An Adaptive Memetic Algorithm with Extended neighborhood Search for the Vehicle Routing Problem with Backhauls and Two-dimensional Loading Constraints

Yangguang Wang[1], Chuang Liu[1], Shengchao Zhou[2], Rui Xu[3] and Huaping Chen[1*]

[1]School of Management, University of Science and Technology of China, Hefei, 230026, Anhui, PR China.
[2]School of Automation, Central South University, Changsha, 410083, Hunan, PR China.
[3]School of Business, Hohai University, Nanjing, 210024, Jiangsu, PR China.

*Corresponding author(s). E-mail(s): hpchen@ustc.edu.cn;
Contributing authors: ygw@mail.ustc.edu.cn; chuang@mail.ustc.edu.cn; zhousc@mail.ustc.edu.cn; rxu@hhu.edu.cn;

Abstract

This paper addresses the well-known capacitated vehicle routing problem with two-dimensional loading constraints (2L-CVRP), as well as one of its extensions called 2L-VRPB. The 2L-CVRP, which consists of two NP-hard problems, namely the capacitated vehicle routing problem and the two-dimensional bin packing problem, is particularly difficult to solve in practice. We propose a metaheuristic algorithm called AMA-ENS, where a memetic algorithm with extended neighborhood search is used for routing, and a hybrid packing method called Hybrid-Search is employed for packing. Extensive computational experiments on the classic 2L-CVRP benchmark instances show that the proposed metaheuristic significantly outperforms the existing state-of-the-art metaheuristics in the literature. New best solutions are found for most instances of the

2L-CVRP benchmark data set. AMA-ENS achieves an average improvement of 1.41% and 0.76% over the existing BKS level for the **2**|UO|L and **2**|UR|L versions of 2L-CVRP respectively. In addition, the detailed computational results for two versions of 2L-VRPB are also reported.

Keywords: Vehicle routing problem, 2L-CVRP, Backhaul, Two-dimensional bin packing, Memetic algorithm

1 Introduction

In real logistics, when people try to consider both vehicle route planning and cargo loading, a tricky combinatorial optimization problem called 2L-CVRP appears. The vehicle routing problem (VRP) and two-dimensional bin packing problem (2L-BPP) are both NP-hard, which have developed into two independent research fields respectively [1]. Admittedly, the combination of these two problems undoubtedly increases the difficulty of solving, but the resultant solutions are more realistic and can better depict actual logistic targets. Overall, the integration of routing and packing is a fairly new research field.

The most representative prototype of the VRP is the capacitated vehicle routing problem (CVRP) [2]. Given a set of customers with different delivery requirements and a fleet of homogeneous vehicles with a fixed load capacity, the CVRP aims to find the minimum cost (i.e., total travelled distance). A common extension of the CVRP is the vehicle routing problem with backhauls (VRPB) [3, 4], which includes two types of customers: linehaul customers whose items are to be delivered, and backhaul customers whose items need to be transported to the distribution center. Generally, VRPB refers to the situation where the backhaul customers are clustered, if not specifically clarified. In this case, all deliveries for linehaul customers, for each vehicle, must be made before any pickups from backhaul customers. However, it is noteworthy that there exists a special situation where the backhaul customers are not clustered, that is, the backhaul customers and linehaul customers are mixed in each route. In order to distinguish these two cases, we call the former more general case backhauls-clustered; for the latter, we call it backhauls-mixed. The significance of the VRPB can be attributed to the effort to reduce distribution costs by utilizing the unused capacity of an empty vehicle returning to a depot [5].

2L-CVRP is obtained by integrating CVRP with two-dimensional bin packing, which was first proposed in [6]. More specifically, in addition to the constraints of the CVRP, it is assumed that the shape of each item is cuboid and the items cannot be stacked because of their fragility or for the requirement of stabilization. In the literature, four variants are proposed depending on two judging conditions: whether the loading mode is unrestricted or sequential, and whether rotation is allowed when the items are loading (unloading)

[7, 8]. To be specific, there are two loading modes: sequential loading and unrestricted loading. For the former, it is assumed that when a customer is being visited, items are unloaded by forklift trucks from the rear door of the vehicle; hence, those items belonging to other customers cannot be moved [9]. For the latter, it is assumed that each vehicle has doors on both sides or items are unloaded by manpower rather than forklift trucks, so rearrangements of items are permitted at customer locations. In addition, items can be loaded with rotation, that is, each item can be rotated by 90 degrees, or without rotation. Here, we use the symbols adopted in previous research [8] to represent these four variants: 2|UO|L, unrestricted + orientated; 2|UR|L, unrestricted + rotated; 2|SO|L, sequential + orientated; 2|SR|L, sequential + rotated.

Similar to the 2L-CVRP, the integration of VRPB and two-dimensional bin packing leads to the 2L-VRPB, which can be regarded as a new realistic extension of the 2L-CVRP. Likewise, there are four variants of 2L-VRPB if we only consider the situation of clustered-backhauls, which are as follows: 2L-VRPB (2|UO|L), 2L-VRPB (2|UR|L), 2L-VRPB (2|SO|L), and 2L-VRPB (2|SR|L).

This paper focuses the 2L-CVRP under the unrestricted loading mode, as well as the 2L-VRPB under the same loading mode. Both the rotated and orientated packing are considered for each problem. The paper is organized as follows. The relevant literature is reviewed in section 2. Section 3 describes the proposed problem in detail. Section 4 presents the adaptive memetic algorithm with extended neighborhood search for the routing, and section 5 gives an introduction to the Hybrid-Search heuristic for packing. Computational experiments on the associated benchmark data sets are reported in section 6. The paper is concluded in section 7.

2 Literature review

The combination of the VRP and two-dimensional packing was first introduced by [6], and this combination is known as the 2L-CVRP. [10] proposed a guided tabu search that combines a guided local search and the rationale of tabu search. To check the feasibility of loading, the authors used a collection of packing heuristics. Later, [11] designed a new packing heuristic on the basis of the packing algorithm in [10]. The packing heuristics for the loading component are integrated into the framework of a simulated annealing algorithm. This algorithm is further developed to solve a new variant of the 2L-CVRP (2L-HFVRP), where customers are visited by a heterogeneous fleet rather than a homogeneous fleet [12]. [13] proposed a simple but effective algorithm by utilizing the technique of biased randomization [14]. More specifically, a biased-randomized version of CWS [15] is used for routing [16], and a biased-randomized version of the best-fit packing heuristic [17] is employed to check the feasibility of loading. Using this biased-randomized technique, [7] also solved a variant of the 2L-CVRP, i.e., the 2L-HFVRP, and obtained better solutions than [12]. [18] proposed a new packing heuristic for the packing component of the 2L-CVRP. To examine the loading feasibility of a given item set, a basic packing heuristic [19] is invoked multiple times on different sequences of the item set. In addition, variable neighborhood search is employed for the routing part. Similar to the design principles of the packing algorithm proposed in [18], the authors later presented a new packing method that also runs on the basis of repeated invocations to one basic packing heuristic [8]. [20] proposed two-dimensional loading time-dependent vehicle routing problem and a bi-objective mathematical model. [21] formulated a model of the 2L-CVRP with stochastic travel times, and presented a hybrid simheuristic algorithm to solve this new model. [22] presented a multi-objective vehicle routing and loading problem with time window constraints, which involves three objectives, namely, the minimization of the total travel, the number of routes to use, and the total number of mixed orders in the same pallet. [23] proposed a split delivery vehicle routing problem with two-dimensional loading constraints (2L-SDVRP), and used an enhanced neighborhood search algorithm to solve the problem. [24] proposed three extensions of the original 2L-CVRP, namely, allowing split delivery (2L-SDVRP), with green requirements (G2LCVRP), and integrating split delivery with green requirements (G2L-SDVRP). [25] considered a stochastic vehicle routing problem where the two-dimensional size (height and width) and the weight of each item is characterized by a discrete probability distribution, and the author further solved the problem by employing an integer L-shaped method.

The VRPB was first proposed by [26] and can be classified into four classes, which are the VRP with clustered Backhauls (the standard VRPB), the VRP with Mixed Backhauls (the mixed VRPB), the VRP with Mixed Pickup and Delivery (VRPMPD) and VRP with Simultaneous Pickup and Delivery (VRP-SPD) [27–29]. In the first and second classes, there are backhaul customers and linehaul customers, and the requirement of each customer is either a delivery or a pickup. In the third and fourth classes, all the customers belong to the same type, and their requirements involve both services. For earlier studies on the problem, the reader can refer to [2], where a comprehensive comparison and analysis of related papers for the standard VRPB are provided. [30] proposed a reactive tabu search metaheuristic that utilizes the saving-insertion and saving assignment heuristics to generate initial solutions, and these solutions are further improved by a reactive tabu search metaheuristic with a reactive concept controlling the search between different neighborhood structures. Later, [31] presented a hybrid operation by combining the reactive tabu search [30] and adaptive memory programming, and the adaptive memory strategy is used to search the unexplored area of the search space.

[3] used a multi-ant colony system (MACS) to deal with the VRPB, and this system utilizes artificial ants to construct new solutions by using pheromone information from previously generated solutions. [32] proposed a local search metaheuristic whose search neighborhood are composed of exchanges of variable-length visiting sequences. In addition, tentative local

search moves are statically encoded and stored in Fibonacci heaps to speed up the search process, and a parameter-free mechanism is used to induce diversification.

[33] presented an effective iterated local search algorithm whose main component is an oscillating local search heuristic. This local search method makes transitions between feasible and infeasible solutions continuously. In addition, the authors performed an extensive statistical analysis to further calibrate the parameters and identify the importance of each component of the proposed algorithm. [4] proposed a biased-randomized version of the classic Clarke and Wright savings (CWS) ([15]) heuristic to generate solutions of the standard VRPB. A skewed probability distribution is utilized to make nondeterministic choices for the savings list of routing arcs each time, and a penalty value is assigned to those arcs from a backhaul customer to a linehaul customer.

In addition to the extensive research on standard VRPB, other variants of VRPB also received great attention. [34] proposed a particle swarm optimization (PSO) algorithm to solve the VRPSPD. [35] presented an adaptive local search method by combining simulated annealing (SA) algorithm and variable neighborhood descent together. The method is capable of make self-turning in the search process and can deal with VRPSPD and VRPMPD efficiently. [36] considered the vehicle problem with backhauls and time windows (VRPBTW), in which the constraints about time windows, backhaul and capacity are involved. Besides, a hybrid algorithm was proposed by integrating SA and tabu search to solve this variant. [37] considered the VRP with backhauls, time windows, and three-dimensional loading constraints, and they proposed a two-phase method to solve this problem. [38] considered the VRPSDP with time windows and three-dimensional loading constraints. To deal with the problem, a large neighborhood search is presented for routing and a collection of packing heuristics is employed for packing.

Compared to the research on the 2L-CVRP, literature is limited on the combination of the VRPB and two-dimensional packing. In the previous relevant literature, [39] were the first to consider such a combination of the VRPB and two-dimensional packing, but no algorithm was presented to solve the problem. [7] also studied this combinatorial optimization problem, and they studied two versions of 2L-VRPB, i.e. 2|SO|L, 2|SR|L. [40] investigated a special version of 2L-VRPB under the sequential loading mode, where linehauls and backhauls are mixed.

3 Problem description

In this paper, 2L-CVRP and 2L-VRPB are both investigated. Since the former can be regarded as a simplified case of the latter, we focus on the description of 2L-VRPB, and that of 2L-CVRP can be obtained directly by ignoring all the backhaul-related constraints. The 2L-VRPB with unrestricted loading mode can be defined as follows. Let $G = (V, E)$ be a complete graph, where $V = 0, 1, \ldots, N$ is the vertex set and $E = (i, j)|i, j \in V, i \neq j$ is the edge set. The vertex set V is composed of three subsets $V = V^0 \cup V^l \cup V^b$: subset V^0 has only one vertex v_0, which represents the central depot, while disjoint subsets V^l and V^b represent linehaul and backhaul customers, respectively. For concise representation, let $V^c = V^l \cup V^b$. For each edge $(i, j) \in E$, the associated travel cost c_{ij} is defined as the direct travel distance from v_i to v_j or from v_j to v_i. In other words, the costs are symmetric, i.e., $c_{ij} = c_{ji}$. In the central depot, a fleet of N_k homogeneous vehicles is available. K denotes the fleet, and k denotes a vehicle in the fleet, i.e. $k \in K$. Each vehicle k has a maximum loading capacity D and a rectangular loading surface $S = W \times L$. Each customer $i \in V^c$ has m_i rectangular items, denoted as I_i, $I_i = \{I_{i1}, I_{i2}, \ldots, I_{im_i}\}$. The width, length, bottom area, and weight of item I_{ir} are $w_{ir}(r = 1, \ldots, m_i)$, $l_{ir}(r = 1, \ldots, m_i)$, $s_{ir}(r = 1, \ldots, m_i)$, and $d_{ir}(r = 1, \ldots, m_i)$, respectively. The total bottom area and total weight of I_i is denoted as s_i and d_i respectively. x_{ijk} is a Boolean variable, which indicates whether edge (i, j) will be visited by the vehicle k.

The goal is to find a set of vehicle routes servicing all customers, such that the total travel cost is minimized. In addition, the constraints can be classified into two types: the first type is about the routing, and the second is about the packing. Both types of constraints are described as follows.

1. Each vehicle starts and finishes its task at the central depot.
2. No more than K vehicles are used to satisfy the customers' demands.
3. Each customer, linehaul or backhaul, is visited exactly once. To satisfy this requirement, all the items belonging to a linehaul customer or a backhaul customer should be loaded into a single vehicle.
4. Every route includes at least one linehaul customer, i.e., a route with no linehaul customers is not permitted.
5. For each route, all the deliveries for linehaul customers must be made before any pickups from backhaul customers.
6. In each route, the vehicle's capacity and loading surface area cannot be exceeded by loaded items.
7. All items assigned to the same vehicle must be loaded without overlapping, and the edges of the items must be parallel to the edges of the vehicle.
8. According to the loading configuration, items can be loaded without rotation (oriented loading mode), or in contrast, a 90-degree rotation is permitted (rotatable loading mode).

In order to describe the mathematical model of 2L-VRPB concisely, we disassemble the model into two parts: the first part describes the constraints related to routing, and actually, it can also be regarded as a mathematical model of VRPB problem; the second part describes in detail the constraints related to packing on a single route. The formulations are adapted from [6, 7]. The following mathematical formulas are the first part of 2L-VRPB.

$$\min \sum_{k \in K} \sum_{i,j \in V, i \neq j} c_{ij} x_{ijk} \tag{1}$$

subject to:

$$\sum_{j \in V^l} x_{0jk} = \sum_{i \in V^c} x_{i0k} \quad \forall k \in K \tag{2}$$

$$\sum_{k \in K} \sum_{i \in V} x_{ijk} = 1 \quad \forall j \in V^c \tag{3}$$

$$\sum_{i \in V} x_{iuk} = \sum_{j \in V} x_{ujk} \quad \forall u \in V^c, \forall k \in K \tag{4}$$

$$\sum_{i \in V^b, j \in V^l} x_{ijk} = 0 \quad \forall k \in K \tag{5}$$

$$\sum_{i \in V^l, j \in V^b} x_{ijk} \leq 1 \quad \forall k \in K \tag{6}$$

$$\sum_{k \in K} \sum_{j \in V^c} x_{0jk} \leq \mathsf{K} \tag{7}$$

$$\sum_{i \in V^l, j \in V} d_i x_{ijk} \leq D \quad \forall k \in K \tag{8}$$

$$\sum_{i \in V^l, j \in V} s_i x_{ijk} \leq S \quad \forall k \in K \tag{9}$$

$$\sum_{i \in V^b, j \in V} d_i x_{ijk} \leq D \quad \forall k \in K \tag{10}$$

$$\sum_{i \in V^b, j \in V} s_i x_{ijk} \leq S \quad \forall k \in K \tag{11}$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in V, \forall j \in V, i \neq j, \forall k \in K \tag{12}$$

The second part is about the constraints of packing feasibility on a single route. The loading surface of a vehicle is defined by Cartesian coordinates, where the domain of the X-axis is between 0 and W and that of the Y-axis is between 0 and L. Hence, the coordinates $(\alpha_{ir}, \beta_{ir})$ indicate, for an item r that belongs to customer i, the position where the bottom-left corner of item r is finally placed. For a route visited by vehicle k, R_k^l represents the set of linehaul customer nodes visited by vehicle k, and R_k^b represents the set of backhaul customer nodes visited by vehicle k. R_k represents the set of all linehaul and backhaul customer nodes visited by vehicle k, $R_k = R_k^l \cup R_k^b$. Binary variable μ_{ir} indicates whether the item r belonging to customer i is allowed to be rotated. Using the variables mentioned above, the description of packing constraints on a single route can be defined as follows.

$$\begin{aligned} & 0 \leq \alpha_{ir} \leq (W - w_{ir})(1 - \mu_{ir}) + (W - l_{ir})\mu_{ir} \quad \wedge \\ & 0 \leq \beta_{ir} \leq (L - l_{ir})(1 - \mu_{ir}) + (L - w_{ir})\mu_{ir} \\ & \forall k \in K, \forall i \in R_k^l, \forall r \in \{1, \ldots, m_i\} \end{aligned} \tag{13}$$

$$\begin{aligned}&\alpha_{ir}+w_{ir}\left(1-\mu_{ir}\right)+l_{ir}\mu_{ir}\leq\alpha_{jr'}\quad\vee\\&\alpha_{jr'}+w_{jr'}\left(1-\mu_{jr'}\right)+l_{jr'}\mu_{jr'}\leq\alpha_{ir}\\&\forall k\in K,\forall i,j\in R_k^l,\forall r\in\left\{1,\ldots,m_i\right\},\forall r'\in\left\{1,\ldots,m_j\right\},I_{ir}\neq I_{jr'}\end{aligned}\tag{14}$$

$$\begin{aligned}&\beta_{ir}+l_{ir}\left(1-\mu_{ir}\right)+w_{ir}\mu_{ir}\leq\beta_{jr'}\quad\vee\\&\beta_{jr'}+l_{jr'}\left(1-\mu_{jr'}\right)+w_{jr'}\mu_{jr'}\leq\beta_{ir}\\&\forall k\in K,\forall i,j\in R_k^l,\forall r\in\left\{1,\ldots,m_i\right\},\forall r'\in\left\{1,\ldots,m_j\right\},I_{ir}\neq I_{jr'}\end{aligned}\tag{15}$$

$$\begin{aligned}&0\leq\alpha_{\mathrm{pu}}\leq\left(W-w_{\mathrm{pu}}\right)\left(1-\mu_{\mathrm{pu}}\right)+\left(W-l_{\mathrm{pu}}\right)\mu_{\mathrm{pu}}\quad\wedge\\&0\leq\beta_{\mathrm{pu}}\leq\left(L-l_{\mathrm{pu}}\right)\left(1-\mu_{\mathrm{pu}}\right)+\left(L-w_{\mathrm{pu}}\right)\mu_{\mathrm{pu}}\\&\forall k\in K,\forall p\in R_k^b,\forall u\in\left\{1,\ldots,m_p\right\}\end{aligned}\tag{16}$$

$$\begin{aligned}&\alpha_{\mathrm{pu}}+w_{\mathrm{pu}}\left(1-\mu_{\mathrm{pu}}\right)+l_{\mathrm{pu}}\mu_{\mathrm{pu}}\leq\alpha_{qu'}\quad\vee\\&\alpha_{\mathrm{qu'}}+w_{\mathrm{qu'}}\left(1-\mu_{\mathrm{qu'}}\right)+l_{\mathrm{qu'}}\mu_{\mathrm{qu'}}\leq\alpha_{\mathrm{pu}}\\&\forall k\in K,\forall p,q\in R_k^b,\forall u\in\left\{1,\ldots,m_p\right\},\forall u'\in\left\{1,\ldots,m_q\right\},I_{\mathrm{pu}}\neq I_{\mathrm{qu'}}\end{aligned}\tag{17}$$

$$\begin{aligned}&\beta_{\mathrm{pu}}+l_{\mathrm{pu}}\left(1-\mu_{\mathrm{pu}}\right)+w_{\mathrm{pu}}\mu_{\mathrm{pu}}\leq\beta_{\mathrm{qu'}}\quad\vee\\&\beta_{\mathrm{qu'}}+l_{\mathrm{qu'}}\left(1-\mu_{\mathrm{qu'}}\right)+w_{\mathrm{qu'}}\mu_{\mathrm{qu'}}\leq\beta_{\mathrm{pu}}\\&\forall k\in K,\forall p,q\in R_k^b,\forall u\in\left\{1,\ldots,m_p\right\},\forall u'\in\left\{1,\ldots,m_q\right\},I_{\mathrm{pu}}\neq I_{\mathrm{qu'}}\end{aligned}\tag{18}$$

The objective function Eq. 1 aims to minimize total transportation costs. Constraint (2) ensures that the number of vehicles departing from the depot is equal to the number of vehicles returning to it. Constraints (3) and (4) ensure that each customer is served only once by one vehicle. Once a vehicle visits a customer, it must also depart from that customer. Constraints (5) and (6) enforce that backhaul customers will be served only after all linehaul customers have been visited. Constraint (7) requires that the number of vehicles used does not exceed the maximum number of vehicles in the fleet. Constraints (8) and (9) guarantee that the total weight and total loading area of the linehaul customers' cargo will not exceed the weight loading capacity and loading surface of the vehicle itself respectively, and constraints (10) and (11) impose the equivalent restrictions on backhaul customers.

Constraints (13), (14), and (15) describe the packing constraints on the linehaul sub-route of each route: constraint (13) guarantees that for any goods to be loaded on the linehaul sub-route (i.e. $I_{ir} \in I_i, i \in R_k^l$), its position cannot cross any edge of the vehicle; constraints (14) and (15) ensure that for any two goods I_{ir} and $I_{jr'}$ to be loaded on the linehaul sub-route (i.e. $I_{ir} \in I_i, i \in R_k^l, I_{jr'} \in I_j, j \in R_k^l, I_{ir} \neq I_{jr'}$), they must not overlap each other. Constraints (16), (17), and (18) impose the same packing constraints on the backhaul-sub route of each route: constraint (16) specifies that any item to be loaded on the backhaul sub-route cannot cross the any edge of the vehicle; constraint (17) and (18) specify that for any two items I_{pu} and $I_{qu'}$ to be loaded on the backhaul sub-route, they cannot overlap each other.

4 The adaptive memetic algorithm with extended neighborhood search for routing

The adaptive memetic algorithm with extended neighborhood search (AMA-ENS) is based on the hybrid genetic search with adaptive diversity control (HGSADC) introduced in [41, 42]. Our adaptation includes two additional state-of-the-art components: a geometrical insert component, which is able to improve the efficiency of routing and accelerate convergence; and a destroy-and-reconstruct component, which, as a more competitive mutation operator than the traditional intra-2opt or inter-2opt operator, enables the algorithm have greater potential to jump out of local optimum and effectively avoid the problem of premature convergence. It is worth noting that, in this paper, the geometrical insert component and destroy-and-reconstruct component run as fast as a traditional mutation operator like swap or relocate due to extra customized designs, including designing tailor-made data structures to accelerate the running process, timely recording the information of nodes and routes modified to avoid massive repeated calculations, etc.

The overview of AMA-ENS is described in Algorithm 1. The main components of the AMA-ENS are described in the remainder of this section: subsection 4.1 gives a brief introduction of the search space and the individual evaluation method; subsection 4.2 presents the process of individual generation; subsection 4.3 describes the improvement process of new individuals, including the local search improvement, the geometrical insert component, and destroy-and-reconstruct component; subsection 4.4 explains the population management mechanism.

4.1 Solution representation

The literature related to the VRP indicates that the efficient exploitation of infeasible solutions may significantly improve the quality of the solutions produced [41, 42]. Specifically, we define the search space of the AMA-ENS as sets of two types of solutions: feasible solutions and infeasible solutions. The latter is obtained by relaxing the constraints of maximum capacity, packing feasibility, or the restrictions related to backhaul customers.

A penalty mechanism is established to promote the transfer of infeasible individuals from the state of infeasible to feasible. Without loss of generality, considering a case where a constraint is violated by one individual, then the corresponding penalty value depends on its deviation from the feasibility of this constraint. Let $R(w)$ denote the set of routes belonging to an individual w. Each route $r \in R(w)$ departs from a depot, visits a sequence of customers, and finally returns to the starting point. For a route r, G_r^l denotes the set of goods belonging to all the linehaul customers, and G_r^b represents the set of goods belonging to all the backhaul customers. The fitness value of individual w, denoted as $\varphi(w)$, is calculated as Eq. 19, which is equal to the sum of the total driving distance and the total penalty value. In this equation, α_c, β_b, and γ_p represent the penalty coefficient of restraints of capacity, backhaul,

Algorithm 1 AMA-ENS.

```
**Construct the initial populations:** $\Lambda_f$, $\Lambda_{inf}$
$\Delta(|\Lambda_f| + |\Lambda_{inf}|) \Leftarrow 0$
**while** the termination condition not met **do**
    $\xi^{'}, \xi^{''} \Leftarrow applyParentsSelection(\Lambda_f, \Lambda_{inf})$
    $\xi \Leftarrow applyCrossover(\xi^{'}, \xi^{''})$
    $S \Leftarrow applySplit(\xi, N_{veh})$
    $S \Leftarrow applyLocalSearch(S, \alpha_c, \beta_b, \gamma_p)$
    **if** $S$ is infeasible **then**
        $S \Leftarrow applyLocalSearch(S, \alpha_c * 10, \beta_b * 10, \gamma_p * 10)$
        **if** $S$ is infeasible **then**
            $\Lambda_f, \Lambda_{inf} \Leftarrow addToPopulation(S)$
            $\Delta(|\Lambda_f| + |\Lambda_{inf}|) \Leftarrow \Delta(|\Lambda_f| + |\Lambda_{inf}|) + 1$
        **end if**
    **end if**
    $S^{'} \Leftarrow S$; $\Upsilon \Leftarrow \Upsilon(S^{'})$;
    **while** the search of $\Upsilon$ not completed **do**
        $S^{'} \Leftarrow applyGeometricalInsert(S^{'}, \Upsilon, \Psi_{geo})$
    **end while**
    $S^{''} \Leftarrow S^{'}$; $\Omega \Leftarrow \emptyset$;
    **while** $|\Omega| < N_{veh}$ **do**
        $S^{''}, \Omega \Leftarrow applyDestroy(S^{''}, \Omega, P_{rv}, P_{wv}, P_{sv}, P_{wr})$
    **end while**
    **while** $\Omega \neq \emptyset$ **do**
        $S^{''}, \Omega \Leftarrow applyReconstruct(S^{''}, \Omega, P_{greedy}, P_{regret}, k_{regret})$
    **end while**
    $\Lambda_f, \Lambda_{inf} \Leftarrow addToPopulation(S^{''})$
    $\Delta(|\Lambda_f| + |\Lambda_{inf}|) \Leftarrow \Delta(|\Lambda_f| + |\Lambda_{inf}|) + 1$
    **if** $\Delta(|\Lambda_f| + |\Lambda_{inf}|) \geq 100$ **then**
        $\alpha_c, \beta_b, \gamma_p \Leftarrow updatePenaltyCoef(\alpha_c, \beta_b, \gamma_p)$; $\Delta(|\Lambda_f| + |\Lambda_{inf}|) \Leftarrow 0$
    **end if**
    **if** $|\Lambda_f| \geq \mu + \lambda$ or $|\Lambda_{inf}| \geq \mu + \gamma$ **then**
        $applySurvivorSelection(\Lambda_f, \Lambda_{inf})$
    **end if**
**end while**
```

and packing, respectively; $c(r)$ denotes the driving distance of route r; $d(G_r^l)$ and $d(G_r^b)$ denote the total weight of goods belonging to G_r^l and G_r^b, respectively; $count(r)$ denotes the number of direct segments violating the constraints related to backhauls in route r; pack(G_r^l) and pack(G_r^b) indicate the number of vehicles required to load the goods belonging to G_r^l and G_r^b, respectively.

$$\varphi(w)=\sum_{r\in R(w)}\left\{\begin{array}{c}c(r)+\beta_b*\mathrm{count}(r)\\ +\alpha_c\left\{\max\left\{0,d\left(G_r^l\right)-D\right\}+\max\left\{0,d\left(G_r^b\right)-D\right\}\right\}\\ +\gamma_p\left\{\max\left\{0,\mathrm{pack}\left(G_r^l\right)-1\right\}+\max\left\{0,\mathrm{pack}\left(G_r^b\right)-1\right\}\right\}\end{array}\right\} \tag{19}$$

4.2 Generation of new individuals

At each iteration, two parents are randomly selected by a binary tournament method from the union of the feasible population Λ_f and infeasible population Λ_{inf}. The giant tour encoding makes it viable to apply recycling crossovers that were initially designed for the traveling salesman problem (TSP). Here, we choose the OX crossover to generate new giant tour ξ on the basis of the giant tours $\xi^{'}$ and $\xi^{''}$ of its parents. Then, a split algorithm from [43] is invoked to generate the corresponding new individual with a complete structure, and the new individual is denoted as S. It is worth noting that each time two giant tours are generated by the OX crossover, and we use a Bernoulli distribution to decide which one is selected as the input of the split algorithm to generate the new offspring.

4.3 Extended neighborhood search

In this paper, the improvement process of routing contains three different components: namely the local search component, the geometrical insert component, and destroy-and-reconstruct component. The combination of these three components equip the AMA-ENS with the state-of-the-art global optimization ability for routing.

The first component is the local search process, which contains four kinds of operators, i.e., swap, relocate, intra-2opt and inter-2opt. When undergoing the local search process, new individuals may be improved or modified for more than once. To be specific, the first execution of the local search, called education, aims to improve the solution quality. The second execution, called repair, attempts to restore the feasibility of the solution, that is, to promote the transformation of the solution from a state of infeasible to feasible. Each time an offspring is generated, education process is invoked automatically to improve its solution quality. Then, if the resulting solution is infeasible, a repair process is performed: the penalty parameters are multiplied by 10, and the local search procedure is restarted.

In addition, the neighborhood pruning strategy is adopted to increase the efficiency of education and repair. A set of promising edges is calculated for each customer node [44]. Only an operator that generates at least one promising edge is further tested. For a given customer v_i, the set $\Gamma(v_i)$ of its correlated customers is defined as the $|\Gamma|$ closest customers with respect to the corresponding similarity. The similarity between two customers are calculated as Eq. 20. Customers included in $\Gamma(v_i)$ can be considered ideal destinations for a direct visit from v_i, so arcs (v_i, v_j) for $v_j \in \Gamma(v_i)$ are regarded as a subset of "promising" arcs. Because of the neighborhood pruning strategy, the neighborhood size is reduced from $O(n^2)$ to $O(|\Gamma|n)$.

$$\text{similarity}(i,j) = \begin{cases} c_{ij} + 10000, & \text{if} i \in V^b \text{ and} j \in V^l \\ c_{ij} + 10000, & \text{if} i \in V^0 \text{ and} j \in V^b \\ c_{ij}, & \text{for other cases} \end{cases} \tag{20}$$

The second component is the geometrical insertion operator, which, to the best my knowledge, is first proposed in this paper. Each route corresponds to a sector area whose radius is equal to the straight-line distance between the depot and the vertex of the route farthest from it. Once a route is modified, its sector area will be corrected automatically. The most remarkable feature of the operator is to measure the similarity between two vertexes through their geometric features, rather than through a specific similarity calculation formula. Specifically, two routes are regarded as similar if their sectors overlap. In order to better describe the geometric characteristics, routes are described in a polar coordinate system with the depot v_0 as the pole. We define the neighborhood space of geometric operators as Υ, and the granularity of the neighborhood space is denoted as Ψ_{geo}. When Ψ_{geo} is equal to 2, each element in the neighborhood space is a route pair composed of two similar routes; when Ψ_{geo} is equal to 3, each element in the neighborhood space is a triad composed of three routes similar to each other. In this paper, Ψ_{geo} is equal to 2. For a neighborhood (i.e. route pair) in the neighborhood space, a vertex is removed for each of the two routes, and then the vertex is inserted into the best position in the other route according to the changes of fitness value. For each route in the route pair, the node to be removed and the position to be inserted depends on the resulting change of fitness value caused by this successive remove and insertion operation, in other words, the operation capable of reducing the fitness value to the greatest extent will be performed.

The third component is the destroy-and-reconstruct component. The execution process of the component includes two stages: in the destroy stage, a subset of vertexes is removed from the current solution and added into the unassigned vertex set Ω; in the reconstruction stage, all the vertexes in Ω are reinserted into the remaining structure of the solution in turn. In the destroy stage, four classical removal operators are employed:

- RANDOM VERTEX (RV): randomly removes a vertex in the current solution.
- WORST VERTEX (WV): first selects three worst vertexes on the basis of their contributions to the total fitness value, and then one of the three vertexes will be randomly selected and removed;
- SIMILAR VERTEXES (SV): randomly selects a vertex and one of its three closest vertexes (according to Eq. 20).
- WORST ROUTE (WR): first selects three worst routes in the current solution, then randomly selects one from these three routes, and removes all vertexes on the selected route.

Compared with the other three removal operators, the worse route (WR) operator causes, to a large extent, greater changes to the current solution, thus leading to greater computational consumption to restore the solution structure. Therefore, we reduce the probability of using the worse route operator. Finally, the probabilities of four removal operators being invoked are $P_{rv} = 0.3, P_{wv} = 0.3, P_{sv} = 0.3, P_{wr} = 0.1$, respectively. These four removal operators are invoked iteratively until at least N_K vertexes are removed from the current solution, i.e. $|\Omega| > N_{veh}$. In the reconstruction stage, two insertion operators are employed with equal probability (i.e. $P_{greedy} = 0.5, P_{regret} = 0.5$) to reconstruct the solution [45, 46]: the first operator is greedy insertion; the second operator is regret insertion whose k-regret value (denoted as k_{regret} in this paper) is equal to 2.

4.4 Mechanism of population management

As mentioned earlier, feasible population Λ_f and infeasible population Λ_{inf} keep evolving during the whole search process. The size of each population is maintained between μ and $\mu + \lambda$, where μ represents the minimum population size and λ denotes the generation size. At the beginning of the search, 4μ individuals are randomly generated and added into the appropriate populations. To avoid the risk of premature convergence and maintain sufficient population diversity, a biased fitness function of the original HGSADC in [41, 42, 44] is used to evaluate the individuals from the perspective of solution quality and its diversity contribution. Here, the diversity contribution value of individual S is calculated as Eq. 21, where Λ_S denotes the subset of its N_{close} closest individuals in the population where it belongs, and $\delta^{bp}(S, S^{'})$ denotes the broken-pairs distance between S and $S^{'}$.

$$\mathrm{Div}(S) = \frac{1}{N_{\mathrm{close}}} \left(\sum_{S' \in \Lambda_S} \delta^{bp}\left(S, S'\right) \right) \tag{21}$$

$$F_{\mathrm{biased}}(S) = \begin{cases} f^{\varphi}_{\Lambda_f}(\varphi(S)) + \left(1 - \frac{N_{\mathrm{elite}}}{|\Lambda_f|}\right) \times f^{d}_{\Lambda_f}(\mathrm{Div}(S)), & \text{if } S \ \in \Lambda_f \\ f^{\varphi}_{\Lambda_{inf}}(\varphi(S)) + \left(1 - \frac{N_{\mathrm{elite}}}{|\Lambda_{inf}|}\right) \times f^{d}_{\Lambda_{inf}}(\mathrm{Div}(S)), & \text{if } S \in \Lambda_{inf} \end{cases} \tag{22}$$

The biased fitness value of individual S is calculated as Eq. 22. It is equal to the weighted sum of its cost rank $f^{\varphi}_{\Lambda}(\varphi(S))$ in terms of the individual fitness (calculated by Eq. 19) and its diversity rank $f^{\varphi}_{\Lambda}(Div(S))$ in terms of the individual diversity contribution (calculated by Eq. 21). In the whole search process, once the size of any population reaches the upper limit $\mu+\lambda$ (presented as $\Delta(|\Lambda_f| + |\Lambda_{inf}|) \geq 100$ in Algorithm 1), a survivor selection process will be triggered automatically. Then, the best μ individuals in terms of their biased fitness (calculated by Eq. 22 are reserved and the remaining individuals are removed. The penalty coefficients related to the constraints of maximum capacity, backhaul customers, and packing feasibility are dynamically adjusted during the execution of the algorithm. The algorithm records and calculates the natural feasibility ratio [42] of each constraint every 100 iterations. Here, let ξ^{para} and ρ^{para} denote the natural feasibility ratio and penalty coefficient of one specific constraint, respectively. The following adjustment (Eq. 23) is then performed every 100 iterations. In the search process, by timely comparing the natural feasibility ratio ξ^{para} with the referential natural feasibility ratio ξ^{ref}, the control strength of the constraint on the current search could be known, which further guides the algorithm to decide whether to tighten or relax the control of the constraint in the following search process.

$$\rho^{\text{para}} = \begin{cases} \min\left\{\rho^{\text{para}} \times 1.20, 200\right\}, & \text{if } \xi^{\text{para}} \leq \xi^{ref} - 0.05 \\ \max\left\{\rho^{\text{para}} \times 0.85, 0.10\right\}, & \text{if } \xi^{\text{para}} \geq \xi^{ref} + 0.05 \end{cases} \tag{23}$$

5 The Hybrid-Search heuristic for the two-dimensional loading problem

Compared with the classical two-dimensional bin packing problem, which aims at packing all items onto as few bins as possible, in the 2L-CVRP or its relevant variants, when a visiting route is generated, the packing feasibility check for this route is equivalent to loading all items into one bin (the vehicle). The effectiveness of packing feasibility check is a bottleneck restricting the improvement of the final solution quality of 2L-CVRP and its related variants. Although the relatively simple packing algorithm can greatly minimize the total running time, it inevitably leads to the generation of low-quality solutions. Therefore, it is necessary to design a bin packing algorithm with strong packing capacity to improve the quality of the final solutions. In view of this, we make a trade-off between the improvement of solution quality and the time consumption, and propose a new packing method called Hybrid-Search algorithm to make feasible packing scheme for each visiting route.

Hybrid-Search algorithm 2 uses a neighborhood search mechanism to repeatedly invoke an internal two-stage packing method, called Two-Stage-Pack: each invocation to the Two-Stage-Pack generates a complete packing scheme according to an input item sequence; if the generated scheme is infeasible, the input sequence is then modified and used as the input for the next invocation to the Two-Stage-Pack algorithm. This process continues until a feasible packing scheme is generated or the maximum number of iterations is reached. It is worth noting that, the Two-Stage-Pack method is neither a strict sequence-based online packing algorithm, nor an offline packing algorithm: in the first stage of packing, it packs the items in strict accordance with the input sequence; in the second stage, the packing order is completely independent of the input sequence.

Overall, Hybrid-Search algorithm first sorts the items according to three sorting rules (descending by area, length, or width) and takes it as the initial input sequence. Then, the Two-Stage-Pack method is invoked based on the sequence. If a feasible packing scheme is found, the algorithm terminates and

Algorithm 2 Algorithm Hybrid-Search

```
Initialize the semi-scheme, scheme, scheme*
for each sorting rule do
    Sort the $seq_{item}$
    Construct an empty scheme with the $seq_{item}$
    semi-scheme ⇐ MaxResidual(scheme, $r_p$)
    scheme* ⇐ BiasedBestFit(semi-scheme, $maxIter$)
    if scheme* is feasible then
        return TRUE
    end if
    scheme ⇐ scheme*
    for $nonImp$ from 1 to $|seq_{item}|$ do
        $item_\alpha$ ⇐ randomSelect(scheme,1/3)
        $item_\beta$ ⇐ randomSelect(scheme,1/2)
        scheme ⇐ swap(scheme, $item_\alpha, item_\beta$)
        semi-scheme ⇐ MaxResidual(scheme, $r_p$)
        scheme ⇐ BiasedBestFit(semi-scheme, $maxIter$)
        if scheme is feasible then
            return TRUE
        else
            if scheme is better than scheme* then
                scheme* ⇐ scheme
                nonImp ⇐ 1
            else
                nonImp ⇐ nonImp+1
            end if
        end if
    end for
end for
return FALSE
```

return TRUE; otherwise, it enters the loop of a neighborhood search, and the number of iterations is controlled by *nonImp*. Here, *nonImp* is set to the number of items to be packed on the current route. At each iteration, new input sequence of the Two-Stage-Pack method is obtained by exchanging two items in the previous sequence (these two items are randomly selected from the first third and first half of the sequence respectively). In the process of the neighborhood search, Hybrid-Search records the current best scheme. Each time the current best scheme is updated, the parameter *nonImp* is reset to 1; otherwise, *nonImp* increases by 1.

In the following part, we focus on the description of the Two-Stage-Pack method invoked inside the Hybrid-Search algorithm. In each evocation to the Two-Stage-Pack, the corresponding complete packing scheme is generated by the first stage and the second stage of the algorithm together: in the first stage, it completes the packing of the first r_p items of the input sequence by using MaxResidual algorithm [47]; The remaining items will be packed in its second stage through the BiasedBestFit algorithm [7], which is obtained by making a biased randomization of the BestFit algorithm [17]. In this paper, we describe the vehicle surface as a rectangular region in a two-dimensional coordinate system. Whenever a new item is placed on the rectangular surface, the remaining free areas is then updated and represented by residual spaces (RSs) that are not included with each other. The definition of residual space (RS) is the same as that in [47, 48]: each RS is the largest rectangle in the remaining free area. That is, each of its edges cannot extend along its direction; otherwise, it either touches the boundary of the vehicle or touches the edge of an item already placed.

5.1 The first stage of Two-Stage-Pack

At each step of the packing process in this stage, the algorithm finds all the feasible placements for the current item, and selects one that minimize the influence on the existing large RSs to the greatest extent possible. Here, a feasible placement consists of three aspects of information: item, RS, and the configuration of the first two. Specifically, in the orientated loading pattern, a selected combination of item and RS corresponds to at most four configurations, that is, the bottom-left, bottom-right, upper-left, or upper-right corner of the two coincides; in the rotatable loading pattern, this combination corresponds to up to eight configurations. That is, the item is placed vertically or horizontally in each corner of the RS.

In order to clearly describe the packing process, we introduce the following symbols: for each execution of a feasible placement p, i.e. item + RS + configuration, the current residual space set, denoted as S, is then updated as $S^{'}_{p}$, and r^{p}_{S} denotes the largest RS in the set S affected by p. In each step of packing, when an item is selected, the algorithm first arranges the elements in the set S in descending order according to the area, and then evaluates all the feasible placement of each RS in reverse order. By default, in order to compare two different placements $p1$ and $p2$, it is necessary to calculate their corresponding newly generated residual space set $S^{'}_{p1}$ and $S^{'}_{p2}$. The comparison of $S^{'}_{p1}$ and $S^{'}_{p2}$ is conducted as follows: firstly, the elements in $S^{'}_{p1}$ and $S^{'}_{p2}$ are arranged in descending order according to the area, and the sorted sequence is represented as $Seq^{'}_{p1}$ and $Seq^{'}_{p2}$ respectively; then the elements in the two sequences are compared one by one. If it first appears that the area of the element in $Seq^{'}_{p1}$ is larger than that of its counterpart in $Seq^{'}_{p2}$, the set $Seq^{'}_{p1}$ is better than $Seq^{'}_{p2}$ (i.e., the position of $p1$ is better than $p2$), and vice versa.

It is noteworthy that, a high-efficient implementation of comparison of different placements is of great necessity. For two placements $p1$ and $p2$, when their corresponding r^{p1}_{S} and r^{p2}_{S} are not the same, $p1$ and $p2$ can be compared directly without calculating their corresponding $S^{'}_{p1}$ and $S^{'}_{p2}$. In addition, before evaluating a new placement p, if the current best placement is p^{*} and the largest RS in the set S affected by p^* is $r_S^{p^*}$, only those RSs whose area are not greater than the area of $r_S^{p^*}$ need to be evaluated.

5.2 The second stage of Two-Stage-Pack

The second stage starts from the termination state of the first stage and keeps on packing the remaining items onto the vehicle. In this stage, it repeats up to $maxIter$ attempts to generate a feasible packing scheme. If a feasible scheme is found in any attempt, it stops immediately and returns the scheme. Otherwise, it returns the one with the largest occupied area. At each step in one attempt, the most bottom-left RS is selected from the current residual space set, and then a geometric distribution with parameter β is employed to select an item from all feasible items and then place it at the bottom-left corner of the RS. It is worth noting that for the selected item and RS, the item only needs to be placed at the bottom-left corner of the RS, regardless of the other three corners, which is different from the first stage. Here, the parameter β is set to a random number ranging from 0.06 to 0.23 as suggested by [7]. In each attempt, if the selected RS cannot place any item, then it is removed from the current residual space set. Whenever an item is placed, the residual space set is updated immediately; meanwhile, the information about the maximum length, width, and area of all elements in the new residual space set, as well as the maximum length, width, and area information of the remaining items, is recorded. This information enables the algorithm to make a preliminary prediction about the feasibility of current scheme and avoids unnecessary calculation.

It is noteworthy that frequent invocations to Hybrid-Search are time-consuming and will dramatically increase the overall computational effort. To avoid unnecessary invocations and improve the efficiency of Hybrid-Search, a special data structure (Hashtable) is used to record the feasibility of the routes already checked.

6 Computational experiments

The numerical experiment includes five parts. In the first part, we introduce the parameter-turning of AMA-ENS and the benchmark instances we used. In the second to fifth parts, we report the computational results of AMA-ENS for two versions of 2L-CVRP, and compare the results with other state-of-the-art metaheuristics in the literature. In the last part, we further study two versions of 2L-VRPB, namely 2L-VRPB (2|UO|L) and 2L-VRPB (2|UR|L), which are the extensions of the 2L-CVRP (2|UO|L) and 2L-CVRP (2|UR|L) respectively. These two variants can be frequently met in practical scenarios but, to the best of our knowledge, they haven't been studied yet. Our work in this part makes up for the vacancy.

The algorithm was implemented in C++ (single thread), and all the experiments were conducted on an ordinary PC equipped with an Intel i7 6700 CPU at 3.4G Hz and 16G RAM.

6.1 Benchmark instances and parameter setting

We use the classic 2L-CVRP benchmark instances to test the performance of our algorithm. This data set was proposed by [49] and has been adopted by most research work on 2L-CVRP and its related variants. Therefore, it is an ideal tool to measure the performance of the algorithm. To be specific, [49] generated 2L-CVRP data set by extending the 36 classic CVRP data set introduced by [2], and the final 2L-CVRP benchmark instances contains 180 instances. In each instance, a customer possesses one or more goods to be delivered. The information of each customer node includes not only its location, but the weight, length, and width of each of his goods. Overall, the 2L-CVRP data set could be divided into five classes, and each class contains 36 instances. In the first class, the length and width of each good are one unit, so all the instances belonging to the first class could be regarded as a benchmark data set for pure CVRP problem. The instances belonging to the remaining four classes were created by generating a certain number of goods with different sizes for each customer node according to certain rules. For more details about the generation of 2L-CVRP data set, the readers are advised to refer to [49].

Before conducting the formal experiments, we made the parameter-turning to ensure AMA-ENS run under the ideal parameter setting. 32 instances were used as the benchmark data set for the parameter-tuning. These instances were obtained by selecting No. 25-32 instances from each of classes 2-5. The parameters of AMA-ENS can be divided into two categories: routing related parameters and packing related parameters. For the former, we adopted the general setting suggested by [42] except for λ and μ, which are set to 10 and 15 respectively, since we found that the reduction of population size does not significantly affect the quality of the solution obtained, but it leads to less time consumption. We focused on the setting of three parameters related to packing, which directly affect the final solution quality: the initial penalty coefficient of packing constraint γ_p, the proportion of items to be loaded in the first stage of Two-Stage-Pack r_p, the maximum number of attempts in the second stage of Two-Stage-Pack $maxIter$. Considering that the influence of γ_p on the algorithm is significantly different from that of r_p and $maxIter$, we first tried to set the value of γ_p. We took the values of r_p and $maxIter$ as 1/3 and 100 respectively, and then tested the performance of AMA-ENS when γ_p is set to 30, 60, 100, and 150 respectively. Each instance was run 10 times under different parameter settings. Here, we evaluated a given parameter setting according to the calculation of $Dev\left(par\right)$ over the 32 instances: $Dev\left(par\right) = \frac{1}{32}\sum\frac{(C)_{i,j,par}-C^*_{i,j}}{C^*_{i,j}}$. In this equation, $C_{i,j,par}$ represents the best solution value found for instance i, j (the i-th instance from class j) in 10 runs, and $C^*_{i,j}$ represents the best solution value found under all different parameter settings. Once the value of r_p is determined, we continued to conduct the following tests: r_p was taken as 1/3 and 1/2 respectively, and $maxIter$ was taken as 50, 100, and 150 respectively. All the six combinations were tested and each was evaluated through the calculation of $Dev\left(par\right)$. After completing the above parameter-turning process, the parameter combination (γ_p, r_p, $maxIter$) was finally set to (100, 1/3, 100).

In the following experimental parts, state-of-the-art metaheuristics for solving 2L-CVRP and its variants in the literature are selected for comparison, including ACO [50], PRMP [51], VNS [18], BR-LNS [7, 13], SA [8]. All the above algorithms have been tested on the 2L-CVRP benchmark instances, and each of them ran 10 times for each instance. To make a fair comparison, our algorithm runs 10 times for each instance within a comparable time setting: the algorithm terminates if 5000 consecutive iterations are conducted without updating the current best solution, or the running time reaches one hour. The best solution for each instance obtained in 10 runs is reported, together with its first occurrence time.

6.2 Results for the first class of 2L-CVRP instances

Table 1 shows the comparison between the results of AMA-ENS and existing advanced metaheuristics for the first class of 2L-CVRP benchmark instances. Figure 1 further clearly presents the gap between the solutions of different metaheuristics and the current BKS on medium-scale and large-scale instances. As can be seen in Table 1, AMA-ENS finds new best solutions for 6 large-scale instances from No. 31 to No. 36, and matches the best-known solutions for 28 instances from the remaining 30 small and medium-sized instances. Meanwhile, the average time consumption of AMA-ENS is the smallest, only 55.74 seconds, which is much less than 460.53 seconds of VNS ([18]) and 448.63 seconds of SA ([8]). Therefore, judging from the solution quality and running time, AMA-ENS is a competitive metaheuristic algorithm for routing aspect.

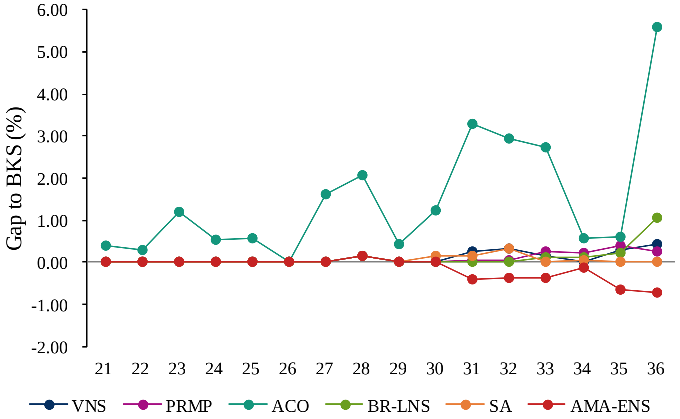

Fig. 1 Gap to the BKS[1] on the large-scale instances of class 1

[a] Gap to BKS (%): Percentage improvement between the solutions and BKS, Gap=100*(solution- BKS)/BKS. The X-axis represents the the current BKS level in the literature.

Table 1 Comparative results on the instances of Class 1

Inst.	BKS[1]	VNS		PRMP		ACO		BR-LNS		SA		AMA-ENS		
		Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Cost	Time (s)	Gap (%)[2]
1	278.73	278.73	0.00	278.73	0.00	278.73	0.10	278.73	0.01	278.73	0.90	278.73	0.00	0.00
2	334.96	334.96	0.00	334.96	0.00	334.96	0.10	334.96	0.00	334.96	0.30	334.96	0.00	0.00
3	358.40	358.40	0.10	358.40	0.00	358.40	0.20	358.40	0.00	358.40	1.00	358.40	0.00	0.00
4	430.88	430.89	0.00	430.88	0.00	430.88	0.30	430.88	0.00	430.89	0.90	430.89	0.00	0.00
5	375.28	375.28	0.00	375.28	0.00	375.28	0.30	375.28	0.00	375.28	0.00	375.28	0.00	0.00
6	495.85	495.85	0.10	495.85	0.00	495.85	0.30	495.85	0.06	495.85	2.50	495.85	0.01	0.00
7	568.56	568.56	0.00	568.56	0.00	568.56	0.20	568.56	0.00	568.56	0.00	568.56	0.00	0.00
8	568.56	568.56	0.00	568.56	0.00	568.56	0.20	568.56	0.00	568.56	0.00	568.56	0.00	0.00
9	607.65	607.65	0.10	607.65	0.00	607.65	0.60	607.65	0.00	607.65	1.10	607.65	0.00	0.00
10	535.74	535.80	0.10	535.80	0.10	535.80	2.30	535.80	5.18	535.80	5.80	535.80	0.01	0.01
11	505.01	505.01	0.00	505.01	0.00	505.01	0.80	505.01	0.18	505.01	0.60	505.01	0.00	0.00
12	610.00	610.00	0.90	610.00	0.20	610.00	1.60	610.00	0.46	610.00	5.40	610.00	0.32	0.00
13	2006.34	2006.34	0.10	2006.34	0.30	2006.34	1.30	2006.34	0.08	2006.34	0.00	2006.34	0.02	0.00
14	837.67	837.67	0.10	837.67	0.10	837.67	4.10	837.67	0.20	837.67	0.00	837.67	0.01	0.00
15	837.67	837.67	0.10	837.67	0.40	837.67	2.80	837.67	0.18	837.67	0.00	837.67	0.01	0.00
16	698.61	698.61	1.10	698.61	0.30	698.61	2.00	698.61	0.16	698.61	4.00	698.61	0.01	0.00
17	861.79	861.79	4.00	861.79	1.60	861.79	3.30	861.79	185.65	861.79	22.20	861.79	1.97	0.00
18	723.54	723.54	1.40	723.54	3.60	723.54	9.50	723.54	1.10	723.54	6.70	723.54	0.02	0.00
19	524.61	524.61	2.00	524.61	2.10	524.61	7.90	524.61	7.29	524.61	9.00	524.61	0.02	0.00
20	241.97	241.97	3.50	241.97	7.20	241.97	55.70	241.97	2.85	241.97	14.60	241.97	0.19	0.00
21	687.60	687.60	74.90	687.60	3.80	690.20	26.70	687.60	164.12	687.60	343.80	687.60	1.78	0.00
22	740.66	740.66	21.20	740.66	2.80	742.91	56.90	740.66	12.63	740.66	101.10	740.66	5.98	0.00
23	835.26	835.26	159.70	835.26	48.70	845.34	55.90	835.26	30.74	835.26	838.00	835.26	3.35	0.00
24	1024.69	1024.69	175.90	1024.69	38.10	1030.25	49.80	1024.69	490.50	1024.69	1250.20	1024.69	7.21	0.00
25	826.14	826.14	332.20	826.14	8.60	830.82	167.50	826.14	44.48	826.14	418.00	826.14	2.90	0.00
26	819.56	819.56	1.70	819.56	11.20	819.56	173.30	819.56	0.77	819.56	1.60	819.56	0.11	0.00
27	1082.65	1082.65	445.50	1082.65	172.30	1100.22	191.00	1082.65	9.50	1082.65	1306.00	1082.65	25.12	0.00
28	1040.70	1042.12	1021.50	1042.12	71.20	1062.23	252.20	1042.12	136.28	1042.12	24.60	1042.12	1.12	0.14
29	1162.96	1162.96	172.90	1162.96	121.90	1168.13	765.00	1162.96	147.85	1162.96	35.90	1162.96	10.25	0.00
30	1028.42	1028.42	1570.00	1028.42	267.50	1041.05	313.90	1028.42	371.68	1029.79	1435.80	1028.42	37.15	0.00
31	1299.21	1302.48	1813.80	1299.56	353.80	1341.89	517.80	1299.21	312.86	1301.03	1884.00	**1293.68**	97.61	**-0.45**
32	1296.18	1300.22	1976.10	1296.91	312.00	1334.26	519.70	1296.18	372.05	1300.30	2006.90	**1291.45**	116.56	**-0.42**
33	1296.13	1298.02	2204.10	1299.55	434.10	1331.69	479.20	1297.50	161.80	1296.13	1884.20	**1291.45**	197.53	**-0.36**
34	708.39	708.39	2125.20	709.82	328.20	712.32	621.40	709.08	554.20	708.66	1658.30	**707.57**	136.36	**-0.12**
35	862.79	865.39	2050.40	866.06	396.30	868.12	1468.20	864.63	382.43	862.79	1611.00	**857.19**	254.25	**-0.65**
36	583.98	586.49	2420.20	585.46	228.90	616.69	1589.80	590.16	560.74	583.98	1276.30	**579.71**	1106.82	**-0.73**
Avg.	769.37	769.80	460.53	769.70	78.20	776.04	203.94	769.69	109.89	769.62	448.63	768.69	55.74	-0.07

Bold indicates the new best solution found.

[1] The best known solution in the literature.

[2] The percentage improvement (%) on the current BKS level.
Lower number equates to better performance.

6.3 Results for the 2|UO|L of 2L-CVRP

Table 2 shows the computational results of AMA-ENS for the 2|UO|L version of 2L-CVRP. New best solutions are found for most instances: among the 144 instances, AMA-ENS finds 100 new best solutions, and matches the best-known solutions for 31 instances. The maximum improvement for a single instance is 6.9%. Compared with the existing best-known level, the average improvement rates of AMA-ENS in class 2 to 5 are 2.48%, 1.72%, and 1.11% respectively. For the class 5, although the quality of the solution remains at the same level as BKS as a whole from the perspective of the average gap (%), 12 new best solutions are still found.

In addition, for each instance from No. 1 to 36, we calculate its average best value from class 2 to 5, and therefore 36 average values are obtained, which can more clearly reflect the differences of solution quality obtained by different metaheuristics. A detailed comparison result is presented in Table 3. Figure 2 further illustrates the gaps between the average best value for the existing BKS and that of the solutions of different metaheuristics. In Figure 2, BKS corresponds to the horizontal axis.

It can be seen that the current BKS is mainly contributed by SA [8], since its corresponding curve is closest to the horizontal axis. Compared with the solution quality of PRMP [51], VNS [18], and SA [8], AMA-ENS achieves the largest improvement on the solution quality. Among the 36 average best values,

Table 2 Results for the 2|UO|L version of 2L-CVRP

Inst.	Class 2			Class 3			Class 4			Class 5		
	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]
1	278.73	278.73	0.00	284.52	**279.49**	**-1.77**	282.95	282.95	0.00	278.73	278.73	0.00
2	334.96	334.96	0.00	352.16	**349.92**	**-0.64**	334.96	334.96	0.00	334.96	334.96	0.00
3	387.70	**379.77**	**-2.05**	394.72	**385.32**	**-2.38**	362.41	**358.40**	**-1.11**	358.40	358.40	0.00
4	430.88	430.89	0.00	430.88	430.89	0.00	447.37	447.37	0.00	430.88	430.89	0.00
5	375.28	375.28	0.00	381.69	**375.28**	**-1.68**	383.87	383.88	0.00	375.28	375.28	0.00
6	495.85	495.85	0.00	497.17	498.16	0.20	498.32	498.32	0.00	495.75	495.85	0.02
7	725.46	**708.61**	**-2.32**	678.75	**660.53**	**-2.68**	700.72	**686.26**	**-2.06**	657.77	657.77	0.00
8	674.55	**664.30**	**-1.52**	738.43	**724.16**	**-1.93**	692.47	**688.32**	**-0.60**	609.90	609.90	0.00
9	607.65	607.65	0.00	607.65	607.65	0.00	621.23	**607.65**	**-2.19**	607.65	607.65	0.00
10	689.68	**665.76**	**-3.47**	615.68	**611.54**	**-0.67**	710.87	**703.64**	**-1.02**	678.66	**678.62**	**-0.01**
11	684.21	**642.78**	**-6.06**	706.73	**698.30**	**-1.19**	784.88	**765.04**	**-2.53**	624.82	624.82	0.00
12	610.57	**610.00**	**-0.09**	610.00	610.00	0.00	614.23	**610.23**	**-0.65**	610.00	610.00	0.00
13	2585.72	**2512.14**	**-2.85**	2436.56	**2370.66**	**-2.70**	2548.06	**2544.09**	**-0.16**	2334.78	**2334.59**	**-0.01**
14	1038.09	**1028.80**	**-0.89**	996.25	**989.08**	**-0.72**	981.00	**954.06**	**-2.75**	871.22	871.22	0.00
15	1013.29	**1002.91**	**-1.02**	1149.99	**1096.97**	**-4.61**	1181.30	**1164.77**	**-1.40**	1159.94	1159.94	0.00
16	698.61	698.61	0.00	698.61	698.61	0.00	703.35	703.35	0.00	698.61	698.61	0.00
17	863.66	**861.79**	**-0.22**	861.79	861.79	0.00	861.79	861.79	0.00	861.79	861.79	0.00
18	1004.99	**983.06**	**-2.18**	1069.45	**1013.72**	**-5.21**	1116.45	**1095.30**	**-1.89**	917.94	917.94	0.00
19	754.53	**715.31**	**-5.20**	771.66	**747.39**	**-3.15**	775.87	**759.63**	**-2.09**	644.59	644.59	0.00
20	524.91	**488.68**	**-6.90**	521.31	**513.53**	**-1.49**	537.56	**533.58**	**-0.74**	470.33	**468.69**	**-0.35**
21	992.83	**965.43**	**-2.76**	1116.58	**1087.87**	**-2.57**	970.37	**959.84**	**-1.09**	873.25	**870.82**	**-0.28**
22	1035.66	**979.29**	**-5.44**	1052.98	**1025.12**	**-2.65**	1045.91	**1044.08**	**-0.17**	930.83	**929.08**	**-0.19**
23	1035.18	**977.37**	**-5.58**	1074.30	**1047.37**	**-2.51**	1071.30	**1045.71**	**-2.39**	930.09	**922.34**	**-0.83**
24	1178.07	**1135.03**	**-3.65**	1080.88	**1073.75**	**-0.66**	1100.76	**1093.55**	**-0.66**	1028.04	1042.37	1.39
25	1407.86	**1334.66**	**-5.20**	1365.37	**1326.00**	**-2.88**	1398.02	**1365.25**	**-2.34**	1150.04	1150.69	0.06
26	1272.87	**1234.91**	**-2.98**	1342.19	**1315.86**	**-1.96**	1390.99	**1375.88**	**-1.09**	1213.04	1216.90	0.32
27	1313.12	**1268.01**	**-3.44**	1369.44	**1341.79**	**-2.02**	1314.05	**1294.35**	**-1.50**	1240.32	**1236.52**	**-0.31**
28	2551.41	**2515.35**	**-1.41**	2592.73	**2557.44**	**-1.36**	2585.92	**2526.69**	**-2.29**	2294.40	2307.46	0.57
29	2196.00	**2134.77**	**-2.79**	2087.15	**2058.57**	**-1.37**	2240.18	**2203.36**	**-1.64**	2127.60	**2126.97**	**-0.03**
30	1803.26	**1726.50**	**-4.26**	1821.83	**1779.06**	**-2.35**	1820.46	**1802.24**	**-1.00**	1521.91	**1517.09**	**-0.32**
31	2254.47	**2188.87**	**-2.91**	2268.64	**2216.04**	**-2.32**	2366.80	**2335.26**	**-1.33**	1987.08	1989.38	0.12
32	2241.02	**2170.92**	**-3.13**	2227.66	**2195.11**	**-1.46**	2252.39	**2228.37**	**-1.07**	1949.34	**1944.97**	**-0.22**
33	2249.68	**2177.36**	**-3.21**	2348.25	**2304.74**	**-1.85**	2373.63	**2344.96**	**-1.21**	1975.14	**1969.26**	**-0.30**
34	1170.77	**1136.53**	**-2.92**	1196.20	**1169.77**	**-2.21**	1193.18	**1183.02**	**-0.85**	1014.76	1015.48	0.07
35	1364.35	**1323.39**	**-3.00**	1436.52	**1406.93**	**-2.06**	1486.29	**1470.64**	**-1.05**	1236.42	**1229.22**	**-0.58**
36	1681.82	**1650.36**	**-1.87**	1757.43	**1738.61**	**-1.07**	1638.66	**1621.28**	**-1.06**	1470.26	1478.07	0.53
Avg.	1125.77	1094.57	-2.48	1137.28	1115.75	-1.72	1149.68	1135.50	-1.11	1026.79	1026.86	-0.01

Bold indicates the new best solution found.

[1] The best known solution in the literature.

[2] The percentage improvement (%) on the current BKS level. Lower number equates to better performance.

33 achieve better values. On average, AMA-ENS achieves an improvement of 1.41% over the existing BKS level.

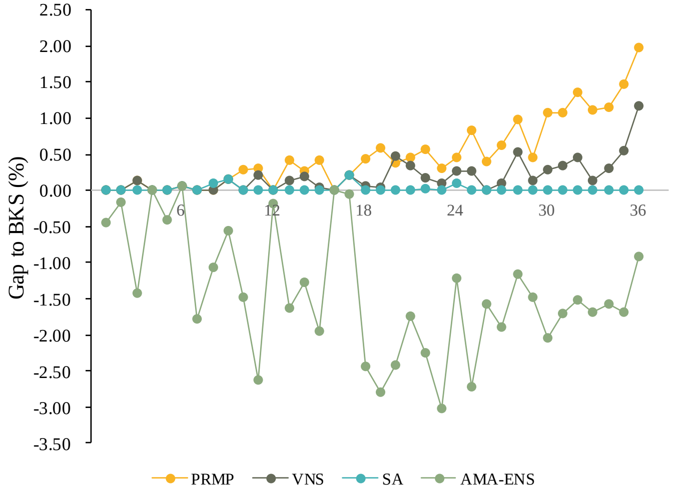

Fig. 2 Gap to the BKS[1] for the 2|UO|L version of 2L-CVRP (averaged over Class 2–5)

[a]Gap to BKS (%): Percentage improvement between the solutions and BKS, Gap=100*(solution- BKS)/BKS. The X-axis represents the the current BKS level in the literature.

Table 3 Comparison for the 2|UO|L version of 2L-CVRP (averaged over Classes 2-5)

Inst.	BKS[1]	PRMP			VNS			SA			AMA-ENS		
		Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]
1	281.23	281.23	0.40	0.00	281.23	1.20	0.00	281.23	5.70	0.00	**279.97**	1.09	**-0.45**
2	339.26	339.26	0.30	0.00	339.26	0.10	0.00	339.26	0.40	0.00	**338.70**	0.36	**-0.17**
3	375.81	376.32	0.40	0.14	376.32	0.90	0.14	375.81	0.70	0.00	**370.47**	2.17	**-1.42**
4	435.00	435.01	0.30	0.00	435.01	0.30	0.00	435.01	1.00	0.00	435.01	1.54	0.00
5	379.03	379.03	1.10	0.00	379.03	1.50	0.00	379.03	0.60	0.00	**377.43**	0.97	**-0.42**
6	496.77	497.04	0.30	0.05	497.04	1.00	0.05	497.04	1.80	0.05	497.04	1.01	0.05
7	690.67	690.67	1.60	0.00	690.67	2.50	0.00	690.67	1.90	0.00	**678.29**	1.47	**-1.79**
8	678.84	678.84	2.60	0.00	678.84	3.80	0.00	679.44	5.30	0.09	**671.67**	1.47	**-1.06**
9	611.05	612.01	1.60	0.16	612.01	1.30	0.16	612.01	2.40	0.16	**607.65**	2.64	**-0.56**
10	674.88	676.75	26.90	0.28	674.92	25.10	0.01	674.88	16.70	0.00	**664.89**	9.31	**-1.48**
11	701.07	703.22	27.20	0.31	702.47	59.90	0.20	701.07	23.70	0.00	**682.73**	3.83	**-2.62**
12	611.20	611.26	1.40	0.01	611.20	3.30	0.00	611.20	6.90	0.00	**610.06**	15.64	**-0.19**
13	2480.73	2491.18	52.70	0.42	2484.16	25.20	0.14	2480.73	22.00	0.00	**2440.37**	10.68	**-1.63**
14	973.23	975.88	164.30	0.27	975.06	295.50	0.19	973.23	53.40	0.00	**960.79**	99.31	**-1.28**
15	1128.18	1132.91	20.10	0.42	1128.60	246.30	0.04	1128.18	445.30	0.00	**1106.15**	100.01	**-1.95**
16	699.79	699.79	4.10	0.00	699.79	1.60	0.00	699.79	4.00	0.00	699.79	19.51	0.00
17	862.26	864.05	2.40	0.21	864.05	4.00	0.21	864.05	18.20	0.21	**861.79**	30.65	**-0.05**
18	1027.45	1031.95	33.00	0.44	1027.98	79.80	0.05	1027.45	160.50	0.00	**1002.50**	81.09	**-2.43**
19	737.40	741.78	24.30	0.59	737.73	250.50	0.04	737.40	206.60	0.00	**716.73**	91.38	**-2.80**
20	513.53	515.44	552.20	0.37	515.92	794.20	0.47	513.53	855.90	0.00	**501.12**	267.26	**-2.42**
21	988.30	992.78	241.50	0.45	991.63	751.10	0.34	988.30	1658.40	0.00	**970.99**	347.64	**-1.75**
22	1017.33	1023.01	166.60	0.56	1019.03	885.20	0.17	1017.56	1740.10	0.02	**994.39**	402.99	**-2.25**
23	1029.32	1032.36	336.80	0.30	1030.40	853.10	0.10	1029.32	1353.20	0.00	**998.20**	369.98	**-3.02**
24	1099.57	1104.64	319.60	0.46	1102.53	572.10	0.27	1100.64	923.10	0.10	**1086.18**	328.10	**-1.22**
25	1330.32	1341.26	921.70	0.82	1333.76	998.70	0.26	1330.32	1833.30	0.00	**1294.15**	490.37	**-2.72**
26	1306.59	1311.79	403.50	0.40	1306.60	1050.60	0.00	1306.59	1466.80	0.00	**1285.89**	503.69	**-1.58**
27	1309.92	1318.04	438.20	0.62	1311.27	874.50	0.10	1309.92	1696.00	0.00	**1285.17**	864.46	**-1.89**
28	2506.12	2530.46	3701.90	0.97	2519.35	2259.20	0.53	2506.12	2222.30	0.00	**2476.74**	2688.55	**-1.17**
29	2163.06	2173.02	1835.70	0.46	2166.14	2232.80	0.14	2163.06	2169.90	0.00	**2130.92**	2812.31	**-1.49**
30	1741.87	1760.59	2151.80	1.07	1746.82	2495.40	0.28	1741.87	1337.10	0.00	**1706.22**	2342.04	**-2.05**
31	2220.22	2244.13	2927.40	1.08	2227.79	2952.90	0.34	2220.22	2080.90	0.00	**2182.39**	2665.39	**-1.70**
32	2167.60	2196.85	3713.80	1.35	2177.66	2648.70	0.46	2167.60	1954.70	0.00	**2134.84**	2603.74	**-1.51**
33	2236.73	2261.68	1964.80	1.12	2239.91	2942.70	0.14	2236.73	1949.90	0.00	**2199.08**	2132.24	**-1.68**
34	1144.14	1157.22	3551.70	1.14	1147.67	2459.60	0.31	1144.14	2472.60	0.00	**1126.20**	3091.90	**-1.57**
35	1380.90	1401.17	2756.50	1.47	1388.55	2620.60	0.55	1380.90	2447.20	0.00	**1357.55**	3119.45	**-1.69**
36	1637.04	1669.44	4245.60	1.98	1656.00	3012.00	1.16	1637.04	2953.50	0.00	**1622.08**	2967.55	**-0.91**
Avg.	1110.46	1118.11	849.84	0.50	1113.23	872.42	0.19	1110.59	891.44	0.02	1093.17	790.88	-1.41

Bold indicates the new best solution found.

[1] The best known solution in the literature.

[2] The percentage improvement (%) on the current BKS level. Lower number equates to better performance.

6.4 Results for 2|UR|L of 2L-CVRP

Table 4 reports the results of AMA-ENS for the 2|UR|L version of 2L-CVRP. AMA-ENS obtains new best solutions for most instances, especially for class 2 to 4: among 144 instances, 82 new best solutions are found, and 55 instances match the corresponding best-known solutions. AMA-ENS achieves a maximum improvement of 4.12% for one single instance. For the instances of class 2 to 4, AMA-ENS achieves significant improvements of 1.6%, 0.87%, and 0.48% respectively. For class 5, the solution quality is basically at the same level as BKS.

Table 4 Results for the 2|UR|L version of 2L-CVRP

Inst.	Class2			Class 3			Class 4			Class 5		
	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]	BKS[1]	AMA-ENS	Gap (%)[2]
1	278.73	278.73	0.00	282.95	**279.49**	**-1.22**	282.95	282.95	0.00	278.73	278.73	0.00
2	334.96	334.96	0.00	352.16	**349.92**	**-0.64**	334.96	334.96	0.00	334.96	334.96	0.00
3	380.35	**371.72**	**-2.27**	385.32	385.32	0.00	358.40	358.40	0.00	358.40	358.40	0.00
4	430.88	430.89	0.00	430.88	430.89	0.00	447.37	447.37	0.00	430.88	430.89	0.00
5	375.28	375.28	0.00	379.94	**375.28**	**-1.23**	383.87	383.88	0.00	375.28	375.28	0.00
6	495.85	495.85	0.00	498.16	498.16	0.00	498.32	498.32	0.00	495.85	495.85	0.00
7	715.02	**699.52**	**-2.17**	664.96	**659.66**	**-0.80**	686.26	686.26	0.00	657.77	657.77	0.00
8	665.17	**664.30**	**-0.13**	738.43	**724.16**	**-1.93**	688.32	688.32	0.00	609.90	609.90	0.00
9	607.65	607.65	0.00	607.65	607.65	0.00	625.10	**607.65**	**-2.79**	607.65	607.65	0.00
10	667.42	**648.94**	**-2.77**	591.61	**584.80**	**-1.15**	703.64	703.64	0.00	678.62	678.62	0.00
11	664.48	**637.12**	**-4.12**	699.35	**685.80**	**-1.94**	771.93	**760.53**	**-1.48**	624.82	624.82	0.00
12	610.00	610.00	0.00	610.00	610.00	0.00	610.23	610.23	0.00	610.00	610.00	0.00
13	2502.65	**2463.55**	**-1.56**	2377.39	**2345.10**	**-1.36**	2533.79	**2500.85**	**-1.30**	2334.59	2334.59	0.00
14	1029.34	**1025.87**	**-0.34**	988.79	988.80	0.00	955.09	**954.06**	**-0.11**	871.22	871.22	0.00
15	1001.51	**1000.68**	**-0.08**	1116.07	**1096.97**	**-1.71**	1164.63	**1164.39**	**-0.02**	1159.94	1159.94	0.00
16	698.61	698.61	0.00	698.61	698.61	0.00	703.35	703.35	0.00	698.61	698.61	0.00
17	861.79	861.79	0.00	861.79	861.79	0.00	861.79	861.79	0.00	861.79	861.79	0.00
18	987.10	**971.48**	**-1.58**	986.30	**985.97**	**-0.03**	1100.52	**1095.12**	**-0.49**	917.94	917.94	0.00
19	723.93	**701.53**	**-3.09**	749.43	**742.27**	**-0.96**	747.03	**739.92**	**-0.95**	644.59	644.59	0.00
20	488.69	**483.60**	**-1.04**	511.46	**510.06**	**-0.27**	533.77	**528.33**	**-1.02**	466.79	466.79	0.00
21	964.49	**944.12**	**-2.11**	1086.72	**1071.20**	**-1.43**	959.82	**952.83**	**-0.73**	870.82	870.82	0.00
22	976.70	**957.04**	**-2.01**	1024.11	**1010.08**	**-1.37**	1041.80	**1033.58**	**-0.79**	928.02	928.02	0.00
23	984.00	**961.68**	**-2.27**	1041.60	**1031.44**	**-0.98**	1047.32	**1032.80**	**-1.39**	922.34	922.34	0.00
24	1140.13	**1112.35**	**-2.44**	1066.15	**1062.81**	**-0.31**	1086.09	**1081.90**	**-0.39**	1042.37	1042.37	0.00
25	1345.89	**1301.14**	**-3.32**	1333.64	**1314.75**	**-1.42**	1366.28	**1357.98**	**-0.61**	1149.66	1149.66	0.00
26	1257.00	**1220.09**	**-2.94**	1311.11	**1295.86**	**-1.16**	1362.22	**1359.75**	**-0.18**	1209.34	1209.34	0.00
27	1271.10	**1242.87**	**-2.22**	1329.33	**1315.87**	**-1.01**	1284.94	**1278.65**	**-0.49**	1231.52	**1222.66**	**-0.72**
28	2491.86	**2439.92**	**-2.08**	2541.02	**2522.66**	**-0.72**	2510.29	**2499.13**	**-0.44**	2276.71	2291.78	0.66
29	2129.10	**2081.81**	**-2.22**	2040.83	**2028.67**	**-0.60**	2199.79	**2193.98**	**-0.26**	2115.53	2116.51	0.05
30	1740.87	**1689.74**	**-2.94**	1767.72	**1753.16**	**-0.82**	1784.14	**1772.14**	**-0.67**	1512.71	1513.42	0.05
31	2162.88	**2120.38**	**-1.96**	2196.26	**2170.36**	**-1.18**	2314.76	**2302.72**	**-0.52**	1968.89	1974.17	0.27
32	2165.96	**2112.12**	**-2.49**	2166.18	**2149.86**	**-0.75**	2206.72	**2181.48**	**-1.14**	1938.96	**1938.42**	**-0.03**
33	2157.23	**2096.28**	**-2.83**	2276.31	**2235.41**	**-1.80**	2318.77	**2308.14**	**-0.46**	1946.51	1951.79	0.27
34	1121.67	**1091.89**	**-2.65**	1165.57	**1148.94**	**-1.43**	1163.96	**1160.03**	**-0.34**	1006.38	1012.15	0.57
35	1310.33	**1282.76**	**-2.10**	1393.90	**1374.65**	**-1.38**	1452.59	**1448.03**	**-0.31**	1224.21	**1218.27**	**-0.49**
36	1625.42	**1595.50**	**-1.84**	1708.05	**1678.23**	**-1.75**	1605.00	**1600.33**	**-0.29**	1457.05	1459.39	0.16
Avg.	1093.45	1072.55	-1.60	1110.55	1099.57	-0.87	1130.44	1124.27	-0.48	1022.76	1023.32	0.02

Bold indicates the new best solution found.
[1] The best known solution in the literature.
[2] The percentage improvement (%) on the current BKS level.
Lower number equates to better performance.

Table 5 compares the best average cost of classes 2–5 of each instance (No. 1-36) against previous metaheuristics for the 2|UR|L version. Figure 3 shows the relative gaps between the best average values of the existing BKS and that of different metaheuristics. Among the 36 results, 31 of them obtain better values, and the remaining 5 results all match the corresponding best-known values. AMA-ENS improves the existing BKS, on average, by 0.76%.

Table 5 Comparison for the 2|UR|L version of 2L-CVRP (averaged over Classes 2–5)

Inst.	BKS[1]	ACO			MS-BR			SA			AMA-ENS		
		Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]	Cost	Time (s)	Gap (%)[2]
1	280.84	281.16	–	0.11	280.84	3.30	0.00	281.13	0.70	0.10	**279.97**	0.48	**-0.31**
2	339.26	341.02	–	0.52	339.26	0.80	0.00	339.26	0.40	0.00	**338.70**	0.39	**-0.17**
3	370.62	372.93	–	0.62	370.62	34.80	0.00	371.62	2.10	0.27	**368.46**	0.91	**-0.58**
4	435.00	435.01	–	0.00	435.00	2.00	0.00	435.00	1.10	0.00	435.01	0.40	0.00
5	378.59	378.59	–	0.00	378.59	30.00	0.00	378.59	0.70	0.00	**377.43**	0.63	**-0.31**
6	497.04	497.04	–	0.00	497.05	3.80	0.00	497.04	1.70	0.00	497.04	0.25	0.00
7	681.00	688.50	–	1.10	681.00	20.30	0.00	681.00	1.20	0.00	**675.80**	0.64	**-0.76**
8	675.45	678.75	–	0.49	675.46	17.00	0.00	675.45	5.10	0.00	**671.67**	0.50	**-0.56**
9	612.01	612.02	–	0.00	612.01	32.30	0.00	612.01	2.80	0.00	**607.65**	1.44	**-0.71**
10	660.43	671.00	–	1.60	667.65	182.00	1.09	660.43	5.40	0.00	**654.00**	2.03	**-0.97**
11	690.56	698.25	–	1.11	690.56	146.50	0.00	690.56	5.20	0.00	**677.07**	4.72	**-1.95**
12	610.06	611.12	–	0.17	611.06	15.30	0.16	610.06	6.80	0.00	610.06	8.22	0.00
13	2437.15	2468.19	–	1.27	2437.15	114.50	0.00	2437.58	4.90	0.02	**2411.02**	5.16	**-1.07**
14	961.11	974.80	–	1.42	968.55	176.50	0.77	961.11	10.90	0.00	**959.99**	12.15	**-0.12**
15	1110.54	1132.49	–	1.98	1112.00	183.50	0.13	1110.54	46.40	0.00	**1105.50**	24.33	**-0.45**
16	699.79	699.79	–	0.00	699.80	26.50	0.00	699.79	5.10	0.00	699.79	8.49	0.00
17	861.79	862.36	–	0.07	861.79	12.50	0.00	861.79	15.40	0.00	861.79	14.34	0.00
18	997.97	1012.19	–	1.42	999.22	182.50	0.13	997.97	23.70	0.00	**992.63**	41.29	**-0.54**
19	716.24	726.96	–	1.50	722.17	268.80	0.83	716.24	168.40	0.00	**707.08**	39.72	**-1.28**
20	500.18	508.69	–	1.70	501.90	210.50	0.34	500.18	143.90	0.00	**497.19**	79.67	**-0.60**
21	971.45	989.24	–	1.83	977.03	348.80	0.57	971.45	1346.40	0.00	**959.74**	123.35	**-1.21**
22	994.77	1008.52	–	1.38	1001.75	297.30	0.70	994.77	1402.00	0.00	**982.18**	147.64	**-1.27**
23	998.81	1024.25	–	2.55	1011.19	420.50	1.24	998.81	903.80	0.00	**987.07**	197.71	**-1.18**
24	1083.69	1098.60	–	1.38	1092.90	213.50	0.85	1083.69	986.40	0.00	**1074.86**	160.60	**-0.81**
25	1298.87	1323.84	–	1.92	1320.27	362.50	1.65	1298.87	1389.20	0.00	**1280.88**	238.94	**-1.39**
26	1284.92	1314.34	–	2.29	1302.52	332.00	1.37	1284.92	1145.90	0.00	**1271.26**	267.51	**-1.06**
27	1279.22	1309.76	–	2.39	1304.14	362.00	1.95	1279.22	1529.90	0.00	**1265.01**	322.06	**-1.11**
28	2454.97	2526.81	–	2.93	2518.51	401.00	2.59	2454.97	1796.60	0.00	**2438.37**	2581.86	**-0.68**
29	2121.31	2175.33	–	2.55	2161.43	417.80	1.89	2121.31	1520.70	0.00	**2105.24**	1955.18	**-0.76**
30	1701.36	1742.15	–	2.40	1742.01	337.80	2.39	1701.36	1717.40	0.00	**1682.12**	893.57	**-1.13**
31	2160.70	2227.74	–	3.10	2204.44	472.80	2.02	2160.70	2244.60	0.00	**2141.91**	1820.12	**-0.87**
32	2119.46	2180.18	–	2.86	2167.61	394.00	2.27	2119.46	2036.50	0.00	**2095.47**	1441.45	**-1.13**
33	2174.71	2239.04	–	2.96	2222.42	459.50	2.19	2174.71	1925.10	0.00	**2147.91**	1732.20	**-1.23**
34	1114.40	1149.87	–	3.18	1142.25	458.00	2.50	1114.40	2399.10	0.00	**1103.25**	1992.90	**-1.00**
35	1345.26	1387.45	–	3.14	1392.05	471.80	3.48	1345.26	2376.00	0.00	**1330.93**	1979.94	**-1.07**
36	1598.88	1670.67	–	4.49	1653.05	375.30	3.39	1598.88	2646.50	0.00	**1583.36**	2754.05	**-0.97**
Avg.	1089.40	1111.63	–	1.57	1104.31	216.33	0.96	1089.45	772.72	0.01	1079.93	523.75	**-0.76**

Bold indicates the new best solution found.

[1] The best known solution in the literature.

[2] The percentage improvement (%) on the current BKS level. Lower number equates to better performance.

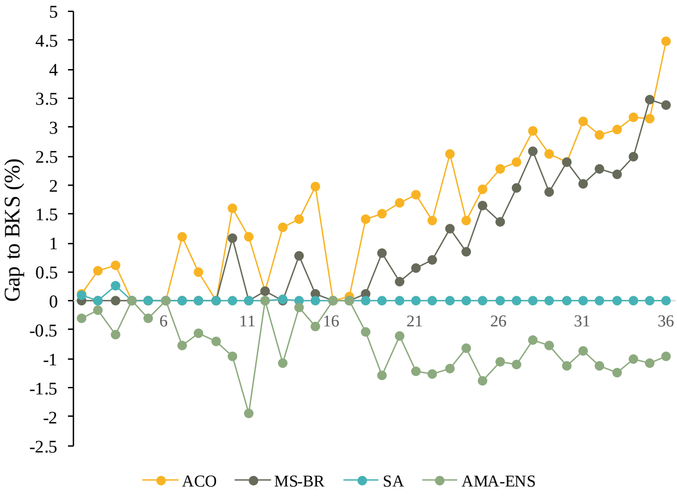

Fig. 3 Gap to the BKS[1] for the 2|UR|L version of 2L-CVRP (averaged over Class 2–5)

[a] Gap to BKS (%): Percentage improvement between the solutions and BKS, Gap=100*(solution- BKS)/BKS. The X-axis represents the the current BKS level in the literature.

6.5 Results for 2L-VRPB

2L-CVRP can be regarded as a special case of 2L-VRPB, that is, all the customers are linehaul customers and there are no backhaul customers. On the other hand, 2L-VRPB problem can be regarded as a special case of 2L-CVRP problem, where the distances from depot to backhaul customers and the distances from backhaul customers to linehaul customers are large enough. Therefore, the algorithm for solving 2L-CVRP can also be used to solve the 2L-VRPB problem without any change. After the efficiency of AMA-ENS is verified by two versions of 2L-CVRP, we use the algorithm to solve the corresponding two versions of 2L-VRPB, i.e. 2L-VRPB (2|UO|L) and 2L-VRPB (2|UR|L), which, to the best of my knowledge, haven't been investigated in the literature. Here, we consider two circumstances for the 2L-VRPB, that is, the ratio of linehaul customers to backhaul customers is 3:1 or 4:1. For each circumstance, we generate the corresponding test data set from the 2L-CVRP instances, which is the same as Domingz et al. 2016. Specifically, for each instance in 2L-CVRP, the last customer for every three (four) customers is set as backhaul customer. The termination condition and parameter setting of AMA-ENS for solving 2L-VRPB problem remain the same as our previous experiments. Each instance is run for ten times with different random seed settings, and the best solution is reported. Tables 6 and 7 presents the detailed computational results for 2L-VRPB with different radio of linehaul/backhaul.

7 Conclusions

We propose a new metaheuristic algorithm called AMA-ENS to solve the well-known 2L-CVRP problem, together with two variants of 2L-VRPB. Experimental results on the classic 2L-CVRP benchmark instances indicate that AMA-ENS achieves a significant improvement on the solution quality and outperforms the existing state-of-the-art metaheuristics. New best solutions are found for most instances of the 2L-CVRP data set, which constitutes the first and most noticeable contribution of our work. Then, we also solve two variants of 2L-VRPB, i.e. 2|UO|L and 2|OR|L, which can be frequently met in daily lives, and report the results of AMA-ENS on the corresponding benchmark data set. This constitutes a second contribution of our work. The competitive edge of AMA-ENS in terms of performance and time consumption, together with its adaptive penalty coefficient mechanism, make it have good versatility and flexibility for a wide class of problems composed of routing and packing. Therefore, AMA-ENS can be extended to solve more complex combinatorial problems, which means that it has greater practical application value in in reality logistics.

Table 6 Results for the 2L-VRPB with a linehaul/backhaul ratio of 3/1

Inst.	Class 1	Class 2		Class 3		Class 4		Class 5	
		2|UO|L	2|UR|L	2|UO|L	2|UR|L	2|UO|L	2|UR|L	2|UO|L	2|UR|L
1	269.32	269.32	269.32	270.16	270.16	269.49	269.49	269.32	269.32
2	303.78	303.78	303.78	303.78	303.78	303.78	303.78	303.78	303.78
3	353.02	354.33	353.02	356.28	356.28	353.02	353.02	353.02	353.02
4	375.34	375.34	375.34	375.34	375.34	375.34	375.34	375.34	375.34
5	373.99	373.99	373.99	378.06	378.06	373.99	373.99	373.99	373.99
6	432.44	432.44	432.44	434.33	434.33	432.44	432.44	432.44	432.44
7	647.79	702.85	702.85	702.85	702.85	703.11	703.11	686.46	686.46
8	647.79	694.15	694.15	715.00	715.00	720.33	720.33	673.28	673.28
9	553.83	553.83	553.83	553.83	553.83	560.17	560.17	553.83	553.83
10	545.77	597.77	597.77	574.43	564.21	665.96	665.96	602.91	602.91
11	545.77	608.29	604.69	608.36	605.35	676.88	673.26	594.40	594.40
12	513.96	516.33	513.96	513.96	513.96	526.68	526.68	513.96	513.96
13	2237.20	2441.59	2414.91	2400.35	2361.88	2415.83	2415.83	2374.03	2374.03
14	757.30	885.81	876.16	889.62	889.31	854.28	851.56	776.24	776.24
15	757.30	881.32	878.66	897.77	896.09	984.58	977.80	1011.93	1011.93
16	604.57	604.57	604.57	604.57	604.57	604.57	604.57	604.57	604.57
17	716.42	716.42	716.42	716.42	716.42	716.42	716.42	716.42	716.42
18	741.53	909.26	888.49	909.81	897.98	974.27	963.87	877.82	877.82
19	572.78	666.03	645.95	690.01	685.77	693.85	681.78	625.99	624.16
20	278.22	426.66	419.08	415.58	411.62	452.52	443.16	385.82	385.82
21	724.40	883.57	869.78	952.12	933.91	849.29	839.97	821.16	820.43
22	754.21	895.93	877.84	913.04	905.46	979.52	975.66	830.20	830.20
23	802.94	907.38	891.97	940.55	931.76	958.85	955.30	867.58	862.30
24	912.39	988.54	964.71	941.11	934.65	979.68	974.25	936.09	933.72
25	847.04	1177.32	1154.96	1154.03	1140.84	1213.88	1203.77	1035.23	1033.76
26	814.91	1081.59	1068.09	1175.87	1155.43	1160.36	1155.36	1082.40	1082.40
27	1002.83	1085.48	1059.25	1185.22	1153.12	1139.13	1117.08	1082.27	1081.62
28	1014.01	1936.03	1908.44	1988.64	1963.52	2003.25	1995.76	1856.82	1855.84
29	1125.36	1750.15	1723.29	1765.92	1750.86	1845.15	1823.99	1781.41	1780.04
30	1051.83	1449.31	1429.55	1526.70	1504.21	1495.79	1488.07	1382.94	1381.62
31	1257.91	1828.35	1777.43	1900.79	1862.67	1945.73	1906.26	1681.45	1673.88
32	1257.91	1821.81	1776.44	1801.80	1773.80	1853.55	1825.15	1646.51	1642.45
33	1243.90	1824.68	1788.07	1907.60	1867.86	1893.41	1885.59	1678.02	1668.63
34	631.44	913.61	899.18	949.86	927.98	973.73	953.93	851.60	846.16
35	791.02	1087.24	1063.74	1133.50	1116.26	1174.23	1160.13	1009.46	1009.94
36	544.20	1266.23	1238.55	1330.89	1290.85	1303.70	1280.57	1196.62	1186.47
Avg.	750.12	950.31	936.41	968.84	956.94	984.08	976.48	912.37	910.92

Table 7 Results for the 2L-VRPB with a linehaul/backhaul ratio of 4/1

Inst.	Class 1	Class 2		Class 3		Class 4		Class 5	
		2|UO|L	2|UR|L	2|UO|L	2|UR|L	2|UO|L	2|UR|L	2|UO|L	2|UR|L
1	263.75	263.75	263.75	263.75	263.75	275.77	275.77	263.75	263.75
2	319.36	319.36	319.36	319.36	319.36	319.36	319.36	319.36	319.36
3	345.90	364.84	364.84	367.40	367.20	345.90	345.90	345.90	345.90
4	382.15	382.15	382.15	382.15	382.15	395.94	395.94	382.15	382.15
5	376.91	385.88	382.59	382.59	382.59	382.59	382.59	382.59	382.59
6	436.89	436.89	436.89	436.89	436.89	451.09	451.09	436.89	436.89
7	623.57	661.08	661.08	675.61	657.16	694.17	694.17	671.38	667.26
8	623.57	660.81	660.81	700.89	698.95	657.16	657.16	631.85	631.85
9	515.48	515.48	515.48	515.48	515.48	518.13	518.13	515.48	515.48
10	517.83	628.99	625.91	574.20	574.20	653.03	653.03	593.90	586.97
11	517.83	612.83	602.12	680.22	679.04	710.49	710.49	581.65	581.65
12	541.30	541.30	541.30	541.30	541.30	550.61	549.61	541.30	541.30
13	2130.12	2453.35	2414.64	2358.87	2329.59	2390.10	2390.10	2293.35	2285.40
14	737.38	877.88	877.88	966.39	938.05	863.34	863.34	842.14	842.14
15	737.38	875.97	864.85	995.45	983.30	963.31	961.81	1024.67	1016.64
16	636.26	636.26	636.26	636.26	636.26	636.26	636.26	636.26	636.26
17	744.66	744.66	744.66	744.66	744.66	744.66	744.66	744.66	744.66
18	754.98	930.86	921.90	966.97	959.79	1011.34	1008.88	845.50	843.06
19	569.53	685.55	665.71	690.56	678.27	717.56	706.73	650.45	648.39
20	279.92	456.62	440.99	448.26	441.34	477.26	470.04	418.20	416.16
21	700.81	851.16	846.00	956.42	939.65	898.85	894.34	815.32	815.32
22	739.02	875.61	858.90	925.07	913.96	917.15	909.75	848.76	848.43
23	788.19	877.44	863.34	940.11	928.74	936.83	927.67	875.95	874.51
24	925.51	1019.79	994.81	955.79	954.71	990.22	986.30	953.52	953.52
25	864.64	1195.50	1160.11	1165.24	1151.66	1212.07	1201.58	1064.45	1063.60
26	813.48	1092.85	1066.92	1200.40	1177.34	1222.12	1208.22	1081.35	1081.35
27	1019.37	1144.84	1123.63	1203.91	1179.87	1173.28	1157.30	1127.97	1125.96
28	1025.16	2112.88	2096.47	2122.45	2096.38	2226.00	2185.96	1956.57	1952.97
29	1176.79	1793.50	1759.73	1795.35	1749.62	1892.01	1879.80	1799.58	1785.87
30	1050.48	1503.37	1462.06	1502.67	1473.12	1513.54	1506.19	1307.73	1305.46
31	1256.42	1860.07	1823.80	1930.26	1900.08	1992.07	1965.43	1745.40	1729.71
32	1256.42	1875.24	1839.49	1872.37	1840.22	1857.82	1818.67	1683.04	1676.14
33	1271.93	1906.66	1862.55	1974.34	1929.37	2043.26	2019.98	1737.57	1714.09
34	655.17	970.39	935.90	989.53	969.11	987.53	974.16	878.66	870.25
35	819.90	1132.35	1103.03	1221.85	1199.03	1241.94	1221.50	1098.91	1094.29
36	567.42	1394.21	1349.22	1467.96	1432.11	1356.15	1325.03	1185.48	1175.81
Avg.	749.60	973.34	957.48	996.42	982.34	1006.08	997.69	924.49	920.98

References

[1] Iori, M., Martello, S.: Routing problems with loading constraints. Top **18**(1), 4–27 (2010)

[2] Toth, P., Vigo, D.: The vehicle routing problem. siam monographs on discrete mathematics and applications, philadelphia. Pennsylvania (2002)

[3] Gajpal, Y., Abad, P.L.: Multi-ant colony system (macs) for a vehicle routing problem with backhauls. European Journal of Operational Research **196**(1), 102–117 (2009)

[4] Brandão, J.: A deterministic iterated local search algorithm for the vehicle routing problem with backhauls. Top **24**(2), 445–465 (2016)

[5] Jacobs-Blecha, C., Goetschalckx, M.: The vehicle routing problem with backhauls: properties and solution algorithms. National Transportation Research Board **13** (1992)

[6] Iori, M., Salazar-González, J.-J., Vigo, D.: An exact approach for the vehicle routing problem with two-dimensional loading constraints. Transportation science **41**(2), 253–264 (2007)

[7] Dominguez, O., Guimarans, D., Juan, A.A., de la Nuez, I.: A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls. European Journal of Operational Research **255**(2), 442–462 (2016)

[8] Wei, L., Zhang, Z., Zhang, D., Leung, S.C.: A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research **265**(3), 843–859 (2018)

[9] Malapert, A., Guéret, C., Jussien, N., Langevin, A., Rousseau, L.-M.: Two-dimensional pickup and delivery routing problem with loading constraints. In: First CPAIOR Workshop on Bin Packing and Placement Constraints (BPPC'08), p. 184 (2008)

[10] Zachariadis, E.E., Tarantilis, C.D., Kiranoudis, C.T.: A guided tabu search for the vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research **195**(3), 729–743 (2009)

[11] Leung, S.C., Zheng, J., Zhang, D., Zhou, X.: Simulated annealing for the vehicle routing problem with two-dimensional loading constraints. Flexible services and manufacturing journal **22**(1), 61–82 (2010)

[12] Leung, S.C., Zhang, Z., Zhang, D., Hua, X., Lim, M.K.: A metaheuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. European Journal of Operational Research **225**(2), 199–210 (2013)

[13] Dominguez, O., Juan, A.A., Faulin, J.: A biased-randomized algorithm for the two-dimensional vehicle routing problem with and without item rotations. International Transactions in Operational Research **21**(3), 375–398 (2014)

[14] Juan, A.A., Faulin, J., Ruiz, R., Barrios, B., Caballé, S.: The sr-gcws hybrid algorithm for solving the capacitated vehicle routing problem. Applied Soft Computing **10**(1), 215–224 (2010)

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

[16] Juan, A.A., Faulín, J., Jorba, J., Riera, D., Masip, D., Barrios, B.: On the use of monte carlo simulation, cache and splitting techniques to improve the clarke and wright savings heuristics. Journal of the Operational Research Society **62**(6), 1085–1097 (2011)

[17] Burke, E.K., Kendall, G., Whitwell, G.: A new placement heuristic for the orthogonal stock-cutting problem. Operations Research **52**(4), 655–671 (2004)

[18] Wei, L., Zhang, Z., Zhang, D., Lim, A.: A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research **243**(3), 798–814 (2015)

[19] Wei, L., Oon, W.-C., Zhu, W., Lim, A.: A skyline heuristic for the 2d rectangular packing and strip packing problems. European Journal of Operational Research **215**(2), 337–346 (2011)

[20] Alinaghian, M., Zamanlou, K., Sabbagh, M.S.: A bi-objective mathematical model for two-dimensional loading time-dependent vehicle routing problem. Journal of the Operational Research Society **68**(11), 1422–1441 (2017)

[21] Guimarans, D., Dominguez, O., Panadero, J., Juan, A.A.: A simheuristic approach for the two-dimensional vehicle routing problem with stochastic travel times. Simulation Modelling Practice and Theory **89**, 1–14 (2018)

[22] Song, X., Jones, D., Asgari, N., Pigden, T.: Multi-objective vehicle routing and loading with time window constraints: a real-life application. Annals of Operations Research **291**(1), 799–825 (2020)

[23] Ji, B., Zhou, S., Samson, S.Y., Wu, G.: An enhanced neighborhood search algorithm for solving the split delivery vehicle routing problem with two-dimensional loading constraints. Computers & Industrial Engineering **162**, 107720 (2021)

[24] Ferreira, K.M., de Queiroz, T.A., Toledo, F.M.B.: An exact approach for the green vehicle routing problem with two-dimensional loading constraints and split delivery. Computers & Operations Research **136**, 105452 (2021)

[25] Côté, J.-F., Gendreau, M., Potvin, J.-Y.: The vehicle routing problem with stochastic two-dimensional items. Transportation Science **54**(2), 453–469 (2020)

[26] Deif, I., Bodin, L.: Extension of the clarke and wright algorithm for solving the vehicle routing problem with backhauling. In: Proceedings of the Babson Conference on Software Uses in Transportation and Logistics Management, pp. 75–96 (1984). Babson Park, MA

[27] Parragh, S.N., Doerner, K.F., Hartl, R.F.: A survey on pickup and delivery problems. Journal für Betriebswirtschaft **58**(2), 81–117 (2008)

[28] Parragh, S.N., Doerner, K.F., Hartl, R.F.: A survey on pickup and delivery problems: Part i: Transportation between customers and depot. Journal für Betriebswirtschaft **58**, 21–51 (2008). https://doi.org/10.1007/s11301-008-0033-7

[29] Toth, P., Vigo, D.: Vehicle routing: problems, methods, and applications (2014)

[30] Osman, I.H., Wassan, N.A.: A reactive tabu search meta-heuristic for the vehicle routing problem with back-hauls. Journal of Scheduling **5**(4), 263–285 (2002)

[31] Wassan, N.: Reactive tabu adaptive memory programming search for the vehicle routing problem with backhauls. Journal of the Operational Research Society **58**(12), 1630–1641 (2007)

[32] Zachariadis, E.E., Kiranoudis, C.T.: An effective local search approach for the vehicle routing problem with backhauls. Expert Systems with Applications **39**(3), 3174–3184 (2012)

[33] Cuervo, D.P., Goos, P., Sörensen, K., Arráiz, E.: An iterated local search algorithm for the vehicle routing problem with backhauls. European Journal of Operational Research **237**(2), 454–464 (2014)

[34] Goksal, F.P., Karaoglan, I., Altiparmak, F.: A hybrid discrete particle

swarm optimization for vehicle routing problem with simultaneous pickup and delivery. Computers & industrial engineering **65**(1), 39–53 (2013)

[35] Avci, M., Topaloglu, S.: An adaptive local search algorithm for vehicle routing problem with simultaneous and mixed pickups and deliveries. Computers & Industrial Engineering **83**, 15–29 (2015)

[36] Küçükoğlu, İ., Öztürk, N.: An advanced hybrid meta-heuristic algorithm for the vehicle routing problem with backhauls and time windows. Computers & Industrial Engineering **86**, 60–68 (2015)

[37] Reil, S., Bortfeldt, A., Mönch, L.: Heuristics for vehicle routing problems with backhauls, time windows, and 3d loading constraints. European Journal of Operational Research **266**(3), 877–894 (2018)

[38] Koch, H., Bortfeldt, A., Wäscher, G.: A hybrid algorithm for the vehicle routing problem with backhauls, time windows and three-dimensional loading constraints. OR spectrum **40**(4), 1029–1075 (2018)

[39] Malapert, A., Guéret, C., Jussien, N., Langevin, A.: Two dimensional pickup and delivery routing problem with loading constraints. Technical report, CIRRELT-2008-37 (2008)

[40] Pinto, T., Alves, C., Valério de Carvalho, J.: Variable neighborhood search algorithms for the vehicle routing problem with two-dimensional loading constraints and mixed linehauls and backhauls. International Transactions in Operational Research **27**(1), 549–572 (2020)

[41] Vidal, T., Crainic, T.G., Gendreau, M., Lahrichi, N., Rei, W.: A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. Operations Research **60**(3), 611–624 (2012)

[42] Vidal, T., Crainic, T.G., Gendreau, M., Prins, C.: A unified solution framework for multi-attribute vehicle routing problems. European Journal of Operational Research **234**(3), 658–673 (2014)

[43] Vidal, T.: Split algorithm in o (n) for the capacitated vehicle routing problem. Computers & Operations Research **69**, 40–47 (2016)

[44] Vidal, T., Crainic, T.G., Gendreau, M., Prins, C.: A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. Computers & operations research **40**(1), 475–489 (2013)

[45] Goeke, D., Schneider, M.: Routing a mixed fleet of electric and conventional vehicles. European Journal of Operational Research **245**(1), 81–99 (2015)

[46] Hiermann, G., Hartl, R.F., Puchinger, J., Vidal, T.: Routing a mix of conventional, plug-in hybrid, and electric vehicles. European Journal of Operational Research **272**(1), 235–248 (2019)

[47] Wang, Y., Chen, L.: Two-dimensional residual-space-maximized packing. Expert Systems with Applications **42**(7), 3297–3305 (2015)

[48] Lai, K.K., Chan, J.W.M.: Developing a simulated annealing algorithm for the cutting stock problem. Computers & Industrial Engineering **32**(1), 115–127 (1997)

[49] Gendreau, M., Iori, M., Laporte, G., Martello, S.: A tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints. Networks: An International Journal **51**(1), 4–18 (2008)

[50] Fuellerer, G., Doerner, K.F., Hartl, R.F., Iori, M.: Ant colony optimization for the two-dimensional loading vehicle routing problem. Computers & Operations Research **36**(3), 655–673 (2009)

[51] Zachariadis, E.E., Tarantilis, C.D., Kiranoudis, C.T.: Integrated distribution and loading planning via a compact metaheuristic algorithm. European Journal of Operational Research **228**(1), 56–71 (2013)