Multi-depot split delivery of batches

Críston Pereira de Souza
Federal University of Ceará
Quixadá, Ceará, Brazil
criston@ufc.br

Andréa Cynthia Santos
LITIS, ISEL, Université Le Havre Normandie
Le Havre, France

ABSTRACT

This study focuses on a vehicle routing problem variant involving multiple depots and split deliveries with discrete deliveries and small integer vehicle capacities and demands. This can be interpreted as the batching of items (k batches per vehicle), and the problem is referred to as k-MD-DSDVRP. An Integer Programming (IP) formulation is proposed, as well as cuts to reduce symmetries. In addition, we discuss the existence of an optimal solution without split cycles and establish bounds for the ratio between the optimal values of k-MD-DSDVRP and MD-DSDVRP. Furthermore, a reformulation of k-MD-DSDVRP as an MDCVRP is presented, followed by a solution approach through RCSP-based map decomposition. Experiments using instances of MDSDVRP and SDVRP from the literature were conducted to evaluate the proposed method, with an analysis of the impact of using batches and a comparison of bounds of k-MD-DSDVRP and MDSDVRP.

KEYWORDS

Vehicle routing, multi-depot, split delivery, logistics, column generation.

1 INTRODUCTION

In this study, we focus on a vehicle routing problem that involves multiple depots and split deliveries. Split deliveries refer to a situation where a customer's demand can be delivered by more than one vehicle. Hence, "delivery" in this context refers to providing a part of a customer's demand supplied by a vehicle to a customer. Compared to the MDSDVRP [11] (a standard definition in the literature for this problem), we assume that the vehicle capacity is a small integer k and that customer demands are at most $k+1$. These additional assumptions can be interpreted as grouping items into "batches." This variant is referred to here as k-MD-DSDVRP.

Motivations for investigating the k-MD-DSDVRP are listed below. Further discussion on the benefits of batching items can be found in [8].

- A MDSDVRP solution can have many tiny deliveries, which can be inconvenient for customers, causing interruptions to receive an insignificant portion of a demand.
- Fractional delivery can also be inconvenient to measure and control its amount on the fly. Using "batches" simplifies the process since supplies rely on integer values.
- From a logistics point of view, it is definitely interesting to simplify the preparation of deliveries through batches with many items.
- In a theoretical perspective, an algorithm for k-MD-DSDVRP can provide upper bounds for MD-DSDVRP. This is shown in Section 4.3.
- When k is small, it is possible to efficiently model the k-MD-DSDVRP as an MDCVRP [12] by creating k replicas of each client. Thus, allowing the use of successful algorithms for the MDCVRP available in the literature.

Contributions. This study brings the following contributions: (i) an IP formulation for the MD-DSDVRP with cuts to reduce symmetries; (ii) a discussion of the existence of optimal solution without split cycles and the relation between the number of splits and the number of routes; (iii) a transformation of instances and solutions between k-MD-DSDVRP and MD-DSDVRP, allowing to establish bounds for the ratio between the optimal values of these problems; (iv) a k-MD-DSDVRP model as an MDCVRP, with and a solution approach through RCSP-based map decomposition; (v) numerical experiments to evaluate the RCSP-based map decomposition and the inclusion of cuts to remove split cycles, using instances of MDSDVRP and SDVRP available in the literature.

The remaining of the manuscript is as follows. Section 2 presents an overview of the CVRP variants closely related to this study. Then, Section 3 defines the k-MD-DSDVRP and describes a proposed IP formulation. Section 4 discusses some properties of optimal k-MD-DSDVRP solutions and a comparison with optimal MD-DSDVRP solutions. Section 5 details an RCSP-based map decomposition approach for solving the k-MD-DSDVRP. Finally, in Section 6, numerical experiments and remarks are presented.

2 LITERATURE REVIEW

The goal of this Section is to provide entry points for articles defining closely related problems. The well-known *Capacitated Vehicle Routing Problem* (CVRP) is defined on an undirected and complete graph $G = (V, E)$ with a set of vertices $V = \{0, 1, \ldots, n\}$, where 0 represents the depot, the others vertices represent clients, and E is the set of edges. An unlimited number of homogeneous vehicles is available, each with a capacity $Q > 0$. A demand $0 < q_i \leq Q$ is associated with each customer i in $V \setminus \{0\}$. Each customer is visited exactly once, and their demand is fully supplied. Each edge $e \in E$ has an associated cost $c_e \geq 0$, satisfying the triangular inequality. The goal is to build a set of minimum-cost routes for the vehicles that meet all customers' demands and respect the vehicles' capacities.

The *Split Delivery Vehicle Routing Problem* (SDVRP) was formally defined by Dror and Trudeau [6, 7]. Unlike the CVRP, the SDVRP allows fractions of a customer's demand to be delivered by different vehicles, such that the sum of the fractions equals the total demand of the customer. In the problems mentioned in this article, we only consider the case in which customer demands are at most the vehicle's capacity, even though some works in the literature investigate the more general case without this constraint.

Gulczynski et al. [10] investigated the SDVRP variant with the additional constraint that each delivery to a customer has a size of at least a certain fraction of its demand and called this problem the *Split Delivery Vehicle Routing Problem with Minimum Delivery Amounts* (SDVRP-MDA). The motivation is to avoid too small deliveries, which generate the inconvenience of managing the receipt of a quantity with little impact on the total expected delivery.

The *Multi-Depot Split Delivery Vehicle Routing Problem* (MDSDVRP) is a generalization of the SDVRP that allows more than one depot, as defined in [11]. Therefore, the set of vertices V is partitioned into a set of depots $D \subset V$ and a set of customers $C = V \setminus D$. In this case, we must decide the routes and which depot each vehicle will depart from. It is also necessary to ensure that each vehicle returns to the depot from which it left. When we add the restriction that exactly one route passes through each customer, we have the so-called *Capacitated Multi-Depot Vehicle Routing Problem* (MDCVRP) [12].

Discrete Split Delivery Vehicle Routing Problem (DSDVRP), proposed by Nakao and Nagamochi [14], considers demands and delivery sizes restricted to positive integers. To clarify the difference, we call this variant of the *Multi-Depot Discrete Split Delivery Vehicle Routing Problem* (MD-DSDVRP).

The use of integer demands and small integer vehicle capacity was investigated by Archetti et al. [2] for the SDVRP. They showed that the problem can be solved in polynomial time using a matching algorithm when the vehicle capacity is 2 and is NP-hard when the vehicle capacity is 3. Furthermore, they showed that in the case of vehicle capacity 3, the optimal value of CVRP (without allowing split deliveries) is at most 3/2 of the optimal value of SDVRP.

3 k-MD-DSDVRP

This study investigates the *Multi-Depot Discrete Split Delivery Vehicle Routing with Small Vehicle Capacity k* (k-MD-DSDVRP). Given a small positive integer k, the k-MD-DSDVRP is a variant of MD-DSDVRP, where vehicle capacity and demands are at most k and $k+1$, respectively. As the deliveries are integers, the vehicle capacity and customer demands are also considered integers. We denote by k-DSDVRP the problem k-MD-DSDVRP with a single depot.

Vehicle capacity, demands, and delivery sizes are integers in k-MD-DSDVRP. Thus, k is a "batch" where several items are grouped into a single one. In this way, the k-MD-DSDVRP can be interpreted as a "discretization" of the MDSDVRP when the batches are constructed respecting the vehicles' capacities and the customers' demands.

3.1 IP formulation

An IP formulation for the MD-DSDVRP was proposed by [16], inspired by the formulation of the split deliveries found in [4], and using the so-called MTZ subtour elimination constraints of [13].

Here, a multiflow formulation is proposed, where subtour elimination constraints found in [5] are used. The problem is defined in a complete directed graph $G = (V, A)$ with a cost $c_{ij} \geq 0$ assigned to each edge $(i, j) \in A$. This cost function satisfies the triangular inequality. G has a set of vertices $V = D \cup C$ divided into two disjoint sets : D (depots) and C (customers). Each customer $i \in C$ has a non-negative integer demand q_i. Moreover, there is a set of homogeneous vehicles R with capacity $Q = k$ to deliver all customer demands. Note that the expression $\sum_{i \in C} \lceil q_i/k \rceil$ is an upper bound on the number of vehicles. The objective is to minimize the total travel cost while fulfilling all customer demands, respecting vehicle capacities, and returning each vehicle to its initial depot. Additionally, the solution must determine the depot of each vehicle.

Equations (1a)–(1i) present an IP formulation for MD-DSDVRP. The binary variable x^r_{ij} is equal to 1 if and only if vehicle $r \in R$ uses arc $(i, j) \in A$. On the other hand, the integer variable y^r_i represents the amount that vehicle $r \in R$ delivers to customer $i \in C$. The objective is to minimize the total costs of the selected arcs, as shown in Equation (1a). Equation (1b) establishes that all vehicles entering a node must also leave it (flow conservation). Equations (1c) and (1d) ensure that each vehicle's route passes through exactly one depot and that there is no cycle on this route containing only customers. The delivery of the entire demand for each customer is guaranteed by Equation (1e). Equation (1f) only allows a vehicle to deliver to a customer if its route passes through it. Finally, the vehicle capacities are guarantee by Equation (1g).

$$\min \sum_{(i,j)\in A} \sum_{r\in R} c_{ij} \cdot x^r_{ij} \tag{1a}$$

$$\sum_{i:(i,v)\in A} x^r_{ij} - \sum_{j:(v,j)\in A} x^r_{ji} = 0 \qquad \forall v \in V,\ \forall r \in R \tag{1b}$$

$$\sum_{d\in D} \sum_{j:(d,j)\in A} x^r_{dj} = 1 \qquad \forall r \in R \tag{1c}$$

$$\sum_{\substack{(i,j)\in A \\ i,j\in S}} x^r_{ij} \leq |S| - 1 \qquad \forall S \subseteq C, r \in R \tag{1d}$$

$$\sum_{r\in R} y^r_i = q_i \qquad \forall i \in C \tag{1e}$$

$$q_j \cdot \sum_{i:(i,j)\in A} x^r_{ij} \geq y^r_j \qquad \forall j \in C,\ \forall r \in R \tag{1f}$$

$$\sum_{i\in C} y^r_i \leq Q \qquad \forall r \in R \tag{1g}$$

$$y^r_i \in \{0, 1, 2, \ldots, q_i\} \qquad \forall i \in C, \forall r \in R \tag{1h}$$

$$x^r_{ij} \in \{0, 1\} \qquad \forall (i, j) \in A, \forall r \in R \tag{1i}$$

3.1.1 Reduction of symmetries. All permutations of the vehicles produce equivalent solutions, thus generating many symmetries. However, it is possible to reduce this problem by ensuring that the order of vehicle indexes corresponds to the order of depot indexes. More precisely, if vehicle r passes through depot d and vehicle r' passes through depot d', with $d < d'$, then $r < r'$. The constraint is provided in Equation (2).

$$\sum_{j\in C} x^{r'}_{dj} + \sum_{j\in C} x^r_{d'j} \leq 1, \qquad \begin{array}{r} \forall (d, d') \in D^2 : d < d', \\ \forall (r, r') \in R^2 : r < r'. \end{array} \tag{2}$$

4 OPTIMAL SOLUTIONS PROPERTIES

In this section, some properties of optimal k-MD-DSDVRP solutions are derived from existing studies. For the sake of clarity, *split cycle* is formally defined in Definition 1. First, we show that there is an optimal solution for the k-MD-DSDVRP without a *split cycle* and define the relation between the number of splits and the number of routes. Dror and Trudeau [7] proved this property for the problem with one depot and fractional deliveries, while Gouveia et al. [9] extended it for multiple depots. The difference from here is that the k-MD-DSDVRP deals with integer deliveries and multiple depots.

In the following, a transformation from MD-DSDVRP instances to k-MD- DSDVRP instances and transformations between solutions of these problems are presented, together with the optimal values correspondence obtained through these transformations. This allows the use of MD-DSDVRP instances from the literature by grouping the items into batches.

4.1 Existence of split cycles

Definition 1. Let R be a set of routes, and a support graph $H = (V, E)$ be an undirected graph where V is the set of customers and an edge $e = (u, v)$ belongs to E iff there exists a route r in R such that r passes through u and v. If C is a cycle of G, the nodes of C form a *split cycle* of R.

Property 2 (Dror and Trudeau, 1990 [7]). *If the edge costs satisfy the triangular inequality, then an optimal solution without split cycles exists for every feasible SDVRP instance.*

The authors in [7] proved this property for the SDVRP, and Gouveia et al. [9] have extended that for the MD-SDVRP. We claim that this property also applies to k-MD-DSDVRP since the exchange argument in the demonstration of [7] can also be extended for discrete deliveries. Note that the limit of k on the vehicle capacity is a characteristic of k-MD-DSDVRP inputs, not of its solutions.

Property 3. *If the edge costs satisfy the triangular inequality, then an optimal solution without split cycles exists for every feasible k-MD-DSDVRP instance.*

The existence of an optimal solution without split cycles does not apply to all CVRP variants. Indeed, Gulczynski et al. [10] showed that SDVRP-MDA instances exist for which all optimal solutions have a split cycle.

4.2 Number of splits and number of routes

The relation between the number of splits and the number of routes (employed vehicles) results from the existence of optimal solution without *split cycles*. This result can be applied to define cuts to remove solutions with split cycles, see Section 5.2.1.

Definition 4. The *number of deliveries n_i of customer i* is the number of routes that deliver a positive amount to i. The *number of splits for customer i* is defined as $n_i - 1$, and the *number of splits of a solution* is the sum of the number of splits among all customers.

Property 5 (Archetti et al., 2006 [3]). *If the edge costs satisfy the triangular inequality, then there is an optimal SDVRP solution where the number of splits is smaller than the number of routes.*

Indeed, Property 5 could be generalized to all sets of routes without a *split cycle*.

Property 6. *If the number of splits is at least the number of routes for some set of routes S, then S has a split cycle.*

Proof. Let $G(S \cup C, E)$ be a bipartite undirected graph where C is the set of customers with at least two deliveries in S, and we have an edge $(r, c) \in E$ if and only if the route $r \in S$ delivers to customer $c \in C$. By contrapositive, assume that S does not have a *split cycle*, and therefore G is acyclic. Thus, the number of edges $|E|$ is less than the number of vertices $|S| + |C|$. Since the number of splits m in S is equal to $|E| - |C|$, we conclude that $m = |E| - |C| < (|S| + |C|) - |C| = |S|$. □

4.3 MD-DSDVRP and k-MD-DSDVRP

In Section 4.3.1, a transformation from MD-DSDVRP instances to k-MD-DSDVRP instances are provided. Then, in Sections 4.3.2 and 4.3.3, the transformations between the resulting solutions are given. These transformations allow us to establish Theorem 7, which allows a correspondance between optimal values for the MD-DSDVRP and k-MD-DSDVRP.

4.3.1 From MD-DSDVRP instances to k-MD-DSDVRP instances. Let I be an instance of MD-DSDVRP with vehicle capacity Q and demand q_i for each customer i. For transforming I into a k-MD-DSDVRP instance I', items of I are grouped into batches of $B = \lfloor Q/k \rfloor$ items. Thus, the vehicle capacity of I' becomes $Q' = k$, and each customer's demand q_i becomes $q'_i = \lceil q_i/B \rceil$. Note that this transformation may produce some demands greater than the vehicle's capacity k (but not exceeding $k+1$), requiring at least two deliveries to these customers. For example, when $k = 3$, $Q = 4$ and $q_i = 4$, in instance I' the batch size is $B = 1$ and customer i has demand $q'_i = 4$, which is greater than the vehicle capacity $Q' = 3$.

4.3.2 From k-MD-DSDVRP solutions to MD-DSDVRP solutions. For every feasible solution S' for k-MD-DSDVRP, q'_i batches are delivered to each customer i. These q'_i batches can transport up to $B \cdot q'_i \geq q_i$ items, enough to satisfy the demand q_i of each customer i. As the sum of deliveries of each vehicle in S' is at most k, the total number of items transported by each vehicle is at most $k \cdot B \leq Q$. Therefore, the routes used in S' are feasible for MD-DSDVRP.

4.3.3 From MD-DSDVRP solutions to k-MD-DSDVRP solutions. The solution transformation preserves MD-DSDVRP routes, in spite of an increase on the number of vehicles. If y_{jr} denotes the amount delivered to customer j by route r of MD-DSDVRP, then this delivery can be made using $\lceil y_{jr}/B \rceil$ batches in k-MD-DSDVRP. Thus, the total number of batches required to make the deliveries of route r is $y'_r = \sum_{j \in C} \lceil y_{jr}/B \rceil$. As each vehicle in k-MD-DSDVRP delivers at most k batches, $\lceil y'_r/k \rceil$ vehicles in k-MD-DSDVRP are required to deliver the demands of each route r.

4.3.4 Comparing optimal values. When comparing the optimal value of two problems, say P_1 and P_2, we denote by $z(P)$ the optimal value of the problem P, and the comparison $z(P_1) \leq z(P_2)$ indicates that the optimal value of P_1 is less than or equal to the optimal value of P_2 whenever both problems have the same instance after the required adjustments.

Theorem 7. *Applying the instance and solution transformations described in sections 4.3.1, 4.3.2 and 4.3.3, we have that*

$$z(\text{MD-DSDVRP}) \leq z(k\text{-MD-DSDVRP}) \leq \left\lceil \frac{\min\{Q,n\}}{k} \right\rceil \cdot z(\text{MD-DSDVRP}), \tag{3}$$

where Q is the vehicle capacity of the MD-DSDVRP instance, and n is the number of customers. Besides, these bounds are tight.

Proof. The first inequality arises from the fact that the optimal k-MD-DSDVRP solution can be transformed into a feasible solution for the corresponding MD-DSDVRP instance without additional cost, as discussed in Section 4.3.2. The second inequality arises from a worst case for the transformation described in Section 4.3.3 when MD-DSDVRP demands are unitary, and each vehicle delivers to $\min\{Q, n\}$ customers. In this case, as the vehicles in k-MD-DSDVRP can deliver batches to a maximum of k vehicles, we need $\lceil \min\{Q, n\}/k \rceil$ vehicles for each route in the solution of MD-DSDVRP. According to the triangular inequality and the solution's optimality, the cost of delivering to a subset of customers on route r is not greater than the cost of r. Therefore, the solution for k-MD-DSDVRP can cost at most $\lceil \min\{Q, n\}/k \rceil$ times the cost of the optimal solution for MD-DSDVRP.

To show that the bounds are tight, consider an instance of MD-DSDVRP where all edges between a client and the single depot has cost one, and the cost of edges between customers is a tiny value ϵ. Therefore, the cost of any possible route converges to 2, remaining to count the number of routes to determine the cost of the solution. A tight example for the first inequality occurs when Q is a multiple of k, and all customers have demand Q. In this case, an optimal solution for both MD-DSDVRP and k-MD-DSDVRP consists of making exclusive deliveries to all customers; that is, each route delivers to only one customer. Note that when Q is a multiple of k, it is not possible to have a customer with demand greater than the vehicle's capacity in the transformation in Section 4.3.1. On the other hand, if the vehicle capacity in MD-DSDVRP is the number of customers and each customer has demand equal to one, then a single vehicle would be able to make all deliveries in MD-DSDVRP. However, it would require $\lceil n/k \rceil = \lceil Q/k \rceil = \lceil \min\{Q, n\}/k \rceil$ vehicles to carry out these deliveries in the k-MD-DSDVRP, from which we conclude that this is a tight example for the second inequality. □

5 RCSP-BASED MAP DECOMPOSITION

This section describes an RCSP-based map decomposition for k-MD-DSDVRP, that is, a column generation where the pricing problem is the *Resource Constrained Shortest Path* (RCSP), and sets of arcs in RCSP are mapped to integer variables of an IP model so that in the final solution each variable has value equals to the number of used mapped arcs in the RCSP solution.

The RCSP is defined over a directed graph $G(V, A)$ where V is the set of nodes, and A is the set of arcs. The set V contains two special nodes, s and d, representing the source and destination nodes, respectively. Each arc $a \in A$ has a cost and a consumption. The cost of a path p is the sum of the costs of its arcs, and the consumption of p is the sum of the consumption of its arcs. Each node $v \in V$ has a lower and upper bound for the available resource when the path enters the v. The cost of a set of paths is the sum of the costs of its paths. The objective of RCSP is to find a set with K paths from s to d of minimum cost such that each path starts with Q units of resource, and the available resource of each path passing through a node $v \in V$ respects the lower and upper bounds of v.

In the following, we define the input graphs for the pricing problem (RCSP) and then present a formulation for the k-MD-DSDVRP using the variables mapped on the arcs of these graphs. For clarity, the notation of Section 3.1 is adopted in the following sections.

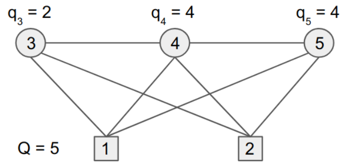
5.1 RCSP Graphs

Let $G^d = (V^d, A^d)$ be a directed graph for each depot $d \in D$. For each customer $i \in C$, we create a set $V_i^d = \{v_{i,1}^d, v_{i,2}^d, \ldots, v_{i,q_i}^d\}$ containing q_i replicas of customer i. Recall that q_i denotes the (integer) demand of customer i. Then, we set $V^d = D \cup \bigcup_{i \in C} V_i^d$, where D is the set of depots. The source and destination nodes are the depot $d \in D$.

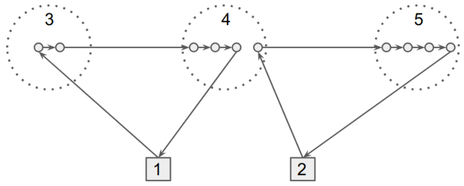
To reduce symmetries, two replicas of the same customer are connected by an arc if and only if they have consecutive indexes. That is, for each replica $v_{i,j}^d$, with $j < q_i$, the only arc going from $v_{i,j}^d$ to another replica of i is the arc $(v_{i,j}^d, v_{i,j+1}^d)$. All other possible arcs between nodes of V^d are included in A^d. The cost between replicas of the same client is zero, and each arc starting from a replica of client i to a replica of a different client j has a cost equal to $c_{i,j}$ (cost of the arc between customers i and j in the input graph). Similarly, the cost of an arc between a depot and a replica of client i is equal to the cost of the corresponding arc between this depot and client i in the input graph. Thus, when a route reaches a given customer, the number of consecutive replicas of this customer served by the route does not affect the route's cost.

A single resource is consumed on each route, representing the use of vehicle capacity. There is a unitary resource consumption whenever the route enters a node other than the depot, as each replica represents a unit of demand delivered to the customer. There is no resource consumption in the depot. We set zero and Q at each node as the accumulated resource consumption limits.

A constraint is added to the IP model to ensure that each client has exactly one delivery. This can be done through the map decomposition described in Section 5.2. Figure 1 provides an example of an k-MD-DSDVRP instance, and an RCSP optimal solution considering the additional constraint of exactly one route passing through each node. In Figure 1-(a), the k-MD-DSDVRP input graph has two depots represented by squares (nodes 1 and 2) and three customers represented by circles (nodes 3, 4, and 5). Each q_i represents the demand of customer i. Edge costs are the Euclidean distances between nodes, and the vehicle capacity Q is 5. An optimal solution with two routes is provided in Figure 1-(b). Assuming additional constraint of exactly one route passing through each node. Each customer i is represented by a dotted circle and is transformed into q_i replicas. Only arcs between consecutive replicas are allowed among replicas of the same customer, all with zero cost. The costs of the remaining arcs come from the input graph. The arcs returning to a depot have a consumption zero, and all the remaining arcs have a consumption one.



(a) k-MD-DSDVRP input graph.



(b) An optimal solution.

Figure 1: Example of k-MD-DSDVRP input graph and corresponding map decomposition optimal solution.

Complexity of transformation. Each client $i \in C$ has q_i replicas in each graph G^d, $d \in D$. Therefore, the resulting total number of nodes is $|D| \cdot \sum_{i \in C} q_i$. Among the replicas of the same customer, there are arcs only between consecutive replicas, but all possible arcs between replicas of different customers are created. Thus, the total number of arcs is $|D| \cdot (\sum_{i \in C}(q_i - 1) + \sum_{i \in C} q_i \sum_{j \in C:\, i \neq j} q_j)$. As $q_i \leq k + 1$ for all $i \in C$, we have that the total number of nodes is at most $(k + 1) \cdot |D| \cdot |C|$, and the total number of arcs is at most $|D| \cdot |C| \cdot (k + (k + 1)^2 \cdot (|C| - 1)) \in \Theta(k^2 \cdot |D| \cdot |C|^2)$.

5.2 Formulation and mapping of variables

Let C' be the set of replicas, and let $G'(C' \cup D, E')$ be an undirected graph where the vertices are all replicas and depots, and an edge $\{u, v\} \in E'$ if and only if $(u, v) \in A^d$ or $(v, u) \in A^d$ for any of the graphs $G^d(V^d, A^d)$, $d \in D$. In Equations (4a)–(4c), a formulation of k-MD-DSDVRP is given, where the decision variables x_e, $e \in E'$, are mapped into arcs of the RCSP graphs G^d, $d \in D$. That is, $x_{\{u,v\}}$ represents the number of routes passing through $\{u, v\}$ in the RCSP solution, taking into account all graphs G^d, $d \in D$. Thus, the formulation minimizes the sum of the costs of the edges used, subject to the condition that exactly one route passes through each replica. Let $\delta(v)$ be the set of edges in E' incident to the replica $v \in C'$, and thus, Constraint (4b) indicates that exactly two edges in $\delta(v)$ are used for each replica $v \in C'$. Note that this constraint affects all graphs G^d, $d \in D$, so that each replica $v \in C'$ is served by route in exactly one of these graphs. As k-MD-DSDVRP does not restrict the number of vehicles, the number of vehicles considered in the RCSP of each graph G^d, $d \in D$, may range from zero to the number of replicas.

$$\min \sum_{e \in E'} c_e \cdot x_e \tag{4a}$$

$$\sum_{e \in \delta(v)} x_e = 2 \qquad \forall v \in C' \tag{4b}$$

$$x_e \in \{0, 1, 2, \ldots\} \qquad \forall e \in E' \tag{4c}$$

Table 1: VRPSolver's non-default parameter values.

parameter	value
RCSPhardTimeThresholdInPricing	25
RCSPmaxNumOfLabelsInEnumeration	500000
RCSPmaxNumOfEnumeratedSolutions	5000000
RCSPrankOneCutsMemoryType	0
CutTailingOffThreshold	0.015
StrongBranchingPhaseOneCandidatesNumber	100
StrongBranchingPhaseOneTreeSizeEstimRatio	0.2
StrongBranchingPhaseTwoCandidatesNumber	3
StrongBranchingPhaseTwoTreeSizeEstimRatio	0.02
GlobalTimeLimit	3600

5.2.1 Valid inequality: split cycle removal. Equation (5) excludes solutions where the number of splits is greater than or equal to the number of routes. Then, by Property 6, only solutions with a split cycle are excluded. Besides, by Property 3, at least one optimal solution is preserved.

Let E_r be the set of all edges $\{u, v\} \in E'$ such that u and v are replicas of the same customer. The number of unused edges in E_r (i.e., with $x_e = 0$) provides the number of splits in the solution, as each unused edge implies an additional route delivering to the customer. Thus, the number of splits can be obtained through the expression $\sum_{e \in E_r}(1 - x_e)$. For example, Figure 1b depicts a solution with one split and one unused arc between the replicas of customer 4. If E_d is the set of arcs incident on any deposit, then the number of routes in the solution can be obtained by the expression $\sum_{e \in E_d} x_e/2$, as each route leaves and enters a single deposit just once.

$$2 \cdot \sum_{e \in E_r}(1 - x_e) \leq \sum_{e \in E_d} x_e - 2 \tag{5}$$

6 COMPUTATIONAL EXPERIMENTS

The experiments were conducted on an Ubuntu 16.04.7 LTS virtual machine with 8 Xeon 2.13GHz cores and 4MB cache, along with 24GB RAM. The VRPSolver 0.4.1 [15] was used for the implementation, with non-default parameters listed in Table 1, and IBM ILOG CPLEX 12.10 as the employed solver. Each replica node of graph G^d, $d \in D$ is defined as a vertex packing set and added a capacity cut separator with a limit of Q. We also assigned branching priority to the x variables.

The RCSP-based map decomposition method presented in Section 5 is tested using two instance sets available in the literature for MDSDVRP and SDVRP. The goal is to evaluate its effectiveness for $k \in \{3, 4, 5\}$. In addition, the impact of adding the cuts suggested in Section 5.2.1 to eliminate split cycles is analyzed.

Instances. Using the procedure described in Section 4.3.1, we adapted 42 instances from [9] to k-MD-DSDVRP and 89 instances from [1] to k-DSDVRP. None of the transformed instances had a demand that exceeded the vehicle's capacity.

6.1 Results

The results of the experiment are presented in Table 2, where the column "problem" is the instance type (k-DSDVRP for instances

Table 2: Summary of results.

problem	SCR	solved	gap	gap_MDSD	time
3-DSDVRP	N	97.7%	0.00%	54.6%	323
3-DSDVRP	Y	96.6%	0.00%	55.1%	288
3-MD-DSDVRP	N	81.0%	0.00%	16.2%	828
3-MD-DSDVRP	Y	81.0%	0.00%	16.2%	850
4-DSDVRP	N	86.4%	0.01%	43.3%	768
4-DSDVRP	Y	85.2%	0.00%	43.5%	771
4-MD-DSDVRP	N	54.8%	0.02%	14.8%	2187
4-MD-DSDVRP	Y	54.8%	0.04%	14.8%	2189
5-DSDVRP	N	72.1%	0.03%	39.3%	1351
5-DSDVRP	Y	70.9%	0.01%	39.4%	1345
5-MD-DSDVRP	N	32.5%	0.03%	12.0%	2667
5-MD-DSDVRP	Y	28.2%	0.07%	11.2%	2806

adapted from SDVRP and k-MD-DSDVRP for instances adapted from MDSDVRP), column "SCR" indicates the inclusion of split cycle removal constraints (with value 'Y' for yes and 'N' for no), column "solved" is the proportion of solved instances, column "gap" is the mean relative difference between the best lower and upper bounds, column "gap_MDSD" is the mean relative difference between the best upper bound and the best reported lower bound for the MDSDVRP, and column "time" is the mean execution time in seconds. A time limit of 1 hour was set for each instance. The following observations can be made:

- The number of solved instances drops quickly for k-MD-DSDVRP as k increases. However, it drops more slowly for k-DSDVRP. This suggests that the search for feasible solutions becomes more difficult as the number of depots increases.
- In all cases where a feasible solution was found, the gap was very small (below 0.1%). This indicates that the solutions produced are very close to being optimal.
- When the quality of the solutions was compared with the lower bounds provided by [9] for MDSDVRP, the k-MD-DSDVRP solution value was, on average, about 14.3% above. This indicates that integer deliveries and the batching of items into few batches per vehicle (3, 4 or 5) did not strongly affect the quality of the solution. However, this average difference was greater for k-DSDVRP, reaching 54.6% for $k = 3$, but this difference decreases whenever k increases.
- The execution time increases with the number of depots and the value of k, as they reflect the number of graphs and the number of replicas per customer, respectively.
- The inclusion of split cycle constraints worsened the average of all metrics, with a relative difference of around 30% for the gap, and less than 2% for the other metrics.

7 CONCLUDING REMARKS

This study investigates the k-MD-DSDVRP, which is convenient for daily logistics by organizing deliveries in batches. Properties are derived for this problem, and a formulation and an RCSP-based map decomposition are proposed. The experimental results show that the proposed map decomposition finds a near-optimal feasible solution for more than 80% of the instances for $k = 3$, but this percentage drops as k increases. It is also possible to conclude that the effect of grouping into batches and discrete deliveries is small (average gap of 14.3%) for instances with multiple deposits.

Several avenues of research are opened, such as investigating the impact of parameter k on the effectiveness of the dynamic programming method proposed in [8] for DSDVRP and the approach presented in [9] for MDSDVRP. Moreover, alternative solution transformations may provide tighter bounds than Theorem 7 for the ratio of $z(k\text{-MD-DSDVRP})$ and $z(\text{MDSDVRP})$. It is also worth investigating whether the demonstration of [2] for the NP-hardness of 3-SDVRP can be adapted to multi-depot and integer deliveries.

REFERENCES

[1] Claudia Archetti, Nicola Bianchessi, and Maria Grazia Speranza. 2011. A column generation approach for the split delivery vehicle routing problem. *Networks* 58, 4 (2011), 241–254.

[2] Claudia Archetti, Renata Mansini, and M. Grazia Speranza. 2005. Complexity and reducibility of the skip delivery problem. *Transportation Science* 39, 2 (2005), 182–187.

[3] Claudia Archetti, Martin W. P. Savelsbergh, and M. Grazia Speranza. 2006. Worst-Case Analysis for Split Delivery Vehicle Routing Problems. *Transportation Science* 40, 2 (May 2006), 226–234. https://doi.org/10.1287/trsc.1050.0117

[4] Claudia Archetti and Maria Grazia Speranza. 2008. The split delivery vehicle routing problem: A survey. *The vehicle routing problem: Latest advances and new challenges* (2008), 103–122.

[5] George Dantzig, Ray Fulkerson, and Selmer Johnson. 1954. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America* 2, 4 (1954), 393–410.

[6] Moshe Dror and Pierre Trudeau. 1989. Savings by Split Delivery Routing. *Transportation Science* 23, 2 (May 1989), 141–145. https://doi.org/10.1287/trsc.23.2.141

[7] Moshe Dror and Pierre Trudeau. 1990. Split delivery routing. *Naval Research Logistics (NRL)* 37, 3 (1990), 383–402.

[8] Weikang Fang, Zailin Guan, Peiyue Su, Dan Luo, Linshan Ding, and Lei Yue. 2022. Multi-Objective Material Logistics Planning with Discrete Split Deliveries Using a Hybrid NSGA-II Algorithm. *Mathematics* 10, 16 (2022), 2871.

[9] Luis Gouveia, Markus Leitner, and Mario Ruthmair. 2023. Multi-Depot Routing with Split Deliveries: Models and a Branch-and-Cut Algorithm. *Transportation Science* 57, 2 (Mar 2023), 512–530. https://doi.org/10.1287/trsc.2022.1179

[10] Damon Gulczynski, Bruce Golden, and Edward Wasil. 2010. The split delivery vehicle routing problem with minimum delivery amounts. *Transportation Research Part E: Logistics and Transportation Review* 46, 5 (Sep 2010), 612–626. https://doi.org/10.1016/j.tre.2009.12.007

[11] Damon Gulczynski, Bruce Golden, and Edward Wasil. 2011. The multi-depot split delivery vehicle routing problem: An integer programming-based heuristic, new test problems, and computational results. *Computers & Industrial Engineering* 61, 3 (2011), 794–804.

[12] Gilbert Laporte. 1984. Optimal solutions to capacitated multidepot vehicle routing problems. *Congressus Nemerantium* 4 (1984), 283–292.

[13] Clair E Miller, Albert W Tucker, and Richard A Zemlin. 1960. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)* 7, 4 (1960), 326–329.

[14] Yoshitaka Nakao and Hiroshi Nagamochi. 2007. A DP-based Heuristic Algorithm for the Discrete Split Delivery Vehicle Routing Problem. *Journal of Advanced Mechanical Design, Systems, and Manufacturing* 1, 2 (2007), 217–226. https://doi.org/10.1299/jamdsm.1.217

[15] Artur Pessoa, Ruslan Sadykov, Eduardo Uchoa, and François Vanderbeck. 2020. A generic exact solver for vehicle routing and related problems. *Mathematical Programming* 183 (2020), 483–523.

[16] Andréa Cynthia Santos. 2016. Shared transportation for macro-distribution in post-disaster relief. In *Proceedings of the 9th Triennial Symposium on Transportation Analysis (TRISTAN 2016)*.