S^{IB}, VNDiters)$;
  if $cost(S^{IB}) < cost(S^{BSF})$ **then**
    $S^{BSF} \leftarrow S^{IB}$;
  $\tau \leftarrow \text{evaporatePheromone}(\tau_{min}, \rho)$;
  $\tau \leftarrow \text{applyPheromone}(\tau_{max}, S^{BSF})$;
return S^{BSF};

4.1.1. ACS and MMAS solution construction

For the EVRPD, each ant represents a complete solution including both EV and Drone routes. In both ACO variants, a single ant starts its journey from the depot adding nodes (satellites or customers, depending on route type) to the route until a further addition violates one of the constraints. The ant then returns to the starting node (Depot or EV) and a second journey starts with the same drone or a new one, depending on the constraints in the case of a drone route, or with a

new EV in the case of EV route. This process is repeated until all customers have been serviced. For all implemented ACO variants, if the generated solution is infeasible, it is discarded and a new one is constructed.

In order to determine which node will be added next in the route, the ACS and HACS utilize the parameter q_0 to control greediness during the solution construction process. Furthermore, parameter β is used to moderate the relative importance of the heuristic information η. Given L_i, the set of customers not visited yet, in the case of drone routes or satellites for EV routes, the following transition rules are applied:

$$j = \begin{cases} \arg\max_{l \in L_i} [\tau_{il}][\eta_{il}]^{\beta}, & \texttt{if} q \leqslant q_0 \\ Z, & \texttt{otherwise} \end{cases} \tag{27}$$

where $q \in [0, 1]$ is a uniformly generated random variable, Z is the node selected according to probability distribution in Eq. (28).

$$p_{ij} = \begin{cases} \frac{[\tau_{ij}][\eta_{ij}]^{\beta}}{\sum_{l \in L_i} [\tau_{il}][\eta_{il}]^{\beta}}, & \texttt{if} j \in L_i \\ 0, & \texttt{otherwise} \end{cases} \tag{28}$$

The MMAS and HMMAS use parameters α and β to moderate the relative importance of both pheromone and the heuristic information, respectively. The probability of an ant moving from the node i to the node j for both the MMAS and HMMAS is presented in Eq. (29).

$$p_{ij} = \begin{cases} \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in L_i} [\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}}, & \texttt{if} j \in L_i \\ 0, & \texttt{otherwise} \end{cases} \tag{29}$$

4.1.2. Pheromone update rules

The ACS variants employ two pheromone update rules, a local update and a global update. In the construction phase, ants remove pheromone from the trails they have followed in order to urge other ants to follow other potentially unexplored paths. The levels are lowered according to the local update rule in Eq. (30):

$$\tau_{ij}^{new} = (1 - \rho)\tau_{ij}^{old} + \rho\tau_0 \tag{30}$$

$$\tau_0 = 1/(n \times C^{IS}) \tag{31}$$

τ_0 is the initial pheromone amount on the trails, based on the initial solution cost C^{IS} and the number of nodes n.

After all the ants construct a solution in the ACS variants, the best-so-far (BSF) ant updates the pheromone levels based on its cost C^{BSF}, using the global update rule in Eq. (32):

$$\tau_{ij}^{new} = (1 - \rho)\tau_{ij}^{old} + \Delta\tau_{ij}^{BSF} \tag{32}$$

$$\Delta\tau_{ij}^{BSF} = 1/C^{BSF} \tag{33}$$

where ρ is the pheromone evaporation factor.

The MMAS variants confine pheromone levels between $[\tau_{min}, \tau_{max}]$ in order to avoid premature convergence. All pheromone trails are initialized at the max values of τ_{max}. τ_{min} and τ_{max} values are given by Eqs. (35) and (34), respectively:

$$\tau_{max} = 1/\rho C^{IS} \tag{34}$$

$$\tau_{min} = \tau_{max}/Q_0 \tag{35}$$

where Q_0 is the parameter adjusting the τ_{min} value.

For the MMAS variants, similarly to the ACS, only the BSF ant lays pheromone on its trail, according to rule in Eq. (36):

$$\tau_{ij}^{new} = (1 - \rho)\tau_{ij}^{old} + \Delta\tau_{ij}^{BSF} \tag{36}$$

$$\Delta\tau_{ij}^{BSF} = 1/C^{BSF} \tag{37}$$

where ρ is the pheromone evaporation factor.

4.2. Variable neighborhood decent

A Variable Neighborhood Decent scheme is utilized as a local search procedure to exploit the solutions generated by the ACO. It is a deterministic variant of the Variable Neighborhood Search framework originally proposed by Mladenović and Hansen (1997) and can be implemented in various ways Mjirda et al. (2017). It is based on the concept of systematically changing neighborhood structures within the search process. For this implementation, a Pipe VND (P-VND) is used, where the search continues in the same neighborhood if it improves the solution. When no further improvement can be made in the particular neighborhood it proceeds in the next neighborhood. This process is repeated until no further improvement can be made by the last neighborhood.

Let $N = \{N_1, N_2, ..N_{k_{max}}\}$be a set of operators that map a given solution to a neighborhood structure $N_k(S)$. Algorithm 3 shows the VND procedure.

Algorithm 3: Variable Neighborhood Descent

```
Data: S, N = {N_1, N_2, ..N_k}, VNDiters
Result: S'
for iter ← 1 to VNDiters do
    for k ← 1 to k_max do
        improved ← True;
        repeat
            S' ← N_k(S);
            if cost(S) < cost(S') then
                S ← S';
            else
                improved ← False;
        until improved = False;
```

To complement the exploitation of solutions, the proposed VND procedure comprises of intra-route and inter-route local search operations, applied to the drone routes and EV routes. Specifically, the following eight neighborhood operators are used:

- **Intra-EV-Intra-Drone Operators: 1–1 Intra-route Swap:** Two customers of a single drone route, swap their positions. Example in Fig. 2. **1–1 Inter-route Exchange:** Two customers, each from a different route of the same drone, exchange their positions. Example in Fig. 3. **1–0 Inter-route Relocation:** A customer is removed from a drone route and is inserted in a different route of the same drone. Example in Fig. 4.
- **Intra-EV-Inter-Drone Operators: 1–1 Inter-route Exchange:** Two customers, each from routes of different drones, exchange their positions. Example in Fig. 5. **1–0 Inter-route Relocation:** A customer is removed from a drone route and is inserted in a route of another drone. Example in Fig. 6.
- **Inter-EV-Inter-Drone Operators: 1–1 Inter-route Exchange:** Same as the intra-route variant, but for drones belonging to different EVs. **1–0 Inter-route Relocation:** Same as the intra-route variant, but for drones belonging to different EVs.
- **EV-route Operator: 2-Opt Intra-route:** The order of the customers in the range of positions in the EV route are reversed. Example in Fig. 7.

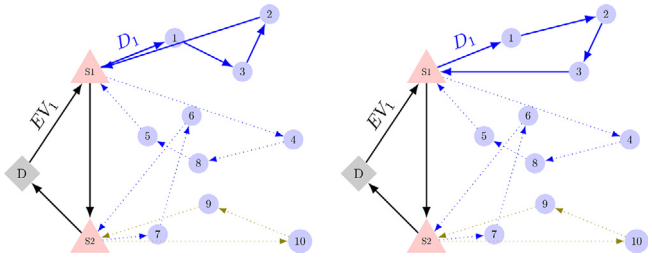

Fig. 2. Intra-EV-Intra-Drone, 1–1 Intra-route Swap.

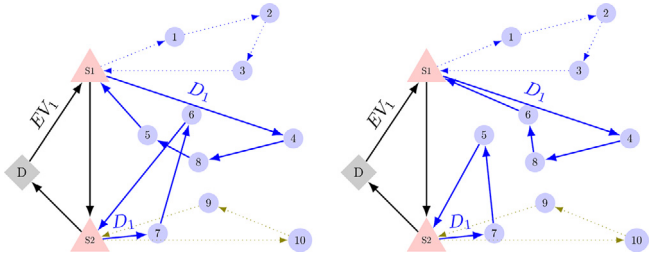

Fig. 3. Intra-EV-Intra-Drone, 1–1 Inter-route Exchange.

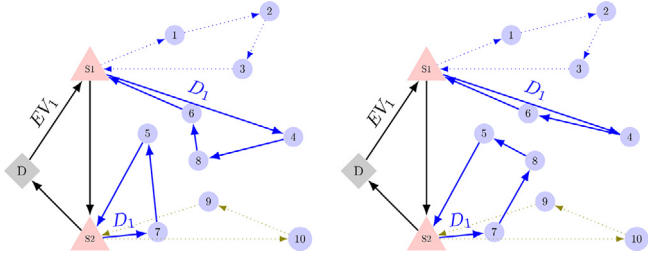

Fig. 4. Intra-EV-Intra-Drone, 1–0 Inter-route Relocation.

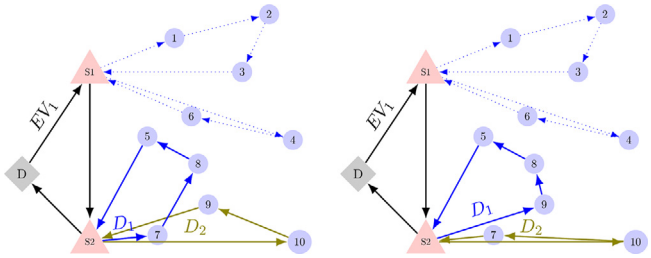

Fig. 5. Intra-EV-Inter-Drone, 1–1 Inter-route Exchange.

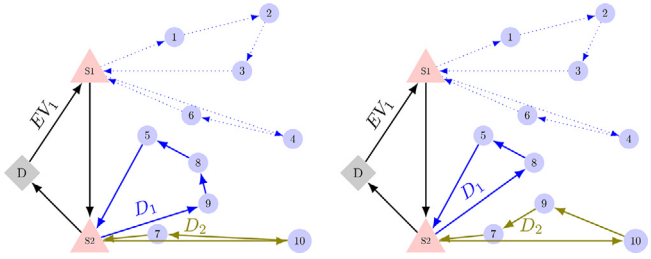

Fig. 6. Intra-EV-Inter-Drone, 1–0 Inter-route Relocation.

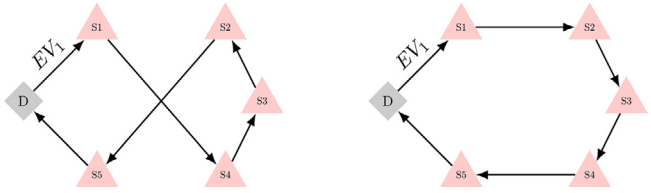

Fig. 7. EV-route, 2-Opt Intra-route.

5. Computational results

5.1. Problem instances

In order to test the algorithms, EVRPD instances were created based on Instance set 2, presented in Perboli et al. (2011) for the Two-Echelon VRP (2E-VRP). These instances consider scenarios ranging from 21 to 50 customers, with one depot location and a number of distribution centers, also called satellite positions.

In order to transform the 2E-VRP instances into EVRPD instances, for each instance, the demand range of its customers is partitioned in three equal parts to correspond with the three-item classes described in SubSection 3.1, essentially forming demand values in the set $\{1, 2, 3\}$.

The locations of the depot, the satellite positions and the locations of the customers are kept the same. For each instance the maximum number of drones per EV, as well as the size of the available EV fleet have been assigned based on the instance characteristics. Values range from 2 to 4 drones per EV and 2 to 3 available EVs.

Further to the benchmark instances created, a practical application is also considered. This case study is performed on 3 scenarios which are based on real-life locations in the city of Chania, Greece. In these examples, quantitative measurements in metric units such as kWh for energy, km for distance and kg for weight are used. The practical case study is presented in detail in SubSection 5.5.

5.2. Parameter settings and sensitivity

The four implemented ACO algorithms have a number of parameters that need to be determined before execution. Table 3 displays the parameters, their description and the values tested.

Table 3
Parameter description and settings.

Parameter	Description	Values tested
	ACS, HACS	
ACOiters	Number of iterations	10000
ρ	Pheromone evaporation factor	0.1
q_0	Controls exploitation during solution construction	$\{0.85, 0.90, 0.95\}$
β	Controls the importance of heuristic information	$\{1.0, 2.0, 3.0\}$
VNDiters	Number of the local search iterations	50
	MMAS, HMMAS	
ACOiters	the number of iterations	10000
Q_0	Used to set the τ_{min} value.	300
ρ	Pheromone evaporation factor	0.02
α	Controls the importance of pheromone trails	1.0
β	Controls the importance of heuristic information	$\{1.0, 2.0, 3.0, 4.0, 5.0\}$
VNDiters	Number of the local search iterations	50

The parameter values regarding the iterations, *ACOiters* and *VNDiters* of the algorithms, were determined by testing different values to find those for which the ants adequately converge on a solution. For the parameters controlling pheromone evaporation and pheromone ranges, the proposed values found in the ACO literature have been used. For the most important parameters, which significantly change the behavior of the algorithms, the recommended ranges in the literature have been used in a sensitivity test to determine the best values for this particular implementation.

Parameters play an important role in the performance of metaheuristic and swarm intelligence algorithms. In the ACS variants, the two most important parameters are β and q_0. For MMAS variants, the most important parameter is β. These parameters control the exploratory and exploitative properties of the corresponding algorithms, and thus, they significantly affect their convergence to the best found solution.

In this subsection, the sensitivity of the four implemented algorithms to the above mentioned parameters, is tested. The values $\{0.85, 0.90, 0.95\}$ and $\{1.0, 2.0, 3.0\}$ were tested for ACS parameters β and q_0, respectively. In the MMAS implementations the values $\{1.0, 2.0, 3.0, 4.0, 5.0\}$ for of MMAS parameter β were tested.

Fig. 8 displays the mean $Gap\%$ from the best solution cost value (BKV) for different values of q_0 and β of ACS and HACS. Both algorithms exhibit similar behaviors in relation to parameter β, which independently of parameter q_0, value $\beta = 1$ is able to obtain the best average gap. For parameter q_0, values 0.85 and 0.90 seem to benefit both algorithms for most values of parameter β. Thus, increased randomness in solution construction is observed to have a positive impact on the overall solution quality. The absolute range of gaps is 0.45 for the ACS and 0.28 for the HACS, making the latter less sensitive to parameters β and q_0.

Fig. 9 displays the mean $Gap\%$ from the best solution cost for different values of β for the MMAS and HMMAS algorithms. In both variants, $\beta = 1$ is able to obtain the best average results, with an absolute percentage difference of 0.06 to the second best value, $\beta = 2$. The absolute range of gaps is 0.21 and 0.17 for the MMAS and HMMAS, respectively. Thus, the HMMAS can be considered slightly less sensitive to parameter β. The gap for all figures is calculated by the formula in Eq. 38: (See Fig. 10)

$$Gap(Value) = 100 \times (Value - BKV)/BKV\% \tag{38}$$

Fig. 8 displays the plots indicating the 95% confidence interval of the $Gap(Cost)\%$ found for each algorithm and parameter setting. Observing the gaps of the obtained results, the sensitivity difference between ACS and HACS is even more prominent, with the ACS having more outlier values further away from the 95% interval. For the MMAS and HMMAS, the differences are not so visible, which is justified since their absolute range difference is small. In all plots, the parameter settings are sorted from worse to best, thus, the best settings for ACS and HACS are $\beta = 1$ and $q_0 = 0.85$, while for the MMAS and HMMAS the best parameter value for β is $\beta = 2$.

5.3. Experimental results

The algorithms are coded in C++ and compiled with GCC 11.2. The experiments are carried using a 2014 Intel®Core i7-4770 CPU (3.40 GHz) with 7.7 GB RAM on the Fedora Workstation 34 OS. Each instance was solved by each algorithm 15 times.

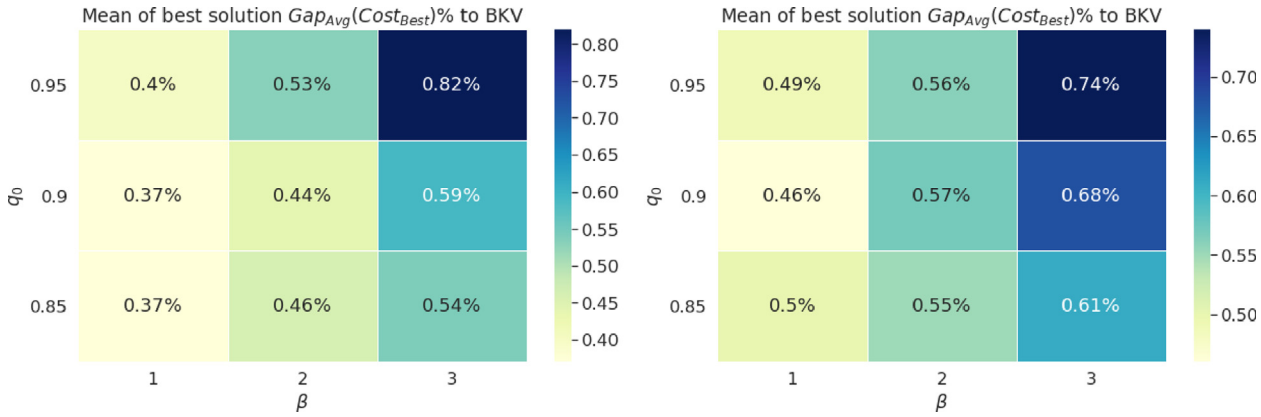

Fig. 8. $Gap_{Avg}(Cost_{Best})$% for ACS (left) and HACS (right).

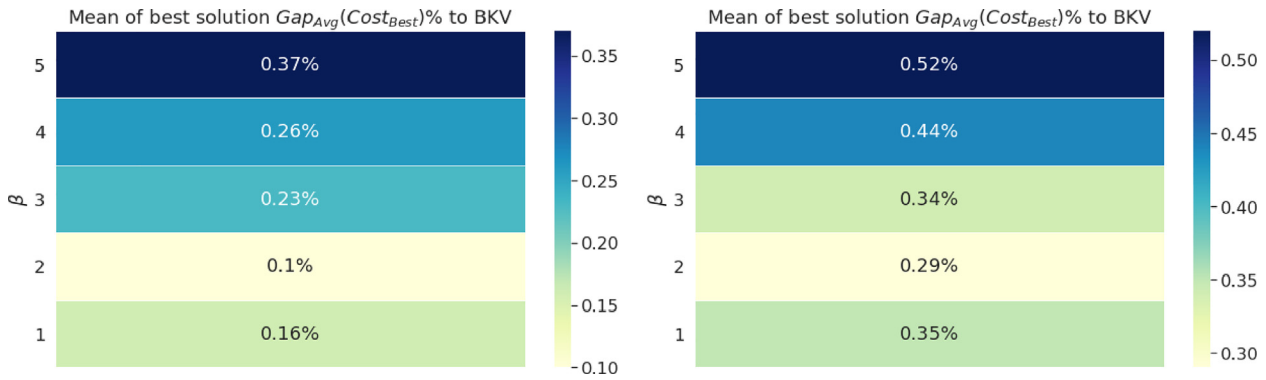

Fig. 9. $Gap_{Avg}(Cost_{Best})$% for MMAS (left) and HMMAS (right).

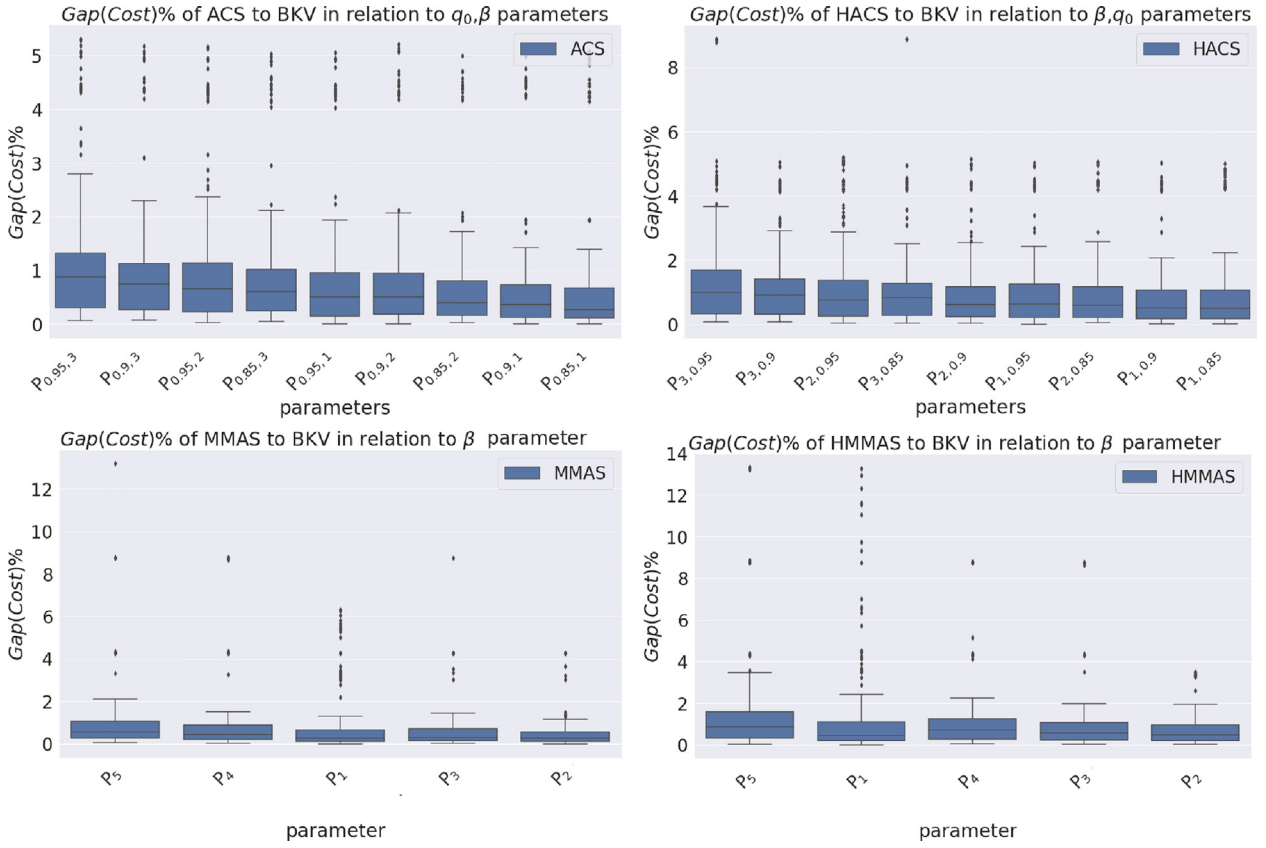

Fig. 10. *Gap*(*Cost*)% to BKV for each algorithm per parameter setting.

Table 4 presents the results obtained by each algorithm on the EVRPD instances. Column 1 denotes the instance name, with the name of the original 2E-VRP instance name found Perboli et al. (2011) and in parenthesis the number of satellite positions, the maximum number per EV allowed and the maximum number of EVs available. Columns 2, 5, 8 and 11 present the best solution values found by each algorithm. In Columns 3, 6, 9 and 12 are the best Gap% results compared to the BKVs for each instance. Columns 4, 7, 10 and 13 indicate the average solution cost obtained.

In terms of solution cost, the ACS and MMAS are observed to obtain better results than their hybridized counterparts, HACS and HMMAS. They are able to obtain more best known values (BKVs) and have significant differences in the average $Cost_{best}$ found. In particular, the ACS and HACS have a difference of 0.07% on average while the MMAS and HMMAS results differ on average by 0.15%. The neighborhood population generating process in HACS and HMMAS incorporates far less randomness than the classical ACS and MMAS transition rules and the population quality strongly depends on the neighborhood structures used. Therefore, a possible explanation for this underperformance, may lay on the limited size of the neighborhoods used for this particular problem. This assumption is reinforced by the observation made in the parameter sensitivity section, where increased randomness in solution construction seemed to translate in better results.

Overall, the MMAS was able to obtain 13 out of the 21 BKVs and is arguably the best performing among the implemented algorithms. Its average results also supports this observation.

The HMMAS, despite being the second best performing algorithm on average, was able to obtain only two BKVs. The ACS, third on average results, was able to surpass the HMMAS in the number of obtained BKVs, with 9 out of 21 instances. Therefore, an argument can be made

Table 4
Computational results for the EVRPD instances (BKV obtained in bold).

	ACS			HACS			MMAS			HMMAS		
Instance (#S,#D,#EV)	$Cost_{best}$	Gap_{best}%	$Cost_{avg}$	$Cost_{best}$	Gap_{best}%	$Cost_{avg}$	$Cost_{best}$	Gap_{best}%	$Cost_{avg}$	$Cost_{best}$	Gap_{best}%	$Cost_{avg}$
n22-k4-s10-14 (2,3,2)	**1144.28**	**0.00**	1149.75	1145.37	0.10	1158.12	**1144.28**	**0.00**	1152.62	1145.37	0.10	1162.67
n22-k4-s11–12 (2,3,2)	**1403.94**	**0.00**	1438.99	1406.06	0.15	1418.46	1404.22	0.02	1438.10	1405.84	0.14	1422.19
n22-k4-s12-16 (2,3,2)	1243.38	0.20	1252.32	1244.10	0.25	1258.47	1241.16	0.02	1246.51	**1240.95**	**0.00**	1248.07
n22-k4-s6-17(2,3,2)	1627.76	1.06	1640.91	1627.93	1.07	1648.49	**1610.70**	**0.00**	1627.89	1614.12	0.21	1632.96
n22-k4-s8-14 (2,3,2)	**1191.20**	**0.00**	1199.01	1194.06	0.24	1207.12	**1191.20**	**0.00**	1218.75	1194.06	0.24	1227.59
n22-k4-s9-19 (2,3,2)	1878.84	0.25	1884.06	**1874.20**	0.00	1883.43	**1874.20**	**0.00**	1879.37	1876.74	0.14	1880.37
n33-k4-s1-9 (2,3,2)	3599.67	0.01	3604.23	3600.27	0.03	3606.09	**3599.16**	**0.00**	3601.95	3599.75	0.02	3604.30
n33-k4-s14-22 (2,3,2)	4035.51	0.06	4040.06	4035.65	0.06	4040.50	**4033.19**	**0.00**	4037.81	4034.78	0.04	4039.27
n33-k4-s2–13 (2,3,2)	3429.00	0.00	3433.03	3429.00	0.00	3435.44	**3428.85**	**0.00**	3432.84	3429.00	0.00	3435.03
n33-k4-s3-17 (2,3,2)	3440.19	4.02	3459.68	3443.84	4.13	3456.86	**3307.26**	**0.00**	3319.26	3313.89	0.20	3334.94
n33-k4-s4-5 (2,3,2)	3795.70	0.00	3799.08	3796.16	0.01	3799.74	**3795.61**	**0.00**	3800.70	3796.47	0.02	3802.41
n33-k4-s7-25 (2,3,2)	3819.82	0.01	3826.87	3821.10	0.04	3827.77	**3819.62**	**0.00**	3825.91	**3819.62**	**0.00**	3827.03
n51-k5-s11–19 (2,3,3)	**3061.89**	**0.00**	3086.29	3067.14	0.17	3094.27	3067.40	0.18	3117.85	3084.38	0.73	3147.20
n51-k5-s11–19-27-47 (4,3,3)	**1916.57**	**0.00**	1928.39	1920.91	0.23	1931.76	1917.73	0.06	1925.49	1921.28	0.25	1932.76
n51-k5-s2-17 (2,3,3)	**2891.04**	**0.00**	2921.97	2902.64	0.40	2928.88	2909.18	0.63	2936.29	2907.18	0.56	2946.38
n51-k5-s2-4–17-46 (4,3,3)	**2895.94**	**0.00**	2922.84	2902.34	0.22	2929.37	2897.17	0.04	2938.39	2916.73	0.72	2953.09
n51-k5-s27-47 (2,3,3)	1918.45	0.05	1928.85	1921.03	0.18	1931.61	**1917.50**	**0.00**	1925.20	1921.10	0.19	1930.25
n51-k5-s32-37 (2,4,3)	**4918.59**	**0.00**	4924.44	4920.40	0.04	4927.77	4921.56	0.06	4939.19	4925.68	0.14	4955.94
n51-k5-s4-46 (2,3,3)	**4170.25**	**0.00**	4181.72	4171.78	0.04	4184.86	4171.00	0.02	4179.80	4176.14	0.14	4187.83
n51-k5-s6-12 (2,3,3)	2545.84	0.19	2561.59	2543.61	0.11	2568.61	**2540.91**	**0.00**	2554.11	2546.96	0.24	2567.47
n51-k5-s6-12–32-37 (4,3,3)	2545.90	0.09	2561.04	2546.75	0.12	2568.80	**2543.73**	**0.00**	2554.55	2549.05	0.21	2570.68
Average	2736.84	0.28		2738.77	0.36		2730.26	0.05		2734.24	0.20	

for both algorithms as the second best performer. The worst performing algorithm in terms of solution cost values is the HACS which was able to obtain only one BKV and has the worst average results by a significant margin. Its average result difference from the MMAS is 0.31%.

Regarding the performance of the algorithms in relation to instance characteristics, it is observed that the MMAS and HMMAS perform better on the smaller n22 and n33 instances than both the ACS and HACS implementations. On the larger n55 instances, the ACS is the best performing algorithm among all with the MMAS second best. The HACS is able to surpass the HMMAS for most n55 instances, and therefore, the HMMAS is the weakest algorithm for these larger instances.

The ACS and MMAS have different approaches regarding intensification and diversification during the search. The observed difference in their effectiveness depending on instance size may be attributed to their respective approaches. The direct feedback by removing pheromone immediately after constructing a solution, of ACS may allow the algorithm to overcome local minimums in larger instances. The MMAS, with its minimum and maximum pheromone levels, has the ability to keep many solution statistically possible, which helps the algorithm to not prematurely converge in the smaller instances.

Table 5 presents the average computational time required for each instance. The MMAS and HMMAS have similar average elapsed time, with the MMAS being slightly faster between the two and also overall. The HACS required more time than the HMMAS on average, but this difference can be attributed mostly to instance *n51-k5-s32-37*, in which the algorithm was trapped in an infeasible region of the solution space. This characteristic is even more prominent in the ACS which has computational times as outliers in 3 instances, and thus, required on

Table 5
Computational time for the EVRPD instances.

	ACS	HACS	MMAS	HMMAS
Instance (#S,#D,#EV)	$T_{avg}(s)$	$T_{avg}(s)$	$T_{avg}(s)$	$T_{avg}(s)$
n22-k4-s10-14 (2,3,2)	24.39	24.63	23.46	24.75
n22-k4-s11–12 (2,3,2)	22.03	23.73	21.81	23.78
n22-k4-s12-16 (2,3,2)	23.82	24.80	21.76	24.58
n22-k4-s6-17(2,3,2)	23.68	24.99	21.82	23.93
n22-k4-s8-14 (2,3,2)	23.33	24.11	22.16	24.06
n22-k4-s9-19 (2,3,2)	23.48	23.03	19.62	22.46
n33-k4-s1-9 (2,3,2)	45.74	43.50	39.76	41.41
n33-k4-s14-22 (2,3,2)	41.04	43.34	40.71	42.65
n33-k4-s2–13 (2,3,2)	45.93	43.25	37.57	40.41
n33-k4-s3-17 (2,3,2)	87.59	47.35	39.86	41.10
n33-k4-s4-5 (2,3,2)	411.43	75.04	36.53	38.12
n33-k4-s7-25 (2,3,2)	42.95	40.12	37.49	39.26
n51-k5-s11–19 (2,3,3)	76.96	60.21	57.25	59.11
n51-k5-s11–19-27-47 (4,3,3)	57.51	58.27	58.40	59.13
n51-k5-s2-17 (2,3,3)	68.52	57.94	56.15	59.25
n51-k5-s2-4–17-46 (4,3,3)	69.62	58.45	56.31	57.99
n51-k5-s27-47 (2,3,3)	58.34	58.31	57.80	59.09
n51-k5-s32-37 (2,4,3)	2181.76	851.37	59.89	60.03
n51-k5-s4-46 (2,3,3)	85.50	57.63	56.21	56.17
n51-k5-s6-12 (2,3,3)	57.28	58.94	56.21	56.81
n51-k5-s6-12–32-37 (4,3,3)	56.99	59.13	56.11	57.17
Average	167.99	73.72	41.75	43.39

average almost 4 times the running time of the MMAS implementations.

Based on the presented results, the MMAS approach is recommended for solving small to medium instances of the EVRPD. For instances with more than 50 customers, ACS is able to obtain the most BKVs with the MMAS very close second, therefore, a recommendation for large instances cannot be made confidently.

5.4. Statistical comparison

In order to verify the observed results, the algorithms are statistically tested against each other in order to determine superiority. Table 6 displays the results of a paired t-test, where the null hypothesis H0 assumes the given algorithms have equal means of results and the alternative hypothesis H1 assumes statistically different means.

The t-test is a parametric statistical test which assumes that the data approximate the normal distribution and have homogeneous or equal variance. Therefore, the non-parametric Wilcoxon signed-rank test is also utilized to further statistically test the significance of performance differences between the algorithms. Table 7 displays the results of the Wilcoxon signed-rank test.

Based on both the t-test and Wilcoxon signed rank test results, we can confidently state that the performances of ACS and MMAS are superior to the HACS and HMMAS, respectively, confirming the observations made on the experimental results subsection. ACS and MMAS are statistically equal, using a 95% confidence level, but the p-value of 0.079 in the Wilcoxon singed rank test, indicates that the MMAS may be considered superior for most cases. These statistical findings support the analysis in the results subsection, where the instance size factor on the performance of the algorithms was discussed. The statistical indifference between ACS and HMMAS is in accordance with the observation that although the ACS was able to obtain more BKVs, the HMMAS had on average best results. The HACS and HMMAS being statistically equally, indicates the split performance dominance, between smaller and larger instances.

Overall, the MMAS is the best choice for solving the EVRPD, and particularly in small instances, while the ACS might be able obtain better results for larger instances. The performance of the hybridized variants are not competitive compared to their respective original versions, but it should be noted that the HMMAS is both faster and better on average results than the ACS.

5.5. Practical application case study

In this section the proposed EVRPD is applied in 3 practical scenarios, which consider a delivery operation in the city of Chania, Greece. Each scenario has 25 customers and 3 satellite locations. The maximum number of EVs which can be used are 2 and the maximum number of drones in each EV is 3.

For this case study, the Volkswagen ABT e-Transporter is considered as the EV model and the Amazon Prime Air UAV is used as the drone model. For the VW ABT e-Transporter technical specifications can be found online Volkswagen-vans.co.uk (2022). For the Amazon Prime Air UAV, the specifications used are found in Jung and Kim (2017). Table 8 describes the specifications considered for each vehicle in this case study.

Since for both vehicles there is no formula to determine the exact consumption rate based on the weight of the payload, the values presented in the last column of Table 8 are based on the data available in the specifications with a pessimistic bias. Thus, instead of considering the nominal range of the drones at 16 km at 2.3 kg payload weight, calculations where based on a 10 km range with the same payload weight.

For the EVs, distances between the depots and satellite locations are based on the road distance estimates given by Google maps, while for the drones, the harvesine distance formula is used on the longitude and latitude coordinates.

Table 9 presents the energy consumption (EC) in Wh of the best solution found by each algorithm for each scenario. Columns 2, 4, 6 and 8 display the total EC while columns 3, 5, 7 and 9 denotes the EC of the drones alone. It is observed for Scenario 1 and Scenario 3, the EC of the EVs accounts for 40% to 46% of the total EC. For Scenario 2, this range is from 56% to 70%. This observation highlights, first of all, the positive environmental impact drones can have when incorporated in supply chains. Although, their per kg energy consumption rate is higher, their light weight makes them more efficient than EVs in delivering small packages to end customers. The results also indicate the importance of optimizing both EV and drone routes in conjunction. Since the two vehicle types are closely coupled, changing a single drone route can have a great impact, positive or negative, to the total energy consumption of the operation. The EVRPD model is able to optimize both, EVs and drones, with respect to their energy consumption, and thus, contributes to cleaner logistics.

Table 6
Paired t-test on the results of the four ACO implementations.

μ_A	μ_B	H_0	t-value	p-value	Result ($\alpha = 0.05$)
2736.845714	2738.778095	$\mu_{ACS} == \mu_{HACS}$	−2.713728	0.013370	Rejected
2736.845714	2730.268095	$\mu_{ACS} == \mu_{MMAS}$	1.019388	0.320192	Not Rejected
2736.845714	2734.242381	$\mu_{ACS} == \mu_{HMMAS}$	0.405119	0.689690	Not Rejected
2738.778095	2730.268095	$\mu_{HACS} == \mu_{MMAS}$	1.315665	0.203169	Not Rejected
2738.778095	2734.242381	$\mu_{HACS} == \mu_{HMMAS}$	0.707664	0.487319	Not Rejected
2730.268095	2734.242381	$\mu_{MMAS} == \mu_{HMMAS}$	−3.457735	0.002487	Rejected

Table 7
Wilcoxon signed-rank test on the results of the four ACO implementations.

$\overline{X}_A$	$\overline{X}_B$	H_0	w-value	p-value	Result ($\alpha = 0.05$)
2736.845714	2738.778095	$\overline{X}_{ACS} == \overline{X}_{HACS}$	29.0	0.004550	Rejected
2736.845714	2730.268095	$\overline{X}_{ACS} == \overline{X}_{MMAS}$	74.0	0.398063	Not Rejected
2736.845714	2734.242381	$\overline{X}_{ACS} == \overline{X}_{HMMAS}$	58.0	0.079322	Not Rejected
2738.778095	2730.268095	$\overline{X}_{HACS} == \overline{X}_{MMAS}$	27.0	0.003592	Rejected
2738.778095	2734.242381	$\overline{X}_{HACS} == \overline{X}_{HMMAS}$	63.0	0.327144	Not Rejected
2730.268095	2734.242381	$\overline{X}_{MMAS} == \overline{X}_{HMMAS}$	10.0	0.000390	Rejected

Table 8
Weight classes of packages.

Model	Weight (kg)	Battery (kWh)	Nominal Range (km)	max Payload (kg)	Approx. Energy Consumption (Wh/km/kg)
VW ABT e-Transoporter 6.1	2204.0	37.3	132 (at 15% max payload)	996.0	0.12007
Amazon Drone	5.5	0.37	16 (at 2.3 kg payload)	14.0	4.743589

Table 9
Case study results.

	ACS		HACS		MMAS		HMMAS	
	Total E.C.	Drone E.C.	Total E.C.	Drone E.C.	Total E.C.	Drone E.C.	Total E.C.	Drone E.C.
Instance	(Wh)	(Wh)	(Wh)	(Wh)	(Wh)	(Wh)	(Wh)	(Wh)
Scenario 1	2103.254	1253.062	2087.092	1236.9	**1841.373**	991.181	**1841.373**	991.181
Scenario 2	2292.075	990.967	2183.1901	682.4351	**2159.331**	658.576	2291.095	989.988
Scenario 3	2487.434	1423.725	**2369.989**	1306.28	2561.616	1497.907	2505.2425	1441.8455

In terms of per algorithm results, the MMAS was able to obtain the best results in 2 out of the 3 scenarios, while HMMAS and HACS obtained the best solution for 1 scenario. This agrees with the overall performance of the algorithms in small instances, where MMAS was able to obtain the majority of best solutions.

Fig. 11 displays the best solutions found for Scenario 1, Scenario 2 and Scenario 3, respectively. Routs drawn with thicker line represent the EV routes while thin lines are the drone routes. All routes belonging to the same drone have the same color.

6. Conclusions

In this paper, the first problem in literature combining electric ground vehicles and UAVs was presented, namely the Electric Vehicle Routing Problem with Drones. This novel problem utilizes these two vehicle types in order to serve customer demand, with the goal of minimizing the energy consumption of operations. This objective not only is in line with the overall green and eco-friendly nature of the vehicles used, but it also allows them to service more customers given their limited range. Electric vehicles will become more common in logistics and in combination with UAVs can provide significant benefits to the supply chain. The EVRPD considers a scenario in which EVs carry drones along with packages from the depot to pre-designated launch/retrieval positions, from where the drones perform the deliveries to customers. Different package weights are considered in order to calculate the drones' payloads, which are then used to calculate the energy consumption of their routes. The same energy consumption principle is also applied to the EVs' routes.

The EVRPD extends the state-of-the-art VRP literature, combining elements from the EVRP, the VRPD and two-echelon VRPs. The EVRPD focuses on the main contributing factor of energy consumption which can be controlled by optimizing the routing of the vehicle, that is, the payload weight.

The scenario considered in the EVRPD with its payload and energy cost elements were described in detail, along with its assumptions. The main contribution of the research to the VRP literature, from a theoretical perspective, is the mathematical formulation of the problem presented, with the detailed explanation of its variables, objective function and constraints.

In order to tackle the complexity of the EVRPD, four Ant Colony Optimization algorithms were developed and tested. The classical ACS and MMAS utilizing a VND local search procedure and two hybrid variants of those, namely the HACS and the HMMAS, based on the ACO-VND framework of Kyriakakis et al. (2021) originally proposed for the Cumulative Capacitated Vehicle Routing Problem. These algorithms follow a neighborhood-generated ant population scheme in order to effectively explore the solution space during solution construction. Additionally, a variable neighborhood descent algorithm is utilized to exploit the generated solutions and further improve the effectiveness of the search. In order to test the implemented ACO algorithms, a set of benchmark instances was created based on the two-echelon data set found in the literature. The instances were altered according to the needs of the EVRPD and its required problem parameters.

The instances were solved by the ACO implementations and their results were compared. The hybrid variants' performance did not meet the expectations and the success the framework had in the CCVRP. This might be attributed to the neighborhood structures utilized not being large enough to effectively search the solution space. The ACS and MMAS, having more randomness in their ant population generating process were able to obtain better solutions than their hybridized counterparts.

Overall, MMAS can be considered the best performing among them, having obtained 13 out of 21 best known values and having the best average results. ACS was second with 9 out of 21 BKVs. For larger instances the ACS was able to obtain more BKVs than MMAS, although it was not as competitive in terms of computational time. Both ACS and HACS had instances in which the algorithms were trapped in the infeasible region, requiring significantly more time to run. The HMMAS, although it was able to reach only 2 out of 21 BKVs, it had better average results than ACS and required on average a quarter of the computational time.

The EVRPD was also applied in 3 practical scenarios which considered a delivery operation in Chania, Greece using the Volkswagen ABT e-Transporter and the Amazon Prime Air UAV. The solutions obtained, with the energy consumption calculated in *Wh* indicated the importance of optimizing both vehicle types in conjunction, as well as, the significant impact they can have in minimizing the total energy consumption of operations. The EVRPD model, by combining EVs and drones in a two-echelon approach, with the goal of minimizing the total energy consumption, can contribute to cleaner logistics.

Future work on the EVRPD, includes variants which incorporate battery swaps for the drones and recharging for the EVs. Although, at the time of writing, recharging times are quite high to be considered during urban delivery operations, as battery technology progresses, problems including fast-charging should be considered for research. Another interesting future work, considers the dynamic version of the EVRPD, in which, due to unpredictable circumstances and conditions, batteries are depleted of energy faster than expected, and thus the routing operation must dynamically respond to these changes.

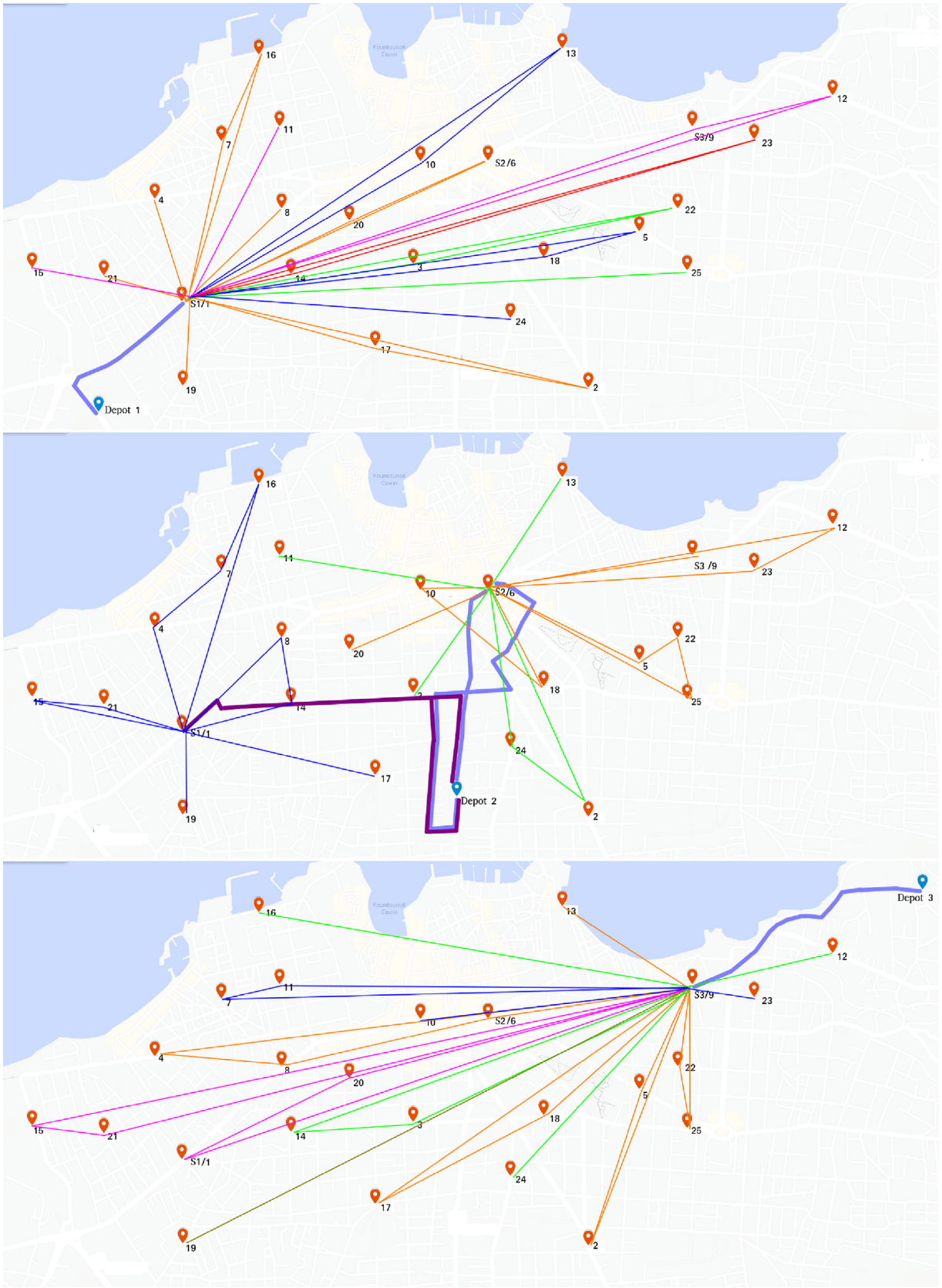

Fig. 11. Case study of Chania. Best solutions found for Scenario 1 (top), Scenario 2 (middle) and Scenario 3 (bottom).

Funding Sources

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Al-dalain, R., Celebi, D., 2021. Planning a mixed fleet of electric and conventional vehicles for urban freight with routing and replacement considerations. Sustain. Cities Soc. 73, 103105.
Basso, R., Kulcsár, B., Sanchez-Diaz, I., 2021. Electric vehicle routing problem with machine learning for energy prediction. Transp. Res. Part B: Methodol. 145, 24–55.
Chakraborty, N., Mondal, A., Mondal, S., 2021. Intelligent charge scheduling and eco-routing mechanism for electric vehicles: A multi-objective heuristic approach. Sustain. Cities Soc. 69, 102820.
Chauhan, D., Unnikrishnan, A., Figliozzi, M., 2019. Maximum coverage capacitated facility location problem with range constrained drones. Transp. Res. Part C: Emerging Technol. 99, 1–18.
Cheikhrouhou, O., Khoufi, I., 2021. A comprehensive survey on the multiple traveling salesman problem: Applications, approaches and taxonomy. Comput. Sci. Rev. 40, 100369.
Cheng, C., Adulyasak, Y., Rousseau, L.-M., 2020. Drone routing with energy function: Formulation and exact algorithm. Transp. Res. Part B: Methodol. 139, 364–387.
Chiang, W.-C., Li, Y., Shang, J., Urban, T.L., 2019. Impact of drone delivery on sustainability and cost: Realizing the uav potential through vehicle routing optimization. Appl. Energy 242, 1164–1175.
Chung, S.H., Sah, B., Lee, J., 2020. Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions. Comput. Oper. Res. 123, 105004.
Coindreau, M.-A., Gallay, O., Zufferey, N., 2019. Vehicle routing with transportable resources: Using carpooling and walking for on-site services. Eur. J. Oper. Res. 279 (3), 996–1010.
Conrad, R.G., Figliozzi, M.A., 2011. The recharging vehicle routing problem. In Proceedings of the 2011 industrial engineering research conference, page 8. IISE Norcross, GA..
Deng, P., Amirjamshidi, G., Roorda, M., 2020. A vehicle routing problem with movement synchronization of drones, sidewalk robots, or foot-walkers. Transp. Res. Procedia 46, 29–36.
Dorigo, M., Gambardella, L.M., 1997. Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evolut. Comput. 1 (1), 53–66.
Dorigo, M., Maniezzo, V., Colorni, A., 1996. Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst., Man, Cybern., Part B (Cybern.) 26 (1), 29–41.
Erdelić, T., Carić, T., 2019. A survey on the electric vehicle routing problem: variants and solution approaches. J. Adv. Transp. 2019..
Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. Transp. Res. Part E 48 (1), 100–114.
Euchi, J., Sadok, A., 2021. Hybrid genetic-sweep algorithm to solve the vehicle routing problem with drones. Phys. Commun. 44, 101236.
Ghelichi, Z., Gentili, M., Mirchandani, P.B., 2021. Logistics for a fleet of drones for medical item delivery: A case study for louisville, ky. Comput. Oper. Res. 135, 105443.
Gonzalez-R, P.L., Canca, D., Andrade-Pineda, J.L., Calle, M., Leon-Blanco, J.M., 2020. Truck-drone team logistics: A heuristic approach to multi-drop route planning. Transp. Res. Part C: Emerging Technol. 114, 657–680.
Gu, Q., Fan, T., Pan, F., Zhang, C., 2020. A vehicle-uav operation scheme for instant delivery. Comput. Ind. Eng. 149, 106809.
Hu, M., Liu, W., Lu, J., Fu, R., Peng, K., Ma, X., Liu, J., 2019. On the joint design of routing and scheduling for vehicle-assisted multi-uav inspection. Fut. Gener. Comput. Syst. 94, 214–223.
Jeong, H.Y., Song, B.D., Lee, S., 2019. Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones. Int. J. Prod. Econ. 214, 220–233.
Jie, W., Yang, J., Zhang, M., Huang, Y., 2019. The two-echelon capacitated electric vehicle routing problem with battery swapping stations: Formulation and efficient methodology. Eur. J. Oper. Res. 272 (3), 879–904.
Jung, S., Kim, H., 2017. Analysis of Amazon Prime Air UAV Delivery Service. J. Knowl. Inform. Technol. Syst. 12, 253–266.
Kancharla, S.R., Ramadurai, G., 2020. Electric vehicle routing problem with non-linear charging and load-dependent discharging. Expert Syst. Appl. 160, 113714.
Kara, İ., Kara, B.Y., Yetis, M.K., 2007. Energy minimizing vehicle routing problem. In Dress, A., Xu, Y., and Zhu, B., editors, Combinatorial Optimization and Applications, Berlin, Heidelberg. Springer, Berlin Heidelberg, pp. 62–71..
Karak, A., Abdelghany, K., 2019. The hybrid vehicle-drone routing problem for pick-up and delivery services. Transp. Res. Part C: Emerging Technol. 102, 427–449.
Keskin, M., Çatay, B., Laporte, G., 2021. A simulation-based heuristic for the electric vehicle routing problem with time windows and stochastic waiting times at recharging stations. Comput. Oper. Res. 125, 105060.
Kitjacharoenchai, P., Min, B.-C., Lee, S., 2020. Two echelon vehicle routing problem with drones in last mile delivery. Int. J. Prod. Econ. 225, 107598.
Kitjacharoenchai, P., Ventresca, M., Moshref-Javadi, M., Lee, S., Tanchoco, J.M., Brunese, P.A., 2019. Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. Comput. Ind. Eng. 129, 14–30.
Kondratenko, G., Kondratenko, Y., Romanov, D., 2006. Fuzzy models for capacitative vehicle routing problems in uncertainty. In: Proc. 17th International DAAAM Symposium, Intelligent Manufacturing and Automation: Focus on Mechatronics & Robotics, pp. 205–206.
Kondratenko, Y., Kondratenko, G., Sidenko, I., Taranov, M., 2021. Fuzzy and Evolutionary Algorithms for Transport Logistics Under Uncertainty Intelligent and Fuzzy Techniques: Smart and Innovative Solutions. In: Kahraman, C., Cevik, Onar S., Oztaysi, B., Sari, I., Cebi, S., Tolga, A. (Eds.), INFUS 2020. Advances in Intelligent Systems and Computing, vol 1197, pp. 1456–1463.
Kyriakakis, N.A., Marinaki, M., Marinakis, Y., 2021. A hybrid ant colony optimization-variable neighborhood descent approach for the cumulative capacitated vehicle routing problem. Comput. Oper. Res. 134, 105397.
Lemardelé, C., Estrada, M., Pagès, L., Bachofner, M., 2021. Potentialities of drones and ground autonomous delivery devices for last-mile logistics. Transp. Res. Part E 149, 102325.
Li, H., Chen, J., Wang, F., Bai, M., 2021. Ground-vehicle and unmanned-aerial-vehicle routing problems from two-echelon scheme perspective: A review. Eur. J. Oper. Res..
Li, H., Wang, H., Chen, J., Bai, M., 2020. Two-echelon vehicle routing problem with time windows and mobile satellites. Transp. Res. Part B 138, 179–201.
Lin, B., Ghaddar, B., Nathwani, J., 2021. Electric vehicle routing with charging/discharging under time-variant electricity prices. Transp. Res. Part C 130, 103285.
Lin, J., Zhou, W., Wolfson, O., 2016. Electric vehicle routing problem. Transp. Res. Procedia 12, 508–521.
Liu, C., Chen, H., Li, X., Liu, Z., 2021. A scheduling decision support model for minimizing the number of drones with dynamic package arrivals and personalized deadlines. Expert Syst. Appl. 167, 114157.
Liu, Y., 2019. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones. Comput. Oper. Res. 111, 1–20.
Luo, Z., Poon, M., Zhang, Z., Liu, Z., Lim, A., 2021. The multi-visit traveling salesman problem with multi-drones. Transp. Res. Part C 128, 103172.
Macias, J.E., Angeloudis, P., Ochieng, W., 2020. Optimal hub selection for rapid medical deliveries using unmanned aerial vehicles. Transp. Res. Part C 110, 56–80.
Macrina, G., Pugliese, L.D.P., Guerriero, F., Laporte, G., 2020. Drone-aided routing: A literature review. Transp. Res. Part C 120, 102762.
Mao, H., Shi, J., Zhou, Y., Zhang, G., 2020. The electric vehicle routing problem with time windows and multiple recharging options. IEEE Access 8, 114864–114875.
Mjirda, A., Todosijević, R., Hanafi, S., Hansen, P., Mladenović, N., 2017. Sequential variable neighborhood descent variants: an empirical study on the traveling salesman problem. Int. Trans. Oper. Res. 24 (3), 615–633.
Mladenović, N., Hansen, P., 1997. Variable neighborhood search. Comput. Oper. Res. 24 (11), 1097–1100.
Mohan, B.C., Baskaran, R., 2012. A survey: Ant colony optimization based recent research and implementation on several engineering domain. Expert Syst. Appl. 39 (4), 4618–4627.
Moshref-Javadi, M., Hemmati, A., Winkenbach, M., 2020. A truck and drones model for last-mile delivery: A mathematical model and heuristic approach. Appl. Math. Model. 80, 290–318.
Moshref-Javadi, M., Winkenbach, M., 2021. Applications and research avenues for drone-based models in logistics: A classification and review. Expert Syst. Appl. 177, 114854.
Murray, C.C., Chu, A.G., 2015. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transp. Res. Part C 54, 86–109.
Murray, C.C., Raj, R., 2020. The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transp. Res. Part C 110, 368–398.
Napoli, G., Micari, S., Dispenza, G., Andaloro, L., Antonucci, V., Polimeni, A., 2021. Freight distribution with electric vehicles: A case study in Sicily. res, infrastructures and vehicle routing. Transp. Eng. 3 (100047).
Nguyen, M.A., Dang, G.T.-H., Há, M.H., Pham, M.-T., 2021. The min-cost parallel drone scheduling vehicle routing problem. Eur. J. Oper. Res..
Nguyen, M.A., Sano, K., Tran, V.T., 2020. A monte carlo tree search for traveling salesman problem with drone. Asian Transport Stud. 6, 100028.
Perboli, G., Tadei, R., Vigo, D., 2011. The two-echelon capacitated vehicle routing problem: Models and math-based heuristics. Transp. Sci. 45 (3), 364–380.
Pina-Pardo, J.C., Silva, D.F., Smith, A.E., 2021. The traveling salesman problem with release dates and drone resupply. Comput. Oper. Res. 129, 105170.
Pugliese, L.D.P., Guerriero, F., Macrina, G., 2020. Using drones for parcels delivery process. Procedia Manuf. 42, 488–497.
Raj, R., Murray, C., 2020. The multiple flying sidekicks traveling salesman problem with variable drone speeds. Transp. Res. Part C 120, 102813.
Rashid, K., Speck, A., Osedach, T.P., Perroni, D.V., Pomerantz, A.E., 2020. Optimized inspection of upstream oil and gas methane emissions using airborne lidar surveillance. Appl. Energy 275, 115327.
Rossello, N.B., Garone, E., 2020. Carrier-vehicle system for delivery in city environments. IFAC-PapersOnLine 53 (2), 15253–15258.
Sacramento, D., Pisinger, D., Ropke, S., 2019. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. Transp. Res. Part C 102, 289–315.

Schermer, D., Moeini, M., Wendt, O., 2019. A matheuristic for the vehicle routing problem with drones and its variants. Transp. Res. Part C 106, 166–204.
Schiffer, M., Klein, P.S., Laporte, G., Walther, G., 2021. Integrated planning for electric commercial vehicle fleets: A case study for retail mid-haul logistics networks. Eur. J. Oper. Res. 291 (3), 944–960.
Schneider, M., Stenger, A., Goeke, D., 2014. The electric vehicle routing problem with time windows and recharging stations. Transp. Sci. 48 (4), 500–520.
Shahzaad, B., Bouguettaya, A., Mistry, S., Neiat, A.G., 2021. Resilient composition of drone services for delivery. Fut. Gener. Comput. Syst. 115, 335–350.
Solesvik, M., Kondratenko, Y., Kondratenko, G., Sidenko, I., Kharchenko, V., Boyarchuk, A., 2021. Fuzzy decision support systems in marine practice. In: 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1–6.
Stützle, T., Hoos, H.H., 2000. Max-min ant system. Fut. Gener. Comput. Syst. 16 (8), 889–914.
Tamke, F., Buscher, U., 2021. A branch-and-cut algorithm for the vehicle routing problem with drones. Transp. Res. Part B 144, 174–203.
Thibbotuwawa, A., Bocewicz, G., Nielsen, P., Zbigniew, B., 2019. Planning deliveries with uav routing under weather forecast and energy consumption constraints. IFAC-PapersOnLine 52 (13), 820–825.
Vidal, T., Laporte, G., Matl, P., 2020. A concise guide to existing and emerging vehicle routing problem variants. Eur. J. Oper. Res. 286 (2), 401–416.
Volkswagen-vans.co.uk (2022, February 1),https://www.volkswagen-vans.co.uk/idhub/content/dam/onehub_nfz/importers/gb/downloads/brochures/abt-e-transporter/abt-e-transporter-brochure.pdf..
Wang, Z., Sheu, J.-B., 2019. Vehicle routing problem with drones. Transp. Res. Part B 122, 350–364.
Werners, B., Kondratenko, Y., 2018. Alternative fuzzy approaches for efficiently solving the capacitated vehicle routing problem in conditions of uncertain demands. In: Berger-Vachon C., Gil Lafuente A., Kacprzyk J., Kondratenko Y., Merigó J., Morabito C. (eds) Complex Systems: Solutions and Challenges in Economics, Management and Engineering. Studies in Systems, Decision and Control, vol 125, Springer, pp. 521–543..
Xiao, Y., Zhang, Y., Kaku, I., Kang, R., Pan, X., 2021. Electric vehicle routing problem: A systematic review and a new comprehensive model with nonlinear energy recharging and consumption. Renew. Sustain. Energy Rev. 151, 111567.
Zang, Y., Wang, M., Qi, M., 2021. A column generation tailored to electric vehicle routing problem with nonlinear battery depreciation. Comput. Oper. Res. 137, 105527.
Zhang, S., Gajpal, Y., Appadoo, S., Abdulkader, M., 2018. Electric vehicle routing problem with recharging stations for minimizing energy consumption. Int. J. Prod. Econ. 203, 404–413.
Zhen, L., Li, M., Laporte, G., Wang, W., 2019. A vehicle routing problem arising in unmanned aerial monitoring. Comput. Oper. Res. 105, 1–11.