IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

A Novel Hybrid Metaheuristic For Solving Automobile Part Delivery Logistics With Clustering Customer Distribution

RUI XU[1], (Member, IEEE), RUI GUO[1], AND QIONG JIA[1]
[1] School of Business, Hohai University, Nanjing 21111, China

Corresponding author: Rui Xu (e-mail: rxu@hhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 71671168/71433003, the Fundamental Research Funds for the Central Universities under Grant B18020204.

ABSTRACT With the continuous maturity of the automobile industry, the cluster network effect has been gradually formed among automobile parts suppliers. How to carry out reasonable and efficient distribution in the cluster network has become one of the greatest challenges in today's automobile manufacturers. In this paper, we conducted a field survey of a real-world automobile logistics company and collected data. After analyzing the relevant transportation modes, we concluded that the problem could be classified as a clustered vehicle routing problem (CluVRP). A CluVRP considers gathering customers according to geographical proximity on the basis of the classical capacitated vehicle routing problem (CVRP). An accompanying problem is that there is a hard constraint that all customers in a cluster must be served consecutively by the same vehicle. Because the CluVRP is NP-hard, a hybrid metaheuristic solution approach was proposed. First, the shortest distance best fit decreasing (SD-BFD) algorithm was used to construct an initial solution. Second, a clustering feature was considered in order to divide the problem into two levels for optimization: the intra-cluster level and the inter-cluster level. For the intra-cluster level, the variable neighborhood search (VNS) was applied; for the inter-cluster level, the Lin-Kernighan (LK) considering dummy point was employed for optimizing. The computational performances of the hybrid metaheuristic algorithms are tested on three sets of instances and compared with the results from previous papers and the company's manually calculated solutions. The results show that the proposed algorithm is excellent in both the operation time and the final solution value, which is significant to improve distribution efficiency, reduce transportation costs and improve customer satisfaction.

INDEX TERMS Clustered vehicle routing problem, Lin-Kernighan, dummy point, variable neighborhood search, shortest distance best fit decreasing.

I. INTRODUCTION

The developments of globalization have enabled the business scope of companies to expand continuously. To keep up with the pace of globalization and develop an enterprise scale rapidly, leaders have paid more attention to the role of logistics. Reasonable and rigorous transportation route scheduling not only reduces logistics costs but also improves customers' satisfaction. The problem of route scheduling is called a vehicle routing problem (VRP), which is a classical combinatorial optimization problem that was first brought forward by Dantzig and Ramser [1]. In a VRP, there are customers each having different demands. A depot provides services to customers by organizing efficient transportation routes. Under certain constraints, the goal is to satisfy the demands of all customers and seek the shortest route distance simultaneously [2].

The purpose of the study was investigated a variant VRP derived from a real-world automobile logistics company in China. In recent years, with the improvement of living standards, automobiles consumption has become more widespread. The increasing sales make a number of automobile companies fiercely compete. It is well known that more than 3000 parts are needed for assembling an automobile, which makes the company cost more in transferring parts from multiple parts manufacturers. In order to ensure reasonable expenditure while making profits, the leaders have a strong willingness to find more optimized ways to reduce transportation costs.

Different from traditional VRP that demand distribution is relatively scattered and irregular, specific requirements from the automobile companies led to a particular constraint: the location of parts manufacturers had already presented a clustering distribution before planning the route schedule. By analyzing existing VRP variants, there are two types of VRPs with clustering features, which are the clustered vehicle routing problem (CluVRP) and the generalized vehicle routing problem (GVRP). The original prototype of both models is the classical capacitated vehicle routing problem (CVRP) with load constraints proposed by Lenstra and Kan [3]. On the basis of CVRP, they are characterized by clustering customers in advance according to some specific clustering principles. These principles are always predefined according to the needs of researchers and enterprises. The most widely used clustering principle is geographical proximity. In the CluVRP, customers belonging to the same cluster must be serviced by the same vehicle. When the vehicle leaves a cluster, it must have served all customers in the cluster according to a scheduled sequence. This is a hard constraint. After that, the vehicle can continue to serve another cluster or return to the depot. The other model with clustering features is the GVRP, which was first proposed by [4]. It is characterized by the fact [5] that a vehicle serves only one customer within a cluster according to the schedule, and other customers will be ignored. The difference between the two problems is that all customers or one customer must be served in each cluster.

In the investigation, we learned that the company divided all suppliers in China into three subranges according to geographic position, and the activities of each subrange are independent. The three subranges include 11, 11 and 24 clusters separately, and Nanjing is the mutual delivery depot. In addition, all customers in each cluster must be served by the same vehicle in order. In summary, the transportation mode of a company can be classified as a CluVRP. Moreover, the practice of clustered distribution of customers is very common in the real world, such as for parcel deliveries and elderly patient transportation and so on [6-10].

A. THE PRACTICAL SIGNIFICANCE OF CLUVRP

Why does the CluVRP play an important role in enterprise logistics? There is one point worth thinking about: Why group customers? According to the investigation of current logistics activities in China, the following two typical scenarios are summarized:

1) MAXIMUM EFFICIENCY

With the rapid development of Chinese logistics, the ranking of logistics volumes has been first in the world continuously since 2014. The Chinese logistics system has gradually formed its own characteristics. Instead of adopting circular delivery, vehicles only run one delivery with the least amount of time. Therefore, the route schedulers prefer to arrange vehicles to serve customers that belong to the same area. From the perspective of the logistics company, the distribution costs are reduced, and the effective compactness of the route is increased. From the perspective of the drivers, it is conducive to increasing familiarity with the route and to speeding up the delivery.

2) RESOURCES LIMITED

Social development has led to increasing logistical requirements day by day, but only limited resources, including human and material resources are available, which often lead to service providers being unable to meet all needs. As pointed out by Charon and Hudry [11], it is a good way to differentiate customers into multiple subranges and to allow multiple service providers to arrange service in a timely manner. In addition to these scenarios, there are also other meaningful scenarios in which the CluVRP can play a valuable role.

B. RELATED WORKS

Although there are a wide range of practical applications, the state-of-the-art literature on VRP with clustering features is quite limited. The CluVRP was first conceived by Sevaux and Sörensen [12] to optimize parcel delivery for an actual courier company. After that, related studies appeared one after another. Barthelemy et al. [13] introduced a positive infinite M penalty in the inter-cluster distance matrix to achieve a hard constraint. Through this step, the CluVRP can be easily converted to CVRP. The problem was solved by means of the algorithm of Clarke and Wright followed by hybrid simulated annealing algorithm. Marc et al. [14] proposed the preprocessing of the inter-cluster distance matrix on Euclidean instances basis and proposed a hybrid algorithm to solve it. Maria Battarra and Daniele Vigo [15] proposed two CluVRP models. The first model is based on the two-index vehicle flow formulation from CVRP, in which the edges in a cluster to any vertex outside the cluster are constrained to two. This model fails to include the special substructure of the clusters and thus they introduced a new and more rigorous formulation model. The authors tried several exact algorithms, including branch and cut; branch and cut and price [16-17]. The experimental results proved that the last option with the best lower bound, was more reasonable and the performance of the branch was also better. In addition, the authors provided a set of benchmark instances with up to 481 vertices that often mentioned in later literatures. Thibaut Vidal and Maria Battarra [18] provided three metaheuristic algorithms, namely, iterative local search [19-20], iterative local search with mixed cluster constraints, and unified hybrid genetic search for solving the CluVRP. Because these three algorithms are exact algorithms, they took a large amount of running time to obtain accurate results at the same time. Time consumption is mainly used in pre-calculation of service order in each cluster.

Bowerman Calamiand and Hall [21] were the first to propose a category of solution termed cluster-first/route-second. They grouped customers into clusters first and then determined the serving order in each cluster. The solution of VRP could be based on clusters rather than customers. Following [21], Defryn and Sorensen [22] decomposed the

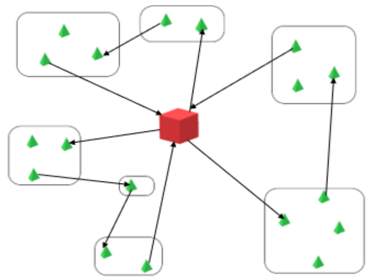
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

CluVRP into a cluster-level routing problem and a customer-level routing problem, as illustrated in Figure 1. Figure 1(a) shows a simple CluVRP solution at the cluster level. Without further planning routes within each cluster, specific driving directions in each cluster are uncertain. Figure 1(b) presents a CluVRP solution at the customer level. The target is to determine the specific driving directions within each cluster after obtaining the partial solution at the cluster level. Figure 1(a) and Figure 1(b) illustrated the two separate but successive subtasks in solving CluVRP. After constructing the first subtask, a partial solution can be obtained. It will become a complete solution after performing the second subtask. Following the two-level decomposition strategy, Exposito-Izquierdo et al. [23] split the CluVRP into a high-level route and a low-level route. The high-level route used a record-to-record(RTR) travel algorithm [24]. When solving a low-level route, the authors introduced three approaches: a mixed integer linear programming approach, the Christofides algorithm, and the Lin-Kernighan heuristic. Because the solution process was separated into two levels, it was not ideal to address the cohesive part of the later solution and only merge the two subproblems. Christof Defryn and Kenneth Sörensen [25] used variable neighborhood search (VNS) [26-28] to perform a two-level iterative search. The VNS not only improved the quality of solutions, but also accelerated the solution speed. Even for large-scale calculations, this algorithm could obtain a better upper bound. In addition, the authors also proposed a practical soft constraint. That is, the vehicles, with no need for travel according to the planned route and the service, can be interrupted in the middle of the distribution process. The vehicles can then provide priority service for customers in other clusters and then return to their own clusters to continue to service.

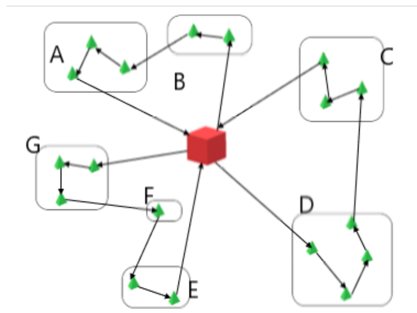
Hintsch and Timo [29] divided the CluVRP into three subproblems: cluster assignment, inter-cluster route planning and intra-cluster route planning, respectively. First, the stage of cluster assignment obtained the service order in each cluster. Next, the authors used the Large Multiple Neighborhood Search to destroy and repair the multiple cluster solutions for optimization purposes and used the variable neighborhood descent (VND) to implement postprocessing. Horvat-Marc et al. [30] proposed a novel approach to decompose the CluVRP into two submodels. Among them, a genetic algorithm (GA) was used to solve the inter-cluster routing problem, and simulated annealing (SA) was used to solve the intra-cluster routing problem. Both of these algorithms are excellent heuristics for solving VRP and the authors made a good connection in the final route merger.

C. CONTRIBUTION

The trend of clustering in automobile parts suppliers is gradually obvious, especially in China. As analyzed above, the related transportation problem can be classified as a CluVRP problem. Although there are some relevant theoretical researches in the existing literature, however, to our best knowledge, there is little study on CluVRP extracted from the real-world background.



(a) CluVRP solution at cluster level.



(b) CluVRP solution at customer level.

FIGURE 1. Illustration of the CluVRP with hard constraints. All three vehicles depart from the depot (red cube) to serve all the eighteen customers (green triangle cone). Customers in a black rectangular box belong to the same cluster, and all customers within the same cluster have to be served consecutively by the same vehicle.

The main contribution of this paper can be divided into two areas: On the one hand, this paper links a realistic scenario with the theoretical CluVRP. We surveyed an automobile logistics company in Nanjing, China, collected relevant operational data and compared experimental results with the company's manually calculated solutions, which uncovers the practical application value of the CluVRP. On the other hand, note that in a CluVRP, the visiting sequence of a vehicle route can be built that consists of two parts: the inter-cluster sequence and the intra-cluster sequence. We thus design corresponding optimizing mechanisms to determine their routes separately on the basis of their distinguishing characteristics. First, a construction algorithm based on the Hausdorff distance between clusters, which measures how far two clusters are from each other, is designed to generate an overall initial solution. Note that a fixed entry-point and exit-point of a visiting sequence are given for each cluster for the intra-cluster routing. Then, an improved Lin-Kernighan (LK) [31] heuristic was developed by combining the concept of a dummy point to optimize the intra-cluster routing and a VNS [26-28] mechanism to achieve the optimization solutions of the inter-cluster routing.

Since the CVRP is an NP-hard problem [32], if each customer in the CVRP is considered to be a cluster that contains only one customer, the CluVRP problem is

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

equivalent to a CVRP; thus, the CluVRP is still an NP-hard problem.

The structure of this paper is organized as follows. Section 2 introduces integer programming formulations for modeling the CluVRP. Subsequently, Section 3 introduces the solution approach for the problem and describes the algorithmic design. Section 4 verifies the effectiveness of the proposed algorithm on three benchmark sets of instances from previous research and a real-world instance from a Chinese logistics enterprise. In addition, different parameters were examined to assess the performance of the algorithm optimization. Finally, Section 5 presents the main conclusive remarks and study directions that are worth considering for future research.

II. MATHEMATICAL MODEL

A. MODEL ASSUMPTION

The CluVRP can be given by a complete directed graph $G = (V, A)$. The problem consists of a single distribution depot denoted as point 0, and a given set of n customer points, $N = \{1,2,3,\dots,n\}$. The demand of customer $i \in N$ is given by a scalar $q_i > 0$. The vehicle fleet $K = \{1,2,3,\dots,|K|\}$ is assumed to be homogeneous, meaning that $|K|$ vehicles are available at the depot. All vehicles have the same capacity $Q > 0$. The travel cost of a vehicle moving from i to j denoted as c_{ij}. Let $V = \{0\} \cup N = \{0,1,2,\dots,n\}$ be the set of vertices. Whereas the arc set $A = \{(i,j) \in V \times V: i \neq j\}$ and arc costs c_{ij} for $(i,j) \in A$. Assuming the problem has $|R|$ clusters, and the cluster set $R = \{1,2,3,\dots,|R|\}$(except depot), $\forall r \in R$ represents a cluster that contains at least one customer. Each customer $i \in N$ belongs to a cluster $r_i \in R$. The set of customers in a cluster r is denoted by $N^r = \{i \in N: r_i = r\}, \forall r \in R$. Consider S to be any subset of V that is different from V. Then, the out-arcs of S is defined as $\delta^+(S) = \{(i,j) \in A: i \in S, j \notin S\}$ and the in-arcs of S is defined as $\delta^-(S) = \{(i,j) \in A: i \notin S, j \in S\}$. Each vehicle departs from the distribution depot and comes back after serving all clusters and customers. The capacity of one cluster must be less than or equal to the maximum vehicle capacity. In addition, when the vehicle leaves a cluster, it must serve all customers belonging to this cluster by scheduled order.

B. MODEL DEFINITION

Decision Variables

$$x_{ijk} = \begin{cases} 1, vehicle\ k\ travels\ from\ vertice\ i\ to\ j \\ 0, otherwise \end{cases} \tag{1}$$

$$y_{ik} = \begin{cases} 1, vertice\ i\ is\ serviced\ by\ vehicle\ k \\ 0, otherwise \end{cases} \tag{2}$$

Objective Function

$$Minimize \sum_{(i,j)\in A} \sum_{k\in K} c_{ij} x_{ijk} \tag{3}$$

Subject to

$$\sum_{k\in K} y_{ik} = 1 \quad \forall i \in N \tag{4}$$

$$\sum_{k\in K} y_{0k} \leq |K| \tag{5}$$

$$y_{0k} \geq y_{ik} \quad \forall i \in N, \forall k \in K \tag{6}$$

$$\sum_{j\in N} x_{ijk} = \sum_{j\in N} x_{jik} = y_{ik} \quad \forall i \in V, \forall k \in K \tag{7}$$

$$\sum_{i\in V} q_i y_{ik} \leq Q \quad \forall k \in K \tag{8}$$

$$\sum_{i\in S} \sum_{j\notin S} x_{ijk} \geq y_{uk} \quad \forall S \subseteq N, \forall u \in S, \forall k \in K \tag{9}$$

$$\sum_{(i,j)\in\delta^+(N^r)} \sum_{k\in K} x_{ijk} = \sum_{(i,j)\in\delta^-(N^r)} \sum_{k\in K} x_{ijk} = 1 \quad \forall r \in R \tag{10}$$

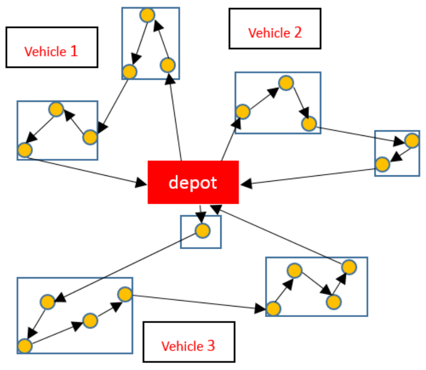
$$x_{ijk} \in \{0,1\} \quad \forall (i,j) \in A, \forall k \in K \tag{11}$$

$$y_{ik} \in \{0,1\} \quad \forall i \in V, \forall k \in K \tag{12}$$

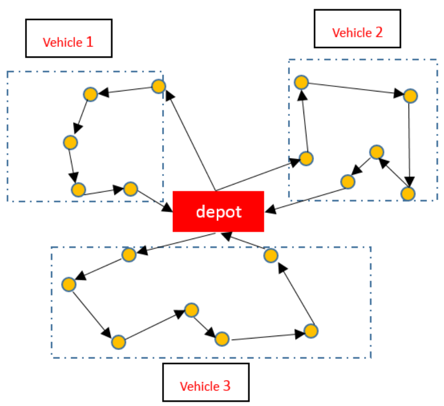
In the proposed model formulation above, there are two decision variables x_{ijk} and y_{ik} . The objective function (3) is to minimize the total travel cost. Equation (4) ensures that each customer must be serviced once. Equation (5) specifies that the total number of vehicles departing from the distribution center will not exceed the total available number of vehicles. Equation (6) ensures that all vehicles used must visit the distribution center. Equation (7) guarantees that the vehicle that arrives at a customer also leaves from the same customer. Equation (8) specifies that the vehicle capacity will not be exceeded. Equation (9) states the classical sub-tour elimination constraints. Equation (10) states that each cluster is visited only once. Equations (11-12) incorporate 0-1 binary integer variables that represent whether the customer is serviced or not and by which vehicle.

III. ALGORITHM DESIGN

The CluVRP problem is a new route planning problem in reality and its characteristic is that customers present a clustering distribution before designing a solving method. The main difference between CluVRP and CVRP can be summarized as follows: The definition of clustering in CluVRP is embodied on the problem-level, while the definition of clustering in CVRP is embodied on the method-level in the problem-solving procedure. Specifically speaking, the clustering effect is determined before designing a solving algorithm for CluVRP and geographically close customers are more likely to be regarded as a cluster. However, the clustering effect is formed during the process of constructing routes in CVRP. The distinguished difference between two transportation solutions produced by CluVRP and CVRP can be seen in Figure 2.



(a) CluVRP transportation solution



(b) CVRP transportation solution

FIGURE 2. Solution difference between CluVRP and CVRP. Solid rectangles in (a) represent real customer clustering on problem-level. Dotted rectangles in (b) represent fictitious clustering on method-level.

CVRP and its variants always consist of two separate but successive subtasks: a) partitioning of customers; b) routing of vehicles [33]. They can be regarded as a hybrid of two well-known combinatorial optimization problems: Bin Packing Problem (BPP) [34] and Traveling Salesman Problem (TSP) [35]. The solving procedure of CVRP and its variants always follows a common two-phase strategy, including the route construction phase and the followed solution improvement or local search phase [36].

For solving the CluVRP, we also followed the two-phase strategy. The first phase is to partition formed clusters into variable vehicles, which was solved through an improved well-known Best Fit Decreasing (BFD) algorithm [37] considering the specific factors in CluVRP. Note that, due to the hard constraint in CluVRP (i.e., Customers belonging to the same cluster must be serviced by the same vehicle. When the vehicle leaves a cluster, it must have served all customers in the cluster according to a scheduled sequence), it is not necessary to construct the visiting sequence in a cluster in this phase, thus a cluster, including at least one customer, can be regarded as a single "customer". While the second phase is to improve the route sequences of various vehicles at cluster-level and customer-level respectively, and there exists a corresponding route improvement process at each level.

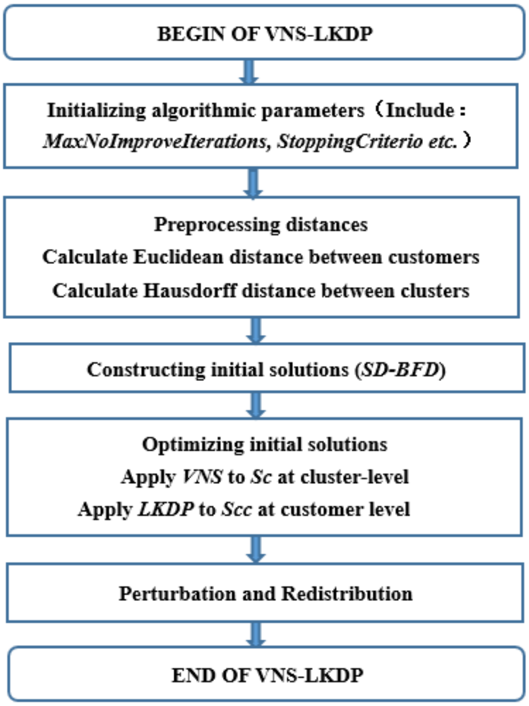


FIGURE 3. The system block diagram of VNS-LKDP.

In this paper, the overall algorithm design involved three submodules as follows:

- The cluster assignment can be regarded as a BPP to construct the initial solution. With the assistance of Hausdorff, we obtain distance matrix between clusters and clarify the entry point and the exit point within the cluster. The matrix and the SD-BFD algorithm can be used to construct an initial solution with a more reasonable interpretation.
- After constructing the initial solution, we consider optimizing it on two levels as the two subproblems can then both be considered TSP. For the cluster level (namely, inter-cluster routing), all the customer demand in a cluster was summed and the cluster is regarded as a single "customer". The problem is thus transformed into a CVRP and optimized with VNS.

  For the customer level (namely, the intra-cluster route), we introduced the concept of a dummy point because the entry-point and exit-point are not the same point and apply the LK heuristic to optimize the service order in each cluster, which is called the Lin-Kernighan with dummy point (LKDP). In summary, the complete hybrid algorithm in this paper is called the VNS-LKDP.

- To balance exploration and exploitation, a perturbation operator was designed. However, it may destroy the structure of the solution and prevent realization of target. Therefore, the redistribution operator to repair a corrupted solution was designed.

The individual section of the hybrid metaheuristic design is shown in Figure 3 concretely.

The outline of hybrid metaheuristic approach is shown in Table 1.

TABLE 1. **Pseudocode of the hybrid metaheuristic approach.**

Algorithm. Pseudocode of the metaheuristic approach
Input: a CluVRP instance to be solved
Output: the best solution S^*
Step0: Determine the initial parameters
nNoImprove ← 0; *stoppingCriterion* ← false;
Initialize the optimal solution S^* ← null;
The target value $f(S^*) \leftarrow +\infty$;
Step1: Constructing initial solution
Calculate two distance matrices;
Clusters allocation forms a partial feasible solution S_0;
Step2: Optimization
while *stoppingCriterion* is false **do**
gotoLKDP ←true;
S_{C1} ← Inter-cluster route optimization on S_0;
while *gotoLKDP* is true **do**
S_C^* ← Intra-cluster route optimization on S_{C1};
if $f(S_C^*) < f(S^*)$ **then**
$S^* \leftarrow S_C^*; f(S^*) \leftarrow f(S_C^*)$;
nNoImprove ← 0;
else
nNoImprove += 1;
if *nNoImprove* = *maxNoImprove* **then**
stoppingCriterion ← true;
break;
end if
end if
if random number $r1$ > *destoryprob1* **then**
Perturb (S_{C1}) ;
Redistribute (S_{C1});
if random number $r2$ < *converse-prob* **then**
gotoLKDP ← false;
end if
else
break;
end if
end while
end while

A. CONSTRUCTING INITIAL SOLUTION

As described above, the solving procedure for CluVRP followed the two-phase strategy and has two interdependent subtask. The first subtask is cluster assignment (i.e., partition clusters into variable vehicles), which can be regarded as a variant BPP problem. Considering the hard constraint in CluVRP, individual customers can be ignored and each cluster with different amount of customers can be regarded as a single "customer".

As shown in Figure 1(a), for an initial solution of CluVRP, only the service order at the cluster level was considered and the specific service order within clusters was ignored. To construct such initial solutions, an improved BFD heuristic, which is a famous BPP solution algorithm, is proposed. The BFD ranks item weights in decreasing order and puts them into fixed size boxes. Under the vehicle's loading capacity constraint, each item selects the box that suits it best. The meaning of "best" always represents a higher loading rate (i.e., a heavier one among candidate items is preferred.) However, in CluVRP, a better solution not only pursues a higher loading rate but also a shorter travel distance; therefore, it is preferred to arrange for the same vehicle to serve two or more geographically close clusters. The meaning of "best" for CluVRP is defined as the shortest distance between a current vehicle and its candidate clusters, which called SD-BFD algorithm.

A simple illustrative description is demonstrated as follows: suppose that all clusters are sorted by weights in descending order and denote as $C_m (m = 1,2,3 \dots M)$. They are loaded into V_K $(k = 1,2,3 \dots K)$ vehicles successively. The symbol d_{mk} represents the distance between C_m and V_K. In the beginning, all vehicles are empty and available from the depot. The biggest cluster C_1 can be assigned into any one of them. Suppose V_1 is selected, the current location of V_1 will move from the depot to C_1. For $m = 2$, d_{2k} is calculated for all in-transit vehicles (including V_1). The vehicle with the shortest distance is selected, and the location of the selected vehicle will move to C_2. Similarly, for $m = 3 \dots M$, the cluster assignment operation will be repeated until all clusters are loaded. It should be noted that the capacity constraint must be satisfied in the process of cluster assignment. If there are multiple vehicles with the same distance, the vehicle with the minimum idle space is preferred. Moreover, a visiting sequence of a vehicle also follows the assigned order during the process of cluster assignment. A small example is shown graphically in Figure 4. The six sequential subplots describe the whole process of cluster allocation and initial driving routes according to the SD-BFD algorithm.

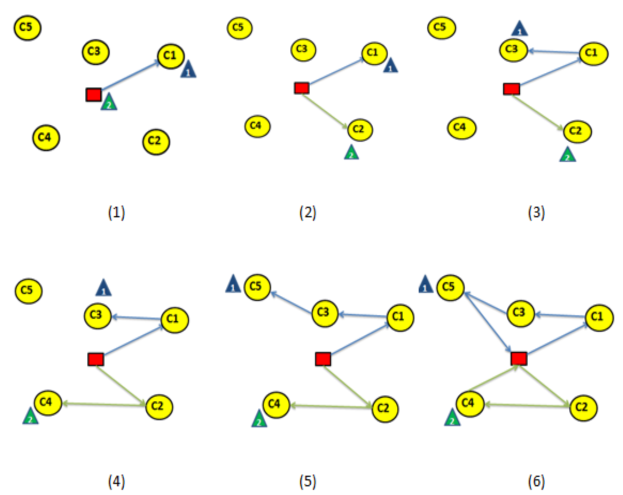


FIGURE 4. A small example illustration on the SD-BFD algorithm. The five yellow dots represent five clusters sorted by their weights($C1 > C2 > C3 > C4 > C5$). The red rectangle represents depot, the blue triangle represents vehicle 1 and the green one represents vehicle 2.

To address this process reasonably, appropriate measurement of distances between clusters must be considered. In CVRP, the distances between customers are symmetrical, and it is suitable to be calculated by Euclidean distance method. However, a cluster is a point set, and the distances between clusters are preferred to be measured by Hausdorff distance method.

The Hausdorff distance describes the distance between two point sets: Suppose there are two point sets $A = \{a_1, a_2, \dots, a_p\}$ and $B = \{b_1, b_2, \dots, b_q\}$. The Hausdorff distance is defined as below:

$$H(A,B) = max(h(A,B), h(B,A)) \tag{13}$$

Among them:

$$h(A,B) = max(a \in A)\, min(b \in B)\|a - b\| \tag{14}$$

$$h(B,A) = max(b \in B)\, min(a \in A)\|b - a\| \tag{15}$$

$\|\cdot\|$is the distance paradigm between point sets A and B. The two-way Hausdorff distance chooses the larger value between $h(A,B)$ and $h(B,A)$, which measures the maximum degree of mismatch between A and B. As shown in Figure 5.

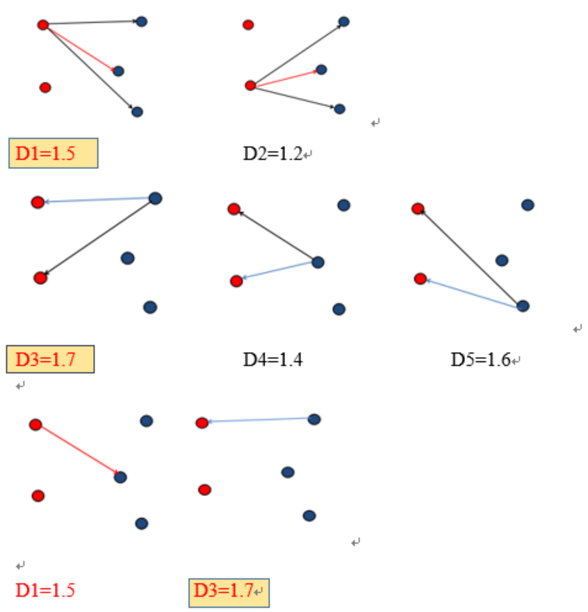


FIGURE 5. Hausdorff distance diagram. In the picture above, the red dot represents point set *A*, the blue dot represents point set *B*, and the red and blue arrows represent the minimum distance of the points in point sets *A* and *B* and the points in the other point set, respectively.

As can be seen, the Hausdorff distance is suitable to be employed for calculating distances between point sets. Another advantage is that there exist explicit entry-point and exit-point when visiting each cluster. Considering the diversity of VNS-LKDP, the parameter *sto-distance* was set. If the generated random number was greater than *sto-distance*, then the Hausdorff distance was used; if not, two customers were randomly selected to calculate their distance.

B OPTIMIZE THE INITIAL SOLUTION

1) INTER-CLUSTER ROUTE OPTIMIZATION

After the execution of the above steps, the initial feasible solution and the service order of each vehicle were determined. However, the performance is not very good.

With the development of optimization technology, more accurate and effective algorithms have been proposed for solving the VRP and related variants. This paper employed VNS to achieve gradual optimization at the cluster level. The VNS algorithm was proposed by [28] and inherited the relevant heuristic characteristics. The VNS is essentially an improved local search, and its key is to convert different neighborhoods. An alternate search using neighborhood structures composed of different actions achieves a good balance between concentration and evacuation. The VNS consists of two main parts:

First, a VND [28] organizes the framework of the VNS. When the search operator does not find a better solution, it will skip to the next neighborhood and continue searching. As shown by the red line in Figure 6. Otherwise, it will jump back to the first neighborhood to search again. As shown by the blue line in Figure 6.

Second, the shaking procedure is essentially a perturbation operator for generating neighborhoods. It is equivalent to the $N_k(k = 1,2, \dots, max)$ neighborhood in Figure 6. The relevant methods, including exact algorithms and heuristics, can all play a good role. The above two processes execute repeatedly until stopping rule is met.

In this paper, VNS was introduced into the solution of the CluVRP, and shaking of neighborhood, local search operator and acceptance rules were designed by optimization goals and metaheuristic strategies.

In a general way, the neighborhood structure construct includes: the number of neighborhood structure sets; the order between neighborhood structures; the strategies for moving between neighborhood structures. The local search operator design is another core part of the VNS. It introduces metaheuristics and strategies, such as first/best improvement strategies. This proposed VNS optimization process used several local neighborhood search operators as described in Table 2, and the complexity of per operator is $O(M^2)$ [25], where M represents the number of clusters.

TABLE 2. Local search operator design.

Single vehicle search operation	
Swap(1,1)	Exchange two clusters in the same vehicle.
Shift(1,0)	Move one cluster and insert it into another position in the same vehicle.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

Ruin	Randomly select i clusters, remove them from the current vehicle to get the partial solution of the problem; then re-insert the removed cluster into the partial solution to get a feasible solution.
2-opt[38]	Destroy two edges of the same vehicle, replace them with two new edges, and close the route.
Double vehicle search operation	
Swap(1,1)	Exchange two clusters in two vehicle routes.
Shift(1,0)	Move a cluster and insert it into a location in another vehicle.
Ruin	Randomly select i clusters, remove them from the current vehicle to get the partial solution of the problem; then re-insert the removed cluster into the partial solution to get a feasible solution.
2-opt	Destroy two edges of the two vehicles, replace them with two new edges, and close the route.

In addition, we adopted the most common acceptance rules. The acceptance rule could avoid the algorithm from falling into local optimum prematurely and ensure the diversity of the solution. Each time the execution reaches the VNS procedure, all neighborhoods will be checked sequentially. When the local search operator can find an improvement solution, the algorithm will jump back to the first neighborhood to search again. Otherwise, it will skip to the next neighborhood and continue searching. It continues until there's no neighborhood to improve the solution then a local optimum is reached at the cluster level.

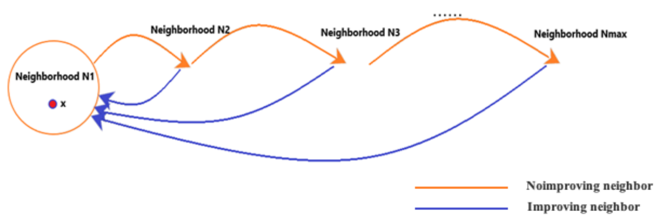


FIGURE 6. **VNS optimization process.**

2) INTRA-CLUSTER ROUTE OPTIMIZATION

The best inter-cluster route contains the service order of each vehicle and fixes the entry point and the exit point within the cluster. However, the service order within each cluster is still unknown. The intra-cluster route is similar to a TSP based on the analysis results. The difference is that in a TSP, vehicles must return back to the entry-point eventually in order to form a closed cycle, but for the situation here this is not required. In other words, the entry-point and the exit-point are not the same point.

To remedy this defect, a dummy point was added in each cluster. The dummy point connects to the entry-point and exit-point only and the connection distances are all zero. Other than this, there is no other way to reach the dummy point. When the incomplete TSP contains a dummy point, the final result must contain the sequence "entry-point A → dummy point O → exit-point G → point F → point E → point D → point C → point B", as shown in Figure 7. Then, we changed the route order, that is, we eliminated dummy point O and divided the route into two parts: "entry-point A" and "exit-point G → point F → point E → point D → point C → point B". Then, we reversed the second half, so that the order becomes "entry-point A → point B → point C → point D → point E → point F → exit-point G", as shown in Figure 8. Now, the entry-point and exit-point are treated as one point, and the model follows the TSP logic.

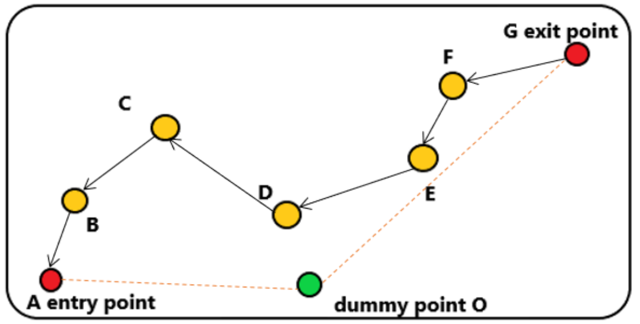


FIGURE 7. **The connection feature of a dummy point within a cluster. It is only connected to entry-point and exit-point and cannot connect to other customer points.**

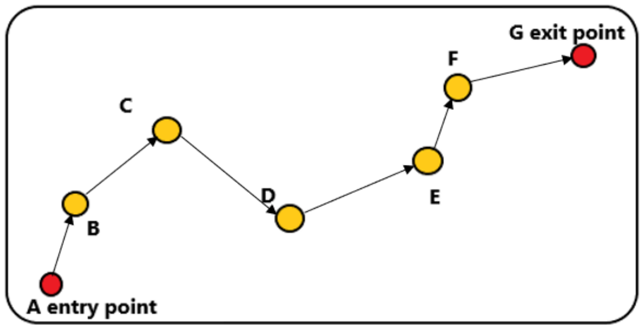


FIGURE 8. **The complete path formed by eliminating the dummy point and reversing partial paths.**

The Lin-Kernighan heuristic is proposed by Lin and Kernighan [31], which is one of the efficient heuristic optimization methods for solving symmetric TSP. The strategy of the candidate solution search is to move nodes from their own communities to other communities or exchange nodes between different communities. In the process of each iteration, the algorithm accepts the best candidate solution.

Since the Lin-Kernighan algorithm can obtain a tradeoff between solution quality and running time for solving large-scale problems [31], the LKDP was adopted to optimize the intra-cluster route problem with dummy point. Moreover, the LKDP was employed for optimizing customers routes in each cluster and the complexity of this process is $O(P^2 logP)$, where P represents the number of customers in a cluster.

C PERTURBATION AND REDISTRIBUTION

The overall optimal solution was obtained by merging the inter-cluster route and intra-cluster route results. To balance exploration and exploitation of VNS-LKDP, the obtained results also need to be augmented with a diversity control operation, which consists of a perturbation operator and a repair operator.

Disturbing customers by a perturbation operation may destroy the integrity of the cluster. Due to the hard constraint, a vehicle has to finish servicing all customers in the same cluster before moving to the next cluster. If the perturbation operation occurs at the customer level, it may lead to some customers are removed from their formed clusters, the integrity of the original cluster would be destroyed, which is contrary to the hard constraint. Therefore, the perturbation was carried out only at the cluster level, and customer order in each cluster remained unchanged. The perturbation operator consists of two parameters, *destoryprob1* and *destoryprob2*. The *destoryprob1* represents the possibility of a cluster being destroyed, and *destoryprob2* stands for the percentage of customers removed when a cluster has been destroyed. The clusters that were removed will be put into a separate table. The repair operators are required to assign the clusters in a separate table to the remaining available vehicles. If there is no suitable vehicle, then redistribution is performed.

After destroying the solution, there may be a situation in which some clusters can have no vehicle assigned. To ensure that all clusters are serviced, we constructed a redistribution operator.

Suppose that there are I vehicles sorted in descending order of the remaining space $W_i(i = 1,2,3 \dots I)$, each vehicle has loaded $N_i(i = 1,2,3 \dots I)$ clusters, and the weight of each cluster is $W_{iQ}(Q = 1,2,3 \dots N_i)$. For each vehicle, we sort W_{iQ} in ascending order. In addition, the weights of the removed clusters in the separate table are sorted as $M_j(j = 1,2,3 \dots J)$ in descending order. The goal is to put all the removed clusters in vehicles, therefore, $M_j(j = 1,2,3 \dots J)$ and $W_i(i = 1,2,3 \dots I)$ are compared in turn. Firstly, M_1 is compared with $W_i(i = 1,2,3 \dots I)$ one by one and selects the vehicle with the highest loading rate after loading M_1. If the loading rate reaches 100%, the vehicle will be removed. After each operation, all vehicles are sorted in descending order according to their remaining spaces. Next, $M_j(j = 1,2,3 \dots J)$ are packed in the same way.

Figure 9 is a small example to illustrate the redistribution operation. For example, at first, M_1 (weight equals 13) is considered to be loaded into vehicle *2* instead of vehicle *1*, because vehicle *2* is full after loading M_1. All vehicles are resorted according to the remaining space in descending order and the order of vehicles becomes *1*, *3*, *4*. Follow the same operation process, M_2, M_3 and M_4 (weights are 12, 8 and 5) are considered to be loaded into vehicle *1*, vehicle *3*, and vehicle *4*.

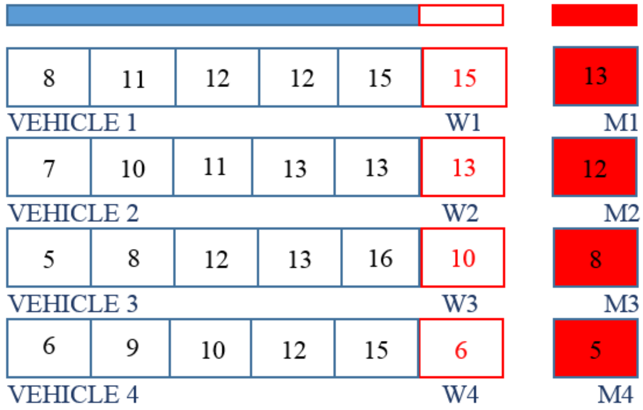


FIGURE 9. **A small example of redistribution operation with four vehicles. Values in blue rectangles represent the loaded capacity of vehicles. Clusters loaded on each vehicle are sorted in ascending order of demand. Values in red rectangles represent the remaining capacity of a vehicle. Vehicles are sorted in descending order of remaining capacity. Values in red filled rectangles represent weights of removed clusters due to perturbation operations. All removed clusters are arranged in descending order of according to weight.**

Note that the sum of vehicles capacity in instances is always greater than the aggregate demand of customers, so all removed clusters can be loaded. In addition, all clusters in a separate table are removed from original vehicles. Even for the cluster with the largest weight in a separate table, there is still a larger remaining space value.

The redistribution operation will be executed until all clusters are loaded, which can fix the broken solution. In this part, the parameter *converse-prob* was set to indicate the possibility of continuing to optimize by LKDP at the customer level after redistribution.

IV. COMPUTATIONAL EXPERIMENT

A INSTANCES DESCRIPTION

The effectiveness of the proposed hybrid algorithm was verified on instances with different customer sizes. The instances were divided into three sets: *A*, *B*, and *C*. The reasons for choosing these three instance sets are as follows: both *A* and *B* were obtained from the previous relevant literature and *C* was a real case data collected from an automobile parts manufacturer in China. The set *A* is an adaptation of CVRP instances and a seed-based algorithm was used to produce a clustering effect [15]. The set *B* is an adaptation of Golden instances proposed by [23], which has five subsets distinguished by the different filling percentage of a vehicle. Note that, compared to the set *A*, *B* takes account for the particular characteristic of "filling percentage" on CluVRP when generating the cluster information.

The first set of instances *A* consists of 79 small and medium-sized instances, and they are named according to the identifier *A-n32-k5-C11-V2*, where *A* represents the type of instance (including five classes: A, B, G, M and P), *n* represents the number of customers, *k* represents the maximum number of customers in a cluster, *C* represents the number of clusters, and *V* represents the number of vehicles.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

The second set of instances *B* was divided by a parameter ρ that represents the filling percentage of a vehicle. There are five different values of ρ (10%, 25%, 50%, 75% and 100%). This parameter constraints the maximum number of clusters that a vehicle can service. The smaller ρ is, the fewer customers there are in each cluster, and the more clusters can be loaded for each vehicle. When ρ equals to 100%, each vehicle can serve at most one cluster.

The third set of instances *C* consisted of operational data collected from a real-world automobile logistics company in Nanjing, China. The company divided suppliers all over China into three subareas for convenience of management. Note that the term "supplier" here denotes "customer" as mentioned above. The activities of each subarea are independent and Nanjing is a common depot. The southwest and north of Anhui are subarea 1, and the south of Zhejiang is subarea 2. Jiangsu and Shanghai are relatively important and dense subarea 3. In each subarea, the company divided clusters by geographical location. The three subareas include 11, 11 and 24 clusters each. The national supplier distribution map is shown in Figure 10. In addition, the supply of each supplier is constantly changing. The company updates demand at intervals of four days. We surveyed information including geographic latitude and longitude coordinates of all suppliers and the demand for each supplier from September 12 to 24.

FIGURE 10. **The parts manufactures distribution of the vehicle logistics company in China. (The three types of subareas are indicated by different colors and different shapes: red four-pointed star belongs to subarea 1, black triangle belongs to subarea 2, green circle belongs to subarea 3).**

B PARAMETERS TEST

To explore more search space of the algorithm and find more satisfactory solutions, the various parameters were set, and the optimal effect of the parameters was analyzed. Lastly, a total of four parameters were involved in the algorithm. The parameters and their best values are shown in Table 3.

The stopping criterion has a direct impact on the global convergence and timeliness of an algorithm. To achieve a tradeoff in both solution quality and calculation time, the algorithm is terminated if no improvement is obtained after *maxNoImprove* consecutive iterations in this study. Through pilot experiments on several types of instances, *maxNoImprove* is set to 1000 for all the test instances. All results are obtained with Intel(R) Xeon(R) CPU E5-2690 V3@2.60 GHz with 32 GB of RAM.

TABLE 3. **Experimental parameters.**

Parameter	Definition	Tested values	Best
destoryprob1	The possibility of destroying solution.	0,0.1,...,1	0.2
destoryprob2	Percentage of the solutions that are randomly destroyed by the perturbation operator.	0,0.1,...,1	0.1
sto-distance	The possibility of calculating the distance between clusters by Hausdorff.	0,0.1,...,1	0.4
converse-prob	The probability of optimizing after diversification.	0,0.1,...,1	0.5

C EXPERIMENTAL RESULTS

For instances A and B, the experimental results were compared with [25]. Although the solution approaches of the two papers are similar, there are still differences in the detailed analysis and solution algorithm.

The computation results of set *A* were shown in Table 4. In, *Instance* describes identifiers of 79 test instances. V_{V-L} represents the sum of all transportation route length calculated by VNS-LKDP proposed in this paper, and $V_{C.D}$ stands for the sum of all transportation route length obtained by the algorithm proposed in [25]. The best result between the two algorithms is recorded as $Best\ known$. Similarly, the fastest computing time is recorded as $T\_Best\ known$. The value of $ap = \frac{(V_{V-L} - Best\ known)}{Best\ known} \times 100\%$, $T\_Gap = \frac{(T_{V-L} - T\_Best\ known)}{T\_Best\ known} \times 100\%$. In the total 79 instances, the VNS-LKDP algorithm obtained 55 best values, 47 best computational time; while the algorithm in [25] obtained 53 best values, 49 best computational time. In addition, the value of average Gap is 0.62% and the value of average T_Gap is 1.57%. Figure 11 shows the number of optimal solutions obtained by both algorithms on different types of instances. As can be seen, both algorithms have similar performance on the instance set *A* with respect to the solution quality and computational time, but VNS-LKDP can obtain more optimal solutions on the first two classes of A and B.

TABLE 4. **Comparison of the results reposted by Christof Defryn et al. [25] and VNS-LKDP on set A.**

Instance	V_{V-L}	$V_{C.D}$	Best known	Gap(%)	T_ Gap(%)
A-n32-k5-C11-V2	522	522	522	0.00	0.00
A-n33-k5-C11-V2	475	482	475	0.00	0.00
A-n33-k6-C11-V2	565	582	565	0.00	0.00
A-n34-k5-C12-V2	554	547	547	1.28	0.00
A-n36-k5-C12-V2	589	589	589	0.00	0.00
A-n37-k5-C13-V2	575	569	569	1.05	0.00

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

A-n37-k6-C13-V2	629	667	629	0.00	0.00
A-n38-k5-C13-V2	507	507	507	0.00	0.00
A-n39-k5-C13-V2	618	618	618	0.00	0.00
A-n39-k6-C13-V2	626	644	626	0.00	0.00
A-n44-k6-C15-V2	756	760	756	0.00	0.00
A-n45-k6-C15-V3	720	712	712	1.12	0.00
A-n45-k7-C15-V3	668	689	668	0.00	0.00
A-n46-k7-C16-V3	681	671	671	1.49	0.00
A-n48-k7-C16-V3	694	683	683	1.61	0.00
A-n53-k7-C18-V3	651	651	651	0.00	0.00
A-n54-k7-C18-V3	742	742	742	0.00	1.23
A-n55-k9-C19-V3	673	688	673	0.00	0.00
A-n60-k9-C20-V3	832	821	821	1.34	0.00
A-n61-k9-C21-V4	686	707	686	0.00	0.00
A-n62-k8-C21-V3	780	824	780	0.00	0.00
A-n63-k10-C21-V4	818	841	818	0.00	0.00
A-n63-k9-C21-V3	921	1036	921	0.00	0.00
A-n64-k9-C22-V3	789	789	789	0.00	0.00
A-n65-k9-C22-V3	805	821	805	0.00	4.98
A-n69-k9-C23-V3	836	837	836	0.00	0.00
A-n80-k10-C27-V4	1009	990	990	1.92	0.00
B-n31-k5-C11-V2	375	376	375	0.00	5.56
B-n34-k5-C12-V2	416	416	416	0.00	5.43
B-n35-k5-C12-V2	563	563	563	0.00	1.45
B-n38-k6-C13-V2	447	541	447	0.00	0.00
B-n39-k5-C13-V2	321	321	321	0.00	0.00
B-n41-k6-C14-V2	478	481	478	0.00	1.22
B-n43-k6-C15-V2	425	415	415	2.41	0.00
B-n44-k7-C15-V3	449	449	449	0.00	7.24
B-n45-k5-C15-V2	511	511	511	0.00	0.00
B-n45-k6-C15-V2	408	410	408	0.00	0.00
B-n50-k7-C17-V3	467	467	467	0.00	3.23
B-n50-k8-C17-V3	676	667	667	1.35	0.00
B-n51-k7-C17-V3	585	585	585	0.00	0.00
B-n52-k7-C18-V3	427	427	427	0.00	0.00
B-n56-k7-C19-V3	435	435	435	0.00	2.77
B-n57-k7-C19-V3	640	634	634	0.95	0.00
B-n57-k9-C19-V3	770	773	770	0.00	0.00
B-n63-k10-C21-V3	728	715	715	1.82	0.00
B-n64-k9-C22-V4	526	526	526	0.00	0.00
B-n66-k9-C22-V3	694	731	694	0.00	0.00
B-n67-k10-C23-V4	626	626	626	0.00	0.00
B-n68-k9-C23-V3	609	697	609	0.00	0.00
B-n78-k10-C26-V4	731	721	721	1.39	6.67
G-n262-k25-C88-V9	3485	3430	3430	1.60	8.01
M-n101-k10-C34-V4	623	607	607	2.64	9.23
M-n121-k7-C41-V3	725	725	725	0.00	0.00
M-n151-k12-C51-V4	848	832	832	1.92	15.94
M-n200-k16-C67-V6	980	941	941	4.14	0.00
P-n101-k4-C34-V2	687	696	687	0.00	11.02
P-n16-k8-C6-V4	253	253	253	0.00	0.00
P-n19-k2-C7-V1	186	186	186	0.00	7.02
P-n20-k2-C7-V1	200	200	200	0.00	0.00
P-n21-k2-C7-V1	190	190	190	0.00	5.77
P-n22-k2-C8-V1	212	202	202	4.95	0.00
P-n22-k8-C8-V4	365	365	365	0.00	0.00
P-n23-k8-C8-V3	288	298	288	0.00	0.00
P-n40-k5-C14-V2	412	396	396	4.04	9.44
P-n45-k5-C15-V2	454	440	440	3.18	0.00
P-n50-k10-C17-V4	504	506	504	0.00	0.00
P-n50-k7-C17-V3	454	460	454	0.00	0.00
P-n50-k8-C17-V3	473	465	465	1.72	1.76
P-n51-k10-C17-V4	538	538	538	0.00	12.01
P-n55-k10-C19-V4	519	512	512	1.37	0.00
P-n55-k15-C19-V6	618	607	607	1.81	0.00
P-n55-k7-C19-V3	474	474	474	0.00	5.23
P-n55-k8-C19-V3	489	473	473	3.38	0.00
P-n60-k10-C20-V4	565	570	565	0.00	1.12
P-n60-k15-C20-V5	624	657	624	0.00	0.00
P-n65-k10-C22-V4	626	626	626	0.00	0.00
P-n70-k10-C24-V4	652	648	648	0.62	0.00
P-n76-k4-C26-V2	597	597	597	0.00	6.16
P-n76-k5-C26-V2	592	592	592	0.00	0.00
#Avergae				0.62	1.57

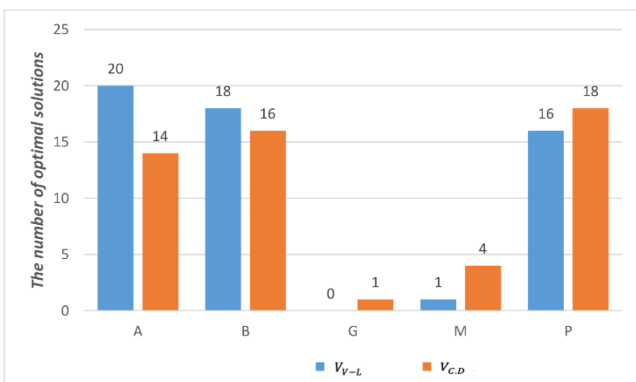


FIGURE 11. **The number of optimal solutions obtained by VNS-LKDP and the algorithm proposed by Christof Defryn et al. (2017) in different classes *A*, *B*, *M*, *G* and *P*.**

Next, the experimental results of B were compared with [25]. As mentioned before, the set B has five subsets classified by the parameter ρ (i.e. filling percentage), which is a distinct characteristic of CluVRP. Tables 5 to 9 compared the results obtained by Christof Defryn et al. [25] and VNS-LKDP on the set B. Every table includes the same 20 subinstances, but the difference lies in the different setting of parameter ρ . The larger ρ means the more customer demand contained in each cluster, and the fewer clusters a vehicle can service. Five ρ values were designed: 10%, 25%, 50%, 75% and 100%, corresponding to Tables 5 to 9. In these tables, *Id*, *n+1*, *Q* represents the instances index, the number of customers, the vehicle capacity, respectively. $V_{C.D}$ represents the calculation of the algorithm in [25]. $Best\ known$ stands for the best results of the two algorithms and $Gap = \frac{(V_{V-L} - Best\ known)}{Best\ known} \times 100\%$. The results showed that when ρ equals 10%, 25% and 50%, the average high-quality solution obtained by VNS-LKDP is 8, and the advantage is not obvious. However, when ρ equals 75%, the algorithm obtained 12 high-quality solutions in 20 instances. When ρ increases to 100%, the algorithm obtained 14 high-quality solutions in 20 instances, which is significantly better than [25]. We calculated the number of $Gap = 0$ and $Gap \geq 0$ obtained by VNS-LKDP under different values of parameter ρ. The results are summarized in Figure 12 and two trend lines were added to show the change of the calculation results under different values of parameter ρ intuitively. As can be seen, as the value of ρ increases, the number of $Gap = 0$ increases as well while the number of $Gap > 0$ decreases, which means the performance of VNS-LKDP is superior with the increasing of the filling percentage. Note that the VNS-LKDP also performed in a balanced manner for different ρ values. Even when ρ was low, it could still obtain excellent solutions on nearly half of the instances. It overcomes the problem of high ρ value yielding almost no high-quality solutions [25].

The reason for better performance is that more customers are included in a cluster, which means that the larger parameter is more significant for large-scale instances. In such cases, the advantage of the LK algorithm is demonstrated. When ρ equals 100%, there are 40-50 customers in a cluster, and one vehicle can service at most one cluster. The situation is close to that of China's express delivery. Regardless of what brand of express delivery, for hierarchical management, they all prefer to build first-level and second-level sites. Almost all first-level sites must serve dozens of second-level sites. One second-level site must serve hundreds of express outlets in a city. If thousands of express outlets across the country are to be clustered into only two or three clusters, the efficiency will be low. Therefore, the major express companies are more inclined to divide 50 or more express outlets into one cluster and arrange one or more vehicles for distribution at the same time. The algorithm designed in this paper is well adapted to the instances with higher ρ value than [25] and is also more suitable for the real China logistic background.

TABLE 5. **Comparison of the results reported by Christof Defryn et al. [25] and VNS-LKDP on set B with ρ =10%.**

Id	n+1	Q	$V_{C.D}$	V_{V-L}	Best known	Gap(%)
1	240	550	6054.42	6138.85	6054.42	1.39
2	320	700	9689.60	9832.42	9689.60	1.47
3	400	900	13963.40	13812.32	13812.32	0.00
4	480	1000	18792.40	17995.62	17995.62	0.00
5	200	900	9570.53	8968.68	8968.68	0.00
6	280	900	11356.97	11380.20	11356.97	0.20
7	360	900	13698.96	13942.73	13698.96	1.78
8	440	900	14193.92	14342.95	14193.92	1.05
9	255	1000	876.09	829.29	829.29	0.00
10	323	1000	927.20	926.12	926.12	0.00
11	399	1000	1157.52	1163.45	1157.52	0.51
12	483	1000	1400.19	1465.83	1400.19	4.69
13	252	1000	1286.55	1183.94	1183.94	0.00
14	320	1000	1374.24	1396.95	1374.24	1.65
15	396	1000	2113.68	2194.92	2113.68	3.84
16	480	1000	2090.63	2132.95	2090.63	2.02
17	240	200	870.02	878.15	870.02	0.93
18	300	200	1120.47	1139.34	1120.47	1.68
19	360	200	1591.16	1647.06	1591.16	3.51
20	420	200	2138.94	2107.59	2107.59	0.00
Best solution			13	7		

TABLE 6. **Comparison of the results reported by Christof Defryn et al. [25] and VNS-LKDP on set B with ρ =25%.**

Id	n+1	Q	$V_{C.D}$	V_{V-L}	Best known	Gap(%)
1	240	550	6198.11	6254.23	6198.11	0.91
2	320	700	9860.23	9941.56	9860.23	0.82
3	400	900	14098.61	14067.43	14067.43	0.00
4	480	1000	17622.14	17602.89	17602.89	0.00
5	200	900	9857.87	9917.76	9857.87	0.61
6	280	900	11411.12	11480.20	11411.12	0.61
7	360	900	12608.30	12549.21	12549.21	0.00
8	440	900	14638.77	14700.93	14638.77	0.42
9	255	1000	744.68	754.65	744.68	1.34
10	323	1000	1000.82	1078.42	1000.82	7.75
11	399	1000	1153.54	1120.76	1120.76	0.00
12	483	1000	1401.79	1467.54	1401.79	4.79
13	252	1000	1055.45	1107.32	1055.45	4.91
14	320	1000	1322.89	1365.33	1322.89	3.21
15	396	1000	1740.86	1734.65	1734.65	0.00
16	480	1000	2136.72	2112.75	2112.75	0.00
17	240	200	830.85	830.85	830.85	0.00
18	300	200	1573.76	1573.76	1573.76	0.00
19	360	200	1571.23	1609.31	1571.23	2.42

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

20	420	200	2086.07	2099.62	2086.07	0.65
Best solution			14	8		

TABLE 7. Comparison of the results reported by Christof Defryn et al. [25] and VNS-LKDP on set B with ρ =50%.

Id	n+1	Q	$V_{C.D}$	V_{V-L}	Best known	Gap(%)
1	240	550	6683.49	6690.65	6683.49	0.11
2	320	700	9983.51	9974.98	9974.98	0.00
3	400	900	13482.81	13543.67	13482.81	0.52
4	480	1000	17813.00	18123.73	17813.00	1.74
5	200	900	8642.96	8642.96	8642.96	0.00
6	280	900	10712.92	10712.92	10712.92	0.00
7	360	900	12981.63	13007.78	12981.63	0.20
8	440	900	13988.32	14002.13	13988.32	0.10
9	255	1000	711.41	731.41	711.41	2.81
10	323	1000	905.97	878.58	878.58	0.00
11	399	1000	1147.83	1130.76	1130.76	0.00
12	483	1000	1328.55	1317.65	1317.65	0.00
13	252	1000	1518.85	1587.66	1518.85	4.53
14	320	1000	1356.84	1289.98	1289.98	0.00
15	396	1000	1859.45	1832.78	1832.78	0.00
16	480	1000	2035.39	2054.43	2035.39	0.94
17	240	200	898.28	938.28	898.28	4.54
18	300	200	1206.60	1289.46	1206.60	6.87
19	360	200	1648.16	1667.24	1648.16	1.16
20	420	200	2319.69	2367.09	2267.09	4.41
Best solution			14	8		

TABLE 8. Comparison of the results reported by Christof Defryn et al. [25] and VNS-LKDP on set B with ρ =75%.

Id	n+1	Q	$V_{C.D}$	V_{V-L}	Best known	Gap(%)
1	240	550	6783.31	6845.23	6783.31	0.91
2	320	700	10291.32	10291.32	10291.32	0.00
3	400	900	13750.00	13797.31	13750.00	0.34
4	480	1000	17306.67	17306.67	17306.67	0.00
5	200	900	8747.93	8799.34	8747.93	0.59
6	280	900	13078.39	13092.44	13078.39	0.11
7	360	900	11636.91	11563.28	11563.28	0.00
8	440	900	14059.73	13892.76	13892.76	0.00
9	255	1000	776.02	745.09	745.09	0.00
10	323	1000	1006.56	987.52	987.52	0.00
11	399	1000	1230.21	1198.94	1198.94	0.00
12	483	1000	1484.29	1500.68	1484.29	1.10
13	252	1000	1185.74	1200.45	1185.74	1.24
14	320	1000	1524.50	1492.76	1492.76	0.00
15	396	1000	1828.22	1810.42	1810.42	0.00
16	480	1000	2272.24	2208.98	2208.98	0.00
17	240	200	1001.09	1001.09	1001.09	0.00
18	300	200	1393.52	1421.54	1393.52	2.01
19	360	200	1952.73	1952.73	1952.73	0.00
20	420	200	2541.01	2531.09	2531.01	0.00
Best solution			11	12		

TABLE 9. Comparison of the results reported by Christof Defryn et al. [25] and VNS-LKDP on set B with ρ =100%.

Id	n+1	Q	$V_{C.D}$	V_{V-L}	Best known	Gap(%)
1	240	550	6325.52	6325.52	6325.52	0.00
2	320	700	9986.65	9975.72	9975.72	0.00
3	400	900	12573.91	12550.30	12550.30	0.00
4	480	1000	16380.45	16336.71	16336.71	0.00
5	200	900	8450.99	8546.33	8450.99	1.13
6	280	900	10895.37	11069.53	10895.37	1.60
7	360	900	11477.87	11469.11	11469.11	0.00
8	440	900	13406.50	13355.37	13355.37	0.00
9	255	1000	708.21	700.10	700.10	0.00
10	323	1000	846.02	842.69	842.69	0.00
11	399	1000	1062.66	1057.00	1057.00	0.00
12	483	1000	1306.57	1305.97	1305.97	0.00
13	252	1000	999.41	1030.32	999.41	3.09
14	320	1000	1227.96	1248.88	1227.96	1.70
15	396	1000	1540.13	1530.25	1530.25	0.00
16	480	1000	1878.36	1924.51	1878.36	2.46
17	240	200	845.47	845.47	845.47	0.00
18	300	200	1215.38	1207.61	1207.61	0.00
19	360	200	1668.83	1668.83	1668.83	0.00
20	420	200	2131.20	2202.12	2131.20	3.33
Best solution			9	14		

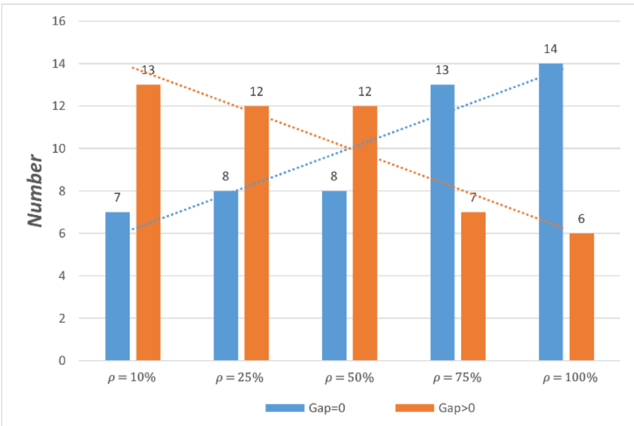


FIGURE 12. The number of $Gap = 0$ and $Gap > 0$ obtained by VNS-LKDP under different values of parameter ρ.

Lastly, we extracted a part of company operational data as instances C. The distribution of parts manufacturers is derived from the real world of China. The instances of three independent subareas were tested by VNS-LKDP and compared with the company's manually calculation. For a complete overview of the detailed results, we can refer to Appendix. In addition, here we defined $Gap = \frac{V_m - V_{V-L}}{V_m} * 100\%$, which represents the improvement percentage of VNS-LKDP compared to the company's manual results. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2931622, IEEE Access

larger the improvement percentage is, the better the performance of VNS-LKDP. We summarized Gap for all the three subareas at different times, which is presented in Figure 13. As can be seen, the subarea 3 is densest and the improvement is also the most obvious, which means that the denser the area, the better VNS-LKDP performs. Table 10 presents a specific example obtained by VNS-LKDP and manual result on Oct. 2. Trip0 to Trip9 represents a total of nine vehicle routes details. For example, in Trip 0, “DIST = 104” represents the length of the transport route is 104, and KC148 KC431 KC534 KC562 KC433 KC431CZ1 represents the specific sequence of the vehicle's transport route.

Through the analysis of external geographical factors and internal transport modes, it was found that the distribution of suppliers is getting denser and each cluster contains more suppliers among the three subareas. As can be seen in Tables 11-13, the performance of the VNS-LKDP algorithm is getting better as well when compared with manual results. The reason is that VNS-LKDP has an obvious advantage in solving CluVRP with large-scale and relatively dense areas.

In addition, from the perspective of time consumption, the maximum running time of VNS-LKDP is 4.67 second, but normally a company's manual calculation needs 2 to 3 hours. The scattered distribution of parts manufacturers makes their distances significantly different, which helps vehicles choose the next service point. The situation in dense distribution areas is the opposite, vehicles may spend more time in route planning. In this case, VNS-LKDP is more advantageous than manual work. The excellent calculation cost certainly makes this algorithm has practical application value. Besides, some of the results obtained by manual calculation were exactly the same as those by VNS-LKDP when solving small-scale problems. The possible reason is that the suppliers’ distribution in these areas is relatively scattered and it is also easy to be calculated with manual experience as well. Even so, VNS-LKDP has incomparable advantages in terms of computing time.

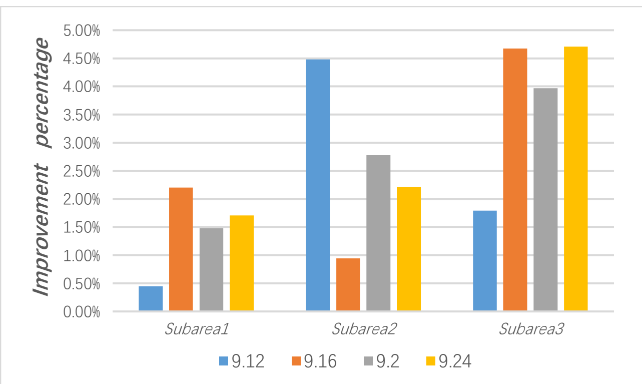


FIGURE 13. **At different times in the three subareas, the improvement percentage of VNS-LKDP compared to the company's manual results.**

TABLE 10. **Comparison of the specific route order obtained by the VNS-LKDP and company's manual calculation in subarea 3 on Oct. 2.**

Instances	Specific route of each vehicle
VNS-LKDP	Trip 0 (DIST=104) =1 KC148 KC534 KC431 KC562 KC433 KC431CZ 1
	Trip 1 (DIST=209) =1 KC612 KC619 KC206 KC664 KC100 KC131 KC149 KC170 KC179 1
	Trip 2 (DIST=93) =1 KC150 KC167 KC389 KC584 KC592 KC660 KC7191
	Trip 3(DIST=213) =1 KC406 KC445 KC617 KC420 KC424 KC659 KC132X KC178 KC163 KC188 1
	Trip 4 (DIST=77) =1 KC240 KC421 KC548TC KC644 KC451 KC456 1
	Trip 5 (DIST=140) =1 KC125 KC103 KC157 KC264 KC345 KC448 KC681 LONGTAN 1
	Trip 6 (DIST=69) =1 KC126 KC171 KC431CS KC190 KC334 KC6841
	Trip 7 (DIST=131) =1 KC005 KC175 KC459 KC553 KC176 KC388 KC607 1
	Trip 8 (DIST=159) =1 KC105 KC124 KC133WX KC208 KC589 KC616 KC762 1
	Trip 9 (DIST=91) =1 KC251YZ KC385 KC390 KC736 1
MANUAL	Trip 0 (DIST=110) =1 KC148 KC534 KC431 KC562 KC433 KC431CZ 1
	Trip 1 (DIST=224) =1 KC612 KC619 KC206 KC664 KC100 KC131 KC149 KC170 KC179 1
	Trip 2 (DIST=93) =1 KC150 KC167 KC389 KC584 KC592 KC660 KC7191
	Trip 3(DIST=213) =1 KC406 KC445 KC617 KC420 KC424 KC659 KC132X KC178 KC163 KC188 1
	Trip 4 (DIST=77) =1 KC240 KC421 KC548TC KC644 KC451 KC456 1
	Trip 5 (DIST=143) =1 KC125 KC103 KC157 KC264 KC345 KC448 KC681 LONGTAN 1
	Trip 6 (DIST=67) =1 KC126 KC171 KC431CS KC190 KC334 KC6841
	Trip 7 (DIST=131) =1 KC005 KC175 KC459 KC553 KC176 KC388 KC607 1
	Trip 8 (DIST=159) =1 KC105 KC124 KC133WX KC208 KC589 KC616 KC762 1
	Trip 9 (DIST=91) =1 KC251YZ KC385 KC390 KC736 1

VII. CONCLUSION

This paper discussed the CluVRP problem derived from a real-world logistics company, which is a new variant of classical CVRP. The demand distribution of CVRP is scattered and irregular, but CluVRP has its particular hard constraint: the geographical location of parts manufacturers had already presented clustering distribution before planning the route schedule and all customers in a cluster have to be served consecutively by the same vehicle. The CluVRP seems to be more in line with the current route planning scenario in world.

For solving CluVRP, we followed the two-phase strategy and disassembled CluVRP into two subproblems. For the first subproblem of cluster assignment, the SD-BFD heuristic was developed by combining with the Hausdorff distance to measure distances between clusters. The second subproblem is to optimize the constructed initial solutions on the inter-cluster level and the intra-cluster level successively. Considering the specific characteristics on each level, VNS and LKDP were adapted to improve the routes respectively. Furthermore, to balance exploration and exploitation, a perturbation operator and a redistribution operator were developed to augment the diversification capacity of VNS-LKDP. The performance of VNS-LKDP has been shown through a number of computational experiments on the classical benchmark instance sets and the real data set from a Chinese logistics company.

There are still many directions for future research. For example, delivery vehicles can be heterogeneous because companies prefer to build multiple models for distribution activities. Second, time window requirements are becoming more stringent, and daily distribution must meet the supplier's time requirements. Third, as proposed in [25], the CluVRP can meet soft constraints, and vehicles can service customers in other clusters in the distribution process and then return to the previous cluster to continue to service. The study of soft constraints has practical significance and saves transportation costs. In addition, in terms of solving algorithms, not only exact algorithms and heuristic algorithms but also design machine learning and deep learning can be explored.

In addition, the clustering method is a fundamental direction that can be studied: This paper was based on clustering, so how is the background of clustering built? There is no mention of this in previous papers. Is the principle of geographical proximity truly the most effective, or is the delineation of cluster boundaries truly as reasonable? There are many methods for clustering such as K-MEANS, which is the oldest hierarchical clustering method, and DBSCAN, which is the most common method and is very popular. If clustering customers can be reasonably executed before building the model, the result may also be improved. These can be expanded into future research directions.

APPENDIX. DETAILED RESULTS

TABLE 11. Comparison of the distribution mileage of the ten time segments in subarea 1 reported by VNS-LKDP and company's manual calculation.

Id	Cluster(Cities)	Supplier Code	9.12	9.16	9.20	9.24
1	An Hui	KC308,KC566,KC733	14	13	15	14
2	Bao Ding, Bei Jing	KC006,KC392,KC550, KC610	15	14	14	18
3	Chong Qing1	KC169,KC18,KC189	23	16	16	19
4	Chong Qing2	KC281,KC505,KC527	14	16	20	22
5	Chong Qing3	KC570,KC731	14	15	11	12
6	Tian Jing1	KC183,KC184,KC266,KC449	20	11	12	12
7	Tian Jing2	KC552,KC588,KC614	10	12	13	12
8	Chang Shan, Cheng Du	KC135,KC549,KC558,KC663	16	12	13	11
9	Wu Han	KC613	6	7	8	4
10	Da Lain	KC278,KC359	9	7	5	5
11	Qing Dao, Yan Tai	KC450,KC286,KC454,KC752	19	16	18	16
	VNS+LKDP		443	400	399	403
	Original company solver results		445	409	405	410
	Time_Consumed		3.78s	4.01s	3.94s	4.13s

**Id*: The index of each subarea.

**Cluster (Cities)*: The cities contained in per cluster.

**Supplier Code:* The parts suppliers contained in per cluster.

**Date (From 9.12 to 9.24):* The demand of parts suppliers at different time periods.

TABLE 12. Comparison of the distribution mileage of the ten time segments in subarea 2 reported by VNS-LKDP and company's manual calculation.

Id	Cluster(Cities)	Supplier Code	9.12	9.16	9.20	9.24
1	Guang Zhou	KC116,KC309,KC310,KC437	25	24	22	22
2	HuiZhou, ZhongShan	KC241,KC618	12	12	12	14
3	Fou Shan	KC323,KC492	10	10	11	10
4	Ning Bo1	KC004,KC118,KC298,KC300	18	19	20	19
5	Ning Bo2	KC528,KC565,KC621	15	15	14	13
6	Ning Bo3	KC685,KC763	10	12	11	14
7	Jia Xing1	KC279	6	6	5	6
8	Jia Xing2	KC409JX,KC412,KC587	9	11	12	11
9	Tai Zhou Hang Zhou	KC240,KC243,KC507,KC575	21	22	22	21
10	Fu Zhou, Xia Meng	KC226,KC230,KC333,KC561	25	26	26	24
11	Zhe Jiang	KC409NJ,KC229	18	17	17	18

	VNS+LKDP	533	526	525	530
	Original company solver results	558	531	540	542
	Time_Consumed	3.10s	3.56s	3.43s	3.55s

**Id*: The index of each subarea.

**Cluster (Cities)*: The cities contained in per cluster.

**Supplier Code:* The parts suppliers contained in per cluster.

**Date (From 9.12 to 9.24):* The demand of parts suppliers at different time periods.

TABLE 13. Comparison of the distribution mileage of the ten time segments in subarea 3 reported by VNS-LKDP and company's manual calculation.

Id	Cluster(Cities)	Supplier Code	9.12	9.16	9.20	9.24
1	Chang Shu	KC126,KC171, KC190,KC431CS	11	12	12	13
2	Chang Zhou1	KC148,KC431	7	7	8	7
3	Chang Zhou2	KC433,KC431CZ	8	8	8	9
4	Chang Zhou3	KC534,KC562	6	5	6	7
5	Jing Jiang	KC334,KC684	10	9	11	13
6	Kun Shan	KC05,KC175,KC459,KC553	16	19	18	18
7	Lian Yungang	KC176	3	4	3	4
8	Nan Jing1	KC103,KC125, KC157,KC264	22	20	22	20
9	Nan Jing2	KC345,KC448	10	8	9	8
10	Nan Jing3	KC681,LONGTAN	8	8	7	9
11	Nan Tong	KC388,KC607	6	5	7	7
12	Shang Hai (Fu Dong)1	KC100,KC131,KC149, KC170, KC179	30	33	32	33
13	Shang Hai (Fu Dong)2	KC206,KC612,KC619, KC664	22	25	26	24
14	Shang Hai (Fu Xi)1	KC150,KC167, KC389,KC584,KC592, KC660	36	33	35	35
15	Shang Hai (Fu Xi)2	KC719	28	29	29	28
16	Su Zhuo1	KC406,KC420,KC424, KC445,KC617,KC659	44	45	50	45
17	Su Zhuo2	KC132X,KC163,KC178, KC188	25	24	24	25
18	Tai Zhou	KC240	8	8	7	7
19	Tai Cang	KC421,KC548TC,KC644	16	15	15	16
20	Wu Xi1	KC105,KC124,KC133WX, KC208	23	23	22	22
21	Wu Xi2	KC589,KC616,KC762	15	17	19	19
22	Yi Zheng1	KC251YZ,KC385,KC390	11	13	13	14
23	Yi Zheng2	KC736	12	12	11	11
24	Zhang JiaGang	KC451,KC456	9	8	8	9
	VNS+LKDP		1314	1325	1356	1335
	Original company solver results		1338	1390	1412	1401
	Time_Consumed		4.12s	4.54s	4.39s	4.67s

**Id*: The index of each subarea.

**Cluster (Cities)*: The cities contained in per cluster.

**Supplier Code:* The parts suppliers contained in per cluster.

**Date (From 9.12 to 9.24):* The demand of parts suppliers at different time periods.

ACKNOWLEDGMENT

The authors are very grateful to the Editor-in-Chief and the anonymous reviewers for their constructive comments and suggestions that have further helped to improve the quality and presentation of this paper. In addition, the authors express their sincere thanks to Christof Defryn, who selflessly provided the program and data that contributed to the success of our work.

REFERENCES

[1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manage. Sci.*, vol. 6, no. 1, pp. 80-91, 1959.

[2] E. Alba and B. Dorronsoro, "Computing nine new best-so-far solutions for capacitated VRP with a cellular genetic algorithm," *Inform. Process. Lett.*, vol. 98, no. 6, pp. 225-230, 2006.

[3] J. K. Lenstra and A. Kan, "Complexity of vehicle routing and scheduling problems," *Networks.*, vol. 11, no. 2, pp. 221-227, 1981.

[4] G. Ghiani and G. Improta, "An efficient transformation of the generalized vehicle routing problem," *Eur. J. Oper. Res.*, vol. 122, no. 1, pp. 11-17, 2000.

[5] P. C. Pop, I. Kara, and A. H. Marc, "New mathematical models of the generalized vehicle routing problem and extensions," *Appl. Math. Model.*, vol. 36, no. 1, pp. 97–107, 2012.

[6] X. Zhao, P. Y. Wu, G. Ren, K. Ji, and W. W. Qian, "Clustering analysis on ridership patterns at subway stations: A case in Nanjing," *J. Urban. Plan. Development.*, vol. 145, no. 2, pp. 97–107, 2019.

[7] K. Lin, F. Xia, and G. Fortino, "Data-driven for clustering multimedia communication in Internet of vehicles," *Future. Gener. Comp. Sy.*, vol. 94, pp. 610-619, 2019.

[8] Y. Wang, X. Ma, Y. Lao, and Y. Wang, "A Fuzzy-based Customer Clustering Approach with Hierarchical Structure for Logistics Network Optimization," *Expert. Syst. Appl.*, vol. 41, no. 2, pp. 521-534, 2014.

[9] Y. Wang, X. Ma, Z. Li, Y. Liu, M. Xu, and Y. Wang, "Profit Distribution in Collaborative Multiple Centers Vehicle Routing Problem," *J. Clean. Prod.*, vol. 144, pp. 203-219, 2017.

[10] Y. Wang, X. Ma, M. Liu, K. Gong, Y. Liu, M. Xu, and Y. Wang, "Cooperation and Profit Allocation in Two-echelon Logistics Joint

Distribution Network Optimization," *Appl. Soft. Comput.*, vol. 56, pp. 143-157, 2017.
[11] I. Charon and O. Hudry, "The noising method: A new method for combinatorial optimization," *Oper. Res. Lett.*, vol. 14, no. 3, pp. 133-137, 1993.
[12] M. Sevaux and K. Sorensen, "Hamiltonian paths in large clustered routing problems," *in Proc. the EU/MEeting 2008 workshop on Metaheuristics for Logistics and Vehicle Routing*, vol. 8, pp. 411–417, 2008.
[13] T. Barthélemy, A. Rossi, M. Sevaux, and K. Sörensen, "Metaheuristic approach for the clustered VRP," *in Proc. the EU/MEeting 10th Anniversary of the Metaheuristics Community-Universitéde Bretagne Sud, France,* 2010.
[14] A. H. Marc, L. Fuksz, P. C. Pop, and D. Danciulescu, "A novel hybrid algorithm for solving the clustered vehicle routing problem," *Lect. Notes. Artif. Int.*, vol. 9121, pp. 679-689, 2015.
[15] M. Battarra, G. Erdoğan, and D. Vigo, "Exact Algorithms for the Clustered Vehicle Routing Problem," *Oper. Res.*, vol. 62, no. 1, pp. 58-71, 2014.
[16] T. Bektaş, G. Erdoğan, and S. Røpke, "Formulations and Branch-and-Cut Algorithms for the Generalized Vehicle Routing Problem," *Transport. Sci.*, vol. 45, no. 3, pp. 299–316, 2011.
[17] Y. Qiu, L. Wang, X. Xu, and P. Pardalos, "Formulations and branch-and-cut algorithms for multi-product multi-vehicle production routing problems with startup cost," *Expert. Syst. Appl.*, vol. 98, pp. 1-10, 2018.
[18] T. Vidal, M. Battarra, A. Subramanian, and G. Erdoğan, "Hybrid metaheuristics for the Clustered Vehicle Routing Problem," *Comput. Oper. Res.*, vol. 58, pp. 87-99, 2015.
[19] A. Subramanian, and M. Battarra, "An iterated local search algorithm for the Travelling Salesman Problem with Pickups and Deliveries," *J. Oper. Res. Soc.*, vol. 64, no. 3, pp. 402-409, 2013.
[20] P. H. V. Penna, A. Subramanian, and L. S. Ochi, "An Iterated Local Search heuristic for the Heterogeneous Fleet Vehicle Routing Problem," *J. Heuristics.*, vol. 19, no. 2, pp. 201–232, 2013.
[21] R. Bowerman, P. Calamiand, and G. B. Hall, "The spacefilling curve with optimal partitioning heuristic for the vehicle routing problem," *Eur. J. Oper. Res.*, vol. 76, no. 1, pp. 128-142, 1994.
[22] C. Defryn and K. Sörensen, "A Two-Level Variable Neighbourhood Search for the Euclidean Clustered Vehicle Routing Problem," *Faculty. Appl. Economics.*, 2015.
[23] C. Expósito-Izquierdo, A. Rossi, and M. Sevaux, "A Two-Level solution approach to solve the Clustered Capacitated Vehicle Routing Problem," *Comput. Ind. Eng.*, vol. 91, pp. 274-289, 2016.
[24] F. Lia, B. Goldenb, and E. Wasilc, "A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem," *Comput. Oper. Res.*, vol. 34, pp. 2734 – 2742, 2007.
[25] C. Defryn and K. Sörensen, "A fast two-level variable neighborhood search for the clustered vehicle routing problem," *Comput. Oper. Res.*, vol. 83, pp. 78-94, 2017.
[26] P. Hansen and N. Mladenovic, "Variable neighborhood search-Principles and applications," *Eur. J. Oper. Res.*, vol. 130, no. 3, pp. 449-467, 2001.
[27] P. Hansen, N. Mladenović, and J. A. M. Pérez, "Variable neighbourhood search: methods and applications," *Ann. Oper. Res.*, vol. 175, no. 1, pp. 367-407, 2009.
[28] N. Mladenovirt and P. Hansent, "Variable neighborhood search," *Comput. Oper. Res.*, vol. 24, no. 11, pp. 1097-1100, 1997.
[29] T. Hintsch and S. Irnich, "Large multiple neighborhood search for the clustered vehicle-routing problem," *Eur. J. Oper. Res.*, vol. 270, no. 1, pp. 118–131, 2018.
[30] P. C. Popa, L. Fukszb, A. H. Marca, and C. Saboa, "A novel two-level optimization approach for clustered vehicle routing problem," *Comput. Ind. Eng*, vol. 115, pp. 304–318, 2018.
[31] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling salesman problem," *Oper. Res.*, vol. 21, no. 2, pp. 498–516, 1973.
[32] M. R. Garey and D. S. Johnson, "Strong NP-completeness results motivation, examples, and implications," *Mathematical. Res*, vol. 25, no. 3, pp. 499-508, 1978.
[33] K. Scheinberg, S. Santanu, V. Daniele, and P.Toth, "Vehicle Routing: Problems, Methods, and Applications," *Soc. Ind. Appl. Mathematics.*, 2014.
[34] K. Fleszar and K. S. Hindi, "New heuristics for one-dimensional bin-packing," *Comput. Oper. Res.*, vol. 29, no. 7, pp. 821-839, 2002.
[35] G. Gutin and A. P. Punnen, "Local Search and Metaheuristic," in *The traveling salesman problem and its variations (combinatorial optimization)*, New York, NY, USA: Kluwer Academic, 2007, pp. 193-205, 2007. [Online]. Available: http://www.doc88.com/
[36] O. Braysy and M. Gendreau, "Vehicle Routing Problem with Time Windows, PartI: Route Construction and local search algorithm," *Transport. Sci.*, vol. 39, no. 1, pp. 104-118.
[37] K. Fleszar and K. S. Hindi, "New heuristics for one-dimensional bin-packing," *Comput. Oper. Res.*, vol. 29, no. 7, p. 821, 2002.
[38] M. G. A. Verhoeven, E. H. L. Aarts, and P. C. J. Swinkels, "A parallel 2-opt algorithm for the Traveling Salesman Problem," *Future. Gener. Comp. Sy.*, vol. 11, no. 2, pp. 175-182, 1995.