Evolutionary Optimization of COVID-19 Vaccine Distribution With Evolutionary Demands

Yu-Jun Zheng, *Senior Member, IEEE*, Xin Chen, Qin Song, Jun Yang, and Ling Wang

***Abstract*—Vaccination uptake has become the key factor that will determine our success in containing the coronavirus pneumonia (COVID-19) pandemic. Efficient distribution of vaccines to inoculation spots is crucial to curtailing the spread of the novel COVID-19 pandemic. Normally, in a big city, a huge number of vaccines need to be transported from central depot(s) through a set of satellites to widely scattered inoculation spots by special-purpose vehicles every day. Such a large two-echelon vehicle routing problem is computationally difficult. Moreover, the demands for vaccines evolve with the epidemic spread over time, and the actual demands are hard to determine early and exactly, which not only increases the problem difficulty but also prolongs the distribution time. Based on our practical experience of COVID-19 vaccine distribution in China, we present a hybrid machine learning and evolutionary computation method, which first uses a fuzzy deep learning model to forecast the demands for vaccines for each next day, such that we can predistribute the forecasted number of vaccines to the satellites in advance; after obtaining the actual demands, it uses an evolutionary algorithm (EA) to route vehicles to distribute vaccines from the satellites/depots to the inoculation spots on each day. The EA saves historical problem instances and their high-quality solutions in a knowledge base, so as to capture inherent relationship between evolving problem inputs to solutions; when solving a new problem instance on each day, the EA utilizes historical solutions that perform well on the similar instances to improve initial solution quality and, hence, accelerate convergence. Computational results on real-world instances of vaccine distribution demonstrate that the proposed method can produce solutions with significantly shorter distribution time compared to state-of-the-arts and, hence, contribute to accelerating the achievement of herd immunity.**

***Index Terms*—Evolutionary optimization, hybrid machine learning and evolutionary computation, novel coronavirus pneumonia (COVID-19), vaccine distribution, vehicle routing.**

I. Introduction

THE NOVEL coronavirus pneumonia (COVID-19), caused by severe acute respiratory syndrome corona-virus 2 (SARS-CoV-2), has infected over 400 million people (by February 2022) and caused significant global social and economic distress. With slowly increasing population immunity and evolutionary selection pressure on the virus, new and highly virulent strains of SARS-CoV-2 are emerging, which could quickly exacerbate the crisis [1]. According to the currently available epidemiological data [2], vaccination uptake has become the key factor that will determine our success in containing the COVID-19 pandemic currently enveloping the world [3]. Therefore, distributing available vaccines to inoculation spots in an efficient and accurate way plays an important role in achieving herd immunity and breaking the chain of transmission of the virus.

The motivation of this article comes from our practice in COVID-19 vaccine distribution in Hangzhou and other cities of Zhejiang Province, East China. During the peak period of vaccination in Hangzhou city, the daily average number of vaccinations exceeds 100 000. Available (newly produced and purchased) vaccines are stored in one or several central depots that are typically located in the center of the city, while inoculation spots are widely scattered throughout the city, most of them being far away from the depots. Therefore, the public health department employs a two-echelon distribution approach that first delivers the vaccines from the depot(s) to a set of satellites (regional facilities), each of which is then responsible for distribution to inoculation spots (customers) in a region by special-purpose vehicles on each day. Such a two-echelon distribution approach can be significantly more efficient than direct distribution from the depot(s) to all customers [4]. However, it still has the following difficulties.

1) The corresponding two-echelon vehicle routing problem (2E-VRP) [5] is an NP-hard problem, and a large problem instance with over ten satellites and hundreds of customers is impossible or very difficult to solve using existing exact or heuristic optimization methods.
2) As the vaccine demand of each spot changes every day, the optimal routing solution for the distribution from depot(s) to satellites and that for the distribution from each satellite to customers also change; therefore, we need to solve a new problem instance every day.
3) The exact demands are hard to determine in an accurate and early manner, while vaccinations are expected to start as early as possible (typically, no later than

Manuscript received 7 August 2021; revised 25 November 2021 and 13 February 2022; accepted 22 March 2022. Date of publication 1 April 2022; date of current version 31 January 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61872123, and in part by the Natural Science Foundation of Zhejiang Province under Grant LR20F030002. *(Corresponding author: Yu-Jun Zheng.)*

Yu-Jun Zheng, Xin Chen, and Qin Song are with the School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China (e-mail: yujun.zheng@computer.org).

Jun Yang is with the College of Public Health, Zhejiang University, Hangzhou 310058, China (e-mail: gastate@zju.edu.cn).

Ling Wang is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: wangling@tsinghua.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TEVC.2022.3164260.

Digital Object Identifier 10.1109/TEVC.2022.3164260

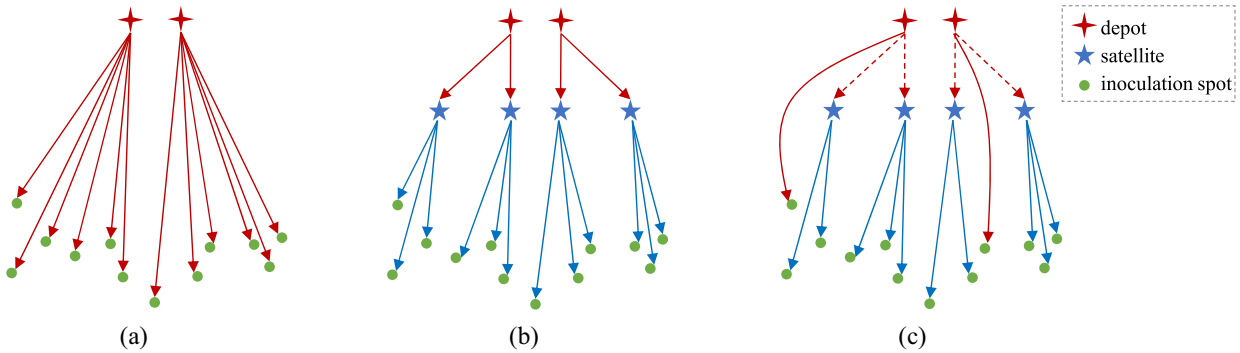

Fig. 1. Comparison of the direct distribution approach, the basic two-echelon distribution approach, and our approach combining demand forecasting, predistribution, and fast distribution. In (c), red arrows denote predistribution of forecasted demands, which can begin much earlier than the final distribution. (a) Direct distribution from depots to inoculation spots. *Challenges:* 1) high complexity and 2) demands are known late, and the delivery begins late. (b) Two-echelon distribution. *Advantages:* Reduced complexity. *Challenges:* demands are known late, and the delivery begins late. (c) Predistribution and two-echelon distribution. *Advantages:* Reduced complexity & early delivery; *Challenges:* Demand forecasting

9:30 A.M.); hence, the time for computing and implementing the distribution solution is very limited.

To address the above difficulties, we propose a hybrid machine learning and evolutionary computation method, which consists of the following steps to plan the vaccine distribution on each day.

1) Use a machine learning approach to forecast the demand of each inoculation spot for the next day.
2) Route vehicles to predistribute the forecasted number of vaccines to the satellite of each region in advance, so as to significantly shorten the actual distribution time for the next day.
3) After obtaining the exact demands, route vehicles to distribute vaccines from the satellites/depots to inoculation spots.

For the first-step task, we propose a fuzzy deep neural network to forecast the demand. The second-step task can be regarded as a basic and relatively small VRP instance that can be efficiently solved using the existing algorithms. The third-step task, however, is a large VRP instance, for which we propose an evolutionary algorithm (EA) that utilizes historical knowledge of vaccine distribution in early days to improve the problem-solving performance on new instances, that is, when initializing a population of solutions to a new problem instance on each day, the EA selects historical solutions to similar instances from the knowledge base, and then adapts these solutions to the new instance to improve initial solution quality. In this way, initial solutions to different instances are continually evolved according to their inputs (demands), and final solutions are obtained by the EA continually evolving the corresponding initial solutions. We find that this strategy can effectively accelerate the convergence and improve the final solution quality. Consequently, our hybrid machine learning and evolutionary computation method has significantly improved the efficiency of vaccine distribution in practice. Fig. 1 illustrates three solution approaches and highlights the advantages of our approach.

The main contributions of this article can be summarized as follows.

1) We introduce machine learning to forecast vaccine demands, so as to enable predistribution and reduce the complexity of the large-scale, complex two-echelon vaccine distribution problem.
2) We utilize historical knowledge of vaccine distribution obtained by the EA in early days to improve the problem-solving performance over time.
3) We apply the proposed hybrid machine learning and evolutionary computation method to real-world vaccine distribution, which significantly improves the effectiveness of epidemic prevention and control.

The proposed method can also be extended to many similar problems, such as medical mask distribution and scheduling of other nonpharmaceutical interventions in epidemics [6] and relief goods distribution in disasters.

The remainder of this article is organized as follows. Section II discusses related work on solution methods for medical supply distribution and, in particular, 2E-VRP. Section III describes the machine learning approach for forecasting the demands for vaccines, Section IV formulates the vaccine distribution problem after forecasting and predistribution, and Section V proposes the EA for solving the problem. Section VI presents the computational results, and Section VII concludes with a discussion.

II. Related Work

The distribution of medical supplies to customers timely and effectively plays a crucial role in response to large-scale emergency events, such as natural disasters and epidemics. In the literature, a medical supply chain is typically modeled as a complex network consisting of many different parties at various stages [7]. Compared to ordinary supply chains, medical supply chains have specific characteristics, including rigorous time constraints and special transportation and storage conditions [8]. Mete and Zabinsky [9] proposed a stochastic optimization method for the storage and distribution of medical supplies under various disaster types and magnitudes. The resulting solutions can suggest loading and routing of vehicles to transport medical supplies. Lei *et al.* [10] studied a problem of personnel scheduling and supplied provisioning in emergency relief operations; they proposed a mathematical programming-based rolling

horizon heuristic for finding near-optimal solutions. Liu and Zhang [11] proposed a dynamic medical logistics model that couples medical demand forecasting and logistics planning to satisfy the demand and minimize the total cost, characterizing decision making at various levels of hospitals, distribution centers, pharmaceutical plants, and the transportation in between them. Büyüktahtakın *et al.* [12] presented a mixed-integer programming epidemics-logistics model for determining the optimal amount, timing, and location of medical resources to minimize the total number of infections and fatalities under a limited budget, which was validated on the case of the 2014–2015 Ebola outbreak in Africa. The model was modified by Liu *et al.* [13] by adapting capacity constraints, and applied to resource planning in the 2009 H1N1 outbreak in China.

In recent years, EAs have been increasingly used to solve large complex medical supply distribution problems. Arabzad *et al.* [14] proposed a multiobjective EA (MOEA) combining the nondominated sorting genetic algorithm (NSGA-II) [15] and parallel neighborhood search to solve a multiobjective location-inventory problem in a distribution center network with the presence of different transportation modes and third-party logistics, simultaneously considering three conflicting objectives including total costs, earliness and tardiness, and deterioration rate. Gan and Liu [16] presented a multiobjective optimization problem considering both the total unsatisfied time and transportation cost in emergency logistics scheduling, and proposed a modified NSGA-II by designing three repair operators to generate improved feasible solutions to the problem. To solve a multiperiod dynamic emergency resource scheduling problem, Zhou *et al.* [17] proposed an MOEA based on decomposition with new evolutionary operators designed according to the intrinsic properties of the problem. Osaba *et al.* [18] modeled a drugs distribution problem with pharmacological waste collection as a clustered VRP with pickups and deliveries, asymmetric variable costs, forbidden roads, and cost constraints; they proposed a discrete bat algorithm, where differences among the bats are calculated based on two different neighborhood structures. To improve the prevention and control of COVID-19, Ling *et al.* [19] studied a problem of integrated civilian-military scheduling of medical supplies, and they proposed an MOEA based on water wave optimization (WWO) metaheuristic [20] to efficiently solve the problem. In order to design an integrated production-distribution-inventory-allocation-location medical supply chain during COVID-19, Goodarzian *et al.* [21] proposed three hybrid metaheuristics: 1) ant colony optimization; 2) fish swarm algorithm; and 3) firefly algorithm, all hybridized with a variable neighborhood search.

In the COVID-19 pandemic, the distribution of emergency medical supplies in a big city normally uses a multiechelon approach. As an extension of the basic VRP, 2E-VRP [5] has a significantly larger solution space than its single-echelon counterpart, and existing exact algorithms are only capable to solve instances with up to 3–6 satellites and tens of customers [22]–[24]. Recently, increasing efforts have been devoted to metaheuristics and EAs for finding near-optimal solutions to large 2E-VRP instances within an acceptable solution time. Hemmelmayr *et al.* [25] proposed an adaptive large neighborhood search (ALNS) heuristic for 2E-VRP by adapting operators to the problem structure. Breunig *et al.* [26] proposed a hybrid metaheuristic combining enumerative local searches with destroy-and-repair heuristics. Grangier *et al.* [27] proposed an ALNS algorithm for a 2E-VRP with satellite synchronization, which is capable of solving instances with up to 200 customers and ten satellites. The ALNS metaheuristic was also used by Enthoven *et al.* [28] to solve a 2E-VRP with covering options and by Li *et al.* [29] for a 2E-VRP with satellite bisynchronization.

To solve a 2E-CVRP with stochastic demands, Wang *et al.* [30] 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.* [31] proposed a hybrid multipopulation GA for a multidepot 2E-VRP, which exhibited good performance on a large family of instances. Yan *et al.* [32] proposed a graph-based fuzzy EA that integrates a graph-based fuzzy satellite-to-customer assignment scheme into an iteratively evolutionary learning process to minimize the total cost of 2E-VRP. Anderluh *et al.* [33] studied a multiobjective 2E-VRP, considering not only the cost but also negative external effects, such as emissions and disturbance; they proposed a metaheuristic that combines a large neighborhood with an ϵ-constraint method to approximate the set of Pareto-optimal solutions to the problem. To solve a real-time 2E-VRP with pickup and delivery that needs to be solved within seconds, Martins *et al.* [34] 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.

There are also many studies on predicting the demands of vaccines as well as other medical supplies, typically using statistical regression methods and neural network models [35]–[37]. However, demand forecasting in public health emergencies like COVID-19 is challenging, mainly because there are no sufficient historical training data. Therefore, some research efforts have been devoted to recent machine learning methods, including unsupervised deep learning, transfer learning, and multitask learning, which requires few or small number of training samples. To support drug procurement in hospitals, Song *et al.* [38] proposed a deep learning model to predict disease morbidities from big data, and then estimated the demands of different drug and determine their optimal combination. Song *et al.* [39] proposed a new tridirectional transfer learning method for predicting the gastric cancer morbidity based on an existing model for predicting the morbidity of another disease in another region by fusing two different directions of transfer learning, which achieves a significantly higher prediction accuracy compared with the state-of-the-art bidirectional transfer learning methods. Yong *et al.* [40] proposed a long short-term memory

(LSTM)-based deep learning model, which can map the input series to a new vector space effectively, to forecast demand for vaccines, the data of which are recorded in the blockchain and used for vaccine production plans. To forecast the ambulance demand for supporting rational and dynamic allocation of ambulances and hospital staffing in Singapore, Lin *et al.* [41] combined several models including LSTM, support vector machine (SVM), and convolutional neural network, to perform prediction based on multinature insights of ambulance demands. Zheng *et al.* [42] presented a co-evolutionary fuzzy deep transfer learning (CoFDTL) method for forecasting the demand of different relief supplies by sharing knowledge among different tasks (e.g., types of disasters, such as earthquake, typhoon, and flood), which effectively overcomes the shortage of data in each task. The fuzzy deep learning model combing unsupervised denoising training and supervised tuning has demonstrated its performance on various learning and classification problems [42]–[45].

III. Fuzzy Machine Learning for Demand Forecasting

On each day, the demands for vaccines come from three main sources: 1) people that have made appointments for vaccination; 2) people that have been assigned with emergency tasks (e.g., medical first aid, visiting epidemic areas, and other high-risk tasks) that need vaccine protection; and 3) people that directly go to the inoculation spots without appointments. The deadline for appointments is 21:00 of the previous day; people without appointments are only admitted before 12:00 of each day.

These demands change on each day. We aim to forecast the demand of each inoculation spot at around 18:00 of each previous day, such that the forecasted number of vaccines can be predistributed to the corresponding satellite in the previous night. Therefore, actually, we need to forecast the number of people who would make appointments during 18:00–21:00, who would be assigned with emergency tasks after 18:00, and who would directly go to the inoculation spot without appointments. We consider the following affecting features.

1) Number of people who have made appointments before 18:00 on the current day.
2) Number of people who have been assigned with emergency tasks that need vaccine protection before 18:00 on the current day.
3) Number of people who have made appointments before 18:00 on each of the previous seven days.
4) Number of people who have made appointments after 18:00 on each of the previous seven days.
5) Number of people who have been assigned with emergency tasks that need vaccine protection before 18:00 on each of the previous seven days.
6) Number of people who have been assigned with emergency tasks that need vaccine protection after 18:00 on each of the previous seven days.
7) Number of people who directly go to the inoculation spot(s) without appointments on each of the previous seven days.
8) Number of residents.
9) Numbers of residents in 0–18, 18–30, 30–50, 50–70, and over 70 years old.
10) Numbers of residents in 0–18, 18–30, 30–50, 50–70, and over 70 years old that have been vaccinated the first dose.
11) Numbers of residents in 0–18, 18–30, 30–50, 50–70, and over 70 years old that have been vaccinated the second dose.
12) Total infection rates (since the outbreak of the epidemic).
13) Current infection rates.

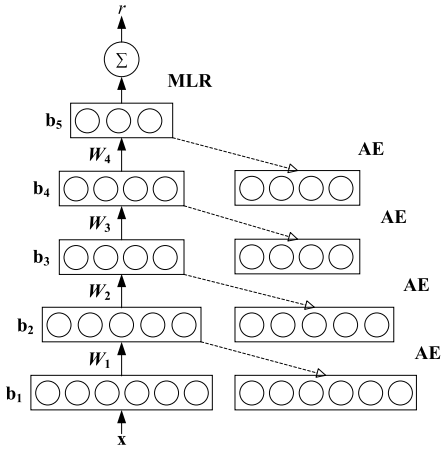

Fig. 2. Structure of the deep learning model, which consists of stacked layers of autoencoders and a regression layer on top of the uppermost autoencoder.

For each of the 55 underlying features, we use the values in the subregion assigned to the current inoculation spot, the whole region assigned to the corresponding satellite, and the whole city. Therefore, an input vector to the forecasting model has $55 \times 3 = 165$ values. We use a fuzzy machine learning forecasting model, which consists of four layers of autoencoders [46] and a multivariate linear regression (MLR) layer on the topmost autoencoder to produce the forecasted demands, including the total number of vaccines demanded and the number of vaccines for people without appointments, as shown in Fig. 2. Each autoencoder consists of an encoder and a decoder. Given a D-dimensional input vector $\mathbf{x}$, the encoder maps it into a D'-dimensional hidden representation $\mathbf{y}$ through a sigmoid activation function s

$$f(\mathbf{x}) = s(W\mathbf{x} + \mathbf{b}) \tag{1}$$

and the decoder maps the hidden representation $\mathbf{y}$ back to a reconstructed vector $\mathbf{x}'$ in the input space

$$f'(\mathbf{y}) = s\left(W^{\mathrm{T}}\mathbf{y} + \mathbf{b}'\right) \tag{2}$$

where W is a $D' \times D$ weight matrix, and $\mathbf{b}$ and $\mathbf{b}'$ are two bias vectors.

In order to effectively learn the uncertain probability distribution over cross-layer units, we use fuzzy model parameters, where each parameter β is an interval-valued Pythagorean fuzzy number (PFN) [47]

$$\beta = \left\langle \left[\mu_\beta^L, \mu_\beta^U\right]; \left[\nu_\beta^L, \nu_\beta^U\right] \right\rangle$$

where $[\mu_\beta^L, \mu_\beta^U]$ and $[\nu_\beta^L, \nu_\beta^U]$ are the membership and nonmembership degrees, respectively, satisfying that $[\mu_\beta^L, \mu_\beta^U] \in [0, 1]$, $[\nu_\beta^L, \nu_\beta^U] \in [0, 1]$, and $(\mu_\beta^U)^2 + (\nu_\beta^U)^2 \leq 1$. This type of fuzzy parameters enables the model to learn both how an input *contributes to* and how it *does not contribute to* the production of the output [48].

The model learning consists of two stages. The first stage is unsupervised training of autoencoders layer by layer, at each layer learning the parameters to minimize the reconstruction error on the training set $\mathcal{X}$

$$\min \mathcal{J}(W, \mathbf{b}, \mathbf{b}') = \sum_{\mathbf{x}\in\mathcal{X}} \Big(\|\mathbf{x}, f'(f(\mathbf{x}))\| + \lambda \|J_f(\mathbf{x})\|_F^2 \Big) \quad (3)$$

where $\|\mathbf{x}, \mathbf{x}'\|$ denotes the distance between each input vector $\mathbf{x}$ and its corresponding reconstructed vector $\mathbf{x}'$, λ is a parameter controlling the strength of penalization, and $\|J_f(\mathbf{x})\|_F^2$ is the Frobenius norm of the Jacobian to penalize the sensitivity of the hidden representation to the input noise [49]

$$\|J_f(\mathbf{x})\|_F^2 = \sum_{1\leq d\leq D} \sum_{1\leq d'\leq D'} \left(\frac{\partial f_{d'}(\mathbf{x})}{\partial x_d} \right)^2. \quad (4)$$

The second stage is supervised training of the whole model to minimize the rooted mean square error (RMSE) on the labeled training set $\mathcal{X}'$

$$\min \mathcal{L} = \frac{1}{|\mathcal{X}'|} \sum_{\mathbf{x}\in\mathcal{X}'} \sqrt{(E(r_\mathbf{x}) - \widehat{r}_\mathbf{x})^2 + w' \big(E(r'_\mathbf{x}) - \widehat{r}'_\mathbf{x}\big)^2} \quad (5)$$

where $\widehat{r}_\mathbf{x}$ and $\widehat{r}'_\mathbf{x}$ are the labels of total demand and the demand for people without appointments in sample $\mathbf{x}$, and $E(r_\mathbf{x})$ and $E(r'_\mathbf{x})$ are the expected values the corresponding outputs of the model, respectively.

We employ the Hessian-free (HF) algorithm [50] for unsupervised training of each layer and use an EA [45] for supervised training of the whole model.

IV. Problem of Two-Echelon Vaccine Distribution

After forecasting the demand of each region, we predistribute the forecasted number of vaccines from depot(s) to the satellite in advance for the next day. In case that the total forecasted number exceeds the total available stock at the depots, we can purchase vaccines from the market or mobilize vaccines from other cities. However, if these vaccines cannot arrive before the next day, the number of vaccines to be distributed must be decreased (in the same ratio for each region, or in different ratios determined by the public health department). Moreover, the public health department can take an order-up-to-level policy to replenish more vaccines (if available) to avoid possible shortages in later days, but discussion of this policy is out of the scope of this article. Anyway, the problem of predistribution is a basic VRP of relatively small size, for which we employ a neighborhood search algorithm [51] to efficiently solve it. Of course, many other heuristics and EAs [52]–[54] can be alternatives.

Here, we focus on vaccine distribution after demand forecasting learning and predistribution. Typically, during 0:00–2:00, we obtain the actual number of vaccines for people with appointments and people with emergency tasks that need vaccine protection for each inoculation spot. We consider the sum of this number and the forecasted number of vaccines for people who would directly go to the inoculation spot without appointments as the actual demand of the inoculation spot. If the number of vaccines available at each satellite is sufficient for the total demand of all inoculation spots in the region, the problem is to distribute vaccines from each satellite to the inoculation spots in the region, which is also a relatively small VRP instance that can be efficiently solved. However, in most cases, there are errors between the forecasted demand and the actual demand, resulting in that some regions are with vaccine shortages while other regions are with vaccine surpluses. In such cases, the problem is to distribute vaccines from satellites/depots to all inoculation spots, i.e., all satellites are regarded as depots. Consequently, the problem is a multidepot VRP that can be formulated as follows (Table I lists the main symbols used in the problem formulation).

TABLE I
Mathematical Variables Used in the Vaccine Distribution Problem

Variable	Description
O	Set of inoculation spots
r_o	Number of vaccines demanded by inoculation spot o ($o \in O$)
D	Set of depots
S	Set of satellites
a_d	Number of vaccines available at depot d ($d \in D$)
a_s	Number of vaccines available at satellite s ($s \in S$)
V_d	Set of vehicles stored available at depot d ($d \in D$)
V_s	Set of vehicles stored available at satellite s ($s \in S$)
C_v	Capacity of vehicle v ($v \in \bigcap_{d\in D} V_d$)
$t(p_1, p_2)$	Travel time between pair (p_1, p_2) ($\forall p_1, p_2 \in O \cup S \cup D$)
$T(o)$	Time at which the inoculation spot o receives vaccines ($o \in O$)
$\widehat{T}$	Deadline for all inoculation spots to start vaccination
O_s	Set of inoculation spots assigned to satellite s ($s \in S$)
m	Number of all vehicles
$\mathbf{x}_i$	Route of vehicle v_i ($1 \leq i \leq m$)
s_i	Depot or satellite from which vehicle v_i departs ($1 \leq i \leq m$)
n_i	Length of the route $\mathbf{x}_i$ ($1 \leq i \leq m$)
$\mathbf{x}_{i,j}$	j-th inoculation spot in the route $\mathbf{x}_i$

Let O be the set of inoculation spots, r_o be the actual vaccine demand of each spot $o \in O$, D be the set of depots, S be the set of satellites, a_d and a_s be the number of vaccines available at each depot $d \in D$ and each satellite $s \in S$, and V_d and V_s be the vehicle set available at each depot $d \in D$ and each satellite $s \in S$, respectively. The travel time between each pair of points p_1 and p_2 is assumed to be known and is denoted as $t(p_1, p_2)$ ($\forall p_1, p_2 \in O \cup D \cup S$). Let V be the set of all m vehicles, and C_v be the capacity of each vehicle $v \in V$ (as we consider heterogeneous vehicles). The problem is to determine the route $\mathbf{x}_i$ for each vehicle $v_i \in V$ (i.e., $\mathbf{x}_i$ is a sequence of inoculation spots assigned to v_i, $1 \leq i \leq m$), such that the vaccine-number-weighted accumulative distribution time is minimized

$$\min f(X) = \frac{\sum_{o\in O} r_o T(o)}{\sum_{o\in O} r_o} \quad (6)$$

where $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$ is the solution vector, $T(o)$ is the time at which spot o receives vaccines, r_o is used as the importance weight of spot o such that spots with larger demands

are expected to receive vaccines (and, hence, start vaccination) earlier, and the constant denominator $\sum_{o\in O} r_o$ is used to make the objective function represent the average weighted distribution time over all spots. Let s_i denote the depot or satellite from which vehicle v_i departs and n_i denote the length of route $\mathbf{x}_i$; $T(o)$ in each route $\mathbf{x}_i$ can be iteratively calculated as follows ($1 \le i \le m$):

$$T(x_{i,1}) = t\big(s_i, x_{i,1}\big) \tag{7}$$

$$T(x_{i,j+1}) = T(x_{i,j}) + t\big(x_{i,j}, x_{i,j+1}\big), \quad 1 \le j < n_i. \tag{8}$$

Here, we regard loading/unloading times as constants and do not explicitly include them in the formulation because they are typically trivial compared to the travel time along the (relatively) long routes. We also omit the effects of other factors, such as traffic conditions and storage limits because vaccine distribution is typically regarded as a critical public health management task and given a high priority in resource assignment and traffic control. These detailed calculations can be easily added without affecting the framework of our algorithm.

For notational simplicity, when needed, we also use $\mathbf{x}_i$ to denote the set of inoculation spots in the route. Each solution X must satisfy the following basic constraints.

1) Each inoculation spot is visited by at most one route

$$\mathbf{x}_i \cap \mathbf{x}_{i'} = \varnothing \quad \forall i, i' \in \{1, 2, \ldots, m\} \wedge (i \ne i'). \tag{9}$$

2) All inoculation spots must be assigned

$$\bigcup_{i=1}^{m} \mathbf{x}_i = O. \tag{10}$$

3) Each route cannot violate the vehicle capacity

$$\sum_{j=1}^{n_i} r_{x_{i,j}} \le C_{v_i} \quad \forall 1 \le i \le m. \tag{11}$$

4) Each inoculation spot must receive vaccines no later than the deadline $\widehat{T}$ (set to 9:30 A.M. in our study)

$$T(x_{i,n_i}) \le \widehat{T} \quad \forall 1 \le i \le m. \tag{12}$$

As vaccine distribution in a big city typically involves a dozen of satellites/depots and hundreds of inoculation spots, the problem has a significantly large solution space.

V. Evolutionary Optimization for Vaccine Distribution

In this section, we describe the EA proposed for the vaccine distribution problem. As the vaccination program often lasts for a period, we need to solve a new problem instance every day. Although different instances have different demands, the demands do not change dramatically over time; moreover, the instances share common features, such as the distribution of depots, satellites, and inoculation spots, available vehicles, and the underlying transportation network. Therefore, we have an opportunity to capture inherent relationship between evolving problem inputs (demands) to solutions, such that historical knowledge of vaccine distribution in early days can be utilized to improve the algorithm performance on new instances, which is the main difference of our EA from existing EAs for VRP. We also use machine learning to infer a threshold of the distance between an inoculation spot and its satellite to reduce the search space and improve search performance.

A. Knowledge Base of Historical Instances and Solutions

We construct a knowledge base, which saves two types of knowledge items. Each first-type item consists of an instance of the vaccine distribution problem and a set of known high-quality solutions to the instance. Given two instances I and I', we evaluate the distance between them based on the differences among their inputs as

$$\mathcal{D}(I, I') = \sum_{s\in S\cup D} \big(a_s - a'_s\big)^2 + w_r \sum_{s\in S} \big(r_s - r'_s\big)^2 \tag{13}$$

where a_s (available number of vaccines) and $r_s = (\sum_{o\in O_s} r_o)$ (demanded number of vaccines) are the inputs to the instance I, a'_s and r'_s are the inputs to the instance I', and w_r is a coefficient equal to or larger than 1 (set to 2 in our study).

Each second-type item consists of a subproblem instance of vaccine distribution from a satellite/depot and a set of known high-quality subsolutions to the subinstance. Given two subinstances I_s and I'_s, we evaluate the distance between them as

$$\mathcal{D}\big(I_s, I'_s\big) = \sum_{o\in O_s\cap O'_s} \big(r_o - r'_o\big)^2 + \sum_{o\in O_s\setminus O'_s} r_o^2 + \sum_{o\in O'_s\setminus O_s} r_o'^2 \tag{14}$$

where O_s and O'_s denote the subsets of inoculation spots assigned to s in I_s and I'_s, and r_o and r'_o are the demands of inoculation spot o in the two subinstances, respectively.

For each instance/subinstance, we save at most N_B solutions/subsolutions, where N_B is a control parameter (set to 10 in our study). A new solution will replace an existing solution in the knowledge base only if it is a new best known solution for the instance/subinstance, or it satisfies two conditions: 1) its objective function value is not larger than twice the value of the best known one and 2) by replacing the nonbest solution that has the minimum average distance to all other solutions in the knowledge base, it will increase this minimum average distance. The distance between two routes $\mathbf{x}_i$ and $\mathbf{x}'_i$ of a vehicle v_i, denoted by $d(\mathbf{x}_i, \mathbf{x}'_i)$, is calculated as follows.

1) Initialize the distance to 0.
2) For each spot in $\mathbf{x}_i$ but not in $\mathbf{x}'_i$, increase the distance by 1.
3) For each spot in $\mathbf{x}'_i$ but not in $\mathbf{x}_i$, increase the distance by 1.
4) Obtain the longest common subsequence of the two routes.
5) For each spot in $\mathbf{x}_i$ and $\mathbf{x}'_i$ but not in the longest common subsequence, calculate the difference between its index and the index of the longest common subsequence in each route; if the difference is not the same in the two routes, increase the distance by 0.5.

For two subsolutions X_s and X'_s to a subinstance of vaccine distribution from a satellite/depot s, their distance $d(X_s, X'_s)$ is calculated as follows.

1) Sort the routes in X_s in nonincreasing order of the number of spots.

$\mathbf{x}_1$ {5,2,3,7} $\mathbf{x}'_1$ {1,2,3}

$\mathbf{x}_2$ {10,6,4} $\mathbf{x}'_2$ {8,6,4,10}

$\mathbf{x}_3$ {1,9,8} $\mathbf{x}'_3$ {5,9,7}

Fig. 3. Example of two solutions to an instance using three vehicles for distribution from a depot to ten spots. 1) For vehicle v_1, spots 5 and 7 are in $\mathbf{x}_1$ but not in $\mathbf{x}'_1$, and spot 1 is in $\mathbf{x}'_1$ but not in $\mathbf{x}_1$, and so we have $d(\mathbf{x}_1, \mathbf{x}'_1) = 3$. 2) For v_2, spot 8 is in $\mathbf{x}'_2$ but not in $\mathbf{x}_2$; spot 10 is in both $\mathbf{x}'_2$ and $\mathbf{x}_2$, but it is not in the longest common subsequence {6, 4}, and its index differences from the subsequence are not the same in the two routes; hence, we have $d(\mathbf{x}_2, \mathbf{x}'_2) = 1.5$. 3) For v_3, spots 1 and 8 are only in $\mathbf{x}_3$ and spots 5 and 7 are only in $\mathbf{x}'_3$, and so we have $d(\mathbf{x}_3, \mathbf{x}'_3) = 4$. The distance between the two solutions is $d(X, X') = \sqrt{3^2 + 5^2 + 1.5^2} = 5.22$.

2) For each route $\mathbf{x}_i$ in X_s, match it with a route $\mathbf{x}'_i$ whose distance to $\mathbf{x}_i$ is the smallest among all unmatched routes in X'_s.
3) Calculated $d(X_s, X'_s)$ as the square root of the distances of all pairs of matched routes.

For two solutions X and X' to an instance of the problem, their distance is calculated as the mean square root of the distances of all pairs of subsolutions in the two solutions

$$d(X, X') = \sqrt{\sum_{s \in S \cup D} d^2(X_s, X'_s)}. \tag{15}$$

Fig. 3 presents an example for calculating the solution distance.

Based on the first-type knowledge, we construct another machine learning model. The input features to the model consist of a_d for all $d \in D$ and a_s and r_s for all $s \in S$ in a problem instance. The model output is the maximum travel time $t^\dagger$ between an inoculation spot and the satellite/depot to which it is assigned in the best known solution $\mathbf{x}^*$ to the instance

$$t^\dagger = \max_{s \in S \cup D} \left\{ \max_{v_i \in V_s} \left\{ \max_{o \in \mathbf{x}^*_i} t(o, s) \right\} \right\}. \tag{16}$$

We use MLR to model this relation. In this study, we train each model instance using samples in the same city for prediction. We can also extend the model for prediction in different cities but, if so, geographical distribution information should also be considered as model inputs. For each new instance, we use the model to predict the $t^\dagger$ value; if the value is smaller than the smallest value in the knowledge base, it is set to the latter. We then use $2t^\dagger$ as a threshold for assigning any inoculation spot to a satellite for the instance, which can significantly reduce the search space of the algorithm.

B. Solution Initialization

To solve a new instance I of the problem, the EA initializes a population of N_P solutions, which are divided into two classes. Each first-class solution is generated based on solutions to similar instances and subinstances in the knowledge base. First, we randomly select a "base" solution X^0 to a similar instance from the knowledge base. The similar instance can be the one that has the minimum distance to I, denoted by $\mathcal{D}^*_I$, or another instance whose distance to I is not larger than $2\mathcal{D}^*_I$. We take the inoculation spot assignment in X^0 for X. Next, for each subinstance of vaccine distribution from a satellite/depot $s \in S \cup D$, we randomly select a "base" subsolution X^0_s to a similar subinstance from the knowledge base and then adapt the routes in X^0_s to the given subinstance using the following procedure.

1) Remove any spot in $O'_s \backslash O_s$ from the routes.
2) Sort all unassigned spots in $O_s \backslash O'_s$ in a nonincreasing order of r_o.
3) For each spot in $O_s \backslash O'_s$, insert it into the route with the earliest completion time at a position, which has the minimum objective function value among all possible positions.

We generate at most $N_P/2$ first-class solutions for the population, but remove any duplicated ones.

Each second-class solution is randomly initialized using the following procedure (where the threshold $2t^\dagger$ is applied).

1) For each satellite s with sufficient vaccines for its region, randomly assign each inoculation spot in the region to a vehicle (with sufficient remaining capacity) at the satellite.
2) For each remaining inoculation spot o, randomly assign it to a vehicle (with sufficient remaining capacity) at a satellite/depot s satisfying that $t(o, s) \leq 2t^\dagger$.
3) For each vehicle, employ the NEH heuristic [55] to initialize its route.
4) For each satellite/depot, continually move an inoculation spot from the route with the maximum completion time to the route with the minimum completion time until doing so cannot improve the solution fitness.

C. Solution Evolution

After initializing the population, the EA iteratively evolves the solutions. At each generation, each solution X in the population produces a child solution X' by performing a series of random local search (RLS) operators including:

1) randomly swapping two spots in a route;
2) randomly moving a spot o from a route $\mathbf{x}_i$ to another $\mathbf{x}_{i'}$, under the condition that the capacity of $v_{i'}$ is not violated, and the distance between o and the satellite/depot of $\mathbf{x}_{i'}$ is not larger than $2t^\dagger$;
3) randomly swapping two spots o and o' between two routes $\mathbf{x}_i$ and $\mathbf{x}_{i'}$, under the condition that the capacity of either vehicle is not violated, and the distance between o and the satellite/depot of $\mathbf{x}_{i'}$ and that between o' and the satellite/depot of $\mathbf{x}_i$ are both not larger than $2t^\dagger$.

The number of RLS operations performed on X is a random number between 0 and $\lambda(X)$, the variability of X calculated as follows:

$$\lambda(X) = \frac{m}{2} \cdot \left(\frac{4}{m}\right)^{(f(X) - f_{\min} + \epsilon)/(f_{\max} - f_{\min} + \epsilon)} \tag{17}$$

where $f_{\min}$ and $f_{\max}$ are the minimum and maximum objective function values among the population, respectively, and ϵ is a small positive number to avoid division by 0. The idea of assigning each solution with a variation is borrowed from evolutionary programming [56], [57], and making the variation inversely proportional to the solution fitness is borrowed

Algorithm 1: EA for the Vaccine Distribution Problem After Demand Forecasting and Predistribution

```
Initialize a population of solutions according to Section V-B;
Let X* be the best among the solutions;
while the stop condition is not met do
    Compute the average distance d̄ from all other solutions to X*;
    foreach X in the population do
        Compute λ(X) according to Eq. (17);
        Let λ be a random number between 0 and λ(X);
        Produce a child X′ by performing λ-step local search operations on X;
        if f(X′) < f(X) then
            Replace X with X′;
            if f(X) < f(X*) or d(X, X*) > d̄ then
                perform ENS on X;
                if f(X) < f(X*) then
                    X* ← X;
        else if X has not been improved during consecutive ĝ generations then
            Replace X with a new solution initialized according to Section V-B;
return X*.
```

from the WWO metaheuristic [20], [58], such that good/bad solutions search small/large areas to balance exploration and exploitation.

If the child X' is better than X, it replaces X in the population. To avoid search stagnation, if a solution X has not produced a better child during a number $\widehat{g}$ of consecutive generations (where $\widehat{g}$ is a control parameter), it will be replaced by a new solution randomly generated using the approach described in Section V-B, where the best known solution obtained by the EA is also considered as an exemplar as knowledge-base solutions.

We also enhance the EA with an extensive neighborhood search (ENS), which is performed on any new solution X' that is better than the current best known solution X^*, or is better than the parent X and its distance to X^* is larger than the average distance to X^* of all other solutions in the population. The ENS has the following neighborhood search operators (using ideas borrowed from [59]).

1) Single-route improvement, 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.
2) Multiroute improvement, 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; any operation cannot violate the distance threshold $2t^{\dagger}$.

Algorithm 1 presents the pseudocode of the EA.

TABLE II
CONTROL PARAMETERS OF THE PROPOSED EA

Parameter	Description	Value
N_P	population size	45
N_B	number of solutions of an instance in the knowledge base	10
$\widehat{g}$	maximum number of non-improvement generations	linearly decreases from 12 to 3 with generation

VI. COMPUTATIONAL RESULTS

We have applied the proposed hybrid machine learning and evolutionary computation method to vaccine distribution in Hangzhou City, Zhejiang Province, China, since April 7, 2021. There were two depots and ten satellites, and the number of inoculation spots slightly changed between 200–240. Before April 7, the public health department used a two-echelon distribution approach (without demand forecasting and predistribution) which employed a GA adapted from [30]. For the application on April 7, we used the data from March 24 to April 6 as the samples to train the fuzzy deep learning model; since then, we retrained the model with the new data on each day. For the EA for optimizing vaccine distribution after demand forecasting and predistribution, we first tuned the control parameters of the algorithm on the simulated instances of the seven days before April 7, resulting in a setting shown in Table II; after each seven days, we retuned the parameters with new instances, but the values did not change much.

Here, we present the application results from April 7 to April 30. To validate the performance of the proposed EA with the knowledge base (denoted by KB-EA), we not only simulate the use of the previous GA-based two-echelon (GA-2E) distribution method on each day's vaccine distribution task (without predistribution) but also employ the following heuristics and metaheuristics to solve each day's instance of the vaccine distribution problem after predistribution.

1) A reactive greedy randomized adaptive search (RGRAS) method [60].
2) A hybrid ant colony-variable neighborhood search (ACO-VNS) algorithm [61].
3) A discrete firefly algorithm with compound neighborhoods (DFA-CNs) [62].
4) A hybrid GA with VNS (GA-VNS) [63].
5) A hybrid VNS with greedy randomized adaptive memory programming search (VNS-GRAMPS) algorithm [64].
6) A variant of our EA without using the knowledge base.

Each comparative algorithm is run for 30 times on each instance. The computational environment is a workstation with an i7-6500 2.5GH CPU, 8-GB DDR4 RAM, and an NVIDIA Quadro M500M card. Fig. 4 presents the objective function values (i.e., average weighted distribution times) obtained by the comparative algorithms on the 24 instances. The average weighted distribution time of the previous two-echelon distribution method ranges between 120–170 min and is around 145 min in average. In comparison, by forecasting and predistributing the demands, all other seven algorithms significantly decrease the average distribution time. Among these seven algorithms, the proposed KB-EA achieves the

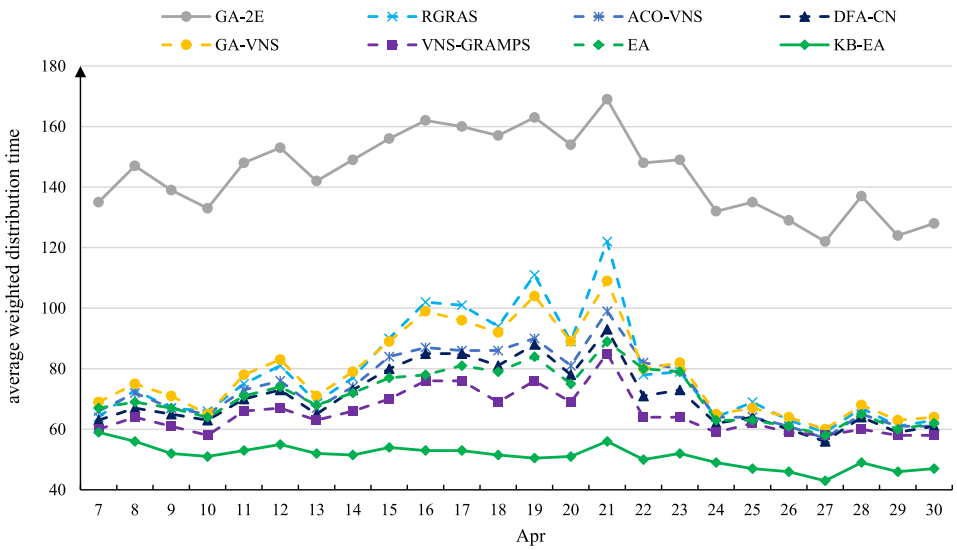

Fig. 4. Objective function values (average weighted distribution times in minutes) obtained by the GA-2E method previously used by the organization, the proposed KB-EA, and the other six comparative algorithms on the instances of vaccine distribution in Hangzhou on 24 days from April 7 to April 30. The results show that the solutions of KB-EA have significantly shorter distribution times (shown in the green line) than other algorithms in all cases.

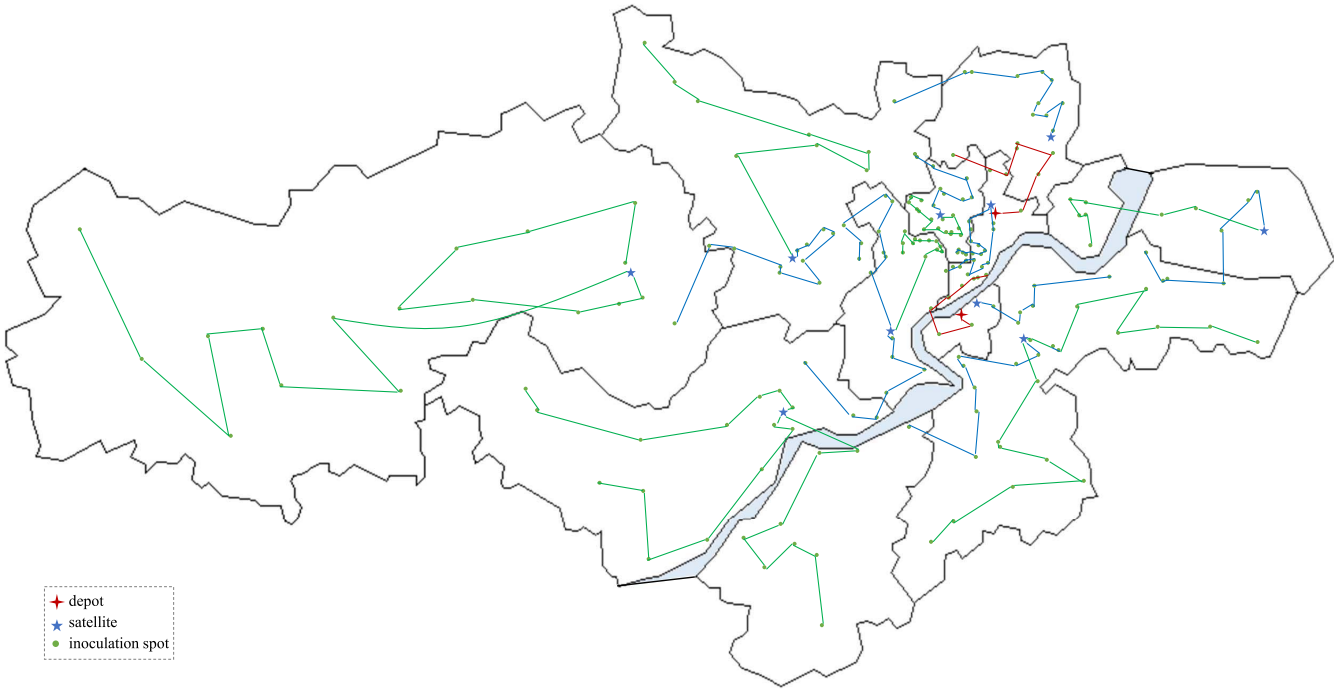

Fig. 5. Distribution solution produced by our method for the instance of April 16. A green line denotes a vehicle route within a predefined region, a blue line denotes a vehicle route which covers inoculation spots cross two or more regions, and a red line denotes a vehicle route directly from the central depot to inoculation spots. The results show that cross-region delivery plays an important role in the solution.

minimum average distribution time of 51.1 min. The key difference between EA and KB-EA is that the former does not utilize the knowledge base for solution initialization; the average distribution time of EA is 71.1 min, which is also significantly longer than that of KB-EA, demonstrating that utilizing high-quality historical solutions in solution initialization effectively accelerates the evolutionary process and therefore improves the final results. Another evidence is that, as we can observe from Fig. 4, although the previous GA-2E method and EA have significantly different performances, they have similar changing forms, that is, their performances change with the instances in a similar way. In comparison, because KB-EA utilizes historical knowledge in initializing solutions for new instances, its performance does not only affected by the instance on hand but also improves over time: the average distribution time of KB-EA on April 7 is the highest among the 24 days and is not significantly from that of the other algorithms, but the time generally decreases from April 7 to April 30, and finally achieves a significant performance advantages over the others. Fig. 5 presents the vehicles routes

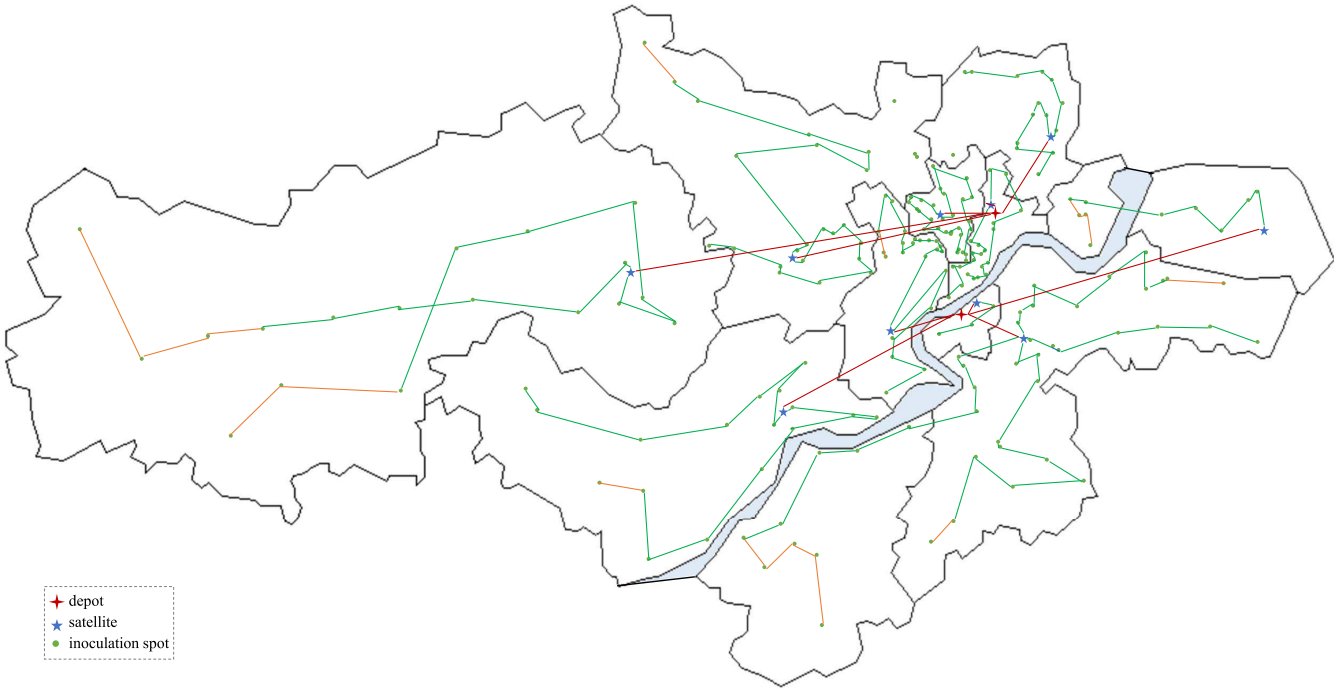

Fig. 6. Distribution solution produced by the 2E-VRP approach without predistribution for the instance of April 16. A red line denotes a depot-to-satellite route, a green line denotes a satellite-to-customer route, while a yellow line following a green line indicates that the corresponding customer(s) would receive vaccines after the deadline.

for vaccine distribution in our solution in April 16: among 23 vehicle routes, 11 routes (in green) cover only inoculation spots inside the corresponding regions, ten (in blue) cover inoculation spots cross two or more regions, and two are from central depots to inoculation spots. The results indicate that, after predistribution, it is necessary and efficient to allow cross-region vaccine distribution. For comparison, Fig. 6 presents the vehicles routes in the solution produced by the EA without predistribution, where inoculation spots on yellow lines would receive vaccines after the deadline $\widehat{T}$. This is because, in the standard 2E-VRP approach, a satellite could begin the delivery only after it receives vaccines from the depot(s) and, hence, some customers have to wait a long time. In our approach, predistribution of vaccines to satellites allows the satellites to begin the delivery much earlier, and there is no customer receiving vaccines after the deadline in our solution.

In general, the performances of the other six algorithms other than KB-EA change over instances in a manner similar to that of GA-2E, as all of them do not utilize historical knowledge. Among these six algorithms, EA does not always show the best performance. The average distribution time of VNS-GRAMPS is only 65.3 min, and that of DFA-CN is around 70.8 min, both being shorter than that of EA. The average distribution times of the other three algorithms are longer than EA, where those of RGRAS and GA-VNS are around 78.5 and 78.4 min, respectively. These two algorithms perform similarly with other algorithms on some relatively simple instances (such as the instances of April 26, 27, 29 and 30), but perform much worse than others on some more difficult instances (such as the instances of April 16–21), which reveals that they are suitable for solving some simple instances, but their performances dramatically deteriorate with the increase of the instance difficulty.

According to Wilcoxon rank-sum tests, on the instance of April 7, the result of KB-EA has no statistical difference with those of VNS-GRAMPS and DFA-CN, but is significantly better than those of RGRAS, ACO-VNS, and GA-VNS; on each of the other 23 instances, the result of KB-EA is significantly better than those of all other algorithms. The results demonstrate that our knowledge-based EA exhibits considerable performance advantages over not only the previous method without predistribution but also popular EAs for the extended VRP after predistribution.

Next, to validate the performance of the fuzzy deep autoencoder (FDAE) forecasting model in combining with KB-EA, we also test the following five forecasting models by combining each of them with the KB-EA.

1) A basic three-layer feedforward artificial neural network (ANN) model.
2) An auto-regressive integrated moving average (ARIMA) model [65].
3) A least square SVM (LSSVM) model [66], [67].
4) A deep multiscale convolutional LSTM (CLSTM) Network model [68].
5) A basic deep autoencoder (DAE) model [46] using the same structure as our FDAE but using standard crisp parameters instead of fuzzy ones.

The structures and parameters of these models are all tuned by the evolutionary optimization method [45]. In Fig. 7, the graph at the top presents the forecasting accuracies of different machine learning models, and the graph at the bottom presents

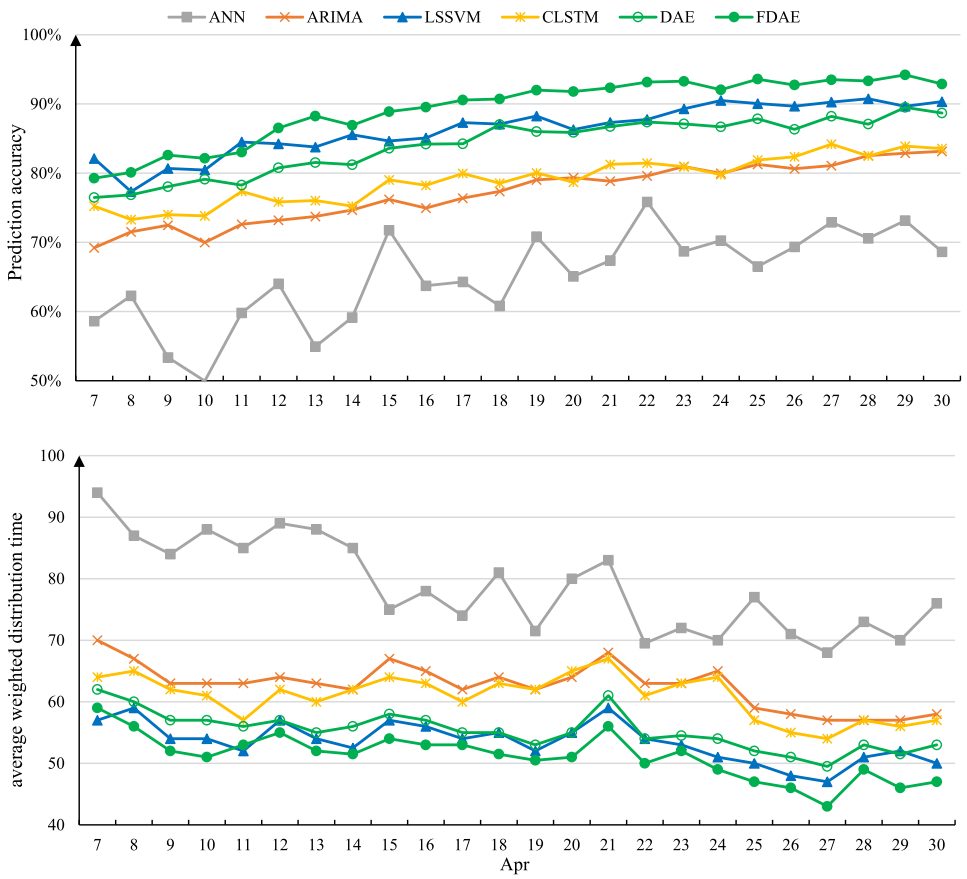

Fig. 7. Comparison of different machine learning models in combining with KB-EA on the 24 instances. The top graph shows the forecasting accuracies of the different forecasting models, and the bottom graph presents the objective function values (average weighted distribution times in minutes) obtained by KB-EA combined with the forecasting models. The results show that on almost all instances, FDAE obtains the highest prediction accuracies, and KB-EA combined with FDAE obtains the shortest distribution times.

the average weighted distribution time (in minutes) obtained by KB-EA combined with these machine learning models. In general, the forecasting accuracies of the models increase over time with the increasing amount of historical data, which consequently improves the distribution performance of the hybrid machine learning and evolutionary computation methods. The forecasting accuracy of the traditional ANN model is the lowest and most unstable, because the number of influence features for demand forecasting is large, and the shallow network structure is inefficient in learning in the high-dimensional feature space. The forecasting accuracies of the other five models and the distribution performances of their combinations with KB-EA are relatively stable. Initially (after training on the data from March 24 to April 6), LSSVM exhibits the highest forecasting accuracy of 82.11%, and FDAE exhibits the second highest accuracy of 79.27%. Since April 8, except on April 11 the accuracy of FDAE is slightly lower than LSSVM, on all other days the accuracy of FDAE is always higher than LSSVM and the other four models. In general, the forecasting performance advantages of FDAE over the other models also increase over time. In particular, the advantage of FDAE over DAE demonstrates that the use of fuzzy parameters effectively improve the model performance, as the input features often contain fuzzy and uncertain information. In terms of the overall or average forecasting accuracy, FDAE exhibits the highest performance that is significantly better than the other models, the performance of LSSVM ranks second, and those of ARIMA and CLSTM are relatively low.

The forecasting performance advantages of FDAE also result in the distribution performance advantages of KB-EA combined with FDAE over KB-EA combined with other machine learning models. As we can observe from the bottom graph in Fig. 7, similar to their forecasting performances, the distribution performance obtained by KB-EA with ANN is the worst and most unstable; the distribution performances obtained by KB-EA with other five models generally increase with time, where those obtained by KB-EA with ARIMA and KB-EA with CLSTM are relatively low, and that obtained by KB-EA with FDAE is the best.

To validate the relationship between forecasting accuracy and distribution time, we also use the performance of KB-EA with FDAE as the benchmark to calculate the ratio of the average weighted distribution time difference to the forecasting accuracy difference of each other model on each instance

$$\text{RAT} = \frac{f - f_b}{(acc_b - acc) \times 100\%} \tag{18}$$

where acc_b is the forecasting accuracy of the benchmark model (FDAE), f_b is the objective function value (average weighted distribution time) obtained by KB-EA with FDAE, and acc

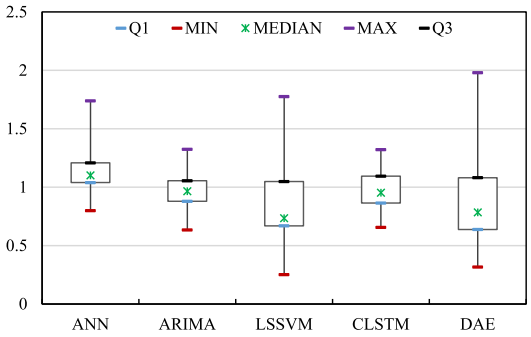

Fig. 8. Values of RAT [defined in (18)] of the comparative models. The results show that if the forecasting accuracy is decreased by one percent, the weighted distribution time will be increased by around one minute, demonstrating the contribution of the forecasting accuracy to the distribution performance.

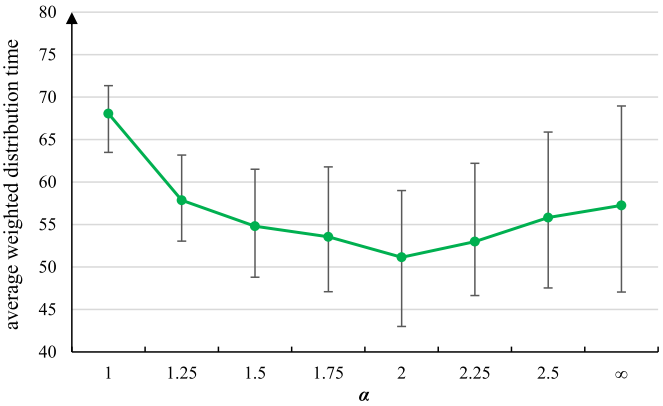

Fig. 9. Objective function values obtained by the proposed EA with different thresholds $\alpha \cdot t^{\dagger}$ ($\alpha = \infty$ indicates without using the threshold). The results show that using $\alpha = 2$ achieves a promising general performance.

and f are the forecasting accuracy and average weighted distribution time of the corresponding comparative model, respectively. Fig. 8 presents the median, maximum, minimum, first quartile (25%) and third quartile (75%) of RAT values of each other model. The results show that the median RAT values of the models are 0.73–1.10, and most RAT values are between 0.64–1.2 (from the first quartile to third quartile), that is, in average, if the forecasting accuracy is decreased by one percent, the weighted distribution time will be increased by around 1 min, and the deviations are generally small. The results clearly demonstrate that the distribution performance is generally proportional to the forecasting performance, which validates the fundamental principle of our hybrid machine learning and evolutionary computation method.

Finally, we test the sensitivity of the EA to the threshold of the distance between an inoculation spot and its satellite. We compare the objective function values obtained by the EA with threshold values of $t^{\dagger}$, $1.25t^{\dagger}$, $1.5t^{\dagger}$, $1.75t^{\dagger}$, $2t^{\dagger}$, $2.25t^{\dagger}$, and $2.5t^{\dagger}$, and without threshold on the test instances. As we can observe from the results shown in Fig. 9, the threshold value of $2t^{\dagger}$ results in the best performance. When the threshold value is too large, the search space increases rapidly, and the probability that the EA finds optimal or near-optimal solutions decreases dramatically; on the contrary, if the threshold is too tight, many high-quality solutions are excluded during the search process and, thus, the algorithm performance reduces. Note that we do not intend to find a very accurate value of the threshold, as the best threshold value changes with instances, and our results show that using $2t^{\dagger}$ is a simple way that performs well on most instances.

VII. Conclusion

In this article, we presented a hybrid machine learning and evolutionary computation method for vaccine distribution in a big city. The method first used fuzzy deep learning to forecast the demands for vaccines for each next day, so as to predistribute the forecasted number of vaccines to satellites in advance, and then used EA to route vehicles to distribute vaccines from the satellites/depots to inoculation spots on each day. The EA utilized the knowledge of historical vaccine distribution to improve initial solution quality to each new instance, and employed a distance threshold to reduce the search space. Computational results on real-world instances in Hangzhou, China, demonstrated the effectiveness and efficiency of the proposed method compared to the previous two-echelon distribution method and some state-of-the-art EAs for VRP. The source code is available at our website http://compintell.cn/en/dataAndCode.html. We are currently extending the distribution problem by allowing using different types of vehicles (including unmanned vehicles) while taking the efficiency of medical staff into consideration [69].

The proposed hybrid machine learning and evolutionary computation method can be useful for many other similar resource distribution problems, such as the medical mask distribution in epidemics, relief goods distribution in disasters, and police deployment in riots, where the demands for resources are often uncertain and change over time. For such problems, demand forecasting and predistribution can significantly reduce the total distribution time, and utilizing historical distribution knowledge can effectively improve the problem-solving performance over time. One limitation of the current method is that the instance/solution measurements depicted in Section V-A are problem specific, and our ongoing work will generalize the measurements to cover a wide range of problems.

References

[1] V. Priesemann, R. Balling, M. M. Brinkmann, S. Ciesek, T. Czypionka, and I. Eckerle, "An action plan for pan-European defence against new SARS-CoV-2 variants," *Lancet*, vol. 397, no. 10273, pp. 469–470, 2021.

[2] R. Markovič, M. Šterk, M. Marhl, M. Perc, and M. Gosak, "Socio-demographic and health factors drive the epidemic progression and should guide vaccination strategies for best COVID-19 containment," *Results Phys.*, vol. 26, Jul. 2021, Art. no. 104433.

[3] M. Piraveenan *et al.*, "Optimal governance and implementation of vaccination programmes to contain the COVID-19 pandemic," *Roy. Soc. Open Sci.*, vol. 8, no. 6, 2021, Art. no. 210429.

[4] S. Anily and A. Federgruen, "Two-echelon distribution systems with vehicle routing costs and central inventories," *Oper. Res.*, vol. 41, no. 1, pp. 37–47, 1993.

[5] J. González-Feliu, G. Perboli, R. Tadei, and D. Vigo, "The two-echelon capacitated vehicle routing problem," Dept. Electron. Comput. Sci. Syst., Univ. Bologna, Rep. DEIS OR.INGCE 2007/2, 2007.

[6] R. Miikkulainen *et al.*, "From prediction to prescription: Evolutionary optimization of nonpharmaceutical interventions in the COVID-19 pandemic," *IEEE Trans. Evol. Comput.*, vol. 25, no. 2, pp. 386–401, Apr. 2021.

[7] L. R. Burns, *The Health Care Value Chain*. San Francisco, CA, USA: Jossey-Bass, 2002.

[8] X. Zhong, H. K. Lee, and J. Li, "From production systems to health care delivery systems: A retrospective look on similarities, difficulties and opportunities," *Int. J. Prod. Res.*, vol. 55, no. 14, pp. 4212–4227, 2017.

[9] H. O. Mete and Z. B. Zabinsky, "Stochastic optimization of medical supply location and distribution in disaster management," *Int. J. Prod. Econom.*, vol. 126, no. 1, pp. 76–84, 2010.

[10] L. Lei, M. Pinedo, L. Qi, S. Wang, and J. Yang, "Personnel scheduling and supplies provisioning in emergency relief operations," *Ann. Oper. Res.*, vol. 235, no. 1, pp. 487–515, 2015.

[11] M. Liu and D. Zhang, "A dynamic logistics model for medical resources allocation in an epidemic control with demand forecast updating," *J. Oper. Res. Soc.*, vol. 67, no. 6, pp. 841–852, 2016.

[12] I. E. Büyüktahtakın, E. Des-Bordes, and E. Y. Kıbış, "A new epidemics–logistics model: Insights into controlling the Ebola virus disease in West Africa," *Eur. J. Oper. Res.*, vol. 265, no. 3, pp. 1046–1063, 2018.

[13] M. Liu, J. Cao, J. Liang, and M. Chen, "Integrated planning for public health emergencies: A modified model for controlling H1N1 pandemic," *J. Oper. Res. Society*, vol. 71, pp. 748–761, May 2020.

[14] S. M. Arabzad, M. Ghorbani, and R. Tavakkoli-Moghaddam, "An evolutionary algorithm for a new multi-objective location-inventory model in a distribution network with transportation modes and third-party logistics providers," *Int. J. Prod. Res.*, vol. 53, no. 4, pp. 1038–1050, 2015.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[16] X. Gan and J. Liu, "A multi-objective evolutionary algorithm for emergency logistics scheduling in large-scale disaster relief," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2017, pp. 51–58.

[17] Y. Zhou, J. Liu, Y. Zhang, and X. Gan, "A multi-objective evolutionary algorithm for multi-period dynamic emergency resource scheduling problems," *Transp. Res. E, Logist. Transp. Rev.*, vol. 99, pp. 77–95, Mar. 2017.

[18] E. Osaba, X.-S. Yang, I. Fister, J. Del Ser, P. Lopez-Garcia, and A. J. Vazquez-Pardavila, "A discrete and improved bat algorithm for solving a medical goods distribution problem with pharmacological waste collection," *Swarm Evol. Comput.*, vol. 44, pp. 273–286, Feb. 2019.

[19] H.-F. Ling, Z.-L. Su, X.-L. Jiang, and Y.-J. Zheng, "Multi-objective optimization of integrated civilian-military scheduling of medical supplies for epidemic prevention and control," *Healthcare*, vol. 9, no. 2, p. 126, 2021.

[20] Y.-J. Zheng, "Water wave optimization: A new nature-inspired metaheuristic," *Comput. Oper. Res.*, vol. 55, no. 1, pp. 1–11, 2015.

[21] F. Goodarzian, A. A. Taleizadeh, P. Ghasemi, and A. Abraham, "An integrated sustainable medical supply chain network during COVID-19," *Eng. Appl. Artif. Intell.*, vol. 100, Apr. 2021, Art. no. 104188.

[22] G. Perboli, R. Tadei, and R. Tadei, "New families of valid inequalities for the two-echelon vehicle routing problem," *Elect. Notes Discrete Math.*, vol. 36, pp. 639–646, Aug. 2010.

[23] R. Baldacci, A. Mingozzi, R. Roberti, and R. W. Calvo, "An exact algorithm for the two-echelon capacitated vehicle routing problem," *Oper. Res.*, vol. 61, no. 2, pp. 298–314, 2013.

[24] N. Dellaert, F. D. Saridarq, T. Van Woensel, and T. G. Crainic, "Branch-and-price–based algorithms for the two-echelon vehicle routing problem with time windows," *Transp. Sci.*, vol. 53, no. 2, pp. 463–479, 2019.

[25] V. C. Hemmelmayr, J.-F. Cordeau, and T. G. Crainic, "An adaptive large neighborhood search heuristic for two-echelon vehicle routing problems arising in city logistics," *Comput. Oper. Res.*, vol. 39, no. 12, pp. 3215–3228, 2012.

[26] U. Breunig, V. Schmid, R. F. Hartl, and T. Vidal, "A large neighbourhood based heuristic for two-echelon routing problems," *Comput. Oper. Res.*, vol. 76, pp. 208–225, Dec. 2016.

[27] P. Grangier, M. Gendreau, F. Lehuédé, and L.-M. Rousseau, "An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization," *Eur. J. Oper. Res.*, vol. 254, no. 1, pp. 80–91, 2016.

[28] D. L. J. U. Enthoven, B. Jargalsaikhan, K. J. Roodbergen, M. A. J. uit het Broek, and A. H. Schrotenboer, "The two-echelon vehicle routing problem with covering options: City logistics with cargo bikes and parcel lockers," *Comput. Oper. Res.*, vol. 118, Jun. 2020, Art. no. 104919.

[29] H. Li, H. Wang, J. Chen, and M. Bai, "Two-echelon vehicle routing problem with satellite bi-synchronization," *Eur. J. Oper. Res.*, vol. 288, no. 3, pp. 775–793, 2021.

[30] K. Wang, S. Lan, and Y. Zhao, "A genetic-algorithm-based approach to the two-echelon capacitated vehicle routing problem with stochastic demands in logistics service," *J. Oper. Res. Soc.*, vol. 68, no. 11, pp. 1409–1421, 2017.

[31] L. Zhou, R. Baldacci, D. Vigo, and X. Wang, "A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution," *Eur. J. Oper. Res.*, vol. 265, no. 2, pp. 765–778, 2018.

[32] X. Yan, H. Huang, Z. Hao, and J. Wang, "A graph-based fuzzy evolutionary algorithm for solving two-echelon vehicle routing problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 1, pp. 129–141, Feb. 2020.

[33] A. Anderluh, P. C. Nolz, V. C. Hemmelmayr, and T. G. Crainic, "Multi-objective optimization of a two-echelon vehicle routing problem with vehicle synchronization and 'grey zone' customers arising in urban logistics," *Eur. J. Oper. Res.*, vol. 289, no. 3, pp. 940–958, 2021.

[34] L. do C. Martins, P. Hirsch, and A. A. Juan, "Agile optimization of a two-echelon vehicle routing problem with pickup and delivery," *Int. Trans. Oper. Res.*, vol. 28, no. 1, pp. 201–221, 2021.

[35] R.-K. Chiu, C.-M. Chang, and Y.-C. Chang, "A forecasting model for deciding annual vaccine demand," in *Proc. Int. Conf. Nat. Comput.*, vol. 7, 2008, pp. 107–111.

[36] J. S. Sahisnu, F. Natalia, F. V. Ferdinand, S. Sudirman, and C.-S. Ko, "Vaccine prediction system using ARIMA method," *ICIC Exp. Lett. B, Appl.*, vol. 11, no. 6, pp. 567–575, 2020.

[37] R. T. Alegado and G. M. Tumibay, "Statistical and machine learning methods for vaccine demand forecasting: A comparative analysis," *J. Comput. Commun.*, vol. 8, no. 10, pp. 37–49, 2020.

[38] Q. Song, Y.-J. Zheng, Y.-J. Huang, Z.-G. Xu, W.-G. Sheng, and J. Yang, "Emergency drug procurement planning based on big-data driven morbidity prediction," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6379–6388, Dec. 2019.

[39] Q. Song, Y.-J. Zheng, W.-G. Sheng, and J. Yang, "Tridirectional transfer learning for predicting gastric cancer morbidity," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 561–574, Feb. 2021.

[40] B. Yong, J. Shen, X. Liu, F. Li, H. Chen, and Q. Zhou, "An intelligent blockchain-based system for safe vaccine supply and supervision," *Int J. Inf. Manag.*, vol. 52, Jun. 2020, Art. no. 102024.

[41] A. X. Lin *et al.*, "Leveraging machine learning techniques and engineering of multi-nature features for national daily regional ambulance demand prediction," *Int. J. Environ. Res. Public Health*, vol. 17, no. 11, p. 4179, 2020.

[42] Y.-J. Zheng, S.-L. Yu, Q. Song, Y.-J. Huang, W.-G. Sheng, and S. Chen, "Co-evolutionary fuzzy deep transfer learning for disaster relief demand forecasting," *IEEE Trans. Emerg. Topics Comput.*, early access, Jun. 1, 2021, doi: 10.1109/TETC.2021.3085337.

[43] Y.-J. Zheng, H.-F. Ling, S.-Y. Chen, and J.-Y. Xue, "A hybrid neuro-fuzzy network based on differential biogeography-based optimization for online population classification in earthquakes," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 4, pp. 1070–1083, Aug. 2015.

[44] Y.-J. Zheng, W.-G. Sheng, X.-M. Sun, and S.-Y. Chen, "Airline passenger profiling based on fuzzy deep machine learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 12, pp. 2911–2923, Dec. 2017.

[45] X.-H. Zhou, M.-X. Zhang, Z.-G. Xu, C.-Y. Cai, Y.-J. Huang, and Y.-J. Zheng, "Shallow and deep neural network training by water wave optimization," *Swarm Evol. Comput.*, vol. 50, pp. 1–13, Nov. 2019.

[46] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 19, J. P. B. Schölkopf and T. Hoffman, Eds. Cambridge, MA, USA: MIT Press, 2007, pp. 153–160.

[47] X. Peng and Y. Yang, "Fundamental properties of interval-valued Pythagorean fuzzy aggregation operators," *Int. J. Intell. Syst.*, vol. 31, no. 5, pp. 444–487, 2016.

[48] Y.-J. Zheng, S.-Y. Chen, Y. Xue, and J.-Y. Xue, "A pythagorean-type fuzzy deep denoising autoencoder for industrial accident early warning," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1561–1575, Dec. 2017.

[49] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 833–840.

[50] J. Martens, "Deep learning via hessian-free optimization," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 735–742.

[51] N. Smiti, M. M. Dhiaf, B. Jarboui, and S. Hanafi, "Skewed general variable neighborhood search for the cumulative capacitated vehicle routing problem," *Int. Trans. Oper. Res.*, vol. 27, no. 1, pp. 651–664, 2020.

[52] J.-Y. Potvin, "State-of-the art review—Evolutionary algorithms for vehicle routing," *INFORMS J. Comput.*, vol. 21, no. 4, pp. 518–548, 2009.

[53] N. R. Sabar, M. Ayob, G. Kendall, and R. Qu, "Grammatical evolution hyper-heuristic for combinatorial optimization problems," *IEEE Trans. Evol. Comput.*, vol. 17, no. 6, pp. 840–861, Dec. 2013.
[54] L. Feng, Y.-S. Ong, M.-H. Lim, and I. W. Tsang, "Memetic search with interdomain learning: A realization between CVRP and CARP," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 644–658, Oct. 2015.
[55] M. Nawaz, E. E. Enscore, and I. Ham, "A heuristic algorithm for the *m*-machine, *n*-job flow-shop sequencing problem," *Omega*, vol. 11, no. 1, pp. 91–95, 1983.
[56] D. B. Fogel, "An introduction to simulated evolutionary optimization," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 3–14, Jan. 1994.
[57] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, Jul. 1999.
[58] Y.-J. Zheng, X.-Q. Lu, Y.-C. Du, Y. Xue, and W.-G. Sheng, "Water wave optimization for combinatorial optimization: Design strategies and applications," *Appl. Soft Comput.*, vol. 83, Oct. 2019, Art. no. 105611.
[59] S. Belhaiza, P. Hansen, and G. Laporte, "A hybrid variable neighborhood tabu search heuristic for the vehicle routing problem with multiple time windows," *Comput. Oper. Res.*, vol. 52, pp. 269–281, Dec. 2014.
[60] R. Cantu-Funes, M. A. Salazar-Aguilar, and V. Boyer, "Multi-depot periodic vehicle routing problem with due dates and time windows," *J. Oper. Res. Soc.*, vol. 69, no. 2, pp. 296–306, 2017.
[61] E. Jabir, V. V. Panicker, and R. Sridharan, "Design and development of a hybrid ant colony-variable neighbourhood search algorithm for a multi-depot green vehicle routing problem," *Transp. Res. D, Transp. Environ.*, vol. 57, pp. 422–457, Dec. 2017.
[62] J. Li *et al.*, "Discrete firefly algorithm with compound neighborhoods for asymmetric multi-depot vehicle routing problem in the maintenance of farm machinery," *Appl. Soft Comput.*, vol. 81, Aug. 2019, Art. no. 105460.
[63] H. Fan, Y. Zhang, P. Tian, Y. Lv, and H. Fan, "Time-dependent multi-depot green vehicle routing problem with time windows considering temporal-spatial distance," *Comput. Oper. Res.*, vol. 129, May 2021, Art. no. 105211.
[64] F. Kocatürk, G. Y. Tütüncü, and S. Salhi, "The multi-depot heterogeneous VRP with backhauls: Formulation and a hybrid VNS with GRAMPS meta-heuristic approach," *Ann. Oper. Res.*, vol. 307, pp. 277–302, Jun. 2021.
[65] G. Mélard and J.-M. Pasteels, "Automatic ARIMA modeling including interventions, using time series expert software," *Int. J. Forecast.*, vol. 16, no. 4, pp. 497–508, 2000.
[66] T. Van Gestel *et al.*, "Financial time series prediction using least squares support vector machines within the evidence framework," *IEEE Trans. Neural Netw.*, vol. 12, no. 4, pp. 809–821, Jul. 2001.
[67] S. Singh, K. S. Parmar, S. J. S. Makkhan, J. Kaur, S. Peshoria, and J. Kumar, "Study of ARIMA and least square support vector machine (LS-SVM) models for the prediction of SARS-CoV-2 confirmed cases in the most affected countries," *Chaos Solit. Fract.*, vol. 139, Oct. 2020, Art. no. 110086.
[68] K. F. Chu, A. Y. S. Lam, and V. O. K. Li, "Travel demand prediction using deep multi-scale convolutional LSTM network," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 1402–1407.
[69] Y.-J. Zheng, Y.-C. Du, H.-F. Ling, W.-G. Sheng, and S.-Y. Chen, "Evolutionary collaborative human-UAV search for escaped criminals," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 217–231, Apr. 2020.

Yu-Jun Zheng (Senior Member, IEEE) received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2010.

He is currently a Professor with Hangzhou Normal University, Hangzhou, China. He has authored over 100 papers in IEEE TRANSACTIONS and other famous journals. His research interests include evolutionary algorithms and their applications in transportation, emergency management, and public health.

Prof. Zheng received the IFORS Prize for Development due to the work on engineering rescuing task scheduling in the 2013 Dingxi Earthquake, China, in 2014. In 2018, he has been selected as a Finalist for the Daniel H. Wagner Prize for Excellence in Operations Research Practice.

Xin Chen received the B.Sc. degree in computer science from Tianjin Agricultural University, Tianjin, China, in 2019. She is currently pursuing the M.Sc. degree in computer science and technology with Hangzhou Normal University, Hangzhou, China.

Her research interests include evolutionary algorithms and intelligent scheduling.

Qin Song received the Ph.D. degree in environmental science and technology from the Zhejiang University of Technology, Hangzhou, China, in 2018.

She is currently a Research Assistant with Hangzhou Normal University, Hangzhou. Her current research interests include machine learning and health informatics.

Jun Yang received the B.S. degree in biochemistry and the M.S. degree in cell biology from Shandong University, Jinan, China, in 1991 and 1994, respectively, and the Ph.D. degree in biochemistry and molecular genetics from Georgia State University, Atlanta, GA, USA, in 2000.

He is currently a Professor with Zhejiang University, Hangzhou, China, and Hangzhou Normal University, Hangzhou. His main research interests are public health management and environmental health.

Ling Wang received the B.Sc. degree in automation and the Ph.D. degree in control theory and control engineering from Tsinghua University, Beijing, China, in 1995 and 1999, respectively.

Since 1999, he has been with the Department of Automation, Tsinghua University, where he became a Full Professor in 2008. He has authored five academic books and over 280 refereed papers. His current research interests include intelligent optimization and production scheduling.

Prof. Wang was a recipient of the National Natural Science Fund for Distinguished Young Scholars of China, the National Natural Science Award (Second Place) in 2014, the Science and Technology Award of Beijing City in 2008, and the Natural Science Award (First Place in 2003 and Second Place in 2007) nominated by the Ministry of Education of China. He is currently the Editor-in-Chief of the *International Journal of Automation and Control* and an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION.