Cooperative trucks and drones for rural last-mile delivery with steep roads

Jiuhong Xiao [a] , Ying Li [a] , Zhiguang Cao [b] , Jianhua Xiao [a,c,*]

a. Research Center of Logistics, Nankai University, Tianjin 300071, China
b. School of Computing and Information Systems, Singapore Management University, 178902, Singapore
c. Laboratory for Economic Behaviors and Policy Simulation, Nankai University, Tianjin 300071, China

Published in Computers & Industrial Engineering (2024) 187, 109849, 2024. DOI: 10.1016/j.cie.2023.109849

Abstract: The cooperative delivery of trucks and drones promises considerable advantages in delivery efficiency and environmental friendliness over pure fossil fuel fleets. As the prosperity of rural B2C e-commerce grows, this study intends to explore the prospect of this cooperation mode for rural last-mile delivery by developing a green vehicle routing problem with drones that considers the presence of steep roads (GVRPD-SR). Realistic energy consumption calculations for trucks and drones that both consider the impacts of general factors and steep roads are incorporated into the GVRPD-SR model, and the objective is to minimize the total energy consumption. To solve the proposed model, an improved adaptive large neighborhood search (IALNS) algorithm is introduced, which incorporates several novel operators designed based on the characteristics of the problem. The effectiveness of the IALNS algorithm and the feasibility of the GVRPD-SR are verified through extensive computational experiments. Furthermore, a detailed sensitivity analysis is conducted on several critical parameters to derive managerial insights.

Keywords: Adaptive large neighborhood search, Cooperative delivery, Energy consumption, Steep roads, Trucks and drones

1. Introduction

With the popularity of online purchasing, business-to-consumer (B2C) e-commerce has accounted for a continuously increasing share of the global retail market. It is worth noting that rural areas have emerged as another booster for this trend due to the internet penetration and the perpetual innovation of business modes. As stated in the China Rural E-commerce Development Report (2021–2022) (China International Electronic Commerce Center, 2022, State Post Bureau of The People's Republic of China., 2022), the rural e-commerce trading volume reached 2.05 trillion RMB in 2021, accounting for 15.66 % of the country with an annual increase of 11.3 %. Likewise, the boom in rural B2C e-commerce is accompanied by a soaring number of delivery parcels in these areas. In China, more than 30 billion parcels were shipped in 2020, and the quantity exceeded 37 billion in 2021 (State Post Bureau of The People's Republic of China, 2022). To accommodate the present situation and provide satisfactory service levels, the logistic network for rural last-mile delivery has been rapidly expanded. However, many remote rural areas are characterized by low population density, rugged terrain, and mountainous landscapes. The current logistic network with pure fossil fuel fleets not only has a poorer delivery efficiency but also has a negative impact on the ecological environment due to their considerable energy consumption (Macrina et al., 2020, Zhang et al., 2022). Hence, rural last-mile delivery calls for other environmentally and efficient alternatives.

Over the past few years, the applications of drones have become one of the most talked-about developments in logistics. Several major companies have actively launched pilot programs to deliver parcels by drones, such as Amazon, DHL, Google, United Parcel Service (UPS), JD.com, and SF Technology (Macrina et al., 2020, Wohlsen, 2014). The benefits of drone-based delivery are apparent in these industrial implementations, including being battery-powered, operating autonomously, and traveling at high speeds while remaining unaffected by traffic congestion. Nonetheless, the limitations of drones in terms of carrying capacity, and flight endurance are also exposed. Consequently, some companies turned attention to the cooperative delivery of trucks and drones, which leverages the advantages of trucks to counteract the shortcomings of drones and vice versa. Specifically, a truck is used as a moving hub for supporting a drone and allowing both to perform delivery tasks. In 2014, AMP electric vehicles and the University of Cincinnati jointly constructed one such system, called the “HorseFly” (Wohlsen, 2014), followed by UPS deploying an electric van-drone system in 2017 (Wang et al., 2017, Kastrenakes, 2017). Meanwhile, the academic community is concerned about the route planning for this cooperative delivery. One of the major research directions is the vehicle routing problem with drones (VRPD)

proposed by Wang et al. (Wang et al., 2017), in which the fleet is equipped with several trucks and one or more drones. They validated the best possible time savings achieved by the cooperative delivery of trucks and drones versus trucks alone. Afterwards, many studies have developed different extensions and variants of the VRPD model (Chiang et al., 2019; Kitjacharoenchai et al., 2020; Poikonen et al., 2017; Schermer et al., 2019a), in which Chiang et al. (Chiang et al., 2019) concluded the environmental friendliness of using drones. All above indicate that the cooperation of trucks and drones is a promising option for rural last-mile delivery.

To actively follow the call of green campaigns, this study concentrates on the energy consumption of cooperative delivery by trucks and drones implemented in rural areas. While a few related studies exist (Chiang et al., 2019), they only assume that the energy consumption of trucks and drones is influenced by the load and the distance traveled on flat roads. In real scenarios for the rural last-mile delivery, however, customers are mostly dispersedly located at different altitudes and connected through steep roads. Trucks and drones need to frequently travel uphill or downhill, consuming a different amount of energy compared to traveling on flat roads. Consequently, ignoring the effect of steep roads can result in suboptimal routing plan, e.g., the planned route becomes undesirable as the battery endurance of the drone is depleted during transit.

In this study, a new green vehicle routing with drones model (GVRPD) is proposed for rural last-mile delivery. The objective is to minimize the energy consumption of drones and trucks. Specially, the realistic energy consumption calculations applicable to trucks and drones are incorporated into the GVRPD (i.e., the GVRPD-SR). They take steep road impact into account in addition to the general impact factors like travel distances, load, and speed. Since the GVRPD-SR is a complex variant of the classical vehicle routing problem (VRP), it is also an essentially NP-hard problem. An improved Adaptive Large Neighborhood Search (IALNS) algorithm is proposed to solve the GVRPD-SR, which comes along with several novel operators designed based on the characteristics of the problem. Finally, a series of experiments are performed in terms of algorithmic performance, model analysis, and managerial discussions to verify the effectiveness of the proposed GVRPD-SR and IALNS algorithm.

The rest of this study is organized as follows. Section 2 provides a literature review on related problems. The formulation of the GVRPD-SR model is presented in Section 3, and the introduction of the IALNS algorithm to solve the model is described in Section 4. Section 5 implements a series of experiments to comprehensively demonstrate the significance of the GVRPD-SR and IALNS algorithm. Section 6 concludes the contributions of this study and discusses the research directions that can be moved forward.

2. Literature review

In this section, the existing literature related to the GVRPD-SR problem is reviewed and discussed.

2.1. Green vehicle routing problems with steep roads

As excessive energy consumption and generated pollution lead to the continuous deterioration of the ecological environment, GVRP as a variant of the classic VRP was first attempted by Sharma and Mathew (Sharma and Mathew, 2011), who use speed-dependent emission functions to minimize emissions of various pollutants in the transportation network. Since then, some researchers have devoted their efforts to VRPs with energy consumption to prevent more environmental damage. Bektaş and Laporte (Bektaş and Laporte, 2011) first introduced a pollution routing problem (PRP) model that considers not only the travel distance but also the amount of greenhouse gas (GHG) emissions, fuel consumption, travel time, and costs. Subsequently, various characteristics of VRPs, such as time windows (Zhu and Hu, 2019), pickup and delivery (Majidi et al., 2018), and multi-depot (Sadati and Çatay, 2021), have been incorporated into GVRP models with energy consumption.

The abovementioned research assumes that the energy consumption rate is constant and has a linear relationship with the distance traveled by vehicles. In fact, as reported by the US Department of Energy (US Department of Energy, 2008), other factors such as load and vehicle speed also have a considerable impact on the amount of energy consumed. Consequently, more and more GVRP models calculate fuel consumption based on these actual factors (Karakostas et al., 2020; Liu et al., 2023; Rauniyar et al., 2019). Several researchers have paid attention to establishing different energy consumption models with consideration of steep roads to estimate more accurate fuel consumption, since the arcs to be traveled are not always distributed over the flat and horizonal terrain in real-world scenario. In Goeke and Scheneider (Goeke and Schneider, 2015), a realistic energy consumption function was designed to assist in route planning for a mixed fleet of electric and conventional vehicles. Especially, the function considers the combined impact of weight, vehicle speed, and steep road. Similarly, Zhou and Lee (Zhou and Lee, 2017) proposed a GVRP model considering vehicle speed as a decision variable. Their objective is to minimize emissions, which are affected by the aforementioned three factors. Schröder and Cabral (Schröder and Cabral, 2019) introduced a three-dimensional routing model based on Geographic Information System (GIS) that considers the impact of steep roads and varying speeds on GHG emissions and fuel consumption. Brunner et al. (Brunner et al., 2021) presented a VRP with steep roads (VRP-SR) that considers steep road and load effects in fuel consumption. In their model, a detailed road network obtained from a GIS was used to estimate the road grade between any two customer locations. Dundar et al. (Dundar et al., 2022) developed a green traveling salesman problem where fuel consumption is calculated by dynamic customer demands and realistic road gradients of the entire road network.

With the rapid development of advanced intelligent technology, a variety of multi-mode delivery services have emerged over the past several years, such as truck-drone joint delivery and autonomous robot-assisted van delivery. The existing research on these delivery modes has mainly concentrated on developing mathematical models aimed at reducing cost and increasing economic benefit. However, the announced worldwide goals of carbon peaking and carbon neutrality urgently call for more environmentally friendly practices in freight transportation. Hence, the PRP studies on traditional single-type vehicle delivery can provide valuable guides and insights from GVRP research into emerging joint delivery modes.

2.2. Trucks and drones cooperative routing problems

Since several major online retailers pioneered the application of drones in delivery services, the academic community has aroused a keen interest in the cooperative delivery of drones and trucks.

One mainstream research direction is the VRPD proposed by Wang et al. (Wang et al., 2017), which is directly relevant to this study. On the one hand, several studies extended the VRPD model by adding new constraints. For example, Poikonen et al. (Poikonen et al., 2017) proposed an extended VRPD model with a constraint on the limited battery life of a drone. Schermer et al. (Schermer et al., 2019a) proposed a multi-drone VRPD model. Sacramento et al. (Sacramento et al., 2019) introduced the VRPD with time limits for vehicles. Li et al. (Li et al., 2020); Coindreau et al. (Coindreau et al., 2021), and Kuo et al. (Kuo et al., 2022) considered time windows for customers. On the other hand, a couple of researchers developed several variants of VRPD from the aspects of the number of drone visits, drone release, and the synchronization between the drone and truck. Schermer et al. (Schermer et al., 2019b) allowed drones to be launched and rendezvoused both at the nodes and some discrete points on arcs. Wang and Sheu (Wang and Sheu, 2019) presented a VRPD model in which the launched drones must return to some predetermined docking locations before being

Table 1
Overview of the related literature.

Literature	Drone(s) per truck	Drone multiple visits	Payload capacity	Time windows	Steep road	Objective function	Solution method
Wang et al. (2017)	M	1	×	×	×	Minimize completion time	Worst-case analysis
Poikonen et al. (2017)	M	1	×	×	×	Minimize completion time	Worst-case analysis
Schermer et al. (2019a)	M	1	×	×	×	Minimize makespan	MILP, Matheuristic
Wang and Sheu (2019)	M	M	√	×	×	Minimize logistics cost	Branch and price
Kuo et al. (2022)	1	1	√	√	×	Minimize traveling cost	GUROBI, VNS
Sacramento et al. (2019)	1	1	√	×	×	Minimizes operational cost	CPLEX, ALNS
Li et al. (2020)	M	M	√	√	×	Minimize the integrated cost	CPLEX, ALNS
Schermer et al. (2019b)	1	1	×	×	×	Minimize makespan	Gurobi, VNS/TS
Salama and Srinivas (2022)	M	1	×	×	×	Minimizes completion time	CPLEX, OTS
Liu et al. (2021)	1	M	√	×	×	Minimize traveling cost	SA-TS
Coindreau et al. (2021)	1	1	×	√	×	Minimize global costs (including truck fuel costs and drone electricity costs)	CPLEX, ALNS
Meng et al. (2023)	1	M	√	×	×	Minimize total cost	MP-ISA
Chiang et al. (2019)	1	1	√	×	×	Minimize CO_2 emissions	GA
This study	1	1	√	×	√	Minimize energy consumption	CPLEX, IALNS

picked up by the trucks. Salama and Srinivas (Salama and Srinivas, 2022) devised a VRPD model that allows the truck to stop at non-customer locations for a drone launch or rendezvous. Meng et al. (Meng et al., 2023) developed a multi-visit VRPD problem that allows drones to provide both pickup and delivery services in each flight. In addition, considering that drones have a much less negative impact on the environment than fossil fuel-powered vehicles, Chiang et al. (Chiang et al., 2019) discussed the positive environmental impact of VRPD by incorporating the evaluation of CO_2 emissions of trucks. In Liu et al. (Liu et al., 2021), an energy consumption model of the route process of the drone was incorporated to analyze the impact of the payload, flying speed, travel distances, and efficiency of the motor. Table 1 summarizes the previous VRPD studies by classifying them based on the main features.

Although the above contributions have led to the development of a range of truck-drone models, their applicability is predominantly tailored to urban last-mile delivery scenarios. This focus is misaligned with the evolving e-commerce that is witnessing a significant surge in demand in rural areas. In fact, the cooperative delivery of trucks and drones has become equally compelling in rural areas. However, the routing plan for trucks and drones determined by existing models might be suboptimal, potentially leading to increased cost, time, or polluting emissions, as they neglect that scattered customers are often connected by steep roads in rural environments. Therefore, the GVRPD-SR model, which takes into account the impact of steep roads, is proposed in this study to investigate the potential value of this mode in rural last-mile delivery from a green perspective.

2.3. Solution methods

As an extension of the well-known VRP that is an NP-hard problem, the VRPD can also be considered NP-hard in a strong sense (Chiang et al., 2019; Sacramento et al., 2019; Wang et al., 2017). Moreover, the varied restrictions described in the preceding section make the problem considerably more intractable to solve. Solution methods proposed in the literature can be categorized as either exact or *meta*-heuristic algorithms. However, exact algorithms struggle to solve the problem in polynomial time as the scale and complexity of the problem increase. Consequently, fewer researchers use exact algorithms as solution methods. Wang & Sheu (Wang and Sheu, 2019) derived a branch-and-cut algorithm for solving the VRPD using a path-based model structure. Kang & Lee (Kang and Lee, 2021) developed an exact algorithm integrating with the logic-based Benders decomposition approach to solve a heterogenous VRPD problem. To solve a two-echelon VRPD problem, Zhou et al. (Zhou et al., 2023) proposed a branch-and-price algorithm that incorporates a bidirectional labeling algorithm for the pricing problem to enhance the search efficiency.

Up to now, *meta*-heuristic algorithms are used as a mainstream

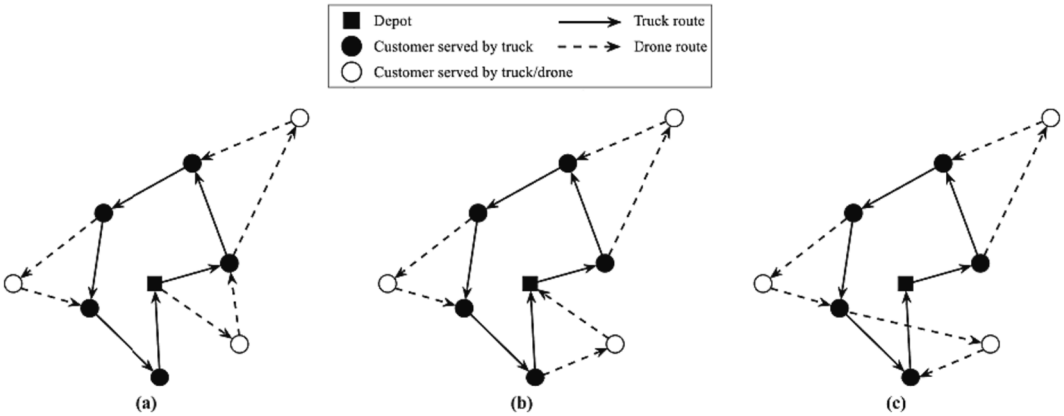

Fig. 1. Cooperation mode of truck and drone.

solution method for the VRPD, as they can find the optimal solution or high-quality approximate optimal solution within an acceptable time. For example, Kuo et al. (Kuo et al., 2022) developed a variable neighborhood search algorithm with a specific solution representation to solve a VRPD problem with time windows. Liu et al. (Liu et al., 2021) combined a simulated annealing algorithm with tabu search to solve a two-echelon VRPD problem. ALNS algorithm has become a mainstream solution method among *meta*-heuristic algorithms for many researchers (Coindreau et al., 2021; Li et al., 2020; Sacramento et al., 2019), as seen from Table 1, since it has superior capabilities with respect to extensibility, balancing between local search and global search, and preventing premature convergence. However, these ALNS algorithms were particularly designed to address the corresponding VRPDs based on unique characteristics of the problem, so they are not the optimal choice for efficiently addressing the proposed GVRPD-SR. In this study, the IALNS algorithm is proposed for the GVRPD-SR, as detailed in Section 4.

3. Problem statement

This section presents the problem definition of the GVRPD-SR as well as its mathematical formulation.

3.1. Problem definition

The GVRPD-SR considers the delivery of parcels from a depot to multiple customers by using a fleet of homogeneous trucks, each of which is equipped with a drone. Customers are located at different altitudes and connected by steep roads with varied road grades. Each customer can be served by either a truck or a drone. Regarding the cooperation between a truck and a drone, as illustrated in Fig. 1(a), they are allowed to depart from the depot independently and return to the depot in tandem. Fig. 1(b) shows that they also can depart from the depot in tandem and return to the depot independently. Fig. 1(c) depicts them departing from and returning to the depot in tandem. During the delivery process, each truck not only performs delivery operations but also acts as a mobile charging station for its own drone. When the truck delivers the parcel at a customer node, the drone can take off with a small parcel from the truck to serve another customer independently. After the drone completes its operation, it can only rendezvous with its truck at a customer node and then swaps a full-charged battery to relaunch or prepare for the next delivery task. Since a drone consumes energy to hover and a long hover may pose operational challenges, it is assumed that the drone cannot arrive before its dedicated truck to keep the model practicable.

The objective of the GVRPD-SR is to determine the main route for the trucks and the adjoint sub-routes for the drones to complete the delivery of all parcels with minimal energy consumption. The other common assumptions, referring to those introduced in Zhen et al. (Zhen et al., 2023), are summarized as follows:

(1) A drone can only serve one customer at a time by conducting back-and-forth flights within its maximum endurance.
(2) A drone can only be launched and retrieved from its dedicated truck at the depot or a customer node, which can occur at a different location.
(3) Each truck is equipped with sufficient batteries for its dedicated drone. As the battery replacement operation is quick, the consumed time is neglectable in the model formulation.
(4) Due to the research focus of this study, the impact of meteorological and external conditions on trucks and drones is ignored. Consequently, the truck and drone maintain a constant speed, making the delivery time only related to travel distance.
(5) The flight angle of a drone is assumed to be parallel to the steep road between the customers.
(6) The required setup time before the launch and retrieval of drones is ignored, as it is extremely short in comparison to travel time.

Table 2
Required notations for the GVRPD-SR model.

Sets	
V	Set of homogeneous trucks $V = \{1, 2, \ldots, e\}$, which is indexed by v
N	Set of all nodes $N = \{0, 1, \ldots, n+1\}$, where 0 and $n+1$ denote the starting and ending depots, respectively
N_c	Set of customers $N_c = \{1, 2, \ldots, n\}$
N_d	Set of departing nodes $N_d = N\backslash\{n+1\} = \{0, 1, \ldots, n\}$
N_a	Set of arriving nodes $N_a = N\backslash\{0\} = \{1, \ldots, n, n+1\}$
F	Set of the possible path for drones, $(i,j,k) \in F$, $i \in N_d$, $j \in \{N_c : j \neq i\}$,$k \in \left\{N_a : k \neq j, k \neq i, cd_{ijk} \leq E, q_j \leq Q_d\right\}$
Parameters	
q_i	Demand of customer $i \in N_c$
Q_t	Maximum load capacity of the truck
Q_d	Maximum load capacity of the drone
cap_b	Maximum battery endurance of the drone
Π_{ij}^t	Time required for a truck to travel from node $i \in N_d$ to node $j \in N_a$
Π_{ij}^d	Time required for a drone to travel from node $i \in N_d$ to node $j \in N_a$
Π_h^d	Time required for a drone to fly vertically at h altitude
ct_{ij}^v	Energy consumption for truck $v \in V$ to travel from node $i \in N_d$ to node $j \in N_a$
cd_{ijk}	Energy consumption for a drone to travel the path $(i,j,k) \in F$
M	An infinite value
Decision variables	
x_{ij}^v	Binary variable, with 1 indicating truck $v \in V$ travels from node $i \in N_d$ to node $j \in N_a$, and 0 otherwise
y_{ijk}^v	Binary variable, with 1 indicating the drone on truck $v \in V$ is launched from node $i \in N_d$, and then serves to node $j \in N_c$, finally rendezvous to truck v or the ending depot at node $k \in N_a$, and 0 otherwise
w_{ij}^v	Integer variable, the load of truck $v \in V$ from node $i \in N_d$ to node $j \in N_a$
τ_{vi}^t	Nonnegative continuous variable, the arrival time of truck $v \in V$ at node $i \in N$
τ_{vi}^d	Nonnegative continuous variable, the arrival time of the drone on truck $v \in V$ at node $i \in N$
μ_i^v	Integer variable, the order of node $i \in N$ in the path of truck $v \in V$
ρ_{ij}^v	Binary variable, with 1 indicating truck $v \in V$ serves node $i \in N$ before node $j \in N$, and 0 otherwise

More formally, the GVRPD-SR can be regarded as a directed graph $G = (N, A)$. In particular, $N = N_0 \cup N_c$ is the node set, where $N_0 = \{0, n+1\}$ represents the depot and N_c={1,…,n} is the set of customers to be served. Each customer $i \in N_c$ has $q_i \in \mathbb{R}_+$demand. Let A denote a set of all possible arcs connecting nodes. A travel distance $d_{ij} \in \mathbb{R}_+$is associated with each arc $a = (i,j) \in A$, which is calculated by the Euclidean distance. It is represented as $d_{ij} = \sqrt{d_{ij}^{'2} + h_{ij}^2}$, where $d_{ij}^{'}$ denotes the horizontal distance and h_{ij} is the vertical distance (Rao et al., 2016). The road grade between any two customers is assumed to be the angle between d_{ij} and $d_{ij}^{'}$. A fleet of homogenous trucks V= {1,…,m}with capacity $Q_t \in \mathbb{N}_+$is dispatched, each equipped with a single drone with capacity $Q_t \in \mathbb{N}_+$. Table 2 defines the notations required to formulate a mixed integer programming (MIP) model of the GVRPD-SR.

3.2. Calculation of energy consumption

3.2.1. Energy consumption model for trucks

Since the energy consumption of a truck is associated with the physical calculation of the power demand for moving a point of mass, the mechanical power formulation presented in (Bektaş and Laporte, 2011) is used as follows.

$$P^T = m_t \cdot g \cdot \sin\alpha \cdot v_t + c_{roll} \cdot m_t \cdot g \cdot \cos\alpha \cdot v_t + \frac{a \cdot c_{air}}{2 \cdot 10^3} \cdot A_t \cdot v_t^3 + \frac{n^{acc} \cdot 0.504}{2 \cdot 10^3 \cdot 3.6} \cdot m_t \cdot v_t^3 \quad (1)$$

where m_t is the truck mass (ton), g is the gravitational constant (m/s^2), α represents the road grade, c_{roll} is the rolling resistance coefficient, a is the air density (kg/m^3), c_{air} denotes the aerodynamical drag coefficient, A_t is the vehicle frontal surface area (m^2), and v_t and ν_t are vehicle speed (m/s

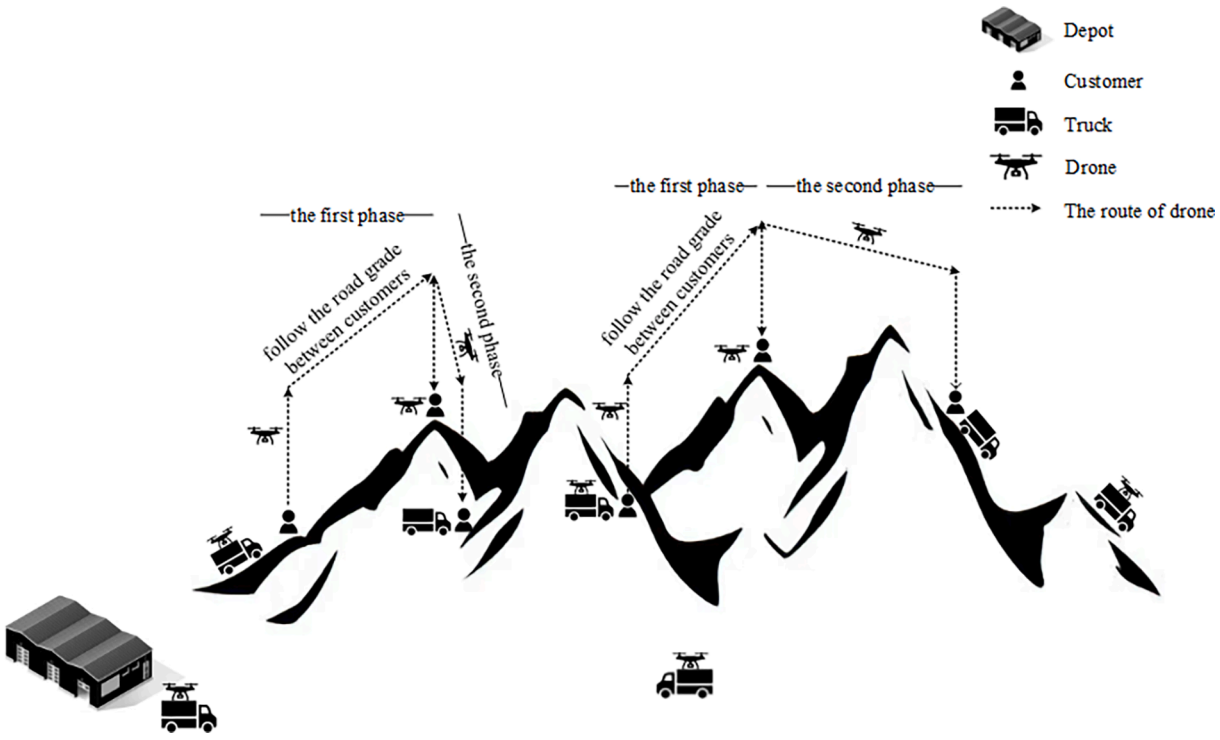

Fig. 2. Flight patterns of a drone.

and km/h, respectively).

The energy consumption model proposed by Kirschstein & Meisel (Kirschstein and Meisel, 2015) is adopted, shown in Eq.(2),

$$F(m_t, d, \nu_t, \alpha) = \frac{d}{\nu_t}\left(f^{idle} + \frac{f^{full} - f^{idle}}{\varsigma \cdot P}\max\left(P^T(m_t, d, \nu_t, \alpha), 0\right)\right) \tag{2}$$

where d is the travel distance (km); f^{idle} and f^{full} denote the fuel consumption rates (L/h) in idle and full-throttle mode, respectively; P is the engine power (kW); The transmission efficiency function ς is set to $\varsigma = 0.9 - 0.72 \cdot e^{-0.077 \cdot \nu_t^{1.41}}$. Eq. (3) is used to convert the energy consumption of trucks to a similar order of magnitude as that of drones as follows,

$$E(m_t, d, \nu_t, \alpha) = F(m_t, d, \nu_t, \alpha) \cdot NHV^{Diesel} \tag{3}$$

where NHV^{Diesel} is the net heating value (kWh/L).

Accordingly, the energy consumption of truck v for arc (i, j) can be calculated as Eq. (4),

$$ct_{ij}^{v} = E\left(m_t + w_{ij}^{v}, d_{ij}, \nu_t, \alpha_{ij}\right) \tag{4}$$

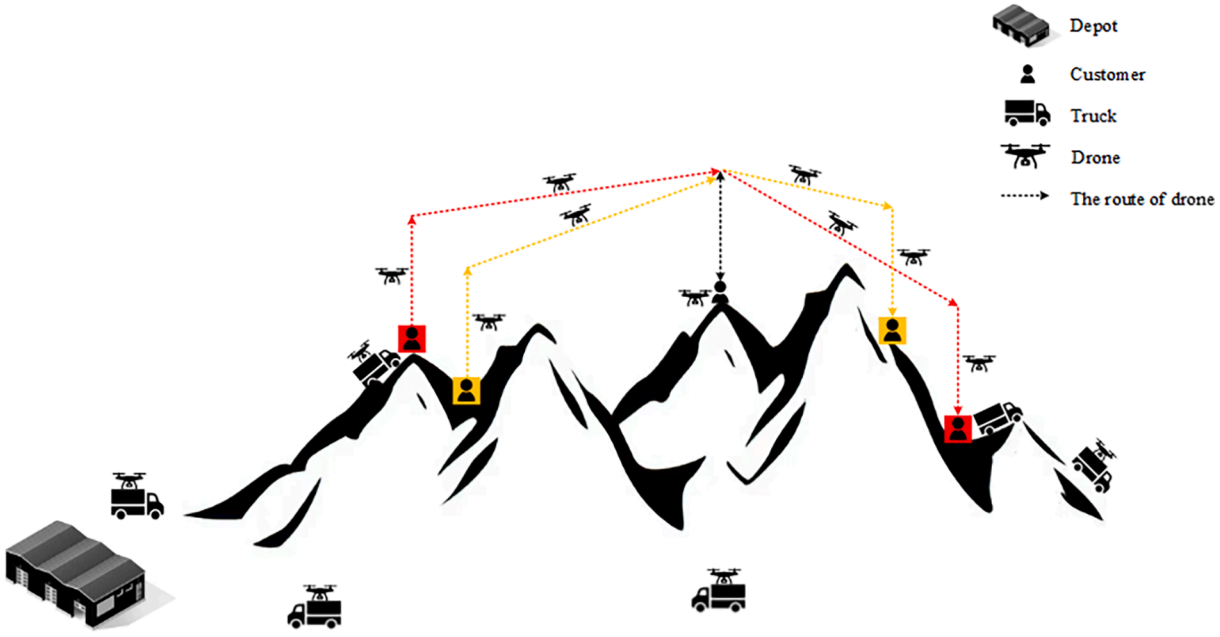

Fig. 3. Drone trips affected by steep roads.

3.2.2. Energy consumption model for drones

For a drone, the model presented in Langelaan et al. (Langelaan et al., 2017) is adopted to figure out the energy consumption of its delivery process. The power required to overcome the air drag of body and rotors, lift, climb, and supply internal auxiliaries is computed by Eq. (5),

$$P^D = P^{body} + P^{rotor} + \kappa \cdot P^{lift} + P^{\text{climb}} + P^{\text{int}} = \left(\frac{1}{2}\cdot a \cdot v_d^3 \cdot A_d \cdot c_{air} + \rho \cdot R \cdot V_d^3 \left(1 + 2\cdot\left(\frac{v_d}{V_d}\right)^2\right)\frac{\sigma \cdot c_{bd}}{8} + K \cdot \omega \cdot T + m_d \cdot g \cdot v_d \cdot \sin\alpha\right) \Big/ 1000 + P^{\text{int}} \tag{5}$$

where a and g are the air density (kg/m^3) and gravitational constant (m/s^2), respectively; A_d denotes the drone frontal surface area (m^2), v_d is the flight speed (m/s), V_d is the blade tip speed (m/s), R represents the rotor disc area, σ is the rotor solidity ratio, c_{bd} is the blade drag coefficient, K is an up-scaling factor, ω is the downwash coefficient, thrust $T = \sqrt{m_d^2 \cdot g^2 + D_{doby}^2 + 2 \cdot D_{body} \cdot m_d \cdot g \cdot \sin\alpha}$ (with $D_{body} = \frac{1}{2} \cdot a \cdot v_d^2 \cdot A_d \cdot c_{air}$), m_d is drone mass (kg), and α is the road grade. See Appendix A for more information on determining V_d, R, σ, c_{bd} and ω.

Fig. 2 depicts an illustration of the flying patterns of a drone. It is assumed that each flight begins with the drone flying vertically at an altitude of 150 m (i.e., $h = 0.15$). Then, the drone ascents, flies parallel to, or descents to the above of its dedicated truck or the serving customer at the same angle as road grade α, followed by a vertical landing. The energy demand of a drone for path $< i, j, k >$ with load q_j can be formally defined as

$$cd_{ijk}(q_j) = \frac{1}{\eta_{eng} \cdot \eta_{trans} \cdot \eta_{char}} \left\{ \begin{array}{l} \left|P^D(m_d + q_j, v_d, \pm 90^\circ)\right| \cdot \frac{h}{v_d} + \left|P^D(m_d + q_j, v_d, \alpha_{ij})\right| \cdot \frac{d_{ij}}{v_d} \\ + \left|P^D(m_d, v_d, \pm 90^\circ)\right| \cdot \frac{h}{v_d} + \left|P^D(m_d, v_d, \alpha_{jk})\right| \cdot \frac{d_{jk}}{v_d} \end{array} \right\} + \frac{1}{\eta_{char}}\left(P^{\text{int}} \cdot t\right), \tag{6}$$

where ν_d denotes flight speed (km/h); t is total time for path $< i, j, k >$, which is calculated by $t = \frac{4 \cdot h + d_{ij} + d_{jk}}{\nu_d}$ (h); η_{eng}, η_{trans}, and η_{char} are engine efficiency, transmission efficiency, and charging efficiency, respectively.

3.2.3. Discussions

The intuitive difference of the GVRPD-SR from existing VRPD models is to minimize the energy consumption of drones and trucks instead of operating costs or delivery makespan. More importantly, the energy consumption calculation of trucks and drones defined above comprehensively depends on the distance, load, speed, and steep road of traversing arcs(i, j). As a result, the GVRPD-SR is essentially represented as an asymmetric graph, whereas the VRPD is a symmetric one. In this sense, the route planning for trucks determined by the VRPD models may consume more energy. Regarding each drone, the route planning will be more sensitive to steep roads due to their limited battery endurance, leading to an increasing amount of computation time. As shown in Fig. 3, in the VRPD model, yellow customer nodes are determined to be the best choice for the drone to rendezvous and be launched on one trip. However, if GVRPD-SR follows the same delivery sequence as VRPD, drones will fly at a large angle uphill and a small angle downhill, consuming increased energy and perhaps even running out of battery on the way back. Consequently, the optimal rendezvous and launch nodes for the drone switch to the red customer nodes. In summary, the introduction of energy consumption in the GVRPD-SR model not only changes the objective but also makes the decision-making process more complicated.

3.3. Mathematical formulation

According to the above definitions and energy consumption model, a mathematical model for GVRPD-SR is formulated as follows.

$$\min \sum_{v \in V}\left(\sum_{i \in N_d}\sum_{j \in N_c}\sum_{k \in N_a} cd_{ijk} \cdot y_{ijk}^v + \sum_{i \in N_d}\sum_{j \in N_a} ct_{ij}^v \cdot x_{ij}^v\right) \tag{7}$$

Subject to:

$$\sum_{j \in N_a} x_{0j}^v \le 1 \forall v \in V \tag{8}$$

$$x_{0,n+1}^v = 0 \forall v \in V \tag{9}$$

$$\sum_{j \in N_a} x_{0j}^v = \sum_{i \in N_d} x_{i,n+1}^v \forall v \in V \tag{10}$$

$$\sum_{i \in N_d} x_{ij}^v = \sum_{k \in N_a} x_{jk}^v \forall v \in V, j \in N_c \tag{11}$$

$$\sum_{j \in N_c}\sum_{k \in N_a} y_{ijk}^v \le 1 \forall v \in V, i \in N_d \tag{12}$$

$$\sum_{i \in N_d}\sum_{j \in N_c} y_{ijk}^v \le 1 \forall v \in V, k \in N_a \tag{13}$$

$$\sum_{v \in V}\sum_{i \in N_d} x_{ij}^v + \sum_{v \in V}\sum_{i \in N_d}\sum_{k \in N_a} y_{ijk}^v = 1 \forall j \in N_c \tag{14}$$

$$2y_{ijk}^v \le \sum_{h \in N_d} x_{hi}^v + \sum_{f \in N_c} x_{fk}^v \forall v \in V, i \in N_c, j \in N_c, k \in N_a \tag{15}$$

$$y_{0jk}^v \le \sum_{i \in N_d} x_{ik}^v \forall v \in V, j \in N_c, k \in N_c \tag{16}$$

$$cd_{ijk} \le cap_b + M \cdot \left(1 - y_{ijk}^v\right) \forall v \in V, i \in N_d, j \in N_c, k \in N_a \tag{17}$$

$$\sum_{j \in N_c}\sum_{k \in N_a} y_{ijk}^v \cdot q_j \le Q_d \forall v \in V, i \in N_d \tag{18}$$

$$w_{0j}^v \geq -Q_t\cdot\left(1-x_{0j}^v\right)+\sum_{h\in N_a}\left(\sum_{i\in N_d}x_{ih}^v\cdot q_h\right)+\sum_{i\in N_c}\left(\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\cdot q_j\right)\forall v\in V, j \in N_c \tag{19}$$

$$w_{0j}^v \leq Q_t\cdot\left(1-x_{0j}^v\right)+\sum_{h\in N_a}\left(\sum_{i\in N_d}x_{ih}^v\cdot q_h\right)+\sum_{i\in N_c}\left(\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\cdot q_j\right)\forall v\in V, j\in N_c \tag{20}$$

$$w_{ij}^v \geq -Q_t\cdot\left(2-x_{fi}^v-x_{ij}^v\right)+\left(w_{fi}^v-q_i-\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\cdot q_j\right)\forall v\in V, i\in N_c, j \in N_c, f\in N_d \tag{21}$$

$$w_{ij}^v \leq Q_t\cdot\left(2-x_{fi}^v-x_{ij}^v\right)+\left(w_{fi}^v-q_i-\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\cdot q_j\right)\forall v\in V, i\in N_c, j\in N_c, f \in N_d \tag{22}$$

$$w_{i,n+1}^v = 0\forall v\in V, i\in N_d \tag{23}$$

$$\tau_{v0}^t = 0\forall v\in V \tag{24}$$

$$\tau_{v0}^d = 0\forall v\in V \tag{25}$$

$$\tau_{vh}^t+\Pi_{hk}^t \leq \tau_{vk}^t+M\cdot\left(1-x_{hk}^v\right)\forall v\in V, h\in N_d, k\in N_a \tag{26}$$

$$\tau_{vh}^t+\Pi_{hk}^t \geq \tau_{vk}^t-M\cdot\left(1-x_{hk}^v\right)\forall v\in V, h\in N_d, k\in N_a \tag{27}$$

$$\tau_{vi}^d+\Pi_{ij}^d+2\cdot\Pi_h^d \leq \tau_{vj}^d+M\cdot\left(1-\sum_{k\in N_a}y_{ijk}^v\right)\forall v\in V, i\in N_d, j\in N_c \tag{28}$$

$$\tau_{vi}^d+\Pi_{ij}^d+2\cdot\Pi_h^d \geq \tau_{vj}^d-M\cdot\left(1-\sum_{k\in N_a}y_{ijk}^v\right)\forall v\in V, i\in N_d, j\in N_c \tag{29}$$

$$\tau_{vj}^d+\Pi_{jk}^d+2\cdot\Pi_h^d \leq \tau_{vk}^d+M\cdot\left(1-\sum_{i\in N_d}y_{ijk}^v\right)\forall v\in V, j\in N_c, k\in N_a \tag{30}$$

$$\tau_{vj}^d+\Pi_{jk}^d+2\cdot\Pi_h^d \geq \tau_{vk}^d-M\cdot\left(1-\sum_{i\in N_d}y_{ijk}^v\right)\forall v\in V, j\in N_c, k\in N_a \tag{31}$$

$$\tau_{vi}^t-M\cdot\left(1-\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\right)-M\cdot\sum_{f\in N_d}\sum_{h\in N_c}y_{fhi}^v \leq \tau_{vi}^d\forall v\in V, i\in N_c \tag{32}$$

$$\tau_{vi}^t+M\cdot\left(1-\sum_{j\in N_c}\sum_{k\in N_a}y_{ijk}^v\right)+M\cdot\sum_{f\in N_d}\sum_{h\in N_c}y_{fhi}^v \geq \tau_{vi}^d\forall v\in V, i\in N_c \tag{33}$$

$$\tau_{vk}^t-\tau_{vi}^t-M\cdot\left(1-\sum_{j\in N_c}y_{ijk}^v\right) \leq \tau_{vk}^d-\tau_{vi}^d\forall v\in V, i\in N_d, k\in N_a \tag{34}$$

$$\tau_{vf}^d \geq \tau_{vk}^d-M\cdot\left(3-\sum_{j\in N_c}y_{ijk}^v-\sum_{o\in N_c}\sum_{n\in N_a}y_{fon}^v-\rho_{if}^v\right)\forall v\in V, i\in N_d, k\in N_c, f\in N_c \tag{35}$$

$$\mu_j^v-\mu_i^v \geq 1-(n+2)\cdot\left(1-x_{ij}^v\right)\forall v\in V, i\in N_d, j\in N_a \tag{36}$$

$$\mu_k^v-\mu_i^v \geq 1-(n+2)\cdot\left(1-\sum_{j\in N_c}y_{ijk}^v\right)\forall v\in V, i\in N_d, k\in N_a \tag{37}$$

$$\mu_j^v-\mu_i^v \geq 1-(n+2)\cdot\left(1-\rho_{ij}^v\right)\forall v\in V, i\in N_c, j\in N_c \tag{38}$$

$$\mu_j^v-\mu_i^v \leq -1+(n+2)\cdot\rho_{ij}^v\forall v\in V, i\in N_c, j\in N_c \tag{39}$$

$$\rho_{oj}^v = \sum_{i\in N_d}x_{ij}^v\forall v\in V, j\in N_a \tag{40}$$

$$0 \leq w_{ij}^v \leq Q_t\forall v\in V, i\in N_d, j\in N_a \tag{41}$$

$$\tau_{vi}^t \geq 0\forall v\in V, i\in N \tag{42}$$

$$\tau_{vi}^d \geq 0\forall v\in V, i\in N \tag{43}$$

$$1 \leq \mu_i^v \leq n+2, \mu_i^v\in\mathbb{Z}^+\forall v\in V, i\in N \tag{44}$$

$$x_{ij}^v\in\{0,1\}\forall v\in V, i\in N_d, j\in N_a \tag{45}$$

$$y_{ijk}^v\in\{0,1\}\forall v\in V, i\in N_d, j\in N_c, k\in N_a \tag{46}$$

$$\rho_{ij}^v\in\{0,1\}\forall v\in V, i\in N, j\in N \tag{47}$$

The objective function (7) minimizes the total energy consumption of serving all customers, including the energy consumption of trucks and drones. Constraint (8) restricts all trucks to leave the depot at most once. Constraint (9) indicates that all trucks are prohibited from traveling between depots. Trucks departing the depot must return to the depot, as given by Constraint (10). The flow conservation constraints for the truck are defined in Constraint (11). A drone can be launched and rendezvoused at most once at each node, as shown in constraints (12)-(13). Constraint (14) guarantees that each customer must be served once. Constraint (15) makes sure that if the drone on truck v is launched at node i and rendezvous with it at node j, then i and j must be visited by truck v. Constraint (16) is a special case of Constraint (15) when the drone is launched from the depot. Constraint (17) states that each trip of the drone cannot exceed its endurance power. The capacity constraint for a drone is given in Constraint (18). Constraints (19)-(23) calculate the load of truck v for each arc (i, j). Constraints (24)-(25) initialize the times for the truck and drone at the beginning of each route. Constraints (26)-(27) are used to calculate the arrival time of trucks, and Constraints (28)-(31) are used for drones. It is worth noting that if successively perform rendezvous operation for the drone's previous trip and launch operation for the drone's new trip at customer node i, the launch time of the drone can be calculated by Constraints (30)-(31). It is independent of the arrival time of the dedicated truck, as the truck is residing at customer node i. Otherwise, when only the launch operation for a drone trip is performed at the customer node, the launch time of the drone needs to be synchronized with its truck. The time synchronization between the truck and its dedicated drone is defined in Constraints (32)-(33). Constraint (34) imposes that the drone cannot arrive before its dedicated truck. Constraint (35) prevents the drone from launching while it is flying. Specifically, the drone is allowed to be newly launched only if the launch time of the drone at another customer node f is later than its previous rendezvous time at customer node k. This is considered by the order in which the truck visits nodes i and f (i.e., $\rho_{if}^v = 1$), if the drone is launched from node i and rendezvoused at node k. Constraints (36)-(37) work as subtour eliminations for trucks and drones, respectively. Constraints (38)-(40) determine the value of ρ_{ij}^v to ensure that the truck does not visit the same route in a different direction, which affect Constraint (35) that needs the customer visit sequence. Constraints (41)-(47) define the values of decision variables.

4. A metaheuristic algorithm for the GVRPD-SR

This section introduces the details of the IALNS algorithm. The standard ALNS framework was first designed by Ropke and Pisinger

(Ropke and Pisinger, 2006), a variant of Large Neighbor Search. It can iteratively improve the given initial solution through several removal and insertion operators as well as a simulated annealing (SA) acceptance function. One removal and one insertion operator performed at each iteration are statistically selected from all operators based on their previous search performance. The pseudo-code for the IALNS algorithm is shown in Algorithm 1.

Algorithm 1: Pseudo-Code for the IALNS algorithm

Input: Initial temperature: T_{start};
Maximum iterations without improvement: $noImpvMax$;
The number of iterations in a segment: $segMax$;
The termination condition: Gen_{max};

Output: The best obtained solution: S^*;

1 $S \leftarrow Initial\ solution(\cdot)$;
2 $S^* \leftarrow S$;
3 $noImpv\ \&\ seg \leftarrow 0$;
4 **while** $gen < Gen_{max}$ **do**
5 Select a removal operator r and an insertion operator $\bar{r}$ based on a roulette-wheel mechanism;
6 $S' \leftarrow \bar{r}(r(S))$;
7 $T = T_{start}(1 - gen/Gen_{max})$;
8 **if** $rand(0,1) < \exp\left(\left(f(S) - f(S')\right)/T\right)$ **then**
9 $S \leftarrow S'$;
10 **if** $f(S) < f(S^*)$ **then**
11 $S^* \leftarrow S$;
12 $noImpv \leftarrow 0$;
13 **else**
14 $noImpv \leftarrow noImpv + 1$;
15 **if** $noImpv > noImpvMax$ **then**
16 $S \leftarrow S^*$;
17 $noImpv \leftarrow 0$;
18 Update scores of r and $\bar{r}$;
19 $seg \leftarrow seg + 1$;
20 **if** $seg = segMax$ **then**
21 Update weights of all operators;
22 $seg \leftarrow 0$;
23 $gen \leftarrow gen + 1$
24 **return** S^*;

More specifically, an initial solution is first constructed and regarded as the current solution S, followed by S being stored as the best solution S^*. Meanwhile, iterations are bundled into segments (lines 1–3). Regarding S, a removal operator and an insertion operator are selected using a roulette-wheel mechanism, and they are then performed to generate a new solution S' (lines 5–6). $f(S)$ represents the objective value of S. Following the updating of a parameter T, the acceptance of S' is determined by a probability function based on the idea of SA to prevent getting trapped prematurely in a local optimum (lines 7–9). Then, the optimal solution S^* and its related record parameter *noImpv* are both updated (lines 10–14). If a predefined number of consecutive iterations has passed without any improvement, S^* is assigned to S (lines 15–17). Using the adaptive weight adjustment mechanism proposed by Ropke and Pisinger (Ropke and Pisinger, 2006), the scores of the performed removal operator and insertion operator are updated based on the quality of S', and a parameter *seg* is also updated (lines 18–19). Note that the weights of each operator used for the roulette-wheel mechanism should be updated at the end of each segment (lines 20–22). The aforementioned steps are repeated until a specified maximum iteration Gen_{max} is reached, after which the optimal solution obtained is output.

Compared to the standard ALNS framework, the proposed IALNS algorithm is tailored and improved based on the characteristics of the GVRPD-SR in the following aspects: 1) As the use of drones to assist with delivery tasks is one of the prominent features of the GVRPD-SR, a heuristic initialization strategy that first determines primary truck routes followed by iteratively adjusting to obtain drone sub-routes is designed. 2) The presence of steep roads is another prominent feature of the GVRPD-SR. Therefore, on the one hand, several classic and powerful operators of the standard ALNS algorithm are modified by incorporating road grades into operating criteria. On the other hand, a new removal operator and two insertion operators are proposed, with a dominant consideration of steep road impacts.

4.1. Initial solution

An initial solution is constructed with a heuristic strategy consisting of three phases. Beginning with a classification for all customers, those that can solely be served by trucks are merged into a set D_t. The remaining customers are formed into a set D_d, which can be served by not only trucks but also drones. Then, Clarke and Wright savings algorithm (Clarke and Wright, 1964) is applied to generate feasible routes for D_t. Specifically, each customer in D_t is independently connected with the depot to form a total of $|D_t|$ initial routes. Let $f(\cdot)$ denote the total energy consumption of a route, $\Delta f_{i,j}$ denote the saving in energy consumption incurred by merging a route (s_{i0}) ending with customer i and another route (s_{0j}) starting with customer j, which can be calculated by $\Delta f_{i,j} = f(s_{i0}) + f(s_{0j}) - f(s_{ij})$. The feasible merge operation with maximum $\Delta f_{i,j}$ is implemented at each iteration, and the procedure terminates when it is no longer possible to merge routes. Thereafter, the unvisited customers are inserted into the present routes one by one with a greedy heuristic, or a new truck route is created separately. At each iteration, a customer is randomly selected from D_d, and the increased energy consumption of all possible insertion positions is determined respecting the capacity constraints, including inserting it into the truck routes or constructing a new sub-route for the drone. After performing the insertion operation with a minimum increased value, this customer is removed from D_d.

4.2. Removal operators

The purpose of removal operators is to diversify the solution space exploration by randomly removing varying number of customers from the current solution. On the one hand, if the number of removed customers is too small, it is difficult for the algorithm to escape the local optimum. On the other hand, if too much part of the solution is destroyed, the subsequent insertion procedure will struggle to reconstruct a high-quality solution. Therefore, the number of customers χ to be removed is calculated by Eq. (48),

$$\chi = \min(\max(\psi_{\min}, \kappa \cdot |N_C|), \psi_{\max}) \tag{48}$$

where $\psi_{\min}$ and $\psi_{\max}$ are the lower and upper bounds for the number of customers to be removed, respectively; κ is responsible for ensuring that χ is appropriate for different scales of instances.

Five removal operators are presented below, four of which are tailored versions of the powerful operators suggested by Ropke and Pisinger (Ropke and Pisinger, 2006) based on the specifics of the problem setting, and the fifth of which is a new one. It is worth noting that drone sub-routes will become invalid when their launch or rendezvous nodes are removed. Furthermore, the launch and rendezvous of a drone can both take place at the same customer node. Thus, the removal of such special truck customers is accompanied by the removal of two extra drone customers. After the removal procedure is completed, there may be one or two removed customers more than the specified number χ.

4.2.1. Random removal

This operator iteratively removes random customers from the current solution until χ customers have been removed, which is beneficial to increasing the search diversity. Notice that it is required to determine whether any drone customers are also removed when removing the truck customer.

4.2.2. Worst removal

It attempts to remove truck customers with higher energy consumption. The removal cost for each customer is the difference between the total energy consumption of the route and the removal of the customer after. A truck customer with $\lfloor y^p |L| \rfloor^{th}$ highest cost is selected to remove, where y is a random number within [0, 1], and p is set to 6. This operator also needs to determine whether any drone customers are also

removed when removing the truck customer. The procedure is repeated until χ customers have been removed.

4.2.3. Shaw removal

It is designed to remove a set of customers that are similar to each other with respect to several criteria. For the customers i and j, the relatedness R_{ij}, comprehensively considering the aspects of distance, steep road, demand, and route, is calculated as follows,

$$R_{ij} = \phi_1 d_{ij} + \phi_2 |\sin\alpha_{ij}| + \phi_3 |q_i - q_j| + \phi_4 l_{ij} \tag{49}$$

where $l_{ij} = 0$ if i and j are on the same route, and 1 otherwise; ϕ_1, ϕ_2, ϕ_3 and ϕ_4 that add up to 1 denote the preference weights for these four criteria, respectively. Moreover, their values are normalized using the max–min method before calculating R_{ij}. A lower value of R_{ij} indicates a higher relatedness.

The procedure of this operator is described as follows: a customer c is first randomly selected and added to the removal list, and the relatedness between c and the remaining unremoved customers is calculated. In accordance with the non-decreasing order of the relatedness value, the selection strategy introduced by Section 4.2.2 is used to determine the next customer to be removed. It is necessary to determine whether any drone customers also need to be added to the removal list when adding the truck customer. This procedure is repeated until the number of removals reaches χ.

4.2.4. Cluster removal

In this operator, customer removal is performed in the area around a random focal customer while considering the steep roads between customers. First, a random customer is selected and removed from the current solution, which is regarded as the focal customer. Then, customers are removed one by one until χ customers have been removed. In each step, one of the two customers closest to the focal customer is randomly selected as the next customer to be removed. The closeness between i and j depends on $|\sin\alpha_{ij}|$. Furthermore, the cluster removal operator employs the same selection strategy as the worst removal operator. Notice that it is required to determine whether any drone customers are also removed when removing the truck customer.

4.2.5. Partial-route removal

Inspired by the cluster removal operator, a partial-route removal operator is designed in this study. It starts with a random route and then divides the customers on this route into two (roughly) equal groups. With the random selection of one group, customers within this group are removed based on the sequence of truck visits. A special case in which the latest removal group is empty may happen. It will trigger the random selection of an undestroyed truck route as the next route to be destroyed. The aforementioned steps of equal grouping, random selection, and sequential removal steps are carried out again. Regarding normal cases, a truck customer is first chosen from the latest removed group at random. Subsequently, one of the two truck customers closest to the customer is selected with equal probability, excluding the truck customers whose routes have been destroyed. The route visiting the selected customer is the next route to be destroyed. Distinguished from the steps of the special case, the group to be removed is determined by the distances between each of the two newly separated groups and the latest removal group, with the smaller one being selected. The distance is measured by the minimum distance between two groups of customers. The above process is repeated until χ customers have been removed. Notice that it is required to determine whether any drone customers are also removed when removing the truck customer. Algorithm 2 outlines the pseudo-code for this operator.

Algorithm 2: Partial-route removal for customers

Input: Current solution: S;
The number of customers to be removed: χ;
Output: The destroyed solution: $\tilde{S}$;

```
r ← RandomRoute(S);
r_1 ← Equally divides r into two groups and select one randomly;
S̃ ← Customers within r_1 are sequentially removed from S based on the order of truck visits;
while removed < χ do
    if r_1 = ∅ then
        r ← Select rondomly one undestroyed truck route;
        r_1 ← Equally divides r into two groups and select one randomly;
    else
        c ← RandomCustomer(r_1);
        c_1 ← Select randomly one of the two truck customers with undestroyed routes closest to c;
        r ← Route(c_1);
        r_2, r_3 ← Equally divides r into two groups;
        r_1 ← MinDistance((r_2, r_3), r_1);
    for c in r_1 do
        c is removed from S;
        if removed > χ then
            break
    S̃ ← S
return S̃;
```

4.3. Insertion operators

Six insertion operators consisting of four tailored operators and two new operators are presented, with the goal of reinserting all removed customers (denoted as set Ω) into the current partial solution to obtain a new feasible solution. Specially, Ω_d is defined as a subset of Ω consisting of truck-only customers (no launch and rendezvous) whose demands are within the capacity of a drone.

4.3.1. Basic greedy insertion

This insertion operator consists of the following two phases: 1) reinserts all removed customers into the existing partial solution as truck visits; 2) attempts to service some customers with drones instead of trucks.

In the first phase, the customers in Ω are randomly selected one by one and then inserted into the partial solution at the best position, based on the principle of least energy consumption. The second phase begins with random selection of a customer c from Ω_d. The changes in energy consumption of all feasible sub-routes using a drone to serve c are calculated, choosing the best sub-route with the maximum positive reduction. The procedure for the second phase is terminated if all customers in Ω_d have been attempted.

4.3.2. Deep greedy insertion

Distinguished from the preceding operator, a customer to be performed insertion is chosen greedily as opposed to randomly in each iteration. In the first phase, the customers in Ω are sorted with ascending order according to their minimum insertion energy consumption. The corresponding optimal insertion position for the first customer is performed, followed by removing this customer from Ω. Regarding the second phase, the operations described in Section 4.2.2 are utilized to select a customer from Ω_d. Subsequently, the selected customer is served by a drone using the same method as presented in Section 4.3.1. The procedure for the second phase is terminated if all customers in Ω_d have been attempted.

4.3.3. Regret insertion

Regret-k operator is developed to avoid the myopic nature of the greedy insertion by incorporating look-ahead information when choosing the customer to be inserted. Specifically, let Δf_i^m denote the change in energy consumption by inserting customer i into its m^{th} greenest route at the best position. In each iteration, the customer i^* to be inserted is determined by $\underset{i\in\Omega}{\operatorname{argmax}} \sum_{g=2}^{k} \Delta f_{ig} - \Delta f_{i1}$. In this study, a regret-4 operator is used as the truck-insertion operator for customers in Ω, i.e., only considering truck services, followed by performing the

Table 3
Parameters of the IALNS and ALNS.

Parameter	Description	IALNS	ALNS
noImpvMax	Non-improvement limit	50	50
$Gen_{\max}$	Iteration limit	1000	1000
T_0	Initial temperature factor	0.5056/0.004	0.0044/0.0004
κ	Degree of destruction	0.15	0.15
γ	Reaction factor	0.90	0.90
$\tau_{\min}/\tau_{\max}$	Lower / Upper bound of removable nodes	(1, 3) / 40	(1, 3) / 40
$\beta_1, \beta_2, \beta_3$	Weight adjustment parameters	{33, 9, 13}	{33, 9, 13}
$\phi_1, \phi_2, \phi_3, \phi_4$	Shaw removal parameters	{0.45, 0.3, 0.15, 0.1}	

second phase of the basic greedy insertion in Section 4.3.1. Notice that if the number of routes that some customers can be inserted into is less than four, then they can be inserted into the route with the fewest customers.

4.3.4. Closest insertion

In this operator, a customer i is randomly selected from Ω, and then the truck route visiting its closest customer is considered as the target insertion route. All feasible insertion positions for customer i using the truck or drone service are determined, and the optimal one is performed. If customer i has any feasible insertion position on the chosen route, it will be still retained in Ω. After all customers in Ω have been attempted, the first phase of the basic greedy insertion in Section 4.3.1 is adopted to deal with the remaining customers.

4.3.5. Nearby-I insertion

The nearby-I insertion operator is designed as one novel special variant of the operator described in Section 4.3.4. to enhance diversity. The differences are rendered on the first phase. On the one hand, the target insertion route is extended to multiple truck routes that visit the first 30 % of customers closest to the selected customer, based on the ascending sorting of distance. On the other hand, the calculation of insertion energy consumption is perturbed by a factor $\varepsilon \in [0.8, 1.2]$.

4.3.6. Nearby-II insertion

Likewise, this new operator is also extended from the operator described in Section 4.3.4. Distinguished from the nearby-I insertion operator, it takes into consideration not only distance but also steepness difference between two nodes. Specifically, the target insertion routes are filtrated through two steps. Firstly, the first half of customers closest to the selected customer are retained based on the ascending sorting of distance. Following that, these customers are re-sorted in ascending order by the absolute value of the steepness difference, and the routes that visit the first 70 % of customers are regarded as the target insertion routes.

5. Computational experiments

All experiments are coded in MATLAB, and the run environment is a PC with an Intel (R) Core 2.80 GHz processor, 24-GB RAM, and a 64-bits Windows 10 operating system.

5.1. Benchmark instances

Since the GVRPD-SR is a new variant of the VRPD, there is no dataset available yet in the literature. Hence, a new benchmark is generated based on that of Sacramento et al. (Sacramento et al., 2019). Appendix B describes the principle for generating instances. Each instance is named "*n. d. m. h*", where n is the number of customers, d represents the dimension of the grid, m is the generic name of the scenarios with different coordinates and demands, and h refers to the generic name of the altitude scenarios.

5.2. Parameters settings

To ensure a fair comparison, the main parameters of the IALNS algorithm are closely followed to the optimal parameter settings for the benchmark algorithm ALNS (Sacramento et al., 2019). The termination condition of both is the maximum number of iterations, whose value is determined by extensive trial-run experiments. Additionally, to prevent a too low temperature for small instances, the initial temperature for these instances is adjusted such that a solution that is 35 % worse than the current solution has a 50 % chance of being accepted. Table 3 summarizes the parameter settings for both algorithms. See Appendix C for the parameter values of the GVRPD-SR model.

5.3. Performance of the IALNS algorithm

5.3.1. Analysis of operators

To discuss the computational performance of each operator, the analysis method of Chen et al. (2021) is adopted on the results obtained from all instances. Let %IBest be the percentage of iterations where a new feasible solution is superior to the current best solution, whereas %

Table 4
Relative performance of each operator (%IBest / %Usage).

Operators	Number of customers						Avg.
	6	10	12	50	100	150	
Removal							
Random	0.0041	0.0079	0.0094	0.0390	0.0441	0.0425	0.0245
Worst	0.0044	0.0098	0.0108	0.0379	0.0534	0.0537	0.0283
Cluster	0.0093	0.0167	0.0151	0.0908	0.1058	0.1219	0.0599
Shaw	0.0087	0.0173	0.0224	0.0649	0.1028	0.1060	0.0537
Partial-route	0.0039	0.0069	0.0080	0.0415	0.0508	0.0611	0.0287
Insertion							
Basic greedy	0.0103	0.0146	0.0127	0.0409	0.0556	0.0637	0.0329
Deep greedy	0.0087	0.0128	0.0128	0.0344	0.0449	0.0544	0.0280
Regret-4	0.0041	0.0080	0.0092	0.0308	0.0238	0.0215	0.0162
Closest	0.0056	0.0159	0.0181	0.0983	0.1045	0.0953	0.0563
Nearby-I	0.0058	0.0080	0.0074	0.0176	0.0293	0.0286	0.0161
Nearby-II	0.0015	0.0145	0.0239	0.0847	0.1220	0.1317	0.0630

Table 5
Experiment results of CPLEX and IALNS for small-scale instances.

Instances	CPLEX		IALNS			Gap	
	z_{best}^{CPLEX}	t^{CPLEX}	z_{best}^{IALNS}	z_{avg}^{IALNS}	t^{IALNS}	Δz_{best}	Δz_{avg}
6.10.1.a	6.4452	**3.69**	6.4452	6.4672	3.71	**0.00 %**	0.34 %
6.10.1.b	4.1907	**2.05**	4.1907	4.2098	2.45	**0.00 %**	0.46 %
6.10.1.c	7.9463	**0.89**	7.9463	7.9463	2.80	**0.00 %**	**0.00 %**
6.10.2.a	8.6190	**3.81**	8.6190	8.6190	5.49	**0.00 %**	**0.00 %**
6.10.2.b	6.7436	**1.78**	6.7436	6.7436	1.91	**0.00 %**	**0.00 %**
6.10.2.c	10.5351	**0.84**	10.5351	10.5351	1.47	**0.00 %**	**0.00 %**
6.20.1.a	9.5469	5.7	9.5469	9.5469	**2.50**	**0.00 %**	**0.00 %**
6.20.1.b	10.2856	**1.61**	10.2856	10.2856	1.72	**0.00 %**	**0.00 %**
6.20.1.c	8.4072	4.22	8.4072	8.4072	**1.88**	**0.00 %**	**0.00 %**
6.20.2.a	8.6407	2.66	8.6407	8.6407	**2.38**	**0.00 %**	**0.00 %**
6.20.2.b	9.0723	**1.78**	9.0723	9.0723	3.19	**0.00 %**	**0.00 %**
6.20.2.c	9.5480	**4.88**	9.5480	9.5818	17.22	**0.00 %**	0.35 %
10.10.1.a	10.7392	25.52	10.7392	10.7392	**4.72**	**0.00 %**	**0.00 %**
10.10.1.b	10.0960	439.25	10.0960	10.1620	**3.62**	**0.00 %**	0.65 %
10.10.1.c	7.6807	72.3	7.6807	7.7567	**11.90**	**0.00 %**	0.99 %
10.10.2.a	9.6332	2334.92	9.6332	9.6746	**36.13**	**0.00 %**	0.43 %
10.10.2.b	8.7486	36.83	8.7486	8.7486	**6.22**	**0.00 %**	**0.00 %**
10.10.2.c	12.3669	1060.09	12.3669	12.3669	**11.09**	**0.00 %**	**0.00 %**
10.20.1.a	10.9534	2522.13	10.9534	10.9534	**6.61**	**0.00 %**	**0.00 %**
10.20.1.b	12.4877	184.89	12.4877	12.5668	**14.51**	**0.00 %**	0.63 %
10.20.1.c	11.5737	316.39	11.5737	11.5943	**6.20**	**0.00 %**	0.18 %
10.20.2.a	14.3125	343.76	14.3125	14.4153	**13.35**	**0.00 %**	0.72 %
10.20.2.b	12.5798	1322.66	12.5798	12.6951	**5.98**	**0.00 %**	0.92 %
10.20.2.c	12.7199	2288.78	12.7199	12.7841	**18.71**	**0.00 %**	0.50 %
12.10.1.a	10.2212	815.02	10.2212	10.3304	**8.99**	**0.00 %**	1.07 %
12.10.1.b	13.4241	3606.88	13.4241	13.5367	**13.42**	**0.00 %**	0.84 %
12.10.1.c	15.3611	479.33	15.3611	15.3799	**37.50**	**0.00 %**	0.12 %
12.10.2.a	10.9683	609.06	10.9683	10.9683	**15.71**	**0.00 %**	**0.00 %**
12.10.2.b	12.4110	7200.00	11.7504	11.7717	**6.22**	**−5.32 %**	**−5.15 %**
12.10.2.c	12.0164	1284.53	12.0164	12.0218	**7.54**	**0.00 %**	0.04 %
12.20.1.a	19.4571	697.99	19.4571	19.5241	**9.15**	**0.00 %**	0.34 %
12.20.1.b	14.3305	649.55	14.3305	14.3305	**17.52**	**0.00 %**	**0.00 %**
12.20.1.c	18.4419	111.53	18.4419	18.4419	**4.18**	**0.00 %**	**0.00 %**
12.20.2.a	14.7331	261.31	14.7331	14.7331	**9.86**	**0.00 %**	**0.00 %**
12.20.2.b	16.7330	905.75	16.7330	16.9612	**26.57**	**0.00 %**	1.36 %
12.20.2.c	16.9098	180.94	16.9098	16.9098	**3.96**	**0.00 %**	**0.00 %**

Note: Bold types denote significantly better results.

Usage represents the percentage of total iterations where a removal / insertion operator is used. A metric %IBest / %Usage can be used to measure which operators are likely to get new best solutions.

The experimental results are displayed in Table 4. For removal operators, the cluster operator and shaw operator, as well as the designed partial-route operator, are the ones that are conducive to finding the new best solutions. In contrast, the other two operators are more responsible for solution diversity. With respect to insertion operators, the nearby-II insertion contributes the most to generating the new best solutions, while the nearby-I insertion and regret-4 insertion contribute more to solution diversity. To this end, the three operators designed based on features of the GVRPD-SR perform admirably and are effective components of the IALNS algorithm.

5.3.2. Experiments with small-scale instances

In this section, the results of IALNS algorithm are compared with those of the commercial software CPLEX to evaluate its solution quality as well as algorithmic efficiency. Considering the computational complexity of the GVRPD-SR, only 36 small-scale instances are tested to ensure that CPLEX can run the mathematical model to optimality within an acceptable time. The maximum execution time for CPLEX is set at 7200 s.

The obtained results are detailed in Table 5. The second and third columns provide information about the optimal solutions (z_{best}^{CPLEX}) obtained from CPLEX and its running time (t^{CPLEX}). Similarly, the next three columns show the results of IALNS algorithm, including the best value (z_{best}^{IALNS}), the average value (z_{avg}^{IALNS}), together with the required running time (t^{IALNS}). The final two columns present the gap between the best (Δz_{best}) and average (Δz_{avg}) values of IALNS algorithm and the optimal solution obtained from CPLEX, which can be computed as in Eqs. (50)-(51).

$$\Delta z_{best} = \left(\frac{z_{best}^{IALNS} - z_{best}^{CPLEX}}{z_{best}^{CPLEX}} \right) \times 100\% \tag{50}$$

$$\Delta z_{avg} = \left(\frac{z_{avg}^{IALNS} - z_{best}^{CPLEX}}{z_{best}^{CPLEX}} \right) \times 100\% \tag{51}$$

From Table 5, it can be observed that IALNS algorithm can always find optimal solutions of equivalent quality to CPLEX and even a superior solution in 12.10.2.b. Moreover, the average value of IALNS algorithm is consistent with the optimal solution of CPLEX in 18 instances, better in 12.10.2.b, and only slightly inferior in the remaining instances. These demonstrate that IALNS algorithm not only can solve all small-scale GVRPD-SR problems to optimality but also has desirable stability.

Regarding the running time, CPLEX shows a slight advantage just in instances with 6 customers. With the increasing number of customers, CPLEX becomes more time-consuming to obtain the optimal solutions, whereas IALNS algorithm only spends a small amount of running time. More specifically, IALNS algorithm obtains optimal solutions for all instances in a couple of seconds to dozens of seconds. On the contrary, the running time of CPLEX to solve certain instances is up to hours. Therefore, IALNS algorithm has excellent solution efficiency in solving the small-scale GVRPD-SR problem.

5.3.3. Experiments with large-scale instances

This section aims to evaluate the effectiveness of the IALNS algorithm in large-scale instances by comparing it with the powerful ALNS.

Table 6
Experiment results of IALNS and ALNS for larger-size instances.

Instances	IALNS			ALNS			Gap	
	z_{best}^{IALNS}	z_{avg}^{IALNS}	std^{IALNS}	z_{best}^{ALNS}	z_{avg}^{ALNS}	std^{ALNS}	Δz_{best}	Δz_{avg}
50.30.1.a	51.84	55.80	2.74	52.95	56.41	**1.41**	**−2.09 %**	**−1.07 %**
50.30.1.b	54.90	58.01	2.07	56.40	58.75	**1.30**	**−2.65 %**	**−1.26 %**
50.30.1.c	56.81	60.28	2.17	58.22	61.36	**1.62**	**−2.42 %**	**−1.77 %**
50.30.2.a	52.77	56.42	2.16	52.84	57.00	**2.01**	**−0.13 %**	**−1.01 %**
50.30.2.b	52.77	54.36	**1.06**	52.25	54.12	1.22	0.99 %	0.44 %
50.30.2.c	57.24	59.72	1.39	58.55	60.15	**1.69**	**−2.24 %**	**−0.72 %**
50.40.1.a	60.38	64.55	2.43	65.94	68.16	**1.48**	**−8.43 %**	**−5.29 %**
50.40.1.b	63.64	65.72	2.46	64.67	68.39	**1.53**	**−1.60 %**	**−3.91 %**
50.40.1.c	63.51	65.28	1.26	63.00	64.83	**1.90**	0.82 %	0.70 %
50.40.2.a	61.67	64.43	2.48	60.98	63.64	**1.27**	1.13 %	1.24 %
50.40.2.b	70.12	71.82	1.17	71.00	72.79	**0.85**	**−1.24 %**	**−1.33 %**
50.40.2.c	66.94	67.72	0.78	66.97	67.91	**0.61**	**−0.05 %**	**−0.28 %**
100.30.1.a	109.45	114.58	3.92	111.08	115.33	**3.82**	**−1.47 %**	**−0.65 %**
100.30.1.b	113.61	116.57	**1.95**	118.38	124.75	3.16	**−4.03 %**	**−6.56 %**
100.30.1.c	103.46	109.66	3.41	106.89	111.02	2.21	**−3.21 %**	**−1.23 %**
100.30.2.a	112.68	114.77	1.40	116.83	120.51	**2.46**	**−3.56 %**	**−4.76 %**
100.30.2.b	102.86	106.19	**2.55**	108.95	111.54	3.05	**−5.59 %**	**−4.80 %**
100.30.2.c	121.37	123.48	1.23	122.68	124.54	**1.80**	**−1.06 %**	**−0.85 %**
100.40.1.a	119.77	122.67	2.34	122.86	125.53	**1.68**	**−2.52 %**	**−2.28 %**
100.40.1.b	113.51	118.12	**3.03**	117.95	124.93	3.84	**−3.76 %**	**−5.44 %**
100.40.1.c	111.34	114.55	2.23	113.94	117.31	**1.91**	**−2.28 %**	**−2.36 %**
100.40.2.a	119.98	123.44	**2.37**	120.90	124.53	2.61	**−0.76 %**	**−0.87 %**
100.40.2.b	113.15	116.88	**2.54**	114.57	120.25	3.82	**−1.24 %**	**−2.80 %**
100.40.2.c	111.25	114.98	2.96	112.77	117.21	**2.71**	**−1.35 %**	**−1.90 %**
150.30.1.a	153.01	156.93	**2.95**	157.40	166.63	5.83	**−2.79 %**	**−5.82 %**
150.30.1.b	161.12	165.18	**2.56**	163.55	171.72	3.73	**−1.49 %**	**−3.81 %**
150.30.1.c	163.06	173.27	4.97	175.70	179.64	**2.02**	**−7.20 %**	**−3.54 %**
150.30.2.a	165.86	170.72	**3.08**	166.99	176.28	6.14	**−0.67 %**	**−3.15 %**
150.30.2.b	206.77	212.81	4.34	213.14	215.93	**2.43**	**−2.99 %**	**−1.45 %**
150.30.2.c	153.20	160.52	**4.22**	161.11	170.56	4.29	**−4.91 %**	**−5.88 %**
150.40.1.a	154.53	158.31	2.58	157.76	160.23	**2.13**	**−2.05 %**	**−1.20 %**
150.40.1.b	170.70	176.94	**3.52**	178.28	185.18	4.96	**−4.25 %**	**−4.45 %**
150.40.1.c	191.29	195.69	**2.82**	199.34	203.24	3.07	**−4.04 %**	**−3.72 %**
150.40.2.a	174.18	181.40	5.10	177.23	183.18	**4.99**	**−1.72 %**	**−0.97 %**
150.40.2.b	181.15	188.33	4.52	185.98	192.14	**4.32**	**−2.60 %**	**−1.98 %**
150.40.2.c	182.97	188.68	2.59	187.68	191.09	**2.03**	**−2.51 %**	**−1.26 %**

Note: Bold types denote significantly better results.

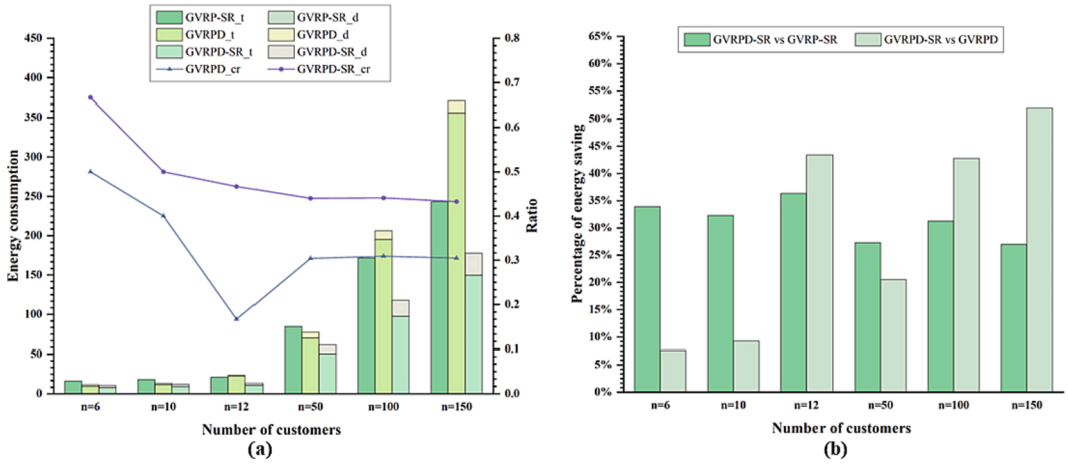

Fig. 4. (a) Detailed results obtained by the GVRP-SR, GVRPD, and GVRPD-SR models. (b) Energy saving percentage of the GVRPD-SR with respect to the GVRP-SR and GVRPD.

Table 6 lists the results obtained by both algorithms for solving the GVRPD-SR model. z_{best}^{IALNS}, z_{avg}^{IALNS}, and std^{IALNS} represent the best value, average value, and standard deviation obtained by the IALNS algorithm, respectively. Similarly, ALNS also has the corresponding three values, i.e., z_{best}^{ALNS}, z_{avg}^{ALNS}, and std^{ALNS}. Furthermore, Δz_{best} and Δz_{avg} show the gap between both the best and average objective values of the two algorithms, which can be computed using the method described in Section 5.3.2. Since ALNS is regarded as the benchmark, a negative value in Δz_{best} or Δz_{avg} means that IALNS algorithm is superior.

It can be seen in Table 6, IALNS algorithm leads the competition in 33 instances, while ALNS algorithm merely gets slight advantages in the rest three. The reason may be that the proposed IALNS algorithm is equipped with several specific operators for dealing with steep roads, rendering its search process more effective. Regarding stability, ALNS algorithm works better than IALNS algorithm overall, especially with 50 customers. With the increasing number of customers, IALNS algorithm

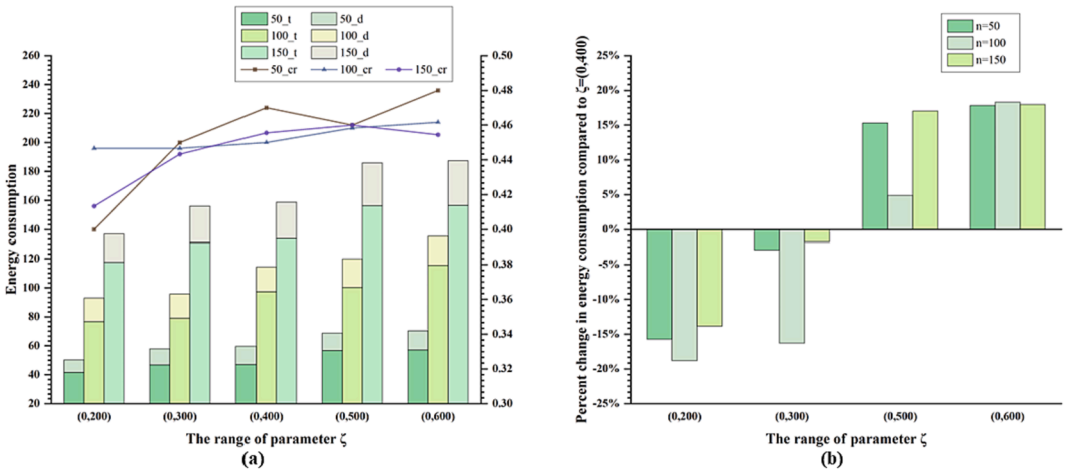

Fig. 5. (a) Decomposed objective of the three instances under each altitude scenario. (b) Percentage change in energy consumption under each altitude scenario compared to the baseline.

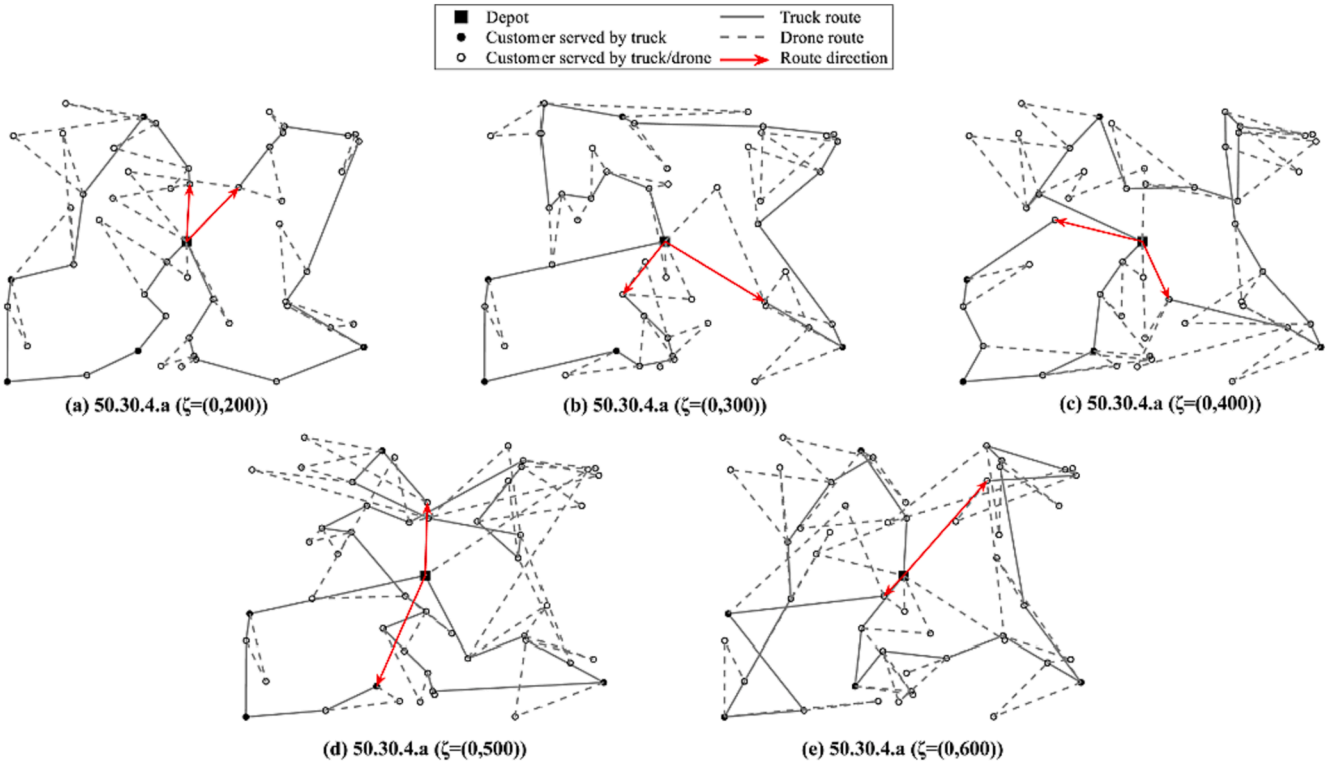

Fig. 6. The optimal solution under each altitude scenario with 50 customers.

can perform more stably than ALNS in some instances.

5.4. Feasibility of the GVRPD-SR

From the perspective of operations research, the GVRPD-SR integrates the classic VRPD and the VRP-SR models, with the objective function shifting from commercial purposes to minimizing energy consumption. Thus, the GVRPD-SR is compared with GVRPD and GVRP-SR on six instances to confirm its feasibility. Specifically, the GVRP-SR employs a pure fleet of conventional fuel trucks, and the energy consumption calculation for each truck mirrors that of the GVRPD-SR. Other formulations of the GVRP-SR can be referred to in the classic VRP-SR (Brunner et al., 2021). For the GVRPD, a fleet of trucks, each equipped with a drone, is dispatched for delivery tasks. Distinguished from the GVRPD-SR, the GVRPD adopts the energy consumption calculations for trucks and drones in the GVRPD-SR, but with the road grade set to 0. Consequently, the delivery operations for the drone are changed correspondingly, ascending vertically to a certain height, flying horizontally, and then descending vertically to the customer node. The delivery operations for trucks and the synchronization for trucks and drones remain consistent with those in the GVRPD-SR. In addition, the instances contain between 6 and 150 customers distributed in areas from 20×20 miles to 40×40 miles, with $t = 3$ and $h = a$. Fig. 4(a) intuitively presents the decomposed objectives obtained by the GVRPD-SR, GVRPD, and GVRP-SR models, and the energy savings of the GVRPD-SR with respect to the GVRP-SR and GVRPD are shown in Fig. 4(b).

The results of the GVRPD-SR and GVRP-SR are firstly analyzed. It can be observed from Fig. 4(a) that the energy consumption of GVRPD-SR is

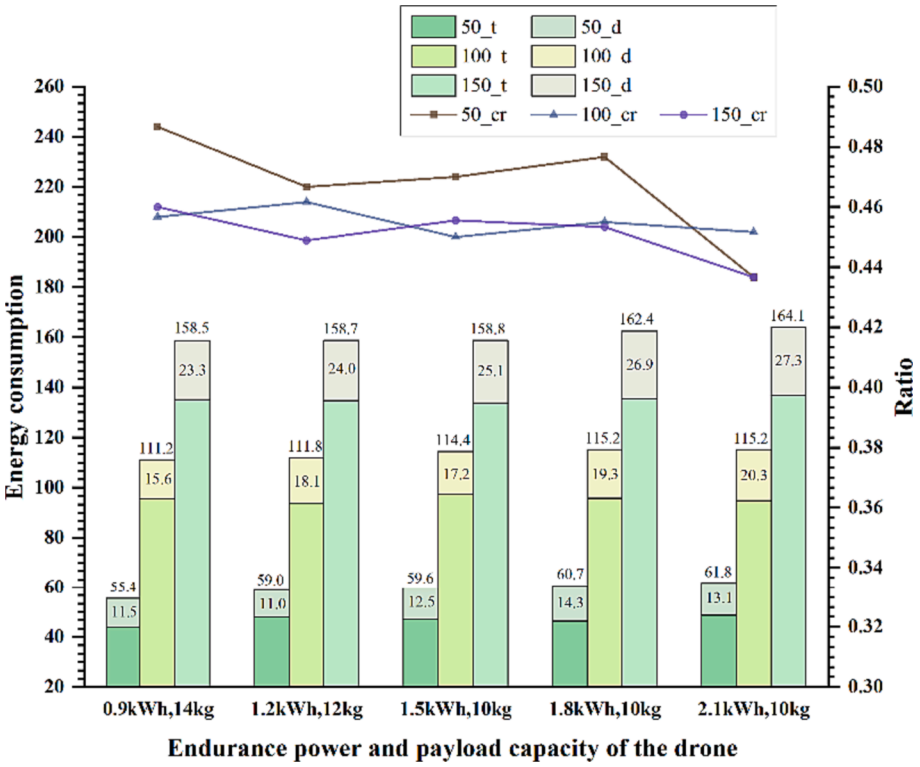

Fig. 7. Detailed results of the three instances under varied endurance power.

significantly lower than that of GVRP-SR in terms of trucks (see GVRPD-SR_t and GVRP-SR_t). Meanwhile, the difference grows as the number of customers increases. The explanation is that drones are preferable to trucks for delivering parcels on steep roads as their energy consumption is not of a similar order of magnitude. Consequently, as the broken line obtained by GVRPD-SR (see GVRPD-SR_cr) indicated that, a certain percentage of customers will be served by drones when drones are integrated into the rural last-mile logistic network. Furthermore, Fig. 4(b) shows that the cooperative delivery of trucks and drones in rural areas can reduce the total energy consumption by around 27 % – 36 %

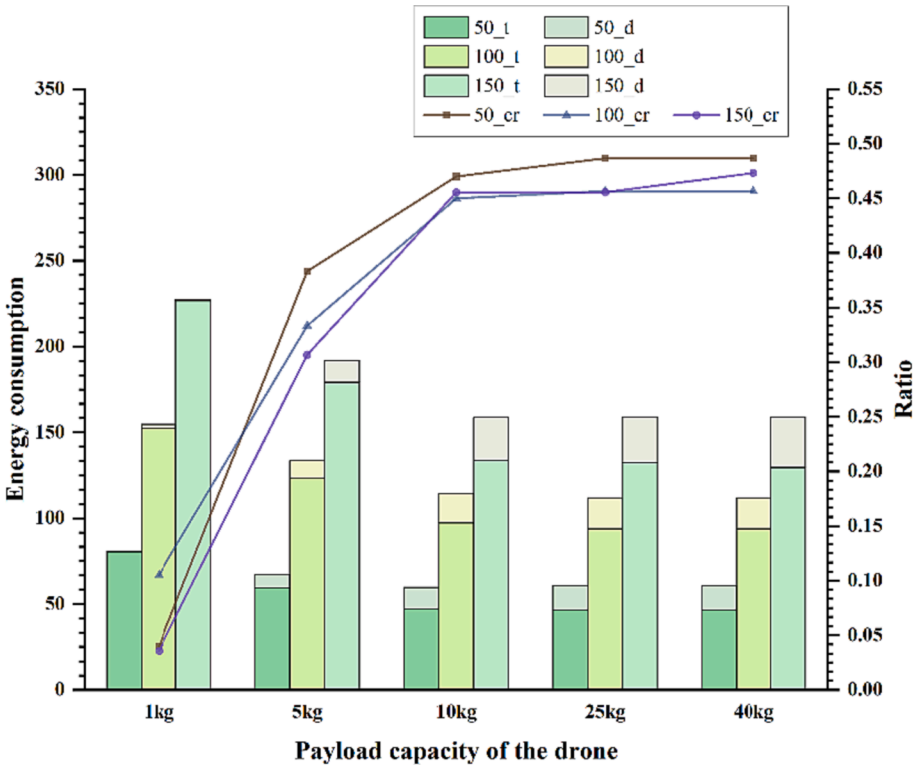

Fig. 8. Detailed results for customers of all sizes under each payload capacity.

Table 7
Traveled distance of the three instances under different payload capacities.

	Payload capacity	Number of customers		
		50	100	150
$Dist_t$	1 kg	138.40	227.99	338.01
$Dist_d$		6.36	29.52	12.89
	5 kg	98.74	177.09	255.19
		108.22	133.99	155.48
	10 kg	82.83	133.18	180.33
		167.99	209.39	297.28
	25 kg	80.73	126.71	183.75
		192.94	220.65	316.15
	40 kg	80.73	126.71	171.27
		192.94	220.65	355.21

Note: $Dist_t$ and $Dist_d$ denote the total travel distance of trucks and drones, respectively.

compared to a pure truck fleet, demonstrating the benefits of using drones.

For the comparison of GVRPD-SR (see GVRPD-SR_t and GVRPD-SR_d) and GVRPD (see GVRPD_t and GVRPD_d), the cumulative bars of Fig. 4(a) show that the optimal planned route will consume more energy when the impact of steep roads is ignored. The reason is evident from the two broken lines, i.e., the customer served ratio of drones in the GVRPD is considerably lower than that of the GVRPD-SR (see GVRPD-SR_cr and GVRPD_cr). On the other hand, Fig. 4(b) depicts the concrete percentage of energy savings between the GVRPD-SR and the GVRPD. In both small-scale and large-scale instances, the energy-saving advantage of GVRPD-SR becomes more prominent as the number of customers increases, with the highest energy savings reaching 52.04 %. As a result, it is necessary for a routing planning model to take into account the impacts of various practical factors.

5.5. Managerial discussions

The main components that constitute the GVRPD-SR are the presence of steep roads during the rural last-mile delivery and the use of drones to assist with the delivery tasks. Therefore, several critical parameters related to the two aspects are selected for sensitivity analysis to derive valuable management insights. The adopted instances are “50. 30. 4. a”, “100. 30. 4. a”, and “150. 30. 4. a”, respectively.

5.5.1. Sensitivity analysis on altitude difference

As explained in Appendix B, ξ is used to adjust the varying altitudes between customers so as to depict the delivery scenario with steep roads. To explore the exogenous impact of rural terrain, the maximum value of ξ is altered based on the set {200, 300, 500, 600}, with $\xi = (0, 400)$ regarded as a baseline.

Fig. 5(a) illustrates the decomposed objective of three instances under each altitude scenario. It can be seen that both trucks and drones have an apparent increasing trend in terms of energy consumption when carrying out delivery tasks on steep roads, as opposed to relatively flat roads. In addition, Fig. 6 illustrates the specific routes of trucks and drones to serve 50 customers in different altitude scenarios. From the ratios of customers served by drones in Fig. 5(a) and Fig. 6, drones deliver parcels to a portion of customers, but trucks remain the main force for last-mile delivery. The reason lies in some technical boundaries of drones, such as their limited endurance power and payload capacity. These limitations also result in the need for drones to be launched and rendezvoused more frequently as the altitude difference increases, as shown in Fig. 6.

Fig. 5(b) shows the changes in the total energy consumption of the four altitude scenarios with respect to the baseline. As seen from Fig. 5 (b), the energy consumption is highly sensitive to road grades, highlighting the importance for a route planning model to take into account the impact of steep roads. Meanwhile, as seen from the cumulative bars of $\xi = (0, 200)$ and $\xi = (0, 600)$, their average percent changes in total energy consumption are 16.12 % and 18.03 %, respectively. Specifically, in the scenario with $\xi = (0, 600)$, the change exceeds 17 % for all three instances. In summary, the cooperative delivery of trucks and drones is an excellent suggestion for the rural last-mile, especially in areas with more mountainous terrain.

5.5.2. Sensitivity analysis on endurance power

The maximum endurance power of each drone is set at 1.5 kWh in Section 5.2. In actual applications, if the maximum endurance power is reduced, the reduced battery weight can be utilized to carry heavier parcels instead. On the other hand, with the progressive development of technology, it is possible for drones to carry heavier batteries to improve their endurance without sacrificing their payload capacity. Hence, this section investigates how endurance power affects the performance of trucks and drones for cooperative delivery in rural areas.

As seen from Fig. 7, the energy consumption of trucks and drones does not go down as the endurance power of drones increases. The total energy consumption of each instance peaks when the endurance power reaches 2.1 kWh. Conversely, scenarios with an endurance power of 0.9 kWh have the lowest energy consumption. In addition, the ratio of customers served by drones at other higher endurance power levels is mostly cut down compared to that at endurance power of 0.9 kWh. These results are unexpected but explainable. On the one hand, a decreased level of endurance power contributes to this result as it loosens the payload capacity restriction for drones. On the other hand, these levels of endurance power may be adequate for most trips by each drone since only one customer is served at a time in this study. Consequently, when the endurance power is set to 0.9 kWh, drones can serve more customers due to their increased capacity, leading to a reduction in total energy consumption. When the endurance power substantially increases but the capacity reduces, drones are inclined to be dispatched to serve remote customers due to their superior endurance power, whereas trucks can serve more customers nearby during drone services with a greater customer ratio. Moreover, drones will consume more energy because of their longer travel distance and relatively heavier self-weight, dominantly contributing to a small increase in total energy consumption. The above analysis may facilitate enterprises tradeoffs between endurance power and payload capacity based on different scenarios.

5.5.3. Sensitivity analysis on payload capacity

The payload capacity of a drone is set to 10 kg in the main configuration. Nonetheless, drones are expected to carry heavier parcels in the future as their hardware keeps improving. This increase is conducive to the expansion of drones’ service scope. Hence, the effect of different payload capacities on route planning is studied.

It can be observed from Fig. 8 that as the payload capacity of drones increases, the total energy consumption gradually decreases while the proportion of drone service grows. This result can be explained by the change in travel distance of trucks and drones listed in Table 7. The increased travel distance of drones means that total energy consumption is reduced by using more drone services to reduce truck detours or avoid traveling on some steeper roads for trucks. It can also be seen from Fig. 8 that the too-small payload capacity of a drone will inhibit the advantages of cooperative delivery of trucks and drones. On the contrary, the change in the two indicators becomes slow when the payload capacity exceeds 10 kg. This is because the median demand of customers is set to 10 kg in all generated instances. If this parameter is set larger, the change may be more obvious. Considering that the deployment cost of drones is expensive, these findings indicate that enterprises should make informed decisions about which kind of drones they use after thoroughly surveying customer demands.

Table C1
Parameters used in the GVRPD-SR model.

Parameter	Description	Value	
		truck	drone
g	gravitational constant $\left(\mathrm{m/s^2}\right)$	9.81	
a	air density $\left(\mathrm{kg/m^3}\right)$	1.225	
A	frontal surface area $\left(\mathrm{m^2}\right)$	$A_t = 6$	$A_d = 0.15$
m (without payload)	tare weight (ton)	$m_t = 2.5$	$m_d = 0.025$
P	engine power (kW)	150	——
f^{idle}	fuel consumption (idle)(L/h)	1	——
f^{full}	fuel consumption (full)(L/h)	25	——
NHV^{Diesel}	net heating value (kWh/L)	10	——
c_{roll}	rolling resistance	0.008	——
c_{air}	air drag	0.65	0.5
n^{acc}	frequency of acceleration	0	——
cap_b	maximum endurance power (kWh)	——	1.5
P^{int}	power internal auxiliaries (kW)	——	0.1
η_{eng}	engine efficiency	——	0.9
η_{trans}	transmission efficiency	——	0.9
η_{char}	charging efficiency	——	0.9
r	rotor radius (m)	——	0.4
n_{rotor}	number rotors	——	8
n_{blades}	number blades	——	3
$\bar{c}$	rotor mean chord	——	0.1
$\bar{c}_l$	blade lift	——	0.4
c_{bd}	blade drag	——	0.075
K	lifting power markup	——	1.15
ν	speed (km/h)	$\nu_t = 40$	$\nu_d = 80$
Q	Maximum load capacity (kg)	$Q_t = 300$	$Q_d = 10$

6. Conclusions

In this study, a new GVRPD-SR model is proposed to explore the prospect of cooperation between trucks and drones for rural last-mile delivery with steep roads. The mathematical formulation of the GVRPD-SR model is extended based on the basic constraints of VRPD models, with the objective to minimize the total energy consumption of the delivery process. Besides, realistic energy consumption calculations for trucks and drones are incorporated into the GVRPD-SR model. They not only consider the impact of general factors but also the impact of a representative feature of rural areas (i.e., abounding steep roads). As the solution approach for the GVRPD-SR model, an ALNS-based algorithm (i.e., IALNS algorithm) is introduced, which includes several novel operators designed based on the characteristics of the proposed problem.

A systematic comparison experiment is conducted from three perspectives, i.e., the efficiency of the IALNS algorithm, the effectiveness of the GVRPD-SR model, and managerial discussion. First of all, the experimental results demonstrate that the IALNS algorithm achieves desirable performance to solve the GVRPD-SR model in instances of varied scales by comparing it with CPLEX and ALNS. Then, comparing the GVRPD-SR with the GVRP-SR model, the obtained results indicate that the advantages of cooperation between trucks and drones are remarkable in rural areas. On average, this cooperative delivery mode results in an impressive energy savings of 31.34 %. Besides, the comparison between the GVPRD-SR and GVRPD reveals that considering the effect of steep roads facilitates planning more environmentally friendly routes for managers. Finally, several managerial insights are extracted from the sensitivity analysis. (1) Road information is essential to planning logistics activities, and the cooperative delivery of trucks and drones is more promising for areas with more mountainous terrain. (2) It is suggested to select a suitable compromise of endurance power and payload capacity for drones based on the demands and location of customers.

In future works, several aspects can be further explored based on the presented research. For instance, other settings, such as multiple drones, pickup and delivery, and allowing drones to visit multiple customers in a single flight, can be added to extend the model. Besides, the realistic dynamics of the delivery process can be taken into account, including weather conditions, unexpected truck failures, and the uncertain time window of customers. Finally, deep (reinforcement) learning and other heuristics can also be utilized to enhance algorithmic performance.

Disclosure statement

The authors report there are no competing interests to declare.

CRediT authorship contribution statement

Jiuhong Xiao: Conceptualization, Methodology, Formal analysis, Writing – original draft, Validation. **Ying Li:** Methodology, Writing – original draft, Software, Validation. **Zhiguang Cao:** Supervision, Writing – review & editing, Formal analysis. **Jianhua Xiao:** Writing – review & editing, Supervision, Funding acquisition, Validation.

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.

Acknowledgements

This work was supported by the National Natural Science Foundation of China [grant numbers 62072258, 61772290]; and the Special Foundation for Philosophy and Social Science Laboratories of Ministry of Education of China [grant number ZX20220103].

Appendix A. Related parameters of drone energy consumption

The energy required by the rotor to overcome air drag depends on the number and size of the rotor and other physical parameters. Let r denote the radius of the rotors and n_{rotor} the number of rotors. The total area R over which the air flows on the rotors is calculated as

$$R = \pi \cdot r^2 \cdot n_{rotor} \tag{A1}$$

The speed of the blade tips V_d relies on the thrust to be exerted and the physical qualities of the blades. Let n_{blades} denote the number of blades per rotor, $\bar{c}$ is the rotor mean chord, and $\bar{c}_l$ represents the mean lift coefficient. V_d can be computed as

$$V_d = \sqrt{\frac{6 \cdot m_d \cdot g}{n_{rotor} \cdot n_{blades} \cdot \bar{c} \cdot \bar{c}_l \cdot \rho \cdot r}} \tag{A2}$$

In addition, the disc solidity ratio σ is defined as $\frac{n_{blades} \cdot \bar{c}}{\pi \cdot r}$. The blade drag coefficient c_{bd} is mainly determined by the airfoil and increases with the thrust coefficient and blade lift coefficient, which is set to 0.075.

Finally, the downwash ω is determined by the following equation:

$$\frac{T}{2 \cdot \rho \cdot R} = \omega \cdot \sqrt{(\omega - \nu_d \cdot \sin\vartheta)^2 + (\nu_d \cdot \cos\vartheta)^2} \tag{A3}$$

where ϑ is the angle of attack, which is calculated by $\vartheta = \arctan\left(\frac{-D_{body} - m_d \cdot g \cdot \sin\alpha}{m_d \cdot g \cdot \cos\alpha}\right)$.

Appendix B. Principles for generating benchmark instances

The newly generated benchmark includes two distinct classes, i.e., small-scale and large-scale instances. The former involves 6, 10, or 12 customers, while the latter comprises 50, 100, or 150 customers. These customers are randomly placed within a $2d \times 2d$ square grid by a uniform distribution $U(-d, d)$. Considering that the realistic rural environment is broad and covers a considerable region, d is set between {10, 20} for small-scale instances and {30, 40} for large-scale instances. In addition, the central depot in all instances is always located at coordinates (0, 0).

In 2020, SF Technology launched an emergency drone delivery service capable of carrying up to 10 kg of parcels at a time. Moreover, UPS stated that the maximum load of a parcel transported by a truck is 68 kg. Following the above limits, let p be a random number ranging from [0, 1] to indicate potential scenarios for small parcel delivery with drones, and the demand (kg) of a customer is given by:

$$q_i = \begin{cases} q_i \in U(0, 10) & \text{if } p < 0.86 \\ q_i \in U(10, 68) & \text{otherwise} \end{cases} \tag{B1}$$

Regarding the altitude for each customer, the Foxholes Shekel function to simulate 3-D terrain of Arshi et al. [44] is modified as follows:

$$h(x, y) = \left\lfloor \xi \times \sum_{i=1}^{n} \frac{C}{(x - a_i)^2 + (y - b_i)^2 + c} \right\rfloor \tag{B2}$$

where n is the total number of customers, and (a_i, b_i) represents the customer coordinate. C and c are both constants that alter the shape of terrains, with values of 2 and 3, respectively. Let $0 \leq \xi \leq 400$ be a random number associated with the difference in the altitudes of customers.

Appendix C. Parameter settings of the GVRPD-SR

According to the recommendation of Kirschstein [45], all parameter settings of the GVRPD-SR model are summarized in Table C1.

References

Bektaş, T., & Laporte, G. (2011). The Pollution-Routing Problem. *Transportation Research Part B: Methodological, 45*(8), 1232–1250.

Brunner, C., Giesen, R., Klapp, M. A., & Flórez-Calderón, L. (2021). Vehicle routing problem with steep roads. *Transportation Research Part A: Policy and Practice, 151*, 1–17.

China International Electronic Commerce Center. China rural e-commerce development report (2021-2022). https://ciecc.ec.com.cn/article/gzdt/202210/.

Chiang, W. C., Li, Y. Y., Shang, J., & Urban, T. L. (2019). Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization. *Applied Energy, 242*, 1164–1175.

Chen, C., Demir, E., & Huang, Y. (2021). An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. *European Journal of Operational Research, 294*(3), 1164–1180.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*(4), 568–581.

Coindreau, M. A., Gallay, O., & Zufferey, N. (2021). Parcel delivery cost minimization with time window constraints using trucks and drones. *Networks, 78*(4), 400–420.

Dundar, H., Soysal, M., Omurgonulsen, M., & Kanellopoulos, A. (2022). A green dynamic TSP with detailed road gradient dependent fuel consumption estimation. *Computers & Industrial Engineering, 168*, Article 108024.

Goeke, D., & Schneider, M. (2015). Routing a mixed fleet of electric and conventional vehicle. *European Journal of Operational Research, 245*(1), 81–99.

Kang, M., & Lee, C. (2021). An exact algorithm for heterogeneous drone-truck problem. *Transportation Science, 55*(5), 1088–1112.

Kastrenakes, J. (2017). UPS has a delivery truck that can launch a drone. http://www.theverge.com/2017/2/21/14691062/ups-drone-delivery-truck-test-completed-video/.

Karakostas, P., Sifaleras, A., & Georgiadis, M. C. (2020). Adaptive variable neighborhood search solution methods for the fleet size and mix pollution location-inventory-routing-problem. *Expert Systems with Applications, 153*, Article 113444.

Kirschstein, T., & Meisel, F. (2015). GHG-emission models for assessing the eco-friendliness of road and rail freight transports. *Transportation Research Part B: Methodological, 73*, 13–33.

Kitjacharoenchai, P., Min, B. C., & Lee, S. (2020). Two-echelon vehicle routing problem with drones in last mile delivery. *International Journal of Production Economics, 225*, Article 107598.

Kuo, R. J., Lu, S. H., Lai, P. Y., & Mara, S. T. W. (2022). Vehicle routing problem with drones considering time windows. *Expert Systems with Applications, 191*, Article 116264.

Langelaan, J. W., Schmitz, S., Palacios, J., & Lorenz, R. D. (2017). Energetics of rotary-wing exploration of Titan. *IEEE Aerospace Conference*, 1–11.

Li, H., Wang, H., Chen, J., & Bai, M. (2020). Two-echelon vehicle routing problem with time windows and mobile satellites. *Transportation Research Part B: Methodological, 138*, 179–201.

Liu, Y., Liu, Z., Shi, J. M., Wu, G. H., & Pedrycz, W. (2021). Two-echelon routing problem for parcel delivery by cooperated truck and drone. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51*(12), 7450–7465.

Liu, Y. M., Yu, Y., Zhang, Y., Baldacci, R., Tang, J. F., Luo, X. G., &,, et al. (2023). Branch-cut-and-price for the time-dependent green vehicle routing problem with time windows. *INFORMS Journal on Computing, 35*(1), 14–30.
Macrina, C., Pugliese, L. D. P., Guerriero, F., & Laporte, G. (2020). Drone-aided routing: A literature review. *Transportation Research Part C: Emerging Technologies, 120*, Article 102762.
Majidi, S., Hosseini-Motlagh, S. M., & Ignatius, J. (2018). Adaptive large neighborhood search heuristic for pollution-routing problem with simultaneous pickup and delivery. *Soft Computing, 22*, 2851–2865.
Meng, S. S., Guo, X. P., Li, D., & Liu, G. Q. (2023). The multi-visit drone routing problem for pickup and delivery services. *Transportation Research Part E: Logistics and Transportation Review, 169*, Article 102990.
Poikonen, S., Wang, X., & Golden, B. L. (2017). The vehicle routing problem with drones: Extended models and connections. *Networks, 70*(1), 34–43.
Rao, W. Z., Liu, F., & Wang, S. B. (2016). An efficient two-objective hybrid local search algorithm for solving the fuel consumption by vehicle routing problem. *Applied Computational Intelligence and Soft Computing, 2016*, 3713918.
Rauniyar, A., Nath, R., & Muhuri, P. K. (2019). Multi-factorial evolutionary algorithm based novel solution approach for multi-objective pollution-routing problem. *Computers & Industrial Engineering, 130*, 757–771.
Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science, 40*(4), 455–472.
Sacramento, D., Pisinger, D., & Ropke, S. (2019). An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transportation Research Part C: Emerging Technologies, 102*, 289–315.
Sadati, M. E. H., & Çatay, B. (2021). A hybrid variable neighborhood search approach for the multi-depot green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review, 149*, Article 102293.
Salama, M. R., & Srinivas, S. (2022). Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites. *Transportation Research Part E: Logistics and Transportation Review, 164*, Article 102788.
Schröder, M., & Cabral, P. (2019). Eco-friendly 3D-Routing: A GIS based 3D-Routing-Model to estimate and reduce CO2-emissions of distribution transports. *Computers, Environment and Urban Systems, 73*, 40–55.
Schermer, D., Moeini, M., & Wendt, O. (2019a). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies, 106*, 166–204.
Schermer, D., Moeini, M., & Wendt, O. (2019b). A hybrid VNS/Tabu search algorithm for solving the vehicle routing problem with drones and en route operations. *Computers & Operations Research, 109*, 134–158.
Sharma, S., & Mathew, T. V. (2011). Multiobjective network design for emission and travel-time trade-off for a sustainable large urban transportation network. *Environment and Planning B: Planning & Design, 38*(3), 520–538.
State Post Bureau of The People's Republic of China. (2022). https://sannong.cctv.com/2022/01/07/.
US Department of Energy. (2008). Fuel economy guide. http://www.fueleconomy.gov.
Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation Research Part B: Methodological, 122*, 350–364.
Wang, X. Y., Poikonen, S., & Golden, B. (2017). The vehicle routing problem with drones: Several worst-case results. *Optimization Letters, 11*, 679–697.
Wohlsen, M. (2014). The next big thing you missed: Amazon's delivery drones could work-they just need trucks. http://www.wired.com/2014/06/the-next-big-thing-you-missed-delivery-drones-launched-from-trucks-are-the-future-of-shipping/.
Zhang, S. Y., Zhou, Z. H., Luo, R., Zhao, R. Z., Xiao, Y. Y., & Xu, Y. C. (2022). A low-carbon, fixed-tour scheduling problem with time windows in a time-dependent traffic environment. *International Journal of Production Research*. https://doi.org/10.1080/00207543.2022.2153940
Zhen, L., Gao, J. J., Tan, Z. Y., Wang, S. A., & Baldacci, R. (2023). Branch-price-and-cut for trucks and drones cooperative delivery. *IISE Transactions, 55*(3), 271–287.
Zhou, Y. J., & Lee, G. M. (2017). A lagrangian relaxation-based solution method for a green vehicle routing problem to minimize greenhouse gas emissions. *Sustainability, 9*(5), 776.
Zhou, H., Qin, H., Cheng, C., & Rousseau, L. M. (2023). An exact algorithm for the two-echelon vehicle routing problem with drones. *Transportation research part B: Methodological, 168*, 124–150.
Zhu, L., & Hu, D. W. (2019). Study on the vehicle routing problem considering congestion and emission factors. *International Journal of Production Research, 57*(19), 6115–6129.