Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Integration of machine learning prediction and heuristic optimization for mask delivery in COVID-19

Xin Chen a, Hong-Fang Yan b, Yu-Jun Zheng a, *, Mumtaz Karatas c

a School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China
b Information Technology Center, Huzhou University, Huzhou Zhejiang 313002, China
c Industrial Engineering Department, Naval Academy, National Defense University, Tuzla 34940, Istanbul, Turkey

A R T I C L E  I N F O

Keywords:
Machine learning
Heuristic optimization
Pre-distribution
Vehicle routing
Tabu search
Variable neighborhood

A B S T R A C T

The novel coronavirus pneumonia (COVID-19) has created huge demands for medical masks that need to be delivered to a lot of demand points to protect citizens. The efficiency of delivery is critical to the prevention and control of the epidemic. However, the huge demands for masks and massive number of demand points scattered make the problem highly complex. Moreover, the actual demands are often obtained late, and hence the time duration for solution calculation and mask delivery is often very limited. Based on our practical experience of medical mask delivery in response to COVID-19 in China, we present a hybrid machine learning and heuristic optimization method, which uses a deep learning model to predict the demand of each region, schedules first-echelon vehicles to pre-distribute the predicted number of masks from depot(s) to regional facilities in advance, reassigns demand points among different regions to balance the deviations of predicted demands from actual demands, and finally routes second-echelon vehicles to efficiently deliver masks to the demand points in each region. For the subproblems of demand point reassignment and two-batch routing whose complexities are significantly lower, we propose variable neighborhood tabu search heuristics to efficiently solve them. Application of the proposed method in emergency mask delivery in three megacities in China during the peak of COVID-19 demonstrated its significant performance advantages over other methods without pre-distribution or reassignment. We also discuss key success factors and lessons learned to facilitate the extension of our method to a wider range of problems.

1. Introduction

During the outbreak of the novel coronavirus pneumonia (COVID-19), there have been huge demands for medical masks, which need to be delivered to numerous demand points (such as hospitals, schools, and housing estates) to protect citizens. For example, during late January and February, 2020, the weekly demands for medical masks were around 10 to 20 million in mega-cities such as Wuhan, Guangzhou, and Hangzhou in China. That is, every day, around 200,000 masks needed to be distributed to thousands to tens of thousands of demand points. Normally, masks are stored in one or multiple depots (e.g., large warehouses in transportation hubs or mask manufactures, where the quantity of masks can be supplemented every day). As the demands are huge and demand points are numerous, it can be very difficult or inefficient to deliver all masks directly from the depots to the demand points. A typical solution is to divide the city into a set of regions, and designate a regional facility (such as a district health station or large drugstore) for transferring masks from the depot(s) to the demand points in each region. In this way, the delivery task is divided into two levels: at the first level, vehicles with high speeds and large capacities (e.g., trucks) are used to transport masks from the depots to the regional facilities; at the second level, short-range vehicles (e.g., minivans) that are relatively more flexible in community streets are utilized to deliver masks from regional facilities to demand points within their responsibility regions, as illustrated in Fig. 1. Such a two-echelon distribution system can significantly decrease the complexity of vehicle routing [1,2].

Nevertheless, the two-echelon vehicle routing problem (2E-VRP) is still an NP-hard problem [3], and solving a large instance with tens of regional facilities and thousands of demand points is also very difficult. Besides, the mask delivery problem in an emergency like COVID-19 has a special challenge caused by the late time at which we can acquire the exact demands of the demand points [4]. In our study, the demands are typically acquired at midnight of the day before delivery or early morning of the delivery day. Therefore, the remaining time for
problem solving and solution implementation is quite limited, which can significantly delay the delivery time.

To address these difficulties and improve the delivery efficiency, based on our practical experience of emergency mask delivery in response to COVID-19 in China, we present a hybrid learning and optimization method, which, on each day, uses deep learning to predict the demand of each region for the next day, and schedules first-level vehicles to transport masks according to the predicted demands to regional facilities in advance. However, as can be expected, the predicted demands can deviate from the actual demands more or less. To balance the deviations, before each delivery day, the demand points are reassigned to regions. If, after reassignment, there are still some regions with mask shortages, we supplement additional masks from depots to these regions. Finally, we schedule the second-level vehicles to deliver masks from each regional facility to its assigned demand points. The overall flowchart of our method is presented in Fig. 2. The main advantage of this method is twofold:

• By the pre-distribution of masks to regions, each region can plan and start second-echelon delivery much earlier, and therefore significantly shorten the overall delivery time.
• The integrated 2E-VRP are divided into the reassignment problem and a set of independent single-echelon VRP instances, which are greatly simpler than the original problem and can be solved simultaneously and more qualitatively within a short response time.

During the peak of COVID-19 epidemic in China, we have first applied the proposed method to emergency medical mask delivery in Hangzhou city. Upon demonstrating the feasibility and success of the approach, we have tested and applied it to other mega-cities in China. Our results indicated that the proposed approach can significantly improve the delivery efficiency and therefore enhance the overall effectiveness of epidemic prevention and control measures. The method has also been adapted and extended for the emergency delivery of other medical supplies, such as disinfectants and vaccines.

The remainder of this paper is organized as follows. Section 2 discusses related work, Section 3 introduces the target problem, Section 4 proposes the hybrid learning and optimization method, Section 5 presents the application results and discusses the lessons learned, and Section 6 concludes.

2. Related work

Generally, the distribution of goods from depots through regional facilities (satellites) to demand points (customers) can be classified as a 2E-VRP. A basic 2E-VRP formulation was introduced by González-Feliu et al. [3], who proposed a branch-and-cut algorithm that solved instances with up to 32 customers and two satellites. The algorithm was improved by Perboli et al. [5–7] by adding valid inequalities in a cutting plane fashion, resulting in solutions to instances with up to 50 customers and four satellites. Baldacci et al. [8] presented a capacitated 2E-VRP (2E-CVRP) formulation and proposed an exact method that decomposes a 2E-CVRP into a set of multi-depot capacitated VRPs [9] with side constraints. A set of exact optimization algorithms, such as branch-and-cut [10], branch-and-price [11], and branch-and-cut-and-price [12], have been proposed for 2E-VRP and tested on instances with...
up to 100 customers and six satellites. However, the considered mask delivery scenario usually involves tens of satellites and thousands of customers, which are obviously beyond the capabilities of these exact algorithms.

In recent years, more efforts have been devoted to heuristics and metaheuristics that are capable of finding near-optimal or sufficiently good solutions to large 2E-VRP instances. Craain et al. [13] presented an adaptive large neighborhood search (ALNS) heuristic for 2E-VRP by adapting the probability distribution to the problem structure. Breunig et al. [15] proposed a hybrid metaheuristic that combines enumeration local searches with destroy-and-repair heuristics. Breunig et al. [16] proposed an ALNS for a 2E-VRP with satellite synchronization, which is capable of solving instances with up to 200 customers and 10 satellites. Li et al. [17] studied a 2E-VRP where vehicle routes on different levels are interacted by time constraints; they proposed a Clarke and Wright heuristic and simulated annealing to solve the problem. ALNS was also used by Li et al. [18] to tackle a 2E-VRP with time windows and mobile satellites, and by Li et al. [19] for a 2E-VRP with satellite bi-synchronization. Belgin et al. [20] proposed a hybrid variable neighborhood descent (VND) and local search algorithm for 2E-VRP with simultaneous pickup and delivery. Liu et al. [21] studied a new variant of 2E-VRP that uses trucks and drones to cooperatively deliver parcels; they proposed a hybrid simulated annealing and tabu search algorithm to optimize both the truck route and the drone route.

To solve a 2E-CVRP with stochastic demands, Wang et al. [22] proposed a genetic algorithm (GA) by designing a simple encoding and decoding scheme, a modified route copy crossover operator, and a satellite-selection-based mutation operator. Results showed that the expected cost obtained by the GA was not greater than that of the best-known solution for each test instance. Zhou et al. [23] proposed a hybrid multi-population GA for a multi-depot 2E-VRP, which demonstrated effectiveness on a large family of instances. Yan et al. [24] proposed a graph-based fuzzy evolutionary algorithm, which integrates a graph-based fuzzy satellite-to-customer assignment scheme into an iteratively evolutionary learning process to minimize the total cost of 2E-VRP. Entchoven et al. [25] presented a 2E-VRP with covering options, where the first-elevation transport goods from a depot to two types of locations, including satellite locations for the second-elevation delivery and covering locations where customers can pick up goods; they proposed a tailored ALNS heuristic to solve the special problem. Anderluh et al. [26] studied a multi-objective 2E-VRP considering not only the cost but also negative external effects such as emissions and disturbances; they proposed a metaheuristic that combines a large neighborhood with an ε-constraint method to approximate the Pareto-optimal solutions to the problem. To solve a real-time 2E-VRP with pickup and delivery that needs to be solved within seconds, do C. Martins et al. [27] proposed a constructive-heuristic-based biased-randomized algorithm using a skewed probability distribution to modify its greedy behavior. Results showed that, using massive parallel computing, the method generates competitive results for instances with up to 150 customers.

In real-world applications, parameters involved in optimization problems are often subject to uncertainty due to a variety of reasons such as estimation errors and lack or disturbance of data; therefore, data-driven optimization that integrates optimization algorithms with machine learning that analyzes and extracts data to hedge against uncertainty and facilitate decision-making has received growing attention [28]. Delage and Ye [29] proposed a general data-driven stochastic programming (also known as distributionally robust optimization) model, where the objective function is defined on both a vector of decision variables and a random vector whose probability distribution need to be learned from uncertainty data by machine learning methods. This optimization paradigm has been popularized in various applications such as production planning and scheduling [30] and shale gas supply chain optimization [31]. For large-scale, expensive-ness optimization problems, data-driven evolutionary algorithms [32] provide a computationally efficient approach by approximating the real objective/constraint functions using surrogate models, which are constructed by learning from offline (historical) or online (real-time) data. Huang et al. [33] proposed an offline data-driven evolutionary algorithm that builds surrogate models in each generation and uses tri-training, a semi-supervised learning method, to generate pseudo labels and updates the surrogate models. Considering the supply of daily necessities and medical items for home-quarantined people, Wu et al. [34] developed a truck-and-drone coordinated delivery system that does not require direct human contact, where a reinforcement learning model is employed to train encoding-decoding policies for constructing truck/drone routes. In [35], the authors proposed an improved variable neighborhood descent method combined with simulated annealing and tabu list to solve contactless parcel delivery using collaborative truck–drone routing. Zhen et al. [36] proposed an offline data-driven evolutionary optimization framework that selects suitable surrogate models from a model pool constructed by four radial basis function models with different smoothnesses. Sadhu et al. [37] proposed an improved firefly algorithm, where the optimal parameter values for each firefly are learned by Q-learning strategy; the hybrid algorithm was examined in the path planning of a robotic manipulator amidst various obstacles. For multi-many-objective optimization, Xu et al. [38] leveraged federated learning for surrogate construction so that multiple clients collaboratively train a radial-basis-function-network as the global surrogate; they proposed a new federated acquisition function to approximate the objective values using the global surrogate and estimate the uncertainty level of the approximated values. Wang et al. [39] proposed a multi-objective evolutionary algorithm that estimates the Pareto optimal set in a reduced decision space, which is learned using principal component analysis from well-converged solutions. Yu et al. [40] proposed a dynamic multi-objective evolutionary algorithm based on a stochastic regression and adaptive clustering, where a regression-based predictor is designed for generating the initial population, and an adaptive reference vector regulator based on clustering is used to track the changes of Pareto optimal front. Very recently, Huang and Gong [41] proposed an alternative way that establishes a contrastive learning model to perform binary classification (which is relatively simpler than regression) to determine whether the absolute fitness of each individual is required. In summary, most existing data-driven optimization methods try to utilize knowledge learned from historical data to improve the problem-solving process, but few utilize knowledge to improve the work flow behind the optimization problem at hand, which can reduce the inherent complexity of the problem.

3. Problem overview

The problem considered in this paper is to deliver medical masks from depots through regional facilities to demand points. The set of demand points (customers) is denoted as \( C \), and the demand of each point \( c \in C \) is \( r_c \). The set of depots is denoted as \( D \); for each \( d \in D \), the available quantity of masks is \( a_d \), and available number of first-level vehicles is \( K_d \). The set \( S \) of regional facilities (satellites) is denoted as \( S \); for each \( s \in S \), the available number of second-level vehicles is \( K_s \). The capacity of each first-level and second-level vehicle is denoted by \( Q_1 \) and \( Q_2 \), respectively. The travel time between each pair of locations \( p_1 \) and \( p_2 \) is assumed to be known and denoted as \( t(p_1, p_2) \), \( \forall p_1, p_2 \in D \cup S \).

The public health department classifies all demand points into five subsets, which are listed in order of priority as follows:

\( C_1 \): Designated COVID-19 hospitals,

\( C_2 \): Shale gas-related hospitals,

\( C_3 \): Hospitals with high patient volume,

\( C_4 \): Hospitals with moderate patient volume,

\( C_5 \): Other hospitals.
Demand points of higher priorities play more important roles in the epidemic prevention and control, and their demands are therefore more urgent. In this study, we assign each point $c$ in the above five subsets with an importance weight $w_c$ of 5, 4, 3, 2, and 1, respectively. Note that it is possible to use more refined weights by implementing weighting methodologies such as the analytic hierarchy process.

After obtaining the actual demands of all points, we may find that the total quantity $A = \sum_{c \in C} a_c$ of masks is smaller than the total demand $R = \sum_{c \in C} r_c$. In this case, it is required that: (1) the demands in $C_1$ should be first satisfied; (2) the satisfaction rates of $C_2$, $C_3$, and $C_4$ should be in non-increasing order; (3) the satisfaction rate of $C_5$ can be zero if the masks are insufficient. According to these principles, we develop the following procedure that is adopted by the public health department to determine the rations for demand points as follows:

1. If $A$ is less than the total amount of demand in $C_1$, allocate all masks to points in $C_1$ with equal satisfaction rates among these points; otherwise, fully satisfy the demands of these highest-priority points;
2. If the remaining quantity is larger than the total amount of demand in $C_2 \cup C_3 \cup C_4$, fully satisfy the demands of these points; otherwise:
   (2.1) Tentatively allocate all remaining masks to these points with equal satisfaction rates;
   (2.1) Move a certain proportion $\rho_1$ of masks from $C_2$ to $C_2$ (the decrement is the same for each point in $C_1$ and the increment is the same for each point in $C_2$), and we use $\rho_1 = 25\%$ in our practice;
3. Allocate the remaining masks to all demand points in $C_3$ with equal satisfaction rates.

In the remainder of this paper, unless otherwise noted, we only consider demand points with actual ration in $C$, and consider $r_c$ to be the ration allocated to demand point $c$.

Representing the time at which the demand point $c$ receives the masks as $T(c)$, we adopt the objective function (1) which seeks to minimize the average weighted time of all demand points:

$$\min f = \frac{1}{|C|} \sum_{c \in C} w_c T(c)$$  \hspace{1cm} (1)

Although this problem seems to match well with the 2E-VRP, we do not utilize the standard 2E-VRP formulation or its variants because we aim to deliver masks to demand points as early as possible so as to control the spread of the epidemic, but the actual demands $r_c$ are known too late. To improve efficiency, we use machine learning to predict the demand of each region, and then deliver the predicted demands to regional facilities in advance. However, the predictions often have errors, and we need to adjust the delivery plan to compensate for the errors. Let $m_i$ be the number of depots, $m_2$ the number of satellites, $K_i$ the number of first-echelon vehicles, $n_i$ the number of demand points in the $i$th region, and $K_i$ the number of second-echelon vehicles in the $i$th region $(1 \leq i \leq m)$, the complexity of the 2E-VRP is $O(K(m_1 + m_2)(\sum_{i=1}^{m} K_i(1 + n_i)!))$. Based on prediction and pre-distribution, the original problem is decomposed into the following sub-problems:

- Reassignment of demand points among regions to handle prediction errors, the complexity of which is $O(m_i(m_i - m_i'))$, where $m_i$ is the number of regions with mask shortage (due to underestimation of the demands);
- Reallocation of masks from depots to supplement the regions with shortages (if exist), which is also a standard VRP with a complexity of $O(K(m_1 + m_2))$;
- Distribution of masks from each regional facility to its demand points, which is a standard VRP for a region without shortage, but is a VRP variant that involves two batches of goods (pre-distributed masks and supplemented masks) for a region with shortage. The complexity of the former is $O(K(1 + n_i))$, while that of the latter is $O(K(m_1 + m_1 + 1 + n_i))$.

The complexity of our method is the sum of the complexities of all subproblems, which is significantly lower than the original problem using multiplication.

### 4. Hybrid learning and optimization for mask delivery

The proposed hybrid learning and optimization method performs the following steps on each day (see Fig. 2):

- Use a machine learning method to predict the demand of each region for the next day;
- Route first-level vehicles to transport the predicted quantities of masks to regional facilities in advance;
- After obtaining the actual demand of each region, calculate the surplus/shortage of masks;
- Try to reassign demand points from regions with shortages to regions with surpluses;
- If there are still regions with shortages, transport masks from depots to these regions to supplement the shortages;
- Route second-level vehicles to deliver masks from each regional facility to its demand points.

Pre-distribution from depots to regional facilities, supplement from depots to regions with shortages, and distributions to demand points in regions without shortages are relatively small- or medium-size one-echelon VRPs that are independent of each other and can be efficiently (and simultaneously) solved using existing algorithms. Therefore, in the remainder of this section, we describe these steps briefly, and place emphasize on designing an efficient algorithm for reassigning demand points among different regions and an improved algorithm for distributing two batches of masks to demand points in a region with shortage.

#### 4.1. Machine learning for demand prediction

Because scheduling mask delivery after obtaining the actual demands is too late, our method first predicts the demand $\hat{R}_s$ of each region $s \in S$, such that we can pre-distribute the predicted demands to the regional facilities in advance to greatly shorten the delivery time. Nevertheless, the organization found that the demands of most regions changed violently, and it was difficult to predict the demands in a relatively accurate manner based on the managers’ experiences.

We believe that the demands change with certain rules affected by factors such as the population, epidemic situation, and historical distribution of masks in each region. Therefore, in this study, we consider 126 features (summarized in Table 1) that affect or probably affect the demand of medical masks in COVID-19. Many machine learning models can be used for this purpose. After testing a set of models (including multivariable linear regression, feedforward neural networks, fuzzy inference systems, etc.), we employ a fuzzy deep contractive autoencoder (FDCAE) [43]. In brief, an FDCAE is stacked layers of autoencoders [44], each of which consists of an encoder for mapping an input vector $x$ into a hidden representation $y$ and a decoder.
where \( \min_\mathcal{S} \) denotes the rooted mean square error:
\[
\hat{f}(x) = s(Wx + b) \\
\hat{f}'(y) = s(W^2y + b')
\]
where \( W \) is a \( D' \times D \) weight matrix, and \( b \) and \( b' \) are two bias vectors. To effectively handle the uncertain relationship between the inputs and outputs, FDCAE uses fuzzy parameters instead of crisp ones.

FDCAE learning consists of two stages. In the first stage, we perform unsupervised training of the autoencoders layer by layer to minimize the reconstruction error:
\[
\min_j \left( \| x, f'(f(x)) \| \right)
\]
where \( x \) is the training set and \( \| x, x' \| \) is the distance between \( x \) and \( x' \).

In the second stage, we perform supervised training of the whole FDCAE to minimize the rooted mean square error:
\[
\min \mathcal{L} = \sqrt{\frac{1}{|\mathcal{X}_l|} \sum_{x \in \mathcal{X}_l} \left( \frac{\| f'(x) - \hat{f}(x) \|}{2} \right)^2}
\]
where \( \mathcal{X}_l \) is the set of labeled training samples, \( \hat{f}(x) \) is the actual output of FDCAE on each input \( x \), and \( f(x) \) is the expected output (label) of \( x \). Please see [43] for more details on the structure and learning mechanisms of FDCAE.

### 4.2. Pre-distribution of predicted demands

Pre-distribution of predicted demands from depots to regional facilities is a VRP with a relatively small number of satellites. For example, during the peak period of COVID-19 in Hangzhou city, there were 13 regional facilities and three depots for mask distribution. In our study, when there are multiple depots, we employ a local search algorithm proposed by Wang et al. [45], which starts from a random feasible solution, uses a perturbed-based local search operator to explore the searching space, and employs six local search operators to exploit the neighborhood of each newly found best solution. When there is one depot, we employ a variable neighborhood search (VNS) algorithm from [46], which uses a constructive heuristic to generate an initial solution, and then uses skewed VNS to iteratively improve the current solution, utilizing the distance function in acceptance criteria phase to improve the exploration of faraway valleys. There are also many other heuristic and metaheuristic algorithms [47–49] that are competitive on instances of this scale.

### 4.3. Reassignment of demand points

Initial assignment of demand points to regional facilities is typically determined by the public health department based on simple principles such as administrative subordination or distance-based clustering. Let \( C_s \) denote the set of demand points initially assigned to each regional facility \( s \in \mathcal{S} \); the predicted demand \( \hat{R}_s \) hardly exactly meets the actual demand \( R_s = \sum_{c \in C_s} r_c \) of the region. We identify two subsets of regional facilities:
- \( S^+ = \{ s \in \mathcal{S} | \hat{R}_s > R_s \} \), i.e., regional facilities with mask surplus; for each \( s \in S^+ \), we calculate the surplus \( \theta(s) = \hat{R}_s - R_s \).
- \( S^- = \{ s \in \mathcal{S} | \hat{R}_s < R_s \} \), i.e., regional facilities with mask shortage; for each \( s \in S^- \), we calculate the shortage \( \theta(s) = R_s - \hat{R}_s \).

In most cases, using surplus masks from facilities in \( S^+ \) to supplement the shortages in \( S^- \) can be significantly more efficient than always using masks from depots to supplement the shortages (we do not consider reassignment of demand points among facilities in \( S^- \)), as doing so would significantly enlarge the solution space, although it might sometimes lead to better solutions). A reassignment solution can be represented by a set of triples, where each triple \( (c, s, s') \) denotes that a demand point \( c \) is reassigned from a facility \( s \in S^- \) to another \( s' \in S^+ \). The aim of reassignment can be considered as minimizing \( \sum_{c \in S^-} \theta(s) + (\sum_{c \in S^+} \theta(s)) \), the sum of absolute values of surpluses and shortages. Ideally, we expect that the sum is zero, i.e., both \( S^+ \) and \( S^- \) are empty. However, this case rarely happens.

For such a problem whose solutions have complex structures, VNS that adaptively utilizes a variety of local search operators to exploit these structures have a strong robust performance. For the reassignment subproblem, we propose a variable neighborhood tabu search (VNTS) heuristic, which uses a greedy constructive heuristic to produce an initial reassignment solution, and then combines VNS with tabu search (for avoiding revisiting recent solutions) to iteratively improve the current solution until the stop condition is met.

Intuitively, a demand point should not be assigned to a region too far away. To avoid such inefficient reassignment, we measure the distance between a demand point \( c \) and a region of \( s' \) in terms of \( r'(c, s') = \min_{s \in \mathcal{S}} \min_{c' \in C_s} r(c, c') \), the minimum travel time between \( c \) and another point \( c' \) among all points in the region, which we call the minimum linking time. We set a threshold \( r' \) on the linking time: if \( r'(c, s') \) is longer than the threshold, \( c \) cannot be reassigned to \( s' \).

The greedy constructive heuristic (the pseudo-code of which is shown in Algorithm 1) reassigns demand points using the following principles:

- Among all regions with shortages, regions with smaller shortages have higher priorities, such that the number of regions in \( S^- \) can be reduced as many as possible.
• Among all demand points in an \( s \in S^e \), points farther from \( s \) have higher priorities, such that the delivery time can be shortened as much as possible.

• When selecting an \( s' \in S^e \) to which a demand point \( c \) will be reassigned, regions with smaller \( t'(c, s') \) have higher priorities, such that the delivery time can be shortened as much as possible.

Algorithm 1: Greedy constructive heuristic to produce an initial reassignment solution.

1. Let \( X \) be an empty set;
2. Sort all \( s \in S^e \) in non-decreasing order of \( \theta(s) \);
3. foreach \( s \in S^e \) do
   4. Sort all \( c \in C \) in non-increasing order of \( \theta(s, c) \);
   5. foreach \( c \in C \) do
      6. Let \( S' = \{ s' \in S^e \mid \theta(s') \geq \theta(s, c), t'(c, s') \leq t'(c, s) \} \);
      7. If \( S' \) is empty then continue;
      8. Let \( s = \arg \min \{ t'(c, s') \} \);
      9. Move \( c \) from \( C_s \) to \( C_{s'} \) and add \( \langle c, s, s' \rangle \) to \( X \);
   10. \( \theta(s') = \min \{ \theta(s'), \theta(s) \} \); return \( \theta(s') = \theta(s) \); remove \( s' \) from \( S' \);
   11. \( \theta(s) = \theta(s') + r_s \); if \( \theta(s) \leq 0 \) then break;
   12. if \( S' \) is empty then break;
5. return \( X \).

Starting from the initial reassignment solution \( X \) obtained by the greedy constructive heuristic, we try to continually improve the solution using the following three local search operators:

LS1. Randomly interchange the reassignment of two points, i.e., select two triples \( \langle c_1, s_1, s'_1 \rangle \) and \( \langle c_2, s_2, s'_2 \rangle \) from \( X \), and change them to \( \langle c_1, s_2, s'_2 \rangle \) and \( \langle c_2, s_1, s'_1 \rangle \); suppose that \( r_{s_1} \geq r_{s_2} \), the precondityion for such an operation is \( \theta(s') \leq (r_{s_1} - r_{s_2}) \leq \theta(s'_2) \).

LS2. Randomly assign a demand point to another facility, i.e., select a triple \( \langle c, s, s' \rangle \) from \( X \) and another facility \( s'_2 \in S^e \), and change the triple to \( \langle c, s_2, s'_2 \rangle \); the precondityion for such an operation is \( \theta(s'_2) \leq \theta(s'_2) \).

LS3. Randomly replace a demand point in the reassignment, i.e., select a triple \( \langle c, s_1, s' \rangle \) from \( X \) and a point \( c' \) belonging to a facility \( s_2 \in S^e \) but not existing in \( X \), and change the triple to \( \langle c', s_2, s'_2 \rangle \); the precondityion for such an operation is \( \theta(s') + r_s \geq r_{c'} \).

The VNTS heurisic adaptively selects among the local search operators to try to find better neighboring solutions based on the past performance of the operators during the search, while avoiding circulation by forbidding recent operations recorded in the tabu list. Initially, the selection probability of each operator is the same; at each iteration, we use the selection probability of each operator to assess its performance during the search, while avoiding cireulation by forbidding recent operations recorded in the tabu list. Initially, \( \sum_{c \in C} \gamma(c) = 1 \), where \( \gamma(c) \) is the remaining shortage of facility \( c \) after reassignment using \( X \). A smaller value of \( f_j(X) \) indicates a better solution. On the right side of Eq. (6), the first item evaluates the number of remaining facilities with shortages, the reduction ratio of shortage of each remaining facility, and its distance to the closest depot; the second item evaluates the difference between the minimum linking time to the reassigned region and the original assigned region of each demand point. We use this surrogate for fast solution comparison in Algorithm 2.

4.4. Supplement the shortage

After reassignment, if there are still regions with shortages, we again schedule first-level vehicles to transport masks from depots to these regional facilities to supplement the shortages. This is normally a small or relatively small VRP instance. Like the pre-distribution subproblem in Section 4.2, here we also use the VNS algorithm [46] when there is one depot and use the local search algorithm [45] when there are multiple depots.

4.5. Second-level delivery in each region

For each region without shortage, delivery of masks from the regional facility to demand points is a single-depot VRP, for which we also employ the VNS algorithm [46]. For each region \( s \) with shortage (i.e., \( s \in S^e \)), the delivery task is a VRP variant where the masks are divided into the first-batch of pre-distributed masks available at time 0 and second-batch of supplementary masks available at a given time \( t_e \). For this two-batch VRP, we partition the \( C_s \) into two subsets \( C_s(1) \) and \( C_s(2) \), where the second batch of masks are only delivered to the second subset. Correspondingly, we can identify two sets of routes:

- Set \( R_s(1) \) of \( J \) routes \( \{ j_1, j_2, \ldots, j_{J_1} \} \), where each route only visits points in \( C_s(1) \), and the total demand of points in the route is not larger than \( Q_z \);
- Set \( R_s(2) \) of \( J \) routes \( \{ j_{J_1+1}, j_{J_1+2}, \ldots, j_{J_1+J_2} \} \), where each route visits zero or more points in \( C_s(1) \) and then visits at least one point in \( C_s(2) \), and the total demand of those points in \( C_s(1) \) is not larger than \( Q_z \).

For each route \( j_e \in R_s(1) \), the time at which the vehicle arrives at the first demand point \( j_{j_e+1} \) is \( T(j_{j_e+1}) = t_{j_e} + t_{j_e} \), and the arrival time at each remaining point is \( T(j_{j_e+1}) = T(j_{j_e}) + r(j_{j_e}) \).
For each route $\gamma_j \in R^{(2)}$, the vehicle needs to return to the regional facility $s$ to carry the second-batch masks. All points in $C^{(2)}_s$ must occur after the revisiting of $s$, but not all points in $C^{(1)}_s$ must occur before the revisiting of $s$ on the route, because it might be better for the vehicle to return to $s$ before visiting all points in $C^{(1)}_s$. Note that if the revisiting time is earlier than $t_r$, the vehicle has to wait at $s$ for the second batch; moreover, the amount of total demand of points after the revisiting cannot be larger than $Q_2$. It is not difficult to test all possible places for revisiting $s$ in the route, and find the place that has the minimum average weighted completion of all points. Hence, we can also obtain the time $T(y_{j,k})$ at which the vehicle arrives at each demand point $y_{j,k}$.

Let $y_{ij} = 0$ denote that demand point $c_i$ is assigned to $C^{(1)}$ and $y_{ij} = 1$ denote that $c_i$ is assigned to $C^{(2)}$ $(1 \leq i \leq n_i)$, $z_j = 1$ denote that route $\gamma_j$ is selected in the solution and $z_j = 0$ otherwise $(1 \leq j \leq J_1 + J_2)$; by extending the formulation by [50], we formulate the two-batch VRP as follows:

$$\text{min } f(y, z) = \frac{1}{n_j} \sum_{j=1}^{J_1+J_2} \sum_{k=1}^{n_j} w_{j,k} T(y_{j,k})$$  
(7)

subject to

$$\sum_{j=1}^{J_1+J_2} a_{ij} z_j = 1, \quad \forall c_i \in C_s$$  
(8)

$$\sum_{j=1}^{J_1+J_2} z_j \leq K_s'$$  
(9)

$$\theta(s) \leq \sum_{i=1}^{n_i} y_{ij} r_{ij} < \theta(s) + \min_{s \in S, n_s} r_{si}$$  
(10)

$$y_{ij} \in \{0, 1\}, \quad j = 1, 2, \ldots, n_s$$  
(11)

$$z_j \in \{0, 1\}, \quad j = 1, 2, \ldots, J_1 + J_2$$  
(12)

where $n_i$ is the number of demand points in route $\gamma_j$, $a_{ij}$ is a binary coefficient indicating whether $c_i$ is visited by $\gamma_j$, (8) requires that each demand point is covered by exactly one route, (9) requires that the number of selected routes cannot exceed the number of available vehicles; and (10) requires that the first-batch masks should be fully utilized before utilizing the second-batch.

To solve the two-batch VRP, we first produce an initial solution using a heuristic shown in Algorithm 3, which selects, among the set $\Gamma$, of all demand point subsets satisfying (10), the subset with the minimum total weight sum as $C^{(1)}_s$, uses VNS [46] to deliver the first batch of masks to $C^{(1)}_s$, and then schedules the vehicles to deliver the second batch of masks to $C^{(2)}_s$ by always assigning the point with the maximum weight to the vehicle with the earliest completion time until all points in $C^{(1)}_s$ have been handled (Lines 7–11 of Algorithm 3).

We then iteratively improve the current solution $(y, z)$ using VNTS, which has a framework similar to Algorithm 2 but uses the following six local search operators for potentially improving a solution to the problem; the first two operators are taken from [51], and the last four operators are proposed for tackling the two batches of delivery (as illustrated in Fig. 3):

- Single-route improvement over $C^{(1)}_s$, which first moves up a point while moving down another point in the route; if no improvement is obtained, swaps two points in the route; these points are limited in $C^{(1)}_s$.

- Multi-route improvement in $C^{(1)}_s$, which first moves a point from one route to another; if no improvement is obtained, swap two points between a pair of routes; if no improvement is obtained, exchange three points between a triple of routes; these points are limited in $C^{(1)}_s$.

- Single-route improvement in $C^{(2)}_s$, which swaps two points in $C^{(2)}_s$ in the route, if the route has at least two such points.

- Multi-route improvement in $C^{(2)}_s$, which removes a point in $C^{(2)}_s$ from one route, and then inserts it into another route as Line 10 of Algorithm 3.

### Algorithm 3: Heuristic to produce an initial solution to the second-level delivery problem.

1. Produce a partition $y$ by selecting $C^{(2)}_s = \min_{c \in C_s} (\sum \omega_c)$ and setting $C^{(1)}_s = C \backslash C^{(2)}_s$.
2. Use the existing algorithm [46] to solve the VRP instance for delivering the first batch of masks to $C^{(1)}_s$; let $z(y)$ be the time at which the vehicle route $z_j$ returns to $s$ in the resulting solution $z$ $(1 \leq j \leq J_1 + J_2)$.
3. If each $z(y)$ is not later than $t_r$, then use the algorithm [46] to solve the VRP instance for delivering the second batch of masks to $C^{(2)}_s$.
5. else
   6. Sort each $c \in C^{(2)}_s$ in non-increasing order of $\omega_c$.
   7. while $C^{(2)}_s$ is not empty do
      8. Let $c$ be the first point in $C^{(2)}_s$.
      9. Let $z$ be the vehicle route that has the earliest completion time.
      10. Insert $c$ into $z$ such that $z$ has the minimum $\sum_{c \in z} \omega_c T(c)$ among all possible insertions.
9. Remove $c$ from $C^{(2)}_s$.
12. return $(y, z)$.

### 5. Results and discussion

#### 5.1. Application results

During the peak of COVID-19, the Hangzhou Municipal Government began to arrange governmental procurement and distribution of medical masks since Feb 1, 2020. From Feb 1 to Feb 4, the manager used a fixed regional distribution method, i.e., the assignment of demand points to regions was fixed, and the first-echelon distribution from depots to regional facilities and the second-echelon delivery from each regional facility to its demand points were scheduled independently. As the first-echelon distribution could begin only after obtaining all actual demands, and second-echelon delivery could only begin after the region received masks from the depots, the overall delivery time was too long.

To improve the delivery efficiency, our hybrid learning and optimization method was applied since Feb 5. The machine learning model was first trained using the demand data from Jan 23 to Feb 4, and then re-trained using the data of each new day. The control parameters $N_{hSize}$ and $TabuLen$ of VNTS was also tuned on the instances constructed based on the data in this period, and the result was that both the parameter values were set to 10. The instance on Feb 5 had three depots, 13 regional facilities, and 1862 demand points. The distribution of which is shown in Fig. 4. The predicted demand and actual demand in each region are illustrated in Fig. 5. After the pre-distribution of predicted demands to regional facilities, there were seven regions with surpluses and six regions with shortages. We used the VNTS heuristic to realign demand points among the regions, the results of which are shown in Fig. 6. After realignment, there was only one region with shortage, for which we used the similar VNTS to route 12 available vehicles for mask delivery, the results of which are shown in Fig. 7.

The instances of Feb 4 and Feb 5 were similar (the latter was even more complex); the average weighted delivery time obtained by the old method on Feb 4 was 789 min; by applying our method on Feb 5, the time decreased to 333 min (as the average weight was around 2.7,
the average waiting time at the demand points decreased from around five hours to two hours). Such a significant time reduction prompted the manager to adopt our method for mask delivery until Feb 13 when sufficient numbers of masks could be bought offline or delivered by commercial express companies.

From Feb 1 to Feb 12, the number of demand points slightly changed between 1750–1960 in Hangzhou. To verify the performance, on each of the last eight days, we simulate the running of the old fixed regional scheduling method as well as the following two methods for comparison:

- An integrated two-echelon distribution method, which schedules the two echelons of distribution in an integrated manner after obtaining all actual demands (without demand prediction and pre-distribution); here we adapt a state-of-the-art multi-population GA [23] to solve the considered 2E-VRP aiming at optimizing the objective function (1).
A pre-distribution and integrated second-echelon distribution method, which also uses machine learning to predict the demands and pre-distributes the demands, but then solves the second-echelon distribution in all regions as an integrated multi-depot VRP (without point reassignment and two-batch delivery), regarding each regional facility with pre-distributed masks as a depot; here we employ the VNS algorithm [46] for the VRP.

To balance between the solution time and delivery time, after testing and tuning, we set the running times of the fixed regional distribution method, integrated two-echelon distribution method,
Swarm and Evolutionary Computation 76 (2023) 101208

X. Chen et al.

pre-distribution and integrated second-echelon distribution method, and our pre-distribution and reassigned distribution method to 2, 17, 19, and 5 min, respectively, which could achieve the best overall result on the eight instances for each method.

For these instances in Hangzhou city, Fig. 8 presents the objective function values, i.e., average weighted delivery time in minutes (where the time at which we obtain all actual demands is set as time zero), obtained by the four methods (line charts over the left vertical axis) and prediction accuracy of the machine learning approach (histograms over the right vertical axis). According to the results, the average weighted delivery time of the old fixed regional distribution method would be 850 min, which was over 2.5 times the 333 min obtained by our method. Moreover, with increasing number of days, the machine learning model predicted the demands more accurately; hence, the performance of our method over the old method increased. During the last four days, the average weighted delivery time of the old method was around three times that of our method. On the last day, the ratio reached 3.28.

On the instances of the last eight days, the integrated two-echelon distribution method exhibited the worst performance, mainly because the 2E-VRP instances with three depots, 13 satellites, and thousands of customers were too large, while the time at which we could use the algorithm to solve the instances was too late. Within such a limited computational time, the algorithm could only explore a very small portion of the solution space and, in most cases, only achieve a slight improvement over the initial greedy solution. As a result, its average weighted delivery time was over four times that of our method, which indicated that directly solving the 2E-VRP in an integrated manner was impractical for such a large-scale operation.

The third method exhibited better performance than the first and second, because it also utilized machine learning to enable the prediction and pre-distribution of demands, which consequently allowed the second-echelon distribution to begin much earlier. Nevertheless, VRP instances for integrated second-echelon distribution were also very large, for which we could not obtain high-quality solutions within the limited computational time. As a result, its average weighted delivery time was around 170% of that of our method.

In comparison, our method not only utilized machine learning to enable the pre-distribution and reduce the first-performance distribution time, but also solved VRP instances for different regions independently. Consequently, our method achieved significant performance improvement over the other three methods on each day. In particular, the improvement over the first two methods (without machine learning) grew with the increasing number of days, because the prediction accuracy improved with increasing data size.

In addition, to test the relationship between the prediction accuracy and delivery performance in our method in a more comprehensive way, on each instance from Feb 5 to Feb 12, we simulated demand predictions with accuracies ranging in [50, 60, 70, 75, 80, 85, 87.5, 90, 92.5, 95, 97.5, 100] percents, and used our method to solve the simulated instances under different prediction accuracies. As we can see from the results shown in Fig. 9, the average weighted delivery time under a prediction accuracy of 50% was around twice to three times that under a prediction accuracy of 90%. That is, when the prediction accuracy was low, pre-distribution of predicted demands could hardly help to improve the delivery performance; with the increase of the prediction accuracy, the average weighted delivery time decreased due to pre-distribution. However, when the prediction accuracy was over 90%, the average weighted delivery time decreased slowly, because the proposed demand point reassignment approach could effectively handle the small amount of surplus/shortage.

As our method exhibited a high delivery efficiency in Hangzhou in early February, it was adopted for emergency mask distribution in Nanjing from Feb 11 to Feb 16 and in Guangzhou from Feb 14 to Feb 20. Similarly, we compared the results of our method with the other three methods in these two cities in Figs. 10 and 11, respectively. In these two cities, our method also exhibited significantly better performance than the other three methods. On instances in Nanjing, the average weighted delivery time obtained by our method was 37%, 25%, and 67% of those obtained by the first three methods; on Feb 16, the quantity of masks as well as the number of demand points were much smaller, that is why the delivery time obtained by the third method was near to that obtained by our method. In Guangzhou, the numbers of regional facilities and demand points were both larger than those in Hangzhou and Nanjing, and hence the performance gap between our method and the integrated two-echelon distribution method was further enlarged; however, the average number of demand points in each region was smaller; hence, the performance gap between our method and the other two methods narrowed. On instances in Guangzhou, the average weighted delivery time obtained by our method was 45%, 16%, and 61% of those obtained by the first three methods.

5.2. Comparative test of VNTS for two-batch delivery

In this subsection, we test the performance of the proposed VNTS heuristic for the two-batch delivery subproblem, which is a key component of our machine learning and heuristic optimization method. For comparison, the following state-of-the-art algorithms for VRP with multiple time windows are adapted to the subproblem by considering the second-batch delivery as special time windows constraints:

Fig. 7. The routes of 12 second-echelon vehicles for two-batch mask delivery masks in region #12 in Hangzhou city, Feb 5. The two routes in green and blue used both batches of masks, while the other ten routes only used the first batch.
Fig. 8. Comparison of the four distribution methods and the prediction accuracy of machine learning in Hangzhou city.

Fig. 9. Estimated delivery performance under different (simulated) demand prediction accuracies.

Fig. 10. Comparison of the four distribution methods and the prediction accuracy of machine learning in Nanjing city.
practices as follows.

As we can see from the results, on instances 1 and 7, there is no statistically significant difference between the results of VNTS and MSGA; on instance 9, there is no statistically significant difference between the results of VNTS and LNS. Except these three cases, the result of VNTS is significantly better than that of each other algorithm on each instance. In summary, VNTS exhibits the best overall performance among the four algorithms on the test set [55]. At each instance, each algorithm is run for 30 times, and nonparametric Wilcoxon rank sum test is used to compare the statistical difference between our VNTS heuristic and each other comparative algorithm. Fig. 12 presents the comparative results, including the median, best (min), worst (max), first quartile (25%), and third quartile (75%) of the objective function values of each algorithm on each instance; a symbol ‘+’ above a box indicates that the result of VNTS is statistically significantly better than that of the corresponding algorithm (at a confidence level of 95%), and a symbol ‘=’ indicates that there is no statistically significant difference between them.

As we can see from the results, on instances 1 and 7, there is no statistically significant difference between the results of VNTS and MSGA; on instance 9, there is no statistically significant difference between the results of VNTS and LNS. Except these three cases, the result of VNTS is significantly better than each other algorithm on each instance. In summary, VNTS exhibits the best overall performance among the four algorithms on the test set, demonstrating that the adaptive integration of the six local search operators in our VNTS algorithmic framework is the most efficient in exploring the solution space of the problem. Particularly, we observe that, given the same number of demand points, if the number of points in \( C_{(2)} \) is larger, the performance advantage of VNTS over the other algorithms becomes more significant, which validates the effectiveness of VNTS’s local search operators designed for improving different kinds of routes in Section 4.5.

5.3. Key success factors and lessons learned

To facilitate the application or extension of the proposed method to more scenarios, e.g., emergency delivery of more medical supplies (such as disinfectants and vaccines) in more countries and regions, we summarize major key success factors and lessons learned from the practices as follows.

- **Effective demand prediction.** Demands for medical supplies for epidemic prevention and control depend on various factors, such as epidemic situation, demographics, public health, and historical demands. In early stage, due to the lack of historical data, we may use some empirical approaches to roughly estimate the demands. However, even with a small number of labeled samples, state-of-the-art deep learning models often provide more accurate predictions, because they can utilize sufficient unlabeled data in unsupervised pre-learning of the joint distribution of influence features. Before applying our FDCAE, we pre-trained the model with hundreds of unlabeled samples from other epidemic-affected cities and then trained it with labeled samples of the previous four days in Hangzhou, and obtained a prediction accuracy of 90.38% on the first day; the accuracy increased with the accumulation of labeled samples in applications; in Guangzhou city, the average prediction accuracy was around 94.56%, and the accuracy on the last day was close to 96%, which greatly facilitated the subsequent pre-distribution to improve the delivery efficiency.

- **Early pre-distribution.** In our practice, immediately after completing the first-echelon distribution for the current day, we began to schedule the pre-distribution for the next day. However, we should consider two situations. The first was that the current available supplies were not enough for pre-distribution; in this case, we also employed the rationing rules described in Section 3 to determine the rations for pre-distribution. The second was that the predicted demand of a region might be abnormally great. We should identify such (rare) cases and re-estimate the demands in a more conservative way.

- **Fast and agile demand point reassignment.** We should compute the reassignment solution immediately after obtaining actual demands. In our practice, the computational time for reassignment was limited to no more than two minutes. However, the reassignment solution might not always be accepted by each region, mainly because regional drivers might be unfamiliar with or inefficient on the roads to some newly assigned demand points. In such rare cases, we had to manually adjust the reassignment scheme. We found that a small threshold \( r^7 \) helped reduce or avoid such cases, but a too-small \( r^7 \) might also severely limit the efficiency of reassignment. In our practice, we set the threshold to 90, 105, and 135 min, respectively.

- **Communication with the manager.** To convince the manager to adopt the proposed method for the first time, we must exhibit the expected performance compared to the old method. On the instance of the first day in Hangzhou, we calculated the expected average delivery time obtained by our method as 40% of that.
obtained by the old method, and presented the average delivery
time of each group of demand points (in each region and with
each importance weight) to the manager, who, after a careful
assessment, decided to adopt the new solution. After application
on the first day, the actual average delivery time obtained by
our method was around 42% of that obtained by the old method
(the increase was mainly because some drivers were not familiar
with some newly reassigned demand points). Given the success in
Hangzhou, the managers of the two other cities were more active
in adopting our new method. However, not all of managers were
inclined to adopt the new method, for example, in some small
and medium-sized cities, the demands were not great, and the
performance improvement would not be so significant. Typically,
if the average delivery time obtained by the new method was over
80% of that obtained by the old method, the manager would not
like to change the method.

6. Conclusion and discussion

In this paper, we propose a new two-echelon delivery system based
on hybrid machine learning and heuristic optimization to improve the
solution quality and efficiency. The method pre-distributes predicted
demands from depots to regional facilities, reassigns demand points
among different regions to balance the deviations of predicted demands
from actual demands, and finally efficiently delivers masks to the
demand points in each region. The key contributions include an effi-
cient heuristic algorithm for reassigning demand points among different
regions and an improved heuristic algorithm for two-batch delivery
in a region with shortage. Application of the proposed method to
emergency medical mask delivery in three megacities in China during
the peak of COVID-19 demonstrate the performance advantages of the
proposed method. Analysis of the success factors and lessons learned
from the practices enables to apply or extend our method to more
emergency delivery problems (such as vaccine distribution and relief good delivery [56]) in more countries and regions. The presented work still has several limitations. First, we do not consider refueling or recharging of the vehicles; in practice we find that electric vehicles are commonly used in short- and medium-distance delivery, and hence we plan to incorporate battery recharging and/or swapping into the problem [57]. Second, the solution time of the VNTS heuristic is still not fast enough, and we are currently studying dynamic online routing which allows new demands being received during the working progress, using neural combinatorial optimization with reinforcement learning [58] to enable fast response to emergencies like COVID-19. In addition, our future research will consider the collaboration of ground vehicles with drones to improve delivery speed as well as reduce human contact [59].

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.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work was supported by National Natural Science Foundation (Grant No. 61872123) and Zhejiang Provincial Natural Science Foundation, (Grant No. LR20F030002) of China.

References

[1] S. Anily, A. Federgruen, Two-echelon distribution systems with vehicle routing costs and central inventories, Oper. Res. 41 (1) (1993) 37–47, http://dx.doi.org/10.1287/opre.41.1.37.
[2] B. Niu, L. Tan, J. Liu, J. Liu, W. Yi, H. Wang, Cooperative bacterial foraging optimization method for multi-objective multi-echelon supply chain optimization problem, Swarm Evol. Comput. 49 (2019) 87–101, http://dx.doi.org/10.1016/j.sweco.2019.05.003.
[3] J. González-Feliu, G. Perboli, R. Tadei, D. Vigo, The Two-Echelon Capacitated Vehicle Routing Problem, Techreport DEIS OR.INGCE 2007/2, Department of Electronics, Computer Science, and Systems, University of Bologna, 2007.
[4] Y.-J. Zheng, S.-Y. Chen, H.-F. Ling, Evolutionary optimization for disaster relief operations: A survey, Appl. Soft Comput. 27 (2015) 553–566, http://dx.doi.org/10.1016/j.asoc.2014.09.041.
[5] G. Perboli, R. Tadei, R. Tadei, New families of valid inequalities for the two-echelon vehicle routing problem, Electron. Not. Discrete Math. 36 (2010) 639–646, http://dx.doi.org/10.1016/j.endm.2010.05.081, iSCO 2010 - International Symposium on Combinatorial Optimization.
[6] G. Perboli, R. Tadei, D. Vigo, The two-echelon capacitated vehicle routing problem: Models and math-heuristic, Transp. Sci. 45 (3) (2011) 364–380, http://dx.doi.org/10.1287/trsc.1110.0369.
[7] G. Perboli, R. Tadei, E. Fadda, New valid inequalities for the two-echelon capacitated vehicle routing problem, Electron. Not. Discrete Math. 64 (2018) 75–84, http://dx.doi.org/10.1016/j.endm.2018.01.009, 8th International Network Optimization Conference - INOC 2017.
[8] R. Baldacci, A. Mingozzi, R. Robiotti, R.W. Calvo, An exact algorithm for the two-echelon capacitated vehicle routing problem, Oper. Res. 61 (2) (2013) 298–314, http://dx.doi.org/10.1287/opre.1120.1153.
[9] K. Venkata Narasimha, E. Kivelevitch, B. Sharma, M. Kumar, An ant colony optimization technique for solving min-max multi-depot vehicle routing problem, Swarm Evol. Comput. 13 (2013) 63–73, http://dx.doi.org/10.1016/j.swevo.2013.05.005.

CRediT authorship contribution statement

Xin Chen: Software, Writing – original draft. Hong-Fang Yang: Data Curation, Validation. Yu-Jun Zheng: Conceptualization, Methodology, Funding acquisition. Muntaz Karatas: Resources, Writing – review & editing.

Swarm and Evolutionary Computation 76 (2023) 101208
