

Split Delivery Vehicle Routing Problem with Minimum Delivery Amounts

Jun Pan^{1,2}, Zhuo Fu^{1*}, Hongwei Chen³

¹ School of Traffic and Transportation Engineering, Central South University, Changsha 410075, China

² School of Science, Central South University of Forestry and Technology, Changsha 410004, China

³ School of Civil Engineering, Hunan City University, Yiyang 413000, China

Corresponding Author Email: zhfu@csu.edu.cn

<https://doi.org/10.18280/jesa.520306>

Received: 10 March 2019

Accepted: 1 June 2019

Keywords:

vehicle routing problem (VRP), minimum delivery amounts, tabu search (TS) algorithm, adaptive search

ABSTRACT

In the split delivery vehicle routing problem with minimum delivery amounts (SDVRP-MDA), the customer demand can be split among different distribution routes, as long as the quantity delivered to the customer each time remains above the minimum threshold. Despite saving transport cost, the split of customer demand may complicate the order processing and disturb the customer. These problems can be solved satisfactorily by the constraint of the minimum delivery amounts. Therefore, this paper puts forward an adaptive tabu search (ATS) algorithm to solve the SDVRP-MDA. Two new neighbourhood structures are adopted, the adaptive adjustment strategy involving both intensification and diversification search is introduced. The proposed method was tested on 32 standard examples, using 4 different splitting coefficients. On average, the test results only deviated from the current best solution by -0.50~0.22. A total of 94 best solutions were derived and 1 new best solution was obtained. This means our method enjoys good adaptability and robustness.

1. INTRODUCTION

The Split Delivery Vehicle Route Problem (SDVRP) is one extension of the classic VRP. Despite the constraint of the VRP that Each customer can only be served by one vehicle, the SDVRP allows multiple vehicles to serve one customer at the same time. Therefore, the customer demands can be split between several routes. Dror and Trudeau [1] and Archetti et al. [2] respectively demonstrated that the number of vehicles used and the total cost of delivery can be reduced under conditions that allow for splitting delivery.

The classic SDVRP assumes that the customer demands can be split into any units. In the practical logistics distribution process, split delivery will complicate the order processing and interfere with the customers many times. These interferences are not desirable for customers. Therefore, the minimum threshold should be specified for the quantity delivered to the customer each time, which has certain practical significance for both the supply and demand sides. Gulczynski et al. [3] first proposed a split delivery vehicle routing problem with minimum delivery amounts (SDVRP-MDA), i.e., the amount of delivery service for each time should not be lower than a minimum amount, but this constraint is expressed as a limit on the delivery amount of money. For instance, Gulczynski et al. [3] mentioned that the American Pizza Hunt has a take-away of about \$8, and in Newfoundland the oil transport company only provides home delivery services for customers with orders greater than \$200. Anthony Han et al. [4] pointed out that the threshold amount of distribution in 7-11, Taiwan's largest supermarket chain was NT\$300. In addition, many e-commerce enterprises charge an additional fee for orders below a certain amount. In summary, further research on the SDVRP-MDA has important theoretical and practical significance.

Gulczynski et al. [3] used the splitting coefficient $p(0 \leq p \leq 1)$ to define the minimum delivery amounts, i.e., the minimum delivery amount of customer i , $MDA_i = \lceil pd_i \rceil$, where d_i is the demand of customer i , and $\lceil \cdot \rceil$ is upward rounding function. Obviously, each customer can be served by $\lfloor 1/p \rfloor$ ($\lfloor \cdot \rfloor$ is a round-down function). When $p > 0.5$, SDVRP-MDA degenerates to a non-splittable VRP; when $p = 0$, the SDVRP-MDA is equivalent to the SDVRP. Thus, the splitting coefficient p is set to the interval (0.0.5) in this paper.

The SDVRP-MDA is mainly studied in two aspects: the property of the solution and the solution algorithm. In terms of the property of the solution, Gulczynski et al. [3] proved that if the distance satisfies the triangle inequality, there may be two identical customer nodes in the two different routes of the optimal solution in the SDVRP-MDA, namely, the 2-split cycle defined by Dror et al. This is not held for the SDVRP, indicating that the structure of the optimal solution in the SDVRP-MDA is more complicated. Xiong et al. [5] through analysis for the best situation of the SDVRP-MAD, proved that when the splitting coefficient $p < 0.5$, the DVRP-MAD can save up to 50 % of the delivery cost, and at $p = 0.5$, it can save up to 1/3.

Compared with abundant research results on the SDVRP, there have been few studies on the SDVRP-MDA solution algorithm. Gulczynski et al. [3] proposed the first hybrid heuristic algorithm for the SDVRP-MDA, which is called as the EMIP-MDA+ERTR: Firstly, the improved clark-wright algorithm was used to construct feasible solutions of the VRP as the initial solutions; then the initial solutions were improved by an endpoint mixed integer program with minimum delivery amounts (EMIP-MDA) to provide the first solutions for the SDVRP-MDA; since the scale of the model increases exponentially with the number of customer points, the model

solves the strategy of controlling the calculation time to obtain a satisfactory solution; finally an enhanced record-to-record travel algorithm (ERTR) was applied to optimize the solution. Numerical experiments showed that EMIP-MDA+ERTR can obtain a satisfactory solution within a reasonable computing time. Besides, Gulczynki also proposed 21 new instances of the SDVRP-MDA. Anthony Fu et al. [4] put forward a multi-start variable neighbourhood search algorithm (SCR+VND). The SCR+VND introduces five kinds of neighbourhood structures, and tests all their 120 sorts to determine the search order of the neighbourhoods. The experimental results show that the SCR+VND is superior to the EMIP-MDA+ERTR in solving efficiency and quality.

The tabu search algorithm (TS) is generally used as a meta heuristic algorithm for solving combinatorial optimization problems. It's also adopted by many scholars to solve the SDVRP. Ho and Haugland [6] first applied the TS algorithm to solve SDVRP with time windows. Archetti et al. [7] proposed a three-stage tabu search algorithm (SPLITABU). Compared with the local search algorithm proposed by Dror and Trudeau [1], the SPLITABU can achieve better results for the test examples. Aleman and Hill [8] presented a tabu search with vocabulary building approach (TSVBA). Berbotto et al. [9] put forward a random granularity tabu search algorithm (RGT). Both TSVBA and RGT enjoy excellent search performance and are comparable to other algorithms for solving the SDVRP. Sicilia et al. [10] proposed a hybrid optimization algorithm based on the tabu search and variable neighborhood search metaheuristic for the rich vehicle routing problem. In view of this, this paper designs an adaptive TS algorithm combining diversification and intensification search according to the characteristics of the SDVRP-MDA.

2. PROBLEM DESCRIPTION AND MATHEMATICAL MODEL

The SDVRP-MDA problem is described as follows. in a logistics distribution system, the distribution centre shall provide distribution services for a group of customers with determined demands; starting from the distribution centre, a fleet of vehicles will serve the customers along the planned route, and meet the two constraints: (1) The load of the vehicle cannot exceed its capacity; (2) The customer demand must be met and can be serviced by multiple vehicles, but the quantity of delivery per time cannot be lower than the given minimum amounts. The goal of the SDVRP-MDA is to minimize the total transport cost.

It assumes that each vehicle starts from the distribution centre to serve the customer along the route in turn, and finally returns to the distribution centre; the vehicle has the same capacity and the number of vehicles is unlimited; the delivery cost of the vehicle is proportional to the distance.

The SDVRP-MDA is defined on a complete graph $G=(V,E)$, where $V = \{0,1, \dots, n\}$ is the vertex set and $E = \{e_{ij}|i, j \in V, i \neq j\}$ is the edge set. Vertex 0 is the distribution centre, and other vertexes $\{1, \dots, n\}$ represent customers; c_{ij} is the travel distance between the customer nodes i and j ; m is the number of vehicles available; Q is the capacity of the vehicle. d_i represents the demand quantity of customer i ; p represents the splitting coefficient; $MDA_i = [pd_i]$, indicating the minimum amount of delivery service for customer i . x_{ijk} is a set of Boolean variables. When the vehicle serving the k th route arrives at the customer point j from the customer point i ,

x_{ijk} is equal to 1; otherwise, it is 0.

$$\min \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ij} x_{ijk} \quad (1)$$

$$\text{s.t.} \quad \sum_{i=0}^n \sum_{k=1}^m x_{ijk} \geq 1 \quad j = 0, \dots, n \quad (2)$$

$$\sum_{i=1}^n x_{ilk} = \sum_{j=1}^n x_{ljk} \quad l = 0, \dots, n; \quad k = 1, \dots, m \quad (3)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \quad k = 1, \dots, m; \quad S \subseteq N \quad (4)$$

$$MDA_i \sum_{j=0}^n x_{ijk} \leq q_{ik} \leq d_i \sum_{j=0}^n x_{ijk} \quad (5)$$

$$i = 1, \dots, n; \quad k = 1, \dots, m$$

$$\sum_{k=1}^m q_{ik} = d_i \quad i = 1, \dots, n \quad (6)$$

$$\sum_{i=1}^n q_{ik} \leq Q \quad k = 1, \dots, m \quad (7)$$

$$x_{ijk} \in \{0, 1\} \quad (8)$$

$$i = 0, \dots, n; \quad j = 0, \dots, n; \quad k = 1, \dots, m$$

In the model, equation (1) is the objective function with the smallest total distance; equation set (2) indicates that each customer is served by at least one vehicle; equation set (3) is the flow conservation constraint, and (4) is the subtour elimination constraints. equation set (5) indicates that the delivery amount of customer i cannot be less than the minimum at each time, and not exceed the demand; equation set (6) indicates that the customer demand must be satisfied; equation set (7) means that the vehicle's load cannot exceed its capacity.

3. TABU SEARCH ALGORITHM

The TS algorithm was proposed by Glover [11]. Its framework is similar to the local neighbourhood search algorithm. The most important feature of the TS algorithm is reflected in introducing the optimization process of the tabu table record and guiding the movement from the current solution to the neighbourhood solution. A large number of numerical experiments show that the TS algorithm is one of the most effective methods for solving the VRPs. In this paper, a new adaptive search TS algorithm was proposed to solve SDVRP-MDA. It's innovated in the aspects as follows: (1) Two new neighbourhood structures are proposed; (2) According to the positional relationship between customer nodes, the similarity between customer nodes and the adjacent route of the customer point are defined; (3) Based on the concept of adjacent route, an adaptive search strategy combining intensification and diversification strategies is proposed.

3.1 Initial solution

The initial solution was generated quickly by the greedy insertion algorithm. First, select the customer node farthest from the depot to generate the first route; then, select the customer node with the smallest insertion cost, and incorporate all or part of the demands into the route; finally, repeat this process until all demands for all customer node were merged into the route. In particular, it depends on the remaining capacity of the vehicle in the service route, the unsatisfied demand of the customer and the minimum split amount to judge whether the demand of the customer point is completely or partially inserted into a route.

let u_i indicate the amount of demand that the customer i has not yet satisfied, and p_r represents the remaining capacity of the vehicle in the route r . If $u_i \leq p_r$, the customer i can be directly inserted into the route r ; otherwise, the unsatisfied demand u_i of the customer i will be split into two parts: one is the service quantity q_{ir} inserted in the route r , and the other is the remaining unserved demand. Considering the limitation of vehicle capacity and minimum delivery amounts, the remaining demand should be $\max\{u_i - p_r, MDA_i\}$, and the quantity of service inserted in route r is:

$$q_{ir} = u_i - \max\{u_i - p_r, MDA_i\} \quad (9)$$

If $q_{ir} < MDA_i$, the minimum delivery amount cannot be met, and the insertion of route r is not allowed in the customer node. The initial solution was constructed in the steps as follows:

Step 1: Initialize the list of unserved customer nodes $L = \{1, 2, \dots, n\}$, and the demand quantity of the customer point service $u_i = q_i (i = 1, \dots, n)$. initialized the route set $R = \Phi$ and the non-insertable route set of each customer node $F_i = \Phi (i = 1, \dots, n), k = 1$;

Step 2: Select the customer node v farthest from the depot in L , and generate the route $r_k = (0 - v - 0)$, $R = R \cup r_k$, $L = L \setminus \{v\}$, $q_{vk} = u_v$, $r_k = Q - q_{vk}$, $u_v = 0$;

Step 3: For each node i in L , update the non-insertable route set F_i . When $u_i > p_k$, q_{ir} was calculated according to formula (9); if $q_{ir} < MDA_i$, then $F_i = F_i \cup \{r_k\}$;

Step 4: Select candidate nodes for insertion. For each point i in L and each route r in $R \setminus F_i$, calculate the minimum insertion cost of the customer point i into the route r :

$$I_{ir} = \min\{c_{ai} + c_{ib} - \lambda c_{ab} | (a, b) \in r\} \quad (10)$$

where, a, b are two adjacent nodes of the route r , and λ is the control parameter, which is 0.7 according to the experiment; i^* and r^* are the customer nodes and routes that cause I_{ir} to reach the minimum, namely:

$$I_{i^*r^*} = \min\{I_{ir} | i \in L, r \in R \setminus F_i\} \quad (11)$$

Step 5: Insert candidate nodes. If $I_{i^*r^*} \leq 2c_{i^*,0}$, i.e., the insertion cost is less than the cost of generating a new route, then i^* is inserted into r^* , and:

$$q_{i^*r^*} = \begin{cases} u_{i^*} - \max\{u_{i^*} - p_{r^*}, MDA_{i^*}\}, & u_{i^*} > p_{r^*} \\ u_{i^*}, & u_{i^*} \leq p_{r^*} \end{cases} \quad (12)$$

$u_{i^*} = u_{i^*} - q_{i^*r^*}$, $p_{r^*} = p_{r^*} - q_{i^*r^*}$, Otherwise, $k = k + 1$; a new route $r_k = (0 - i^* - 0)$ is generated; $R = R \cup r_k$,

$$p_k = Q - u_{i^*}, \quad u_{i^*} = 0;$$

Step 6: If $u_{i^*} = 0$, then $= L \setminus \{i^*\}$. If $L = \emptyset$, return to step 3; otherwise the iteration is terminated.

3.2 Definition of adjacent routes

Toth and Vigo [12] found that the edge with a large cost value c_{ij} has a low probability of being in a high-quality solution, and discarding these edges does not affect the effect of the solution. In order to improve the efficiency of the search and avoid the invalid search process, the adjacent route of the customer point was defined. First, taking the two factors of delivery cost and relative position between customer nodes into account, the similarity between customer nodes was defined according to formula (13).

$$s_{ij} = \frac{I}{\frac{c_{ij}}{c_i^{max}} + \frac{\theta_{ij}}{\bar{\theta}}} \quad (13)$$

where, c_i^{max} is the maximum distance from the customer node i to other nodes; with reference to Aleman et al. [13], the angle of the route was defined; θ_{ij} is the position angle between the customer nodes i and j ; (x_i, y_i) is the position coordinates of the customer point i . Accordingly, the route angle was defined as $\theta_r = \max\{\theta_{ij} | (i, j) \in r\}$; $\bar{\theta}$ is the average value of the route angle in the initial solution.

Definition: For the route r , if r contains at least one of the q customer nodes nearest to the customer point node i , then route r is called an adjacent route of customer node i , and A_i indicates the set of adjacent routes for customer node i .

3.3 Neighbourhood structure

In order to improve the global search ability of the algorithm, this paper uses a hybrid neighbourhood search strategy to define six neighbourhood structures (N1, ..., N6). Among them, N1, ..., N4 are the neighbourhood search structure between routes, which are executed between adjacent routes of the customer nodes, and the route operation with the largest cost saving value is selected for each iteration; N5 is the neighbourhood search structure within the route; N6 is a hybrid random neighbourhood search structure. N3 and N4 are extensions of the proposed the SDVRP neighbourhood structure shift* by Alam, which are two new neighbourhood structures.

(1) Relocation (N₁)

Given a customer node $i \in r_s$ and a route $r_k \in A_i$, the customer node i was removed from the current route r_s and then inserted into r_k . This operation is feasible on the condition that the service quantity q_{is} of the customer node i in the route r_s does not exceed the remaining load p_k of the route r_k .

(2) 2-opt* (N₂)

This neighborhood structure was introduced by Potvin and Rousseau [14]. Given two customer nodes $i \in r_s$ and $j \in r_k$, r_s and r_k are adjacent routes of j and i respectively, and the nodes after node i in r_s and the nodes after the point j in route r_k were exchanged. This operation is feasible on condition that the total service amount of the customer nodes in the route after the exchange does not exceed the capacity of the vehicle.

(3) Ejection chain (N₃)

Ejection chains is a novel neighborhood structure defined by Glover [15]. Rego [16] Brandão [17] and Euchi [18] adopt the idea to the local search for the VRP. A ejection chain can be viewed as a series of triplets, each consisting of three consecutive nodes in a route. And the neighborhood solution is obtained by moving a node to the position occupied by another. Inspired by the idea, this paper proposed the new neighborhood structure. Given an route $r_k \in A_i$ between customer point $i \in r_s$ and i , i was deleted from the current route r_s , and r_k was inserted; then, the node j was removed from the route r_k to another route $r_t \in A_j$. This operation is feasible on the condition that when the customer node i removed from r_k to r_s , the capacity of the service vehicle in r_s was overloaded, but the service amount q_{jk} of the customer j in r_k was greater than or equal to the overload amount; when j was removed from r_k , the vehicle capacity in r_s can be restored, and r_t has a customer point j with sufficient remaining capacity. This operator is illustrated in Figure 1.

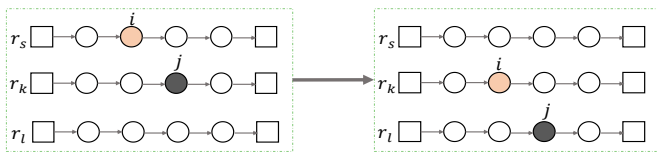


Figure 1. Ejection chain

(4) Split ejection chain, (N₄)

This neighborhood structure extends ejection chain by permitting the demand of the customer involved in the move. Given an adjacent route $r_k \in A_i$ of the customer nodes $i \in r_s$ and i , i was removed from the current route r_s , and r_k was inserted; then an adjacent route $r_t \in A_j$ ($r_t \neq r_s$) between the point $j \in r_k$ and j was selected, and j was splitted between the routes r_k and r_t . This operation is feasible on the condition that after the customer point i moved from r_k to r_s , the vehicle capacity of the service in r_s was overloaded, and the remaining capacity of r_t is greater than or equal to the overload of r_s , but less than the service amount q_{jk} of the route r_k for j . Therefore, part of the service amount at route r_k is moved into r_t to meet the capacity limit of the service route vehicle. This operator is illustrated in Figure 2.

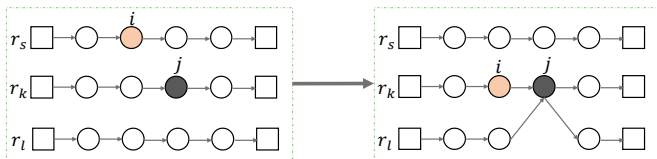


Figure 2. Split ejection chain

(5) 2-opt, (N₅)

The 2-opt in the route was proposed by Lin and Kernighan [19], which is the switching operation of the arc in the route. Two arcs that are not adjacent in the route were deleted and another two arcs were added.

(6) Mixed neighbourhood, (N₆)

One of the N₁, N₃, and N₄ was randomly selected for the neighbourhood operation.

3.4 Intensification and diversification search strategies

Intensification and diversification search are two important

aspects of the TS algorithm. Intensification search is applied to fully search the neighbourhood of the best solution for finding a better solution, while diversification search is to broaden the search area and jump out of the local optimum. Among the six neighbourhood structures designed in Section 3.3 above, N₁, N₂, N₃, N₄, N₅ are used for the former search, and N₆ is for the latter. In the intensification search phase, when the customer node and the neighbourhood structure were selected, the customer node and each of its neighbouring routes were operated accordingly, and the neighbourhood solution with the maximum saving cost was selected as a candidate solution; in the diversification search phase, for the selected customer node, one route was randomly selected a route from its adjacent route set in each iteration to perform a neighbourhood operation. The entire search process adaptively adjusts these two search strategies by combining the current best solution and the number of consecutive iteration steps without improving the best solution. If a better solution is not found after a certain number of iterations in the current search area, then a diversification search strategy is executed; in this search phase, if the best solution is updated, then the intensification search is assumed.

3.5 Tabu duration and aspiration criterion

In order to avoid the search of local loops, the TS algorithm records the attributes of the solutions in the tabu list, and in the next iteration, uses the information in the tabu list to prohibit or conditionally select these searched nodes. If the neighbourhood operation N₁ is performed and the operated customer node is i , then in the next θ step the customer node i is not allowed to execute the N₁ operation; if the neighbourhood operation N₂, N₃, N₄ is performed and the associated customer nodes are respectively i and j , then for the next θ step the customer nodes (i , j) prohibits the execution of the corresponding neighbourhood operation. With reference to Archetti et al. [7] the tabu duration θ was randomly taken within the interval $[\sqrt{10}, \sqrt{10+l}]$; if $n+g < 100$ (n represents the number of customer nodes, g is the number of routes in the current solution), then $l = n+g$; if $n+g \geq 100$, $l = \frac{3}{2}(n+g)$. The aspiration criterion is that if the delivery cost of the current tabu solution is less than the current best solution, the current solution is exempt.

3.6 Algorithm framework

The termination criterion of the algorithm is that the total number of iterations reaches a given maximum value, or that the current best solution is not improved within a given number of consecutive iterations.

According to the update rule of the current solution, if the objective function value of the best solution in the candidate solution set is lower than the current best solution, the best candidate solution is selected as the current solution, and the current best solution is updated, otherwise from the top 5 best non-tabu solutions of the candidate solution set, one is randomly selected as the current solution.

The relevant parameters in the algorithm are denoted as:

Iter: number of current iterations

maxIter: total number of maximum iterations

consIter: consecutive iterations without improving the current best solution

max-consIter: maximum number of consecutive iterations

that allow the current best solution to be unimproved
diver-consIter: iteration threshold for transferring to a diversity search

Step 1: Initialize the parameters and set *Iter*:=0;

Step 2: Generate an initial solution and use it as the current solution *s* and the current best solution *s**;

Step 3: If *consIter* < *diver - consIter*, transfer to the intensification search, and randomly select a neighbourhood structure from *N*₁, *N*₂, *N*₃, *N*₄, *N*₅ to search; otherwise, switch to diversification search and use *N*₆ for neighbourhood search;

Step 4: Update the current solution *s* under the condition that the Tabu rule and the aspiration criteria are met.

If $f(s) < f(s^*)$, then $s^* := s$, *consIter*:=0;
 otherwise *consIter*:= *consIter* + 1;

Step 5: Update the Tabu list and the adjacent route set of the customer nodes;

Step 6: If *Iter* < *maxIter* or *consIter* < *max - consIter*, returns to step 3; otherwise, the algorithm terminates.

4. NUMERAL EXPERIMENTS

4.1 Test instances

The performance of this algorithm was tested on two sets of standard examples. The first set of examples (set B) includes the test examples of the SDVRP proposed by Belenguer, and the second set (set G) was constructed by Gulczynski et al. [3] according to the test examples of SDVRP proposed by Chen et al. [20] In set G, customer nodes were symmetrically distributed, and the approximate optimal solutions could be visualized and generated by pre-designed route patterns. In order to keep the routes in these approximate optimal solutions unchanged, Gulczynski designed a systematic method which can modify the customer demand according to different splitting coefficients. This algorithm was programmed with Matlab R2014a and implemented on the i5-2430M, 2.4GHz processor, and 4GB memory microcomputer. It's also compared with the results of the EMIP-MDA+ERTR algorithm proposed by Gulczynski et al. [3] and the SRC+VND algorithm by Anthony Fu et al. [4]

4.2 parameter value

The test parameters of the algorithm were set as follows: (1)

The threshold *q* for selecting the adjacent route. When the number of customer nodes is $n < 50$, $q = 10$; when $n \geq 50$, $q = 15$;
 (2) Total number of maximum iterations *maxIter*. it's set *maxIter*=1000+10*n* by experiments; (3) The maximum number of consecutive iterations *max-consIter*=5*n* that allows the current best solution to be unimproved.

4.3 Test results

Table 1~4 list the test results of the algorithms in Set B under the four splitting coefficients $p=0.1, 0.2, 0.3$, and 0.4 . The ATS stands for the adaptive TS algorithm proposed in this paper. The proposed ATS is compared with two other representative algorithms. SRC+VND denoted the multi-start variable neighbourhood algorithm (Anthony Fu et al.) which contains five kinds of neighbourhood structures. Anthony Fu et al. [4] tested all the 120 sorting sequences, and SRC+VND represents the variable neighbourhood algorithm with the optimal neighbourhood search order; SRC+ *VND*_{all} is a variable neighbourhood algorithm that runs all 120 neighbourhood sorts. EMIP-MDA+ERTE is the hybrid heuristic algorithm proposed by Gulczynski et al. [3] BKS is provided by Gulczynski et al. [3] The calculation time of the algorithms is in seconds.

It can be found from Table 1~4 that *SRC+VND*_{all} almost obtained all the best solutions with highest solution quality, but it took the longest calculation time and had the lowest operation efficiency; all test results of EMIP-MDA+ERTE are inferior to the current best solution; SRC+VND had a calculating example (in bold in the table) achieving the current best results; the ATS proposed in this paper output two current best solutions, and the results of one example (S101D2_100, $p=0.1$) is better. In terms of the evaluation deviation (Aver. gap) of solution results from the current best solution, the deviation of ATS under the four splitting coefficient is -0.42 %, -0.56 %, -0.27 %, -0.50 %, respectively, which are all below the average deviation of EMIP-MDA+ERTE. Under the condition of $p=0.1$ and 0.3 , the average deviation of ATS is higher than that of SRC+VND, while under other conditions, it is lower than that of SRC+VND. Generally speaking, ATS differ less to SRC+VND. Due to different programming language and test platform, the solution time of the algorithm also vary, making it difficult to directly compare. From the last column of Table 1~4, it can be seen that the ATS calculation time is within an acceptable time range.

Table 1. Results for the SDVRP-MDA test for Set B: $p=0.1$

<i>p</i>	Set B	BKS	EMIP-MDA+ERTE ^a	SRC+VND ^b	SRC+VND _{all} ^b		ATS ^c		
			cost	cost	Time ^b (s)	cost	Time ^b (s)	cost	time ^c (s)
0.1	S51D2_50	717.35	717.35	713.30	3.99	709.77	439.65	711.25	50.94
	S51D3_50	969.99	969.99	953.17	4.84	949.78	561.03	956.31	33.77
	S51D4_50	1588.91	1588.91	1569.89	6.68	1569.89	775.82	1578.40	62.88
	S51D5_50	1373.98	1373.98	1344.68	5.49	1335.55	666.33	1353.20	33.89
	S51D6_50	2225.51	2225.51	2197.64	7.96	2197.04	916.53	2221.70	75.64
	S76D2_75	1106.86	1106.86	1100.90	15.85	1095.53	1782.50	1108.40	172.24
	S76D3_75	1446.48	1457.40	1448.39	20.78	1440.56	2362.41	1445.10	502.93
	S76D4_75	2118.86	2123.16	2095.95	25.96	2095.95	3040.63	2121.90	435.90
	S101D2_100	1398.13	1398.13	1407.35	48.20	1395.42	5600.83	1391.70	1002.56
	S101D3_100	1929.96	1930.86	1900.60	46.29	1899.66	5404.45	1931.00	437.80
	S101D5_100	2862.14	2862.34	2851.33	53.01	2839.49	6325.75	2868.90	879.00
	Average	1612.56	1614.04	1598.47	21.73	1593.51	2534.18	1607.98	335.23
	Avg. gap (%)		0.09	-0.87		-1.23		-0.42	
	BKS improved		-	9		11		7	

^a P4, 3.0GHz, ^b i7-3770, 3.4GHz, ^c i5-2430M, 2.4GHz

Table 2. Results for the SDVRP-MDA test for Set B: $p=0.2$

p	Set B	BKS	EMIP-MDA+ERTE [6] ^a		SRC+VND ^b		SRC+VND _{all} ^b		ATS ^c	
			cost	cost	Time ^b (s)	cost	Time ^b (s)	cost	time ^c (s)	
0.2	S51D2_50	717.35	717.35	712.59	3.79	709.77	424.37	714.32	48.03	
	S51D3_50	969.99	969.99	959.49	4.26	955.45	511.86	960.20	25.43	
	S51D4_50	1593.69	1593.69	1589.32	6.19	1578.15	709.12	1603.10	30.73	
	S51D5_50	1377.99	1377.99	1359.22	4.66	1351.12	553.29	1353.30	25.06	
	S51D6_50	2285.37	2285.37	2240.14	5.07	2240.14	619.05	2245.20	73.99	
	S76D2_75	1116.64	1116.64	1116.07	13.53	1104.27	1621.75	1105.23	158.12	
	S76D3_75	1446.48	1446.48	1450.04	17.07	1445.63	2095.96	1450.30	157.96	
	S76D4_75	2118.86	2118.86	2112.62	22.89	2103.78	2837.47	2106.12	736.53	
	S101D2_100	1398.87	1398.87	1401.57	44.32	1396.78	5544.72	1399.50	389.45	
	S101D3_100	1929.96	1929.96	1917.56	43.31	1904.09	5319.77	1917.60	687.78	
	S101D5_100	2862.14	2862.14	2866.29	42.96	2861.88	5379.26	2866.20	535.15	
	Average	1619.76	1619.76	1611.36	18.91	1604.64	2328.78	1611.01	260.75	
	Avg. gap (%)		0	-0.52		-0.99		-0.56		
	BKS improved		-	8		11		7		

Table 3. Results for the SDVRP-MDA test for Set B: $p=0.3$

p	Set B	BKS	EMIP-MDA+ERTE ^a		SRC+VND ^b		SRC+VND _{all} ^b		ATS ^c	
			cost	Time ^b (s)	cost	Time ^b (s)	cost	time ^c (s)	Time ^b (s)	
0.3	S51D2_50	717.35	717.35	716.17	3.09	711.56	370.07	712.52	63.52	
	S51D3_50	969.99	969.99	966.68	4.32	959.16	508.91	968.15	27.30	
	S51D4_50	1597.89	1597.89	1597.10	4.43	1595.40	528.17	1601.40	41.27	
	S51D5_50	1383.71	1383.71	1356.64	4.02	1349.55	488.59	1353.40	76.10	
	S51D6_50	2301.51	2301.51	2307.14	2.84	2307.14	357.24	2323.30	40.42	
	S76D2_75	1116.64	1116.64	1118.41	14.20	1108.20	1676.14	1118.20	402.00	
	S76D3_75	1453.17	1453.25	1446.81	15.23	1446.81	1912.85	1449.90	267.65	
	S76D4_75	2151.49	2151.49	2123.06	17.88	2123.06	2255.93	2138.60	322.76	
	S101D2_100	1398.88	1401.85	1407.19	40.76	1396.99	5108.96	1403.90	371.11	
	S101D3_100	1939.96	1939.96	1921.27	37.44	1918.41	4678.31	1925.60	830.99	
	S101D5_100	2901.76	2901.76	2887.90	38.45	2873.36	4691.72	2899.80	474.15	
	Average	1630.21	1630.49	1622.58	16.61	1617.24	2052.44	1626.80	265.21	
	Avg. gap (%)		0.02	-0.43		-0.82		-0.27		
	BKS improved		-	8		21		6		

Table 4. Results for the SDVRP-MDA test for Set B: $p=0.4$

p	Set B	BKS	EMIP-MDA+ERTR ^a		SRC+VND ^b		SRC+VND _{all} ^b		ATS ^c	
			cost	Time ^b (s)	cost	Time ^b (s)	cost	Time ^b (s)	cost	time ^c (s)
0.4	S51D2_50	717.35	717.35	717.31	3.10	713.60	367.89	713.56	51.49	
	S51D3_50	978.41	978.41	962.21	3.70	959.16	445.00	969.66	34.14	
	S51D4_50	1621.30	1612.30	1619.87	3.40	1613.06	405.11	1612.30	75.17	
	S51D5_50	1389.32	1389.32	1373.82	3.32	1360.01	399.73	1363.10	66.05	
	S51D6_50	2402.35	2402.35	2395.33	1.37	2395.33	174.85	2402.30	30.63	
	S76D2_75	1116.64	1116.64	1113.78	13.51	1112.58	1656.92	1116.10	315.40	
	S76D3_75	1453.17	1453.17	1453.43	14.21	1446.88	1712.80	1454.60	211.06	
	S76D4_75	2167.27	2167.27	2143.90	16.50	2129.97	2035.99	2148.20	458.08	
	S101D2_100	1398.88	1398.88	1403.70	39.42	1397.75	4862.50	1398.80	554.73	
	S101D3_100	1942.94	1942.94	1935.08	31.09	1922.10	3938.04	1929.80	756.74	
	S101D5_100	2904.61	2904.61	2898.98	32.15	2890.76	3950.67	2900.90	594.70	
	Average	1643.93	1643.93	1637.95	14.71	1631.02	1813.59	1637.21	286.20	
	Avg. gap (%)		0.00	-0.38		-0.82		-0.50		
	BKS improved		-	8		10		10		

Table 5~7 give the test results of the algorithm in Set G. The best solution (BKS) in the tables is the sum of the route lengths of the solutions generated based on the pre-designed route patterns. Since the calculation times of EMIP-MDA+ERTE and SRC+VND under different splitting coefficients p are approximately equal, the calculation time under the condition of $p=0.1$ was only listed in Table 5 and Table 6. Test results consistent with the best solution were marked in bold. The rounding error of the data may cause slight deviations between the test results and the best solutions. These test results were indicated by underlined boldfaces in the tables. From the test results of Table 5~7, it can be seen that ATS can find the best

solution when the customer nodes do not exceed 144; with the enlargement of the problem scale, the test results were compared with the best solution to find that the average deviation of ATS does not exceed 0.22 %, while that of EMIP-MDA+ERTE is between 0.93 % and 1.42 %, indicating that ATS can obtain better calculation results.

The test results show that the calculation performance of ATS is significantly superior to that of EMIP-MDA+ERTE, but slightly worse than SRC+VND. The neighbourhood search order of SRC+VND is preferably selected according to the test results of the example in advance, and the calculation effect of such a search order may vary due to different examples.

However, compared with SRC+VND, the neighbourhood search of ATS is more flexible, and the intensification or diversification strategy can be adaptively adjusted according

to the calculation effect. Therefore, the ATS algorithm is of greater applicability and robustness.

Table 5. Results for the SDVRP-MDA test for Set G: $p=0.1$

Set G	BKS	$p=0.1$					
		EMIP-MDA+ERTR ^a		SRC+VND ^b		ATS ^c	
		cost	time (s)	cost	time (s)	cost	time (s)
MD1_8	228.28	228.28	1.06	228.28	0.09	228.28	0.53
MD2_16	720.00	720.00	291.81	720.00	0.11	720.00	12.57
MD3_16	430.58	430.58	5.52	430.58	0.11	430.58	1.24
MD4_24	631.05	631.05	227.62	631.05	0.22	631.05	6.78
MD5_32	1402.40	1402.43	644.03	1402.40	0.47	1402.40	17.32
MD6_32	831.24	831.24	644.95	831.24	0.47	831.24	16.48
MD7_40	3588.28	3588.28	645.52	3588.30	0.86	3588.28	65.23
MD8_48	5040.00	5060.00	646.91	5040.00	1.37	5040.00	68.23
MD9_48	2044.20	2074.12	645.16	2044.20	1.47	2044.20	107.55
MD10_64	2684.88	2691.69	648.84	2684.90	3.29	2684.88	154.68
MD11_80	13200.00	13220.00	662.44	13200.00	6.08	13200.00	14.36
MD12_80	7150.58	7182.93	659.81	7150.58	5.99	7150.59	211.54
MD13_96	10042.40	10111.80	667.02	10042.40	10.90	10042.40	256.17
MD14_120	10711.06	10845.90	1644.98	10711.07	23.45	10711.07	564.53
MD15_144	15004.20	15181.70	1640.47	15004.22	43.09	15004.20	574.89
MD16_144	3631.30	3755.70	1625.83	3631.30	40.09	3750.10	568.32
MD17_160	26362.46	26628.46	1654.72	26362.36	52.03	26381.24	587.54
MD18_160	14200.90	14477.80	1639.72	14200.92	70.28	14219.34	543.89
MD19_192	19964.86	20432.20	1649.31	19964.87	126.77	20033.00	675.21
MD20_240	39484.23	40202.50	1691.84	39484.21	219.84	39484.23	1000.60
MD21_288	11645.50	12014.60	1656.14	11645.47	488.91	11740.00	1135.72
Average	8999.92	9129.06	933.03	8999.90	52.18	9015.09	313.49
Avg. gap (%)		0.93		0.00		0.22	
BKS found		6		21		15	

Table 6. Results for the SDVRP-MDA test for Set G: $p=0.2$

Set G	BKS	$p=0.2$					
		EMIP-MDA+ERTR ^a		SRC+VND ^b		ATS ^c	
		cost	time (s)	cost	time (s)	cost	time (s)
MD1_8	228.28	228.28		228.28		228.28	0.53
MD2_16	720.00	720.00		720.00		720.00	12.57
MD3_16	430.58	430.58		430.58		430.58	1.24
MD4_24	631.05	631.05		631.05		631.05	6.78
MD5_32	1402.40	1402.40		1402.40		1402.40	17.32
MD6_32	831.24	831.24		831.24		831.24	16.48
MD7_40	3588.28	3588.28		3588.28		3588.28	65.23
MD8_48	5040.00	5060.00		5040.00		5040.00	68.23
MD9_48	2044.20	2063.50		2044.20		2044.20	107.55
MD10_64	2684.88	2704.89		2684.88		2684.88	154.68
MD11_80	13200.00	13280.00		13200.00		13200.00	214.36
MD12_80	7150.58	7182.93		7150.58		7150.59	211.54
MD13_96	10042.40	10131.60		10042.40		10042.40	256.17
MD14_120	10711.06	10733.10		10711.07		10711.80	564.53
MD15_144	15004.20	15116.40		15004.22		15004.20	574.89
MD16_144	3631.30	3865.24		3631.30		3753.10	568.32
MD17_160	26362.46	26519.50		26362.36		26383.12	587.54
MD18_160	14200.90	14559.20		14200.92		14218.20	543.89
MD19_192	19964.86	20300.40		19964.87		20033.00	675.21
MD20_240	39484.23	40102.30		39484.21		39484.23	1000.60
MD21_288	11645.50	12438.60		11645.47		11740.00	1135.72
Average	8999.92	9137.55		8999.92		9015.31	323.02
Avg. gap (%)		1.17		0.00		0.22	
BKS found		7		21		15	

Table 7. Results for the SDVRP-MDA test for Set G: $p=0.3$ and 0.4

Set G	BKS	$p=0.3$				$p=0.4$			
		EMIP-MDA+ERTR ^a SRC+VND ^b		ATS ^c		EMIP-MDA+ERTR ^a SCR+VND ^b		ATS ^c	
		cost	cost	cost	time (s)	cost	cost	cost	time (s)

MD1_8	228.28	228.28	228.28	228.28	0.53	228.28	228.28	228.28	0.40
MD2_16	720.00	720.00	720.00	720.00	12.57	720.00	720.00	720.00	12.57
MD3_16	430.58	430.58	430.58	430.58	1.24	430.58	430.58	430.58	1.24
MD4_24	631.05	631.05	631.05	631.05	6.78	631.05	631.05	631.05	6.78
MD5_32	1402.40	1402.40	1402.40	1402.40	17.32	1414.75	1402.40	1402.40	17.32
MD6_32	831.24	831.24	831.24	831.24	16.48	830.26	831.24	831.24	16.48
MD7_40	3588.28	3588.28	3588.28	3588.28	65.23	3588.28	3588.28	3588.28	65.71
MD8_48	5040.00	5040.00	5040.00	5040.00	68.23	5040.00	5040.00	5040.00	69.13
MD9_48	2044.20	2063.50	2044.20	2044.20	107.55	2059.03	2044.20	2044.20	107.45
MD10_64	2684.88	2710.64	2684.89	2684.89	154.68	2708.80	2684.88	2684.88	124.51
MD11_80	13200.00	13334.10	13200.00	13200.00	214.36	13240.00	13200.00	13200.00	216.33
MD12_80	7150.58	7170.58	7150.58	7150.59	211.54	7260.01	7150.58	7150.59	210.11
MD13_96	10042.40	10112.40	10042.40	10042.40	256.17	10233.50	10042.40	10042.40	267.14
MD14_120	10711.06	10836.30	10711.07	10711.07	564.53	10865.10	10711.07	10711.80	533.60
MD15_144	15004.20	15172.10	15004.22	15004.20	574.89	15202.80	15004.22	15004.20	590.12
MD16_144	3631.30	3962.67	3631.30	3631.30	568.32	3445.50	3443.69	3631.13	565.41
MD17_160	26362.46	26646.50	26362.36	26381.24	587.54	26904.70	26362.35	26383.12	587.54
MD18_160	14200.90	14420.20	14200.92	14218.20	543.89	14447.60	14200.92	14218.20	543.89
MD19_192	19964.86	20355.70	19964.87	20033.00	675.21	20608.90	20032.12	20043.76	675.21
MD20_240	39484.23	40018.30	39484.21	39484.23	1000.60	40551.40	39484.21	39484.23	1010.53
MD21_288	11645.50	12652.90	11645.47	11740.00	1135.72	11909.10	11457.88	11772.14	1137.12
Average	8999.92	9158.46	8999.92	9009.38	323.02	9158.08	8985.25	9011.54	321.84
Avg.gap (%)		1.42	0.00	0.06		1.00	-0.06	0.08	
BKS found		8	21	17		7	21	17	

5. CONCLUSIONS AND PROSPECT

In the logistics process, the split of customer demand is beneficial to internal logistics operations, but it usually leads to additional order processing and interference with customers. The SDVRP-MDA can effectively solve this dilemma. However, compared with the SDVRP, the current research on SDVRP-MDA is not sufficient enough. For this, the adaptive TS algorithm was proposed in this paper to solve the SDVRP-MDA. The proposed method was tested on 32 standard benchmarks, using 4 different splitting coefficients. In the 128 examples tested, the average deviation between the test results and the current best solution ranged from -0.50 to 0.22; 94 best solutions were obtained, and a new best solution was found. This algorithm is innovated in the following aspects:

- (1) Two new neighbourhood structures are proposed;
- (2) According to the positional relationship between the customer nodes, the similarity between the customer nodes and the adjacent route of the customer node are defined;
- (3) Based on the concept of adjacent routes, an adaptive search strategy combining intensifying and diversifying strategies is proposed;
- (4) The algorithm can adaptively adjust between intensification and diversification search, which has better adaptability and robustness.

Although this algorithm has achieved certain results, further research is needed for the VRPs with minimum delivery constraints due to its strong applicability and complexity. It's expected that the proposed adaptive search strategy and neighbourhood structure shall be applied to other issues related to the SDVRP, which can develop more practical computing tools for other related solutions.

ACKNOWLEDGEMENTS

This study was supported by the Natural Science Foundation of China (Project No. 71271220).

The authors would like to thank all colleagues and students who contribute to this study. We are grateful to Dr. Shengbing

Che and Qiong Long who as the reviewers provide some constructive comments.

We thank the editor and series editor for constructive criticisms of an earlier version of this chapter. The errors, idiocies and inconsistencies remain our own.

REFERENCES

- [1] Dror, M., Trudeau, P. (1989). Savings by split delivery routing. *Transport*, 23(3): 141-145. <http://dx.doi.org/10.1287/trsc.23.2.141>
- [2] Archetti, C., Savelsbergh, M.W.P., Speranza, M.G. (2006). Worst-case analysis for split delivery vehicle routing problems. *Transportation Science*, 40(2): 226-234. <http://dx.doi.org/10.1287/trsc.1050.0117>
- [3] Gulczynski, D., Golden, B., Wasil, E. (2010). The split delivery vehicle routing problem with minimum delivery amounts. *Transportation Research Part E: Logistics and Transportation Review*, 46(5): 612-626. <http://dx.doi.org/10.1016/j.tre.2009.12.007>
- [4] Han, A.F.W., Chu, Y.C. (2016). A multi-start heuristic approach for the split-delivery vehicle routing problem with minimum delivery amounts. *Transportation Research Part E: Logistics and Transportation Review*, 88: 11-31. <http://dx.doi.org/10.1016/j.tre.2016.01.014>
- [5] Xiong, Y., Gulczynski, D., Kleitman, D., Golden, B., Wasil, E. (2013). A worst-case analysis for the split delivery vehicle routing problem with minimum delivery amounts. *Optimiz. Lett.*, 7(7): 1597-1609. <http://dx.doi.org/10.1007/s11590-012-0554-9>
- [6] Ho, S.C., Haugland, D. (2004). A tabu search heuristic for the vehicle routing problem with time windows and split deliveries. *Comput. Oper. Res.*, 31(12): 1947-1964. [http://dx.doi.org/10.1016/S0305-0548\(03\)00155-2](http://dx.doi.org/10.1016/S0305-0548(03)00155-2)
- [7] Archetti, C., Speranza, M.G., Hertz, A. (2006). A tabu search algorithm for the split delivery vehicle routing problem. *Transportation Science*, 40(1): 64-73. <http://dx.doi.org/10.1287/trsc.1040.0103>
- [8] Aleman, R.E., Hill, R.R. (2010). A tabu search with vocabulary building approach for the vehicle routing

- problem with split demands. *Int. J. Meta heurist.*, 1(1): 55-80.
<http://dx.doi.org/10.1504/IJMHEUR.2010.033123>
- [9] Berbotto, L., García, S., Nogales, F. (2014). A randomized granular tabu search heuristic for the split delivery vehicle routing problem. *Ann. Oper. Res.*, 222(1): 153-173. <http://dx.doi.org/10.1007/s10479-012-1282-3>
- [10] Sicilia, J.A., Quemada, C., Royo, B. (2016). An optimization algorithm for solving the rich vehicle routing problem based on variable neighborhood search and tabu search metaheuristics. *Journal of Computational & Applied Mathematics*, 291(C): 468-477. <https://doi.org/10.1016/j.cam.2015.03.050>
- [11] Glover, F. (1989). Tabu search. *ORSA Journal of Computing*, 1(3): 190-205. <http://dx.doi.org/10.1287/ijoc.1.3.190>
- [12] Toth, P., Vigo, D. (1993). The granular tabu search and its application to the vehicle routing problem. *INFORMS Journal on Computing*, 15(4): 333-346. <http://dx.doi.org/10.1287/ijoc.15.4.333.24890>
- [13] Aleman, R.E., Zhang, X.H., Hill, R.R. (2010). An adaptive memory algorithm for the split delivery vehicle routing problem. *Journal Heuristics*, 16(4): 441-473. <http://dx.doi.org/10.1007/s10732-008-9101-3>
- [14] Potvin, J.Y., Rousseau, J.M. (1995). An exchange heuristic for routing problems with time windows. *J. Oper. Res. Soc.*, 46(12): 1433-1446. <http://dx.doi.org/10.2307/2584063>
- [15] Glover, F. (1992). New ejection chain and alternating path methods for traveling salesman problems, *Computer Science and Operations Research*, 18: 491-507. <http://dx.doi.org/10.1016/B978-0-08-040806-4.50037-X>
- [16] Rego, C. (2001). Node-ejection chains for the vehicle routing problem: Sequential and parallel algorithm. *Parallel Computing*, 27: 201-222. [https://doi.org/10.1016/S0167-8191\(00\)00102-2](https://doi.org/10.1016/S0167-8191(00)00102-2)
- [17] Brandão, J. (2018). Iterated local search algorithm with ejection chains for the open vehicle routing problem with time windows. *Computers & Industrial Engineering*, 120: 146-159. <https://10.1016/j.cie.2018.04.032>
- [18] Euchi, J. (2017). The vehicle routing problem with private fleet and multiple common carriers: Solution with hybrid metaheuristic algorithm. *Vehicular Communications*, 9: 97-108. <https://10.1016/j.vehcom.2017.04.005>
- [19] Lin, S., Kernighan, B.W. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Oper. Res.*, 21(2): 498-516. <http://dx.doi.org/10.1287/opre.21.2.498>
- [20] Chen, S., Golden, B., Wasil, E. (2007). The split delivery vehicle routing problem: Applications, algorithms, test problems, and computational results. *Networks*, 49(4): 318-329. <http://dx.doi.org/10.1002/net.20181>