

A Column Generation Based Heuristic for the Split Delivery Vehicle Routing Problem with Time Windows

Pedro Munari¹ and Martin Savelsbergh²

¹Federal University of Sao Carlos, Brasil, munari@dep.ufscar.br

²Georgia Institute of Technology, USA, martin.savelsbergh@isye.gatech.edu

Abstract

The vehicle routing problem with time windows (VRPTW) is one of the most studied variants of routing problems. We consider the Split Delivery VRPTW (SDVRPTW), an extension in which customers can be visited multiple times, if advantageous. While this additional flexibility can result in significant cost reductions, it also results in additional modeling and computational challenges. Indeed, the branch-and-price algorithms used to successfully solve VRPTW instances require substantial modifications before they can be used to solve SDVRPTW instances, and then often exhibit inferior performance when compared to branch-and-cut algorithms for solving SDVRPTW instances (whereas branch-and-price algorithms tend perform better than branch-and-cut algorithms for the VRPTW). We propose a new route-based formulation for the SDVRPTW that differs fundamentally from others presented in the literature. It is the first formulation in which the number of decision variables related to delivery quantities as well as the number of constraints are polynomial in the number of customers. We use this formulation as the basis for a column generation based heuristic that produces high-quality solutions for a wide range of benchmark instances with 50 and 100 customers and vehicle capacity equal to 50 and 100. It finds many new best-known solutions, and, for the first time, we report upper bounds for all 100-customer instances.

Keywords. Vehicle routing; Time windows; Split delivery; Column generation; Branch-and-cut.

1 Introduction

Time-constrained routing and scheduling has been a popular, productive, important, and ever-expanding area of research. For an early, but still relevant, introduction and overview, see Desrosiers et al. (1995). Within this class of problems, the vehicle routing problem with time windows (VRPTW) is the most widely studied variant and generalizes the (capacitated) vehicle routing problem by restricting the time during which customers can be served. For introductions and overviews, see Cordeau et al. (2002), Braysy and Gendreau (2005a,b) and Baldacci et al. (2012). Desrochers et al. (1992) were the first to propose an effective exact method for the VRPTW. The method uses the column generation (CG) technique, having a set partitioning formulation as the master problem, and a resource constrained shortest path problem (RCSP) as the subproblem. The method is so powerful that after almost 30 years it remains the basis of current state-of-the-art exact approaches for the VRPTW and related variants, e.g., Ropke and Cordeau (2009),

Desaulniers (2010), Desrosiers and Lübbecke (2011), Ropke (2012), Munari and Gondzio (2013, 2015), and Pecin et al. (2017).

The Split Delivery VRPTW (SDVRPTW) is a variant that, somewhat surprisingly, has received far less attention. It further generalizes the VRPTW by allowing vehicles to deliver a fraction of the total demand of a customer (Archetti et al. 2008, Archetti and Speranza 2012, Irnich et al. 2014). Hence, customers may be visited multiple times if beneficial, by different vehicles, a feature commonly observed in practice in city and humanitarian logistics (Cattaruzza et al. 2017, Al Theeb and Murray 2017). This can result in cost reductions, but introduces additional challenges to modeling and solving the problem (Archetti et al. 2006, Gendreau et al. 2006, Desaulniers 2010, Archetti et al. 2011, Ozbaygin et al. 2018, Hernández-Pérez and Salazar-González 2019, Bianchessi and Irnich 2019, Munari and Savelsbergh 2019). The main challenge is that the traditional mechanisms used to model the vehicle capacity constraints fail to guarantee a feasible solution when multiple visits are allowed. Indeed, when using the traditional modeling techniques, when two or more vehicles visit the same customer, load may be transferred from one vehicle to the other (which is not allowed in the SDVRPTW). There have been various efforts to overcome this difficulty, which are summarized below.

Bianchessi and Irnich (2019) propose a branch-and-cut (BC) method for the SDVRPTW based on a two-index commodity-flow formulation. It is a relaxed formulation, as there may be load transfers among vehicles at customers with multiple visits. Hence, to cut-off solutions with this behavior, the authors resort to the so-called *no-good* cuts in their BC method, similarly as proposed by Archetti et al. (2014) for the variant without time windows. Every time a candidate solution is found, its feasibility is checked by solving an auxiliary route-based formulation defined on a residual network. Additionally, the authors propose new classes of valid inequalities and resort to well-known ones to strengthen the linear relaxation of the problem and speed up their BC method. The results of computational experiments with benchmark instances show that their method is competitive with state-of-the-art methods, including branch-price-and-cut approaches. The authors report proven optimal solutions for 23 instances for the first time.

Munari and Savelsbergh (2019) introduces three novel compact formulations that, for the first time, fully model the SDVRPTW without using variables indexed by vehicles or number of visits. These formulations promote a significantly superior performance of general-purpose integer programming solvers with respect to other compact formulations. Additionally, the authors propose a BC method that starts with a relaxed formulation similar to the one proposed by Bianchessi and Irnich (2019) and then extends this formulation by gradually adding variables and constraints (originally defined in one of the introduced compact formulations) to guarantee that vehicles do not transfer loads at common customers. Their BC method also relies on well-known valid inequalities to strengthen linear relaxations. Computational experiments using benchmark instances indicate the superior performance of their simple, yet effective BC method with respect to other exact approaches for several split delivery variants, with and without time windows. The authors reported proven optimal solutions for 89 instances for the first time, in addition to present improved lower and upper bounds for many other instances.

Previous to these approaches, a few authors have focused on branch-and-price (BP) methods for the SDVRPTW. Gendreau et al. (2006) extend the set partitioning formulation commonly used for traditional VRP variants to further include decision variables related to the delivery quantities of each route to each customer. In addition to the possibly huge number of new variables, the resulting model has additional constraints for each feasible route in the problem, and hence the

number of constraints is also exponential in terms of the number of customers. The advantage of their approach is that the CG subproblem remains a resource constrained elementary shortest path problem (RCESPP) as in the VRPTW, solved by the labeling algorithm.

Desaulniers (2010) proposes an improved BP method that relies on a different formulation and a more sophisticated pricing problem. Instead of having separate variables for routes and delivery quantities, his formulation is based on extreme delivery patterns that combine routes and their corresponding delivery quantities. Hence, the CG subproblem has to determine not only a feasible route, but also the delivery quantities to each customer in this route. These quantities can assume only extreme values at the subproblem level and hence the actual delivery quantities are implicitly defined at the master problem level by means of convex combinations. Although the subproblem becomes more challenging, the advantage is that the number of constraints in the master problem is polynomial. Later on, the method was further enhanced by Archetti et al. (2011), who introduced valid inequalities to strengthen the formulation and embedded a Tabu Search heuristic to speed up the resolution of the subproblem.

The contributions of our research are threefold. First, we introduce a novel route-based formulation for the SDVRPTW with commodity-flow variables and a number of constraints that is polynomial in the number of customers. The proposed formulation can be used to model other split delivery variants, but such extensions are left for future research. Second, we propose a CG-based heuristic which starts by solving modified VRPTW instances and then uses the routes obtained to populate the proposed formulation, which is then solved by an effective BC method. Because we work with a limited set of columns in the formulation, there is no guarantee that an optimal solution is found. However, the approach quickly provides high-quality solutions. In fact, we are the first to report upper bounds for all 100-customer instances. Finally, we give new best known solutions for a number of benchmark instances with 50 and 100 customers. As is usual in most studies of the SDVRPTW, we assume that the size of the demands and the capacity of the vehicles are integer values and that the triangle inequality holds for costs and travel times.

The remainder of the paper is organized as follows. In Section 2, we define the notation and introduce the new route-based formulation with commodity-flow variables. In Section 3, we describe the proposed CG-based heuristic, while in Section 4, we present the results of computational experiments with this method, carried out with benchmark instances from the literature. Finally, we draw our conclusions and discuss ideas for future work in Section 5.

2 A new route-based formulation for the SDVRPTW

Consider a set of customers $\mathcal{C} = \{1, \dots, n\}$ with demand $d_i \in \mathbb{Z}_+$ for each $i \in \mathcal{C}$, a single depot, represented by 0 and $n + 1$, and a set of homogeneous vehicles $\mathcal{M} = \{1, \dots, m\}$ with capacity $Q \in \mathbb{Z}_+$. Each vehicle starts its route at 0 and returns to $n + 1$ after visiting the last customer in the route. We represent the problem on a graph $G = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N} = \mathcal{C} \cup \{0, n + 1\}$ is the set of nodes, and $\mathcal{A} = \{(i, j) : i, j \in \mathcal{C}, i \neq j\} \cup \{(0, i) : i \in \mathcal{C}\} \cup \{(i, n + 1) : i \in \mathcal{C}\}$ is the set of arcs. A cost c_{ij} is associated with each arc $(i, j) \in \mathcal{A}$; we assume the costs c_{ij} satisfy the triangle inequality. Given a node $i \in \mathcal{N}$, let $\delta_i^- = \{(j, i) \in \mathcal{A}\}$ denote the set of its incoming arcs, and $\delta_i^+ = \{(i, j) \in \mathcal{A}\}$ denote the set of its outgoing arcs.

When a vehicle $k \in \mathcal{M}$ visits a customer $i \in \mathcal{C}$ it can deliver any integer quantity less than or equal to $\min\{d_i, Q\}$ and hence customers may be visited by more than one vehicle. Additionally, any service at customer $i \in \mathcal{C}$ incurs a service time s_i and has to start within a time window $[e_i, \ell_i]$.

A vehicle can arrive at $i \in \mathcal{C}$ before e_i , but it has to wait until e_i to start the service at the customer. Vehicles depart from the depot no earlier than e_0 and have to return to the depot no later than ℓ_{n+1} . A travel time t_{ij} is associated with each arc $(i, j) \in \mathcal{A}$; we assume that the travel times t_{ij} satisfy the triangle inequality.

The SDVRPTW seeks to find a least-cost set of routes serving all customer demands. Since we assume integer demands and vehicle capacity, and that the triangle inequality holds for costs and travel times, there exists an optimal solution with the following properties: (1) routes are elementary, i.e., a route never visits a node more than once, (2) two routes have at most one node in common, (3) arcs between two customer nodes are traversed at most once, (4) if arc (i, j) is traversed, then arc (j, i) is not, and (5) delivery quantities are positive integers (Archetti et al. 2014, Desaulniers 2010, Dror and Trudeau 1989). These properties are commonly used to reduce the solution space and, hence, speed up solution approaches.

Let \mathcal{R} be a set of feasible routes that start at 0, visit a subset of customers exactly once and finish at $n+1$, while satisfying vehicle capacity and customer time windows. We further define the following subsets of \mathcal{R} : \mathcal{R}_i contains the routes that visit customer $i \in \mathcal{C}$, \mathcal{R}_{ij} contains the routes that traverse arc $(i, j) \in \mathcal{A}$; and \mathcal{R}^* contains the single-customer routes, i.e., $\{((0, i), (i, n+1)) \mid i \in \mathcal{C}\}$. Additionally, let c_r denote the total travel cost of route $r \in \mathcal{R}$, computed using the arc costs c_{ij} of the arcs traversed in the route. Consider the following decision variables:

$$\begin{aligned} \lambda_r &= \text{the number of times a route } r \in \mathcal{R} \text{ is used;} \\ f_{ij} &= \text{load on the vehicle that traverses arc } (i, j) \in \mathcal{A} \text{ right after servicing node } i; \\ z_i &= \text{number of visits to node } i. \end{aligned}$$

While route-based variables λ_r are widely used in BP approaches for routing problems, the use of load flow variables f_{ij} is less common, as the vehicle load is typically handled at the subproblem level. These load flow variables have been successfully used in BC approaches for the SDVRPTW and related variants (Archetti et al. 2014, Ozbaygin et al. 2018, Bianchessi and Irnich 2019, Munari and Savelsbergh 2019). In addition to strengthening the linear relaxation, they also enable an effective way of checking the feasibility of solutions to relaxed commodity flow formulations, i.e., whether load is transferred from one vehicle to another at a customer with multiple visits. Finally, although redundant, the variables z_i help to strengthen the formulation, as shown in Archetti et al. (2014).

Using the sets, parameters, and decision variables defined above, we introduce the following route-based commodity-flow formulation of the SDVRPTW:

$$\min \quad \sum_{r \in \mathcal{R}} c_r \lambda_r \quad (2.1)$$

$$\text{s.t.} \quad \sum_{r \in \mathcal{R}_i} \lambda_r = z_i, \quad i \in \mathcal{C} \cup \{0\}, \quad (2.2)$$

$$\sum_{(i,j) \in \delta_i^+} f_{ij} = \sum_{(j,i) \in \delta_i^-} f_{ji} + d_i, \quad i \in \mathcal{C}, \quad (2.3)$$

$$f_{ij} \leq Q \sum_{r \in \mathcal{R}_{ij}} \lambda_r, \quad (i, j) \in \mathcal{A}, \quad (2.4)$$

$$f_{ij} \geq f_{hi} - Q(1 - \sum_{r \in \mathcal{R}_{hi} \cap \mathcal{R}_{ij}} \lambda_r), \quad (h, i), (i, j) \in \mathcal{A} : i \in \mathcal{C}, \quad (2.5)$$

$$\sum_{i \in \mathcal{C}} z_i \leq n + m - 1, \quad (2.6)$$

$$\bar{m} \leq z_0 \leq m, \quad (2.7)$$

$$1 \leq z_i \leq m, \quad i \in \mathcal{C}, \quad (2.8)$$

$$f_{0i} = 0, \quad i \in \mathcal{N}, \quad (2.9)$$

$$f_{ij} \geq 0, \quad (i, j) \in \mathcal{A}, \quad (2.10)$$

$$z_i \in \mathbb{Z}_+, \quad i \in \mathcal{N}, \quad (2.11)$$

$$\lambda_r \in \mathbb{Z}_+, \quad r \in \mathcal{R}^*, \quad (2.12)$$

$$\lambda_r \in \{0, 1\}, \quad r \in \mathcal{R} \setminus \mathcal{R}^*. \quad (2.13)$$

The objective function (2.1) seeks to minimize the total travel costs of the selected routes. Constraints (2.2) relate variables λ_r and z_i , while Constraints (2.3) enforce vehicle load flow balance at each customer. Note that we take a pickup perspective when modeling vehicle load and hence vehicles depart empty from the depot, which is enforced by Constraints (2.9). The formulation still models a delivery setting, since the vehicle load flow can be interpreted as the residual capacity in the vehicle. Constraints (2.4) impose vehicle capacity and avoid a positive vehicle load on arc (i, j) if there is no vehicle traversing that arc.

Constraints (2.5) model the *regularity property* (Ozbaygin et al. 2018, Munari and Savelsbergh 2019) which avoids load transfers between vehicles visiting the same customer. In the adopted pickup perspective, a solution is said to satisfy the regularity property if, for each route, the flow on arc (h, i) traversed by the route is less than or equal to the flow in the subsequent arc (i, j) traversed by the route. Hence, we only need to ensure that $f_{hi} \leq f_{ij}$ every time these three nodes are visited consecutively in the same route. Note that due to properties (2) and (3) presented at the beginning of this section, we can assume that no two routes traverse the same sequence of arcs. Constraints (2.6)–(2.8) enforce lower and upper bounds on the number of customer visits, where $\bar{m} = \lceil \sum_{i \in \mathcal{C}} d_i / Q \rceil$ is the minimum number of vehicles needed to meet customer demands. Constraints (2.10)–(2.12) state the domains of the decision variables.

This formulation differs fundamentally from other route-based formulations in the literature. Indeed, the formulation by Gendreau et al. (2006) has decision variables that explicitly represent the amount delivered on each route to each customer. As the number of routes can be huge, there can also be a huge number of these variables in the formulation. Furthermore, each such variable also requires a constraint. Notice that in our formulation, the number of load flow variables and the number of constraints is polynomial in the number of customers. The primary difference with the formulation of Desaulniers (2010) is that our pricing problem is the same RCESPP as in the pricing problem for the VRPTW, as we do not define delivery patterns at the pricing problem level.

3 A column generation-based heuristic

Although BP approaches are currently the basis of the most efficient solution approaches for the VRPTW (as well as for several other vehicle routing variants), this is not the case for the SD-VRPTW. One reason is the additional challenges encountered in formulating the problem, i.e., accommodating multiple visits to customers and accounting for delivery quantities. Another reason, perhaps the more important one, is that the pricing problem becomes harder to solve. Indeed, when visiting a customer, a vehicle can deliver any quantity less than or equal to the customer's

demand and this may result in routes with many customers. Long routes with many visits adversely affect the performance of a labeling algorithm, as the number of feasible routes may increase dramatically, resulting in too many non-dominated states (Desaulniers et al. 2008). Therefore, the state-of-the-art for solving the SDVRPTW are BC approaches.

Route-based formulations may still offer benefits for the SDVRPTW, however, as we will demonstrate in the remainder of the paper. The benefits arise from the following observations. First, route-based formulations tend to have stronger linear relaxations than arc-based formulations and hence result in smaller search trees (Lübbecke and Desrosiers 2005, Munari et al. 2017). Second, using route-based formulations often leads to earlier discovery of high-quality solutions. Finally, the performance of a labeling algorithm can be managed when developing a heuristic by controlling the maximum length (or visits) of the routes. In what follows, we exploit these observations to develop a CG-based heuristic for the SDVRPTW.

The heuristic consists of two phases. In the first phase, we use a BP algorithm to solve a sequence of VRPTW instances imposing a relatively short time limit. Each VRPTW instance has the same data as the SDVRPTW instance, except that the vehicle capacity is increased by a positive amount; the amount increases with each instance in the sequence. The purpose of this phase is to generate a set of feasible routes that will be used to initialize formulation (2.1)–(2.13). Then, in the second phase – the main phase, this formulation is solved by a BC method, and no more columns are generated. In this way, we exploit the strengths of BP and BC: a BP algorithm is used to solve VRPTW instances to generate feasible routes, while a BC algorithm is used to solve the SDVRPTW instance applied to a formulation initialized with these routes.

Fig. 1 illustrates the approach. In our implementation, we limit the number of VRPTW instances solved in the first phase to three. Preliminary experiments have shown that this results in a good compromise between time and solution quality. In the first VRPTW instance, we use the original vehicle capacity $Q_1 = Q$. In the subsequent VRPTW instances, we choose $Q_2 > Q_1$ and $Q_3 > Q_2$. Note that the columns generated for a given VRPTW instance are feasible for the subsequent VRPTW instance and hence we re-use the columns. Furthermore, any feasible solution to the first VRPTW instance is also a feasible solution to the SDVRPTW and provides an initial incumbent for the BC algorithm (often one that is already of good quality).

4 Computational study

We carried out computational experiments with the proposed CG-based heuristic using benchmark instances widely used in the literature. The goal is to assess whether the proposed two-phase heuristic, combining BP and BC algorithms, can produce high-quality solutions in a short amount of time (compared to what is possible with existing state-of-the-art algorithms).

Similar to what has been done in previous papers on the SDVRPTW, we use the benchmark instances of Solomon (1987), originally proposed for the VRPTW. We consider instances from the six available classes: C1 and C2 (clustered customers), R1 and R2 (randomly positioned customers), RC1 and RC2 (a mix of clustered and randomly positioned customers). Classes C2, RC1 and RC2 have 8 instances each, while classes C1, R1 and R2 have 9, 12 and 11 instances each, respectively. We replicate each instance by setting the number of customers to 50 and 100 and vehicle capacities to 50 and 100, resulting in 224 instances: 56 for each combination of $n \in \{50, 100\}$ and $Q \in \{50, 100\}$. We assume Euclidean distances truncated after the first decimal place. Since it does not guarantee that the triangle inequality is satisfied, we modify the distances using the lengths of the shortest paths

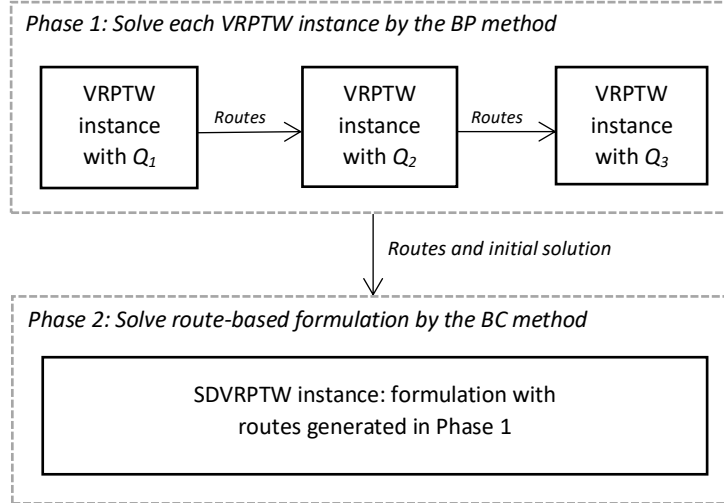


Figure 1: Main steps of the proposed CG-based heuristic.

obtained by the Floyd-Warshall algorithm. Travel times and costs are set equal to the resulting distances.

The CG-based heuristic was coded in C++ and relies on the BP framework of Munari and Gondzio (2013) designed for the VRPTW, and the BC algorithm of Munari and Savelsbergh (2019) designed for the SDVRPTW. The BP algorithm used to solve VRPTW instance is based on the primal-dual column generation method and relies on interior point solutions to stabilize column and cut generation (Gondzio et al. 2013, 2016). The BP algorithm is called three times in the first phase of the heuristic, and in the second and third calls, we initialize the master problem with the columns obtained in previous call.

The BC algorithm uses the Concert Library of the IBM CPLEX Optimization Studio v.12.9 to manage the search tree and the callback functions that separate and add valid inequalities. The method relies on the CVRPSep package to generate capacity cuts, while the separation of regularity property cuts uses the `MatchArcs` procedure (Archetti et al. 2014). Differently from Munari and Savelsbergh (2019), we define these cuts using λ_r variables, and hence no additional arc-based variables are needed when adding these cuts. Another difference is that we do not separate feasibility/optimality and connectivity cuts, as they did not appear to offer significant benefits in preliminary experiments with the CG-based heuristic.

All experiments were run on a Linux PC with Intel Xeon Skylake 2.1 GHz processor (16 cores) and 128 GB of memory. We impose a time limit of 120 seconds for the solution of a VRPTW instance by the BP algorithm. Furthermore, we set $Q_1 = Q$, $Q_2 = Q + 10$, and $Q_3 = Q + 20$. We impose a time limit of 3600 seconds for the solution of the SDVRPTW instance by the BC algorithm and, hence, the total time of the CG-based heuristic can reach up to 3960 seconds. This gives the CG-based heuristic a small advantage (360 seconds to produce an initial feasible solution), but it still provides a meaningful comparison in terms of the performance of the BC component.

Table 1 summarizes the results of the experiments for instances with 50 customers, for vehicle capacities $Q = 50$ and $Q = 100$. We compare the performance of the CG-based heuristic to the state-of-the-art BC algorithms of Bianchessi and Irnich (2019) and Munari and Savelsbergh (2019)

using their published results obtained with a time limit of 3600 seconds. For each method, we report the best lower and upper bounds obtained (columns LB and UB), the gap (column Gap), i.e., the relative difference between the lower and upper bounds as a percentage, and the total running time in seconds (column T_{total}). For the CG-based heuristic, we can only present an upper bound, but also report the time at which this upper bound was found (column T_{best}), the relative deviations between the upper bound obtained by the CG-based heuristic and the upper bounds obtained by the other algorithms (columns ΔUB_{BI} and ΔUB_{MS}), and the number of times that the heuristic found a better upper bound than the other algorithms (columns $\#_{BI}$ and $\#_{MS}$). For each value of Q and each instance class, we present the average (arithmetic mean) values over all instances in the class. Detailed results for each instance are presented in Tables 3 and 4 in the appendix.

Table 1: Summary of the results obtained by state-of-the-art BC methods and the proposed CG-based heuristic on the SDVRPTW benchmark instances with $n = 50$.

Q	$Class$	Bianchessi and Irnich (2019)				Munari and Savelsbergh (2019)				CG-based heuristic						
		LB	UB	Gap	T_{total}	LB	UB	Gap	T_{total}	UB	T_{total}	T_{best}	ΔUB_{BI}	ΔUB_{MS}	$\#_{BI}$	$\#_{MS}$
50	C1	1013.4	1013.4	0.00%	119.3	1013.4	1013.4	0.00%	122.4	1016.0	363.2	353.9	0.26%	0.26%	0	0
	R1	1074.2	1099.0	2.38%	2708.8	1079.2	1095.6	1.49%	2624.0	1096.5	1326.6	739.5	-0.28%	0.05%	3	5
	RC1	1699.6	1699.6	0.00%	3.0	1699.6	1699.6	0.00%	651.2	1700.0	152.2	137.1	0.03%	0.03%	0	0
	C2	1149.1	1160.5	1.00%	3118.4	1157.2	1157.2	0.00%	832.0	1161.1	358.4	294.7	0.05%	0.33%	2	0
	R2	1054.9	1080.2	2.45%	2979.4	1063.2	1076.1	1.20%	2757.1	1075.6	1764.3	967.4	-0.45%	-0.06%	5	4
	RC2	1699.6	1699.6	0.00%	3.3	1699.6	1699.6	0.00%	741.6	1700.1	157.0	138.0	0.03%	0.03%	0	0
100	C1	584.4	584.4	0.00%	308.9	583.7	588.3	0.76%	925.1	586.9	400.8	390.9	0.43%	-0.25%	0	2
	R1	768.9	814.6	6.33%	3000.9	774.8	826.4	6.41%	3008.1	806.7	396.0	352.0	-1.08%	-2.55%	8	10
	RC1	940.5	940.5	0.00%	292.0	940.5	940.5	0.00%	150.8	940.9	326.1	322.9	0.04%	0.04%	0	0
	C2	686.1	686.1	0.00%	981.1	686.1	686.1	0.00%	280.7	686.5	362.6	332.0	0.06%	0.06%	0	0
	R2	719.8	739.3	2.78%	3251.5	721.5	756.1	4.57%	3066.6	738.6	540.8	477.4	-0.08%	-2.38%	4	8
	RC2	934.8	934.8	0.00%	12.9	934.8	934.8	0.00%	65.3	936.3	358.7	355.5	0.15%	0.15%	0	0

The results in Table 1 show that the CG-based heuristic performs well and quickly provides good-quality solutions. On average, the upper bounds are never more than 0.43% higher than the upper bounds provided by the BC algorithms of Bianchessi and Irnich (2019) and Munari and Savelsbergh (2019). We observe too that the CG-based heuristic has good performance on the instances with random customer locations (classes R1 and R2) for both values of the vehicle capacity, while the BC algorithms show the largest gaps for these instances because their valid inequalities are less effective for instances with random customer locations. The CG-based heuristic produced a solution with a value better than the value of at least one of the solutions produced by the BC algorithms for 34 instances, and obtained new best known solutions for 17 instances.

Next, we consider the instances with 100 customers. As no results for these instances have been published, we solved them using the BC algorithm of Munari and Savelsbergh (2019), using the same settings that were used to solve the instances with 50 customers and imposing a time limit of 3600 seconds. A summary of the results can be found in Table 2. When an approach reaches the time limit of 3600 seconds for all instances in a class, we put ‘tl’ in column T_{total} . Detailed results can be found in Tables 5 and 6 in the appendix. We see that the relative performance of the CG-based heuristic is even better. In addition to the statistics reported previously, we report the gap between the upper bound found by the CG-based heuristic and the lower bound obtained by the BC algorithm (Gap_{MS}). Furthermore, we also present T_{best} for the BC algorithm of Munari

and Savelsbergh (2019), as it reached the time limit for almost all instances.

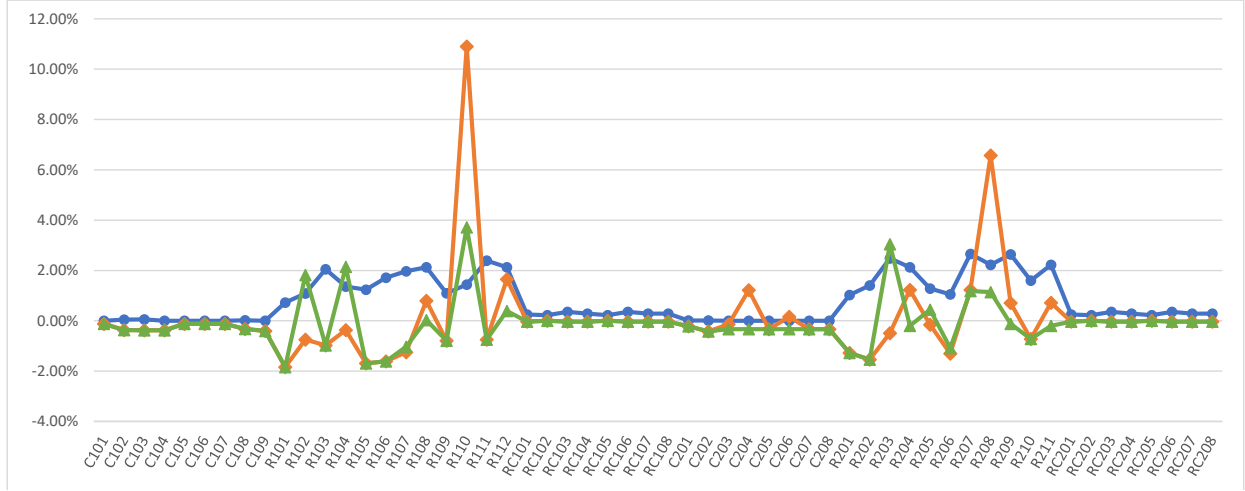
Table 2: Summary of the results obtained by a state-of-the-art BC algorithm and the proposed CG-based heuristic on the SDVRPTW benchmark instances with $n = 100$.

Q	$Class$	Munari and Savelsbergh (2019)					CG-based heuristic					
		LB	UB	Gap	T_{total}	T_{best}	UB	T_{total}	T_{best}	ΔUB_{MS}	Gap_{MS}	$\#_{MS}$
50	C1	2445.7	2493.0	1.88%	tl	2204.2	2463.4	710.8	584.5	-1.20%	0.72%	6
	R1	1846.6	2081.8	11.26%	tl	610.4	1947.9	3953.0	1310.9	-6.92%	5.21%	12
	RC1	2704.8	2988.7	9.43%	tl	1377.6	2783.1	tl	2313.8	-7.40%	2.81%	8
	C2	2522.9	2707.1	6.66%	tl	1339.5	2547.6	3301.9	1121.7	-6.26%	0.97%	8
	R2	1823.3	2079.3	12.28%	tl	1020.5	1928.0	tl	1488.7	-7.86%	5.43%	11
	RC2	2698.9	3109.4	12.93%	tl	783.4	2775.2	tl	1807.4	-12.08%	2.75%	8
100	C1	1377.5	1423.9	3.13%	2680.2	928.4	1389.0	480.6	449.8	-2.52%	0.83%	4
	R1	1214.2	1436.3	15.69%	3305.6	451.6	1321.9	3190.6	1730.2	-8.93%	8.36%	11
	RC1	1574.1	1823.1	13.58%	tl	842.3	1675.1	3412.1	1955.0	-8.95%	6.01%	8
	C2	1430.9	1595.8	10.31%	tl	764.1	1462.3	1372.1	772.5	-9.14%	2.14%	8
	R2	1140.1	1359.2	16.12%	tl	673.8	1252.8	tl	2086.6	-8.51%	9.03%	11
	RC2	1557.2	1834.8	15.02%	tl	945.6	1648.4	tl	1792.6	-11.35%	5.51%	8

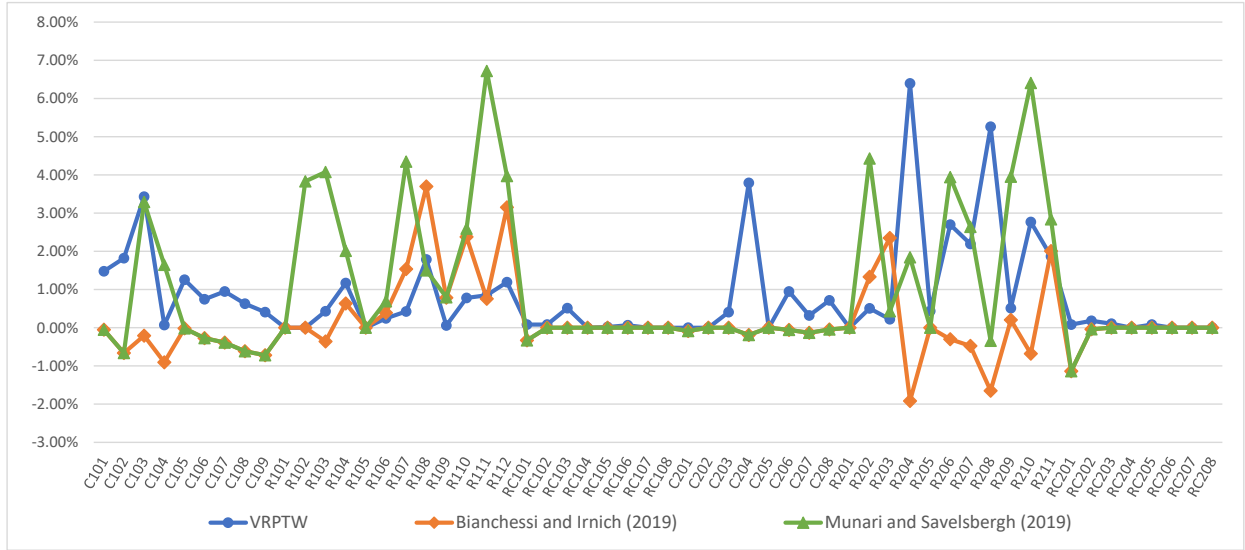
We see that the CG-based heuristic found better upper bounds than the BC algorithm for at least some instances in every instance class, and significantly better upper bounds for many instances. The smallest average improvement is seen for class C1 (1.20%) and the largest average improvement is seen for class RC2 (12.08%), both for $Q = 50$. As a consequence, the gaps for these upper bounds are considerably smaller than the gaps for the upper bounds produced by the BC algorithm. Furthermore, the CG-based heuristic found new best known solutions for 103 instances with 100 customers, demonstrating that route-based approaches can indeed be effective for split delivery vehicle routing variants, especially when used in combination with BC algorithms.

The plots in Figures 2 and 3 show the relative gaps of the cost of the solutions produced by the various algorithms with respect to the cost of the solution produced by the CG-based heuristic. More specifically, for each algorithm, we plot $(\overline{UB} - UB_H)/UB_H$, where \overline{UB} is the cost of the solution produced by the algorithm and UB_H is the cost of the solution produced by the CG-based heuristic. For comparison, we also present the gaps of the solution to the first VRPTW instance that is solved as part of the CG-based heuristic (as it also provides a feasible solution for the SDVRPTW). Figure 2 presents plots for the instances with 50 customers and 3 presents plots for the instances with 100 customers.

Interestingly, we observe that the solution to the VRPTW already provides quite a good solution to the SDVRPTW as well, although better solutions are found by the CG-based heuristic. The latter indicates that the columns generated with vehicle capacities Q_2 and Q_3 are useful and that some of these are selected by the CG-based heuristic. This is more evident for instances with a larger number of customers ($n = 100$) and a larger vehicle capacity ($Q = 100$). Finally, we observe that the cost of the solution produced by the other algorithms can be more than 5% higher than the cost of the solution produced by the CG-based heuristic, while the cost of the solution produced by the other algorithms is never more than 2% lower than the cost of the solution produced by the CG-based heuristic.



(a) $n = 50$, $Q = 50$.



(b) $n = 50$, $Q = 100$.

Figure 2: Relative gap of the cost of the solution to the first VRPTW instance solved in Phase 1 (VRPTW) and the cost of the solutions produced by two state-of-the-art BC methods to the cost of the solution produced by the CG-based heuristic for instances with 50 customers.

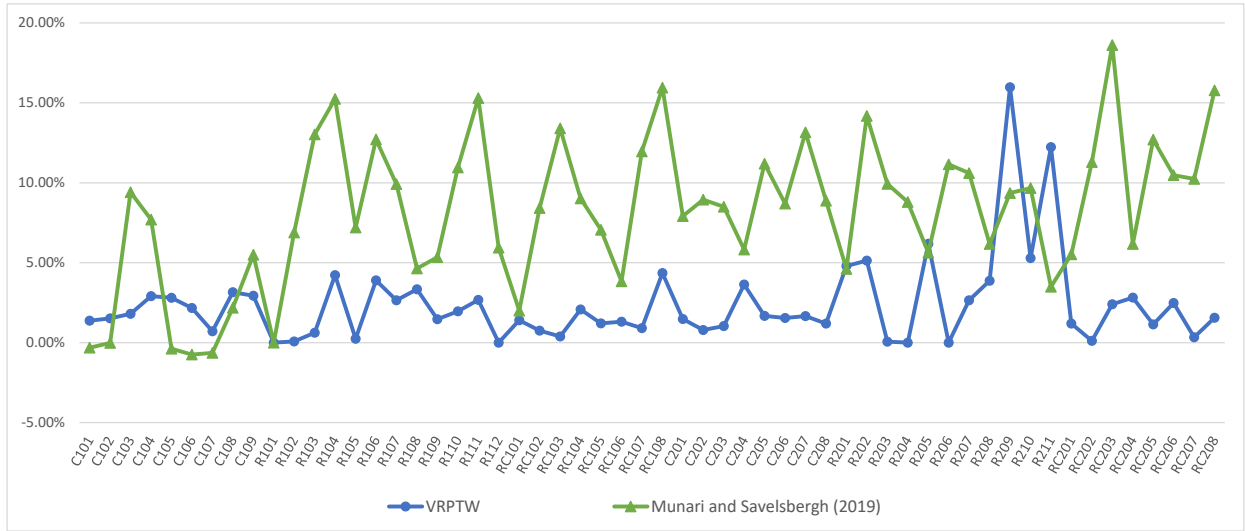
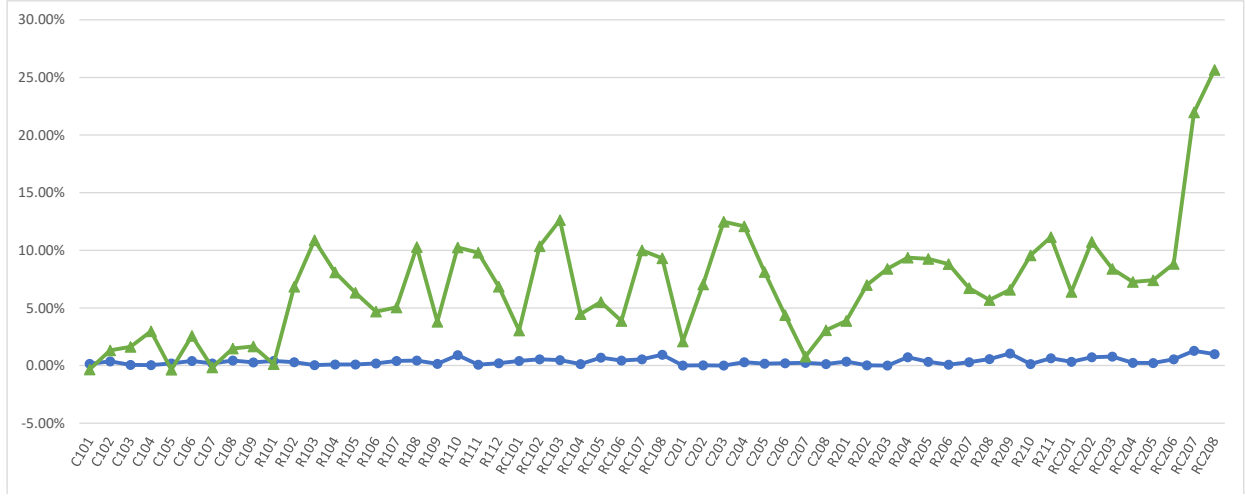


Figure 3: Relative gap of the cost of the solution to the first VRPTW instance solved in Phase 1 (VRPTW) and the cost of the solution produced by a state-of-the-art BC method to the cost of the solution produced by the CG-based heuristic for instances with 100 customers.

5 Conclusions

We have introduced a new route-based formulation for the SDVRPTW and proposed a column generation based heuristic that exploits it. The formulation is the first in which the number of decision variables related to delivery quantities as well as the number of constraints are polynomial in the number of customers. The heuristic is a two-phase approach that exploits the strength of branch-and-price algorithms for the VRPTW and the strength of branch-and-cut algorithms for the SDVRPTW. A computational study demonstrates that the heuristic produces high-quality solutions in a reasonable amount of time. New best solutions were found for many standard benchmark instances.

These encouraging results suggest that a natural future research topic is the design, implementation, and testing of a branch-price-and-cut method for solving the proposed route-based formulation. The idea of gradually relaxing the vehicle capacity, as incorporated in the heuristic, may be useful in this context as well, as it avoids having to battle the increasing difficulty of solving the pricing problem. Another interesting research direction is to verify the performance of the proposed approaches on other split delivery variants, particularly on split-pickup split-delivery vehicle routing problems (Casazza et al. 2018, 2019) as they represent challenging variants that can potentially benefit from these approaches.

Acknowledgments

This work was supported by the Sao Paulo Research Foundation (FAPESP) [grant numbers 18/23555-1, 16/01860-1 and 13/07375-0] and the National Council for Scientific and Technological Development (CNPq) [grant number 304601/2017-9].

References

- Al Theeb, N. and Murray, C. (2017). Vehicle routing and resource distribution in postdisaster humanitarian relief operations. *International Transactions in Operational Research*, 24(6):1253–1284.
- Archetti, C., Bianchessi, N., and Speranza, M. G. (2014). Branch-and-cut algorithms for the split delivery vehicle routing problem. *European Journal of Operational Research*, 238(3):685–698.
- Archetti, C., Bouchard, M., and Desaulniers, G. (2011). Enhanced branch and price and cut for vehicle routing with split deliveries and time windows. *Transportation Science*, 45(3):285–298.
- Archetti, C., Savelsbergh, M., and Speranza, M. G. (2006). Worst-case analysis for split delivery vehicle routing problems. *Transportation Science*, 40(2):226–234.
- Archetti, C., Savelsbergh, M., and Speranza, M. G. (2008). To split or not to split: That is the question. *Transportation Research Part E*, 44:114–123.
- Archetti, C. and Speranza, M. G. (2012). Vehicle routing problems with split deliveries. *International Transactions in Operational Research*, 19(1-2):3–22.
- Baldacci, R., Mingozzi, A., and Roberti, R. (2012). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research*, 218(1):1–6.
- Bianchessi, N. and Irnich, S. (2019). Branch-and-cut for the split delivery vehicle routing problem with time windows. *Transportation Science*, 53(2):442–462.
- Braysy, O. and Gendreau, M. (2005a). Vehicle routing problem with time windows, Part I: Route construction and local search algorithms. *Transportation Science*, 39(1):104–118.

- Braysy, O. and Gendreau, M. (2005b). Vehicle routing problem with time windows, Part II: Metaheuristics. *Transportation Science*, 39(1):119–139.
- Casazza, M., Ceselli, A., Chemla, D., Meunier, F., and Wolfler Calvo, R. (2018). The multiple vehicle balancing problem. *Networks*, 72(3):337–357.
- Casazza, M., Ceselli, A., and Wolfler Calvo, R. (2019). A route decomposition approach for the single commodity split pickup and split delivery vehicle routing problem. *European Journal of Operational Research*. DOI: 10.1016/j.ejor.2019.07.015.
- Cattaruzza, D., Absi, N., Feillet, D., and González-Feliu, J. (2017). Vehicle routing problems for city logistics. *EURO Journal on Transportation and Logistics*, 6(1):51–79.
- Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M. M., and Soumis, F. (2002). VRP with time windows. In Toth, P. and Vigo, D., editors, *The Vehicle Routing Problem*, pages 157–193. SIAM.
- Desaulniers, G. (2010). Branch-and-price-and-cut for the split-delivery vehicle routing problem with time windows. *Operations Research*, 58(1):179–192.
- Desaulniers, G., Lessard, F., and Hadjar, A. (2008). Tabu search, partial elementarity, and generalized k-path inequalities for the vehicle routing problem with time windows. *Transportation Science*, 42(3):387–404.
- Desrochers, M., Desrosiers, J., and Solomon, M. (1992). A new optimization algorithm for the vehicle routing problem with time windows. *Operations Research*, 40(2):342–354.
- Desrosiers, J., Dumas, Y., Solomon, M. M., and Soumis, F. (1995). Chapter 2: Time constrained routing and scheduling. In *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, pages 35 – 139. Elsevier.
- Desrosiers, J. and Lübbecke, M. E. (2011). Branch-price-and-cut algorithms. In Cochran, J. J., Cox, L. A., Keskinocak, P., Kharoufeh, J. P., and Smith, J. C., editors, *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc.
- Dror, M. and Trudeau, P. (1989). Savings by split delivery routing. *Transportation Science*, 23(2):141–145.
- Gendreau, M., Dejax, P., Feillet, D., and Gueguen, C. (2006). Vehicle routing with time windows and split deliveries. *Technical Report - Laboratoire Informatique d’Avignon*, 851.
- Gondzio, J., Gonzalez-Brevis, P., and Munari, P. (2013). New developments in the primal-dual column generation technique. *European Journal of Operational Research*, 224(1):41–51.
- Gondzio, J., González-Brevis, P., and Munari, P. (2016). Large-scale optimization with the primal-dual column generation method. *Mathematical Programming Computation*, 8(1):47–82.
- Hernández-Pérez, H. and Salazar-González, J.-J. (2019). Optimal solutions for the vehicle routing problem with split demands. In *International Conference on Computational Logistics*, pages 189–203. Springer.
- Irnich, S., Schneider, M., and Vigo, D. (2014). Chapter 9: Four variants of the vehicle routing problem. In *Vehicle Routing: Problems, Methods, and Applications, Second Edition*, pages 241–271. SIAM.
- Lübbecke, M. E. and Desrosiers, J. (2005). Selected topics in column generation. *Operations Research*, 53(6):1007–1023.
- Munari, P., Dollevoet, T., and Spliet, R. (2017). A generalized formulation for vehicle routing problems. *arXiv preprint arXiv:1606.01935*.
- Munari, P. and Gondzio, J. (2013). Using the primal-dual interior point algorithm within the branch-price-and-cut method. *Computers & Operations Research*, 40(8):2026 – 2036.
- Munari, P. and Gondzio, J. (2015). Column generation and branch-and-price with interior point methods. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, 3(1).
- Munari, P. and Savelsbergh, M. (2019). Compact formulations for split delivery routing problems. Technical report, MS01/2019, Operations Research Group, Production Engineering Department, Federal University of Sao Carlos.
- Ozbaygin, G., Karasan, O., and Yaman, H. (2018). New exact solution approaches for the split delivery vehicle routing problem. *EURO Journal on Computational Optimization*, 6(1):85–115.

- Pecin, D., Contardo, C., Desaulniers, G., and Uchoa, E. (2017). New enhancements for the exact solution of the vehicle routing problem with time windows. *INFORMS Journal on Computing*, 29(3):489–502.
- Ropke, S. (2012). Branching decisions in branch-and-price-and-cut algorithms for vehicle routing problems. In *International Workshop on Column Generation, Bromont, Canada*.
- Ropke, S. and Cordeau, J.-F. (2009). Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Science*, 43(3):267–286.
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265.

Appendix

Detailed results of the computational experiments reported in Section 4. The instances and meaning of the column headers are the same as in the body of the paper. We use ‘tl’ in column T_{total} to indicate that the time limit was reached by the corresponding solution approach.

Table 3: Results of the state-of-the-art BC methods and the CG-based heuristic on the SDVRPTW instances with $n = 50$ and $Q = 50$.

<i>Instance</i>	Bianchessi and Irnich (2019)				Munari and Savelsbergh (2019)				CG-based heuristic				
	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>UB</i>	<i>T_{total}</i>	<i>T_{best}</i>	ΔUB_{BI}	ΔUB_{MS}
C101	1015.80	1015.80		31.00	1015.80	1015.80		55.99	1017.00	298.13	292.21	0.12%	0.12%
C102	1013.00	1013.00		96.00	1013.00	1013.00		226.06	1016.70	371.50	363.64	0.36%	0.36%
C103	1012.30	1012.30		456.00	1012.30	1012.30		298.07	1016.10	381.22	361.94	0.37%	0.37%
C104	1010.20	1010.20		229.00	1010.20	1010.20		88.36	1014.00	370.26	360.82	0.37%	0.37%
C105	1015.80	1015.80		15.00	1015.80	1015.80		60.69	1017.00	368.74	360.63	0.12%	0.12%
C106	1015.80	1015.80		42.00	1015.80	1015.80		57.44	1017.00	368.62	360.86	0.12%	0.12%
C107	1015.80	1015.80		44.00	1015.80	1015.80		103.09	1017.00	369.42	360.48	0.12%	0.12%
C108	1011.80	1011.80		88.00	1011.80	1011.80		135.73	1015.10	369.97	363.65	0.33%	0.33%
C109	1010.10	1010.10		73.00	1010.10	1010.10		76.60	1014.20	371.16	360.71	0.40%	0.40%
R101	1190.70	1190.70		8.00	1190.70	1190.70		169.11	1213.00	37.27	31.89	1.84%	1.84%
R102	1108.87	1114.20	0.48%	tl	1101.76	1143.00	3.61%	tl	1122.60	67.79	62.91	0.75%	-1.82%
R103	1067.48	1081.50	1.31%	tl	1068.91	1081.50	1.16%	tl	1092.20	493.09	434.44	0.98%	0.98%
R104	1034.88	1059.70	2.40%	tl	1044.29	1086.40	3.88%	tl	1063.60	3023.39	648.23	0.37%	-2.14%
R105	1132.30	1132.30		694.00	1132.30	1132.30		2831.49	1151.80	1690.26	881.33	1.69%	1.69%
R106	1080.20	1080.20		226.00	1080.20	1080.20		229.85	1097.90	683.74	644.68	1.61%	1.61%
R107	1053.16	1063.90	1.02%	tl	1058.35	1066.30	0.75%	tl	1077.40	874.84	278.06	1.25%	1.03%
R108	1029.12	1064.00	3.39%	tl	1043.89	1055.90	1.14%	tl	1055.60	1357.03	1139.00	-0.80%	-0.03%
R109	1074.06	1081.80	0.72%	tl	1081.80	1081.80		2239.87	1090.40	504.34	457.40	0.79%	0.79%
R110	1034.78	1184.60	14.48%	tl	1045.62	1107.80	5.61%	tl	1068.10	2675.87	1913.03	-10.91%	-3.72%
R111	1061.50	1061.50		2777.00	1061.50	1061.50		817.14	1069.50	952.97	873.96	0.75%	0.75%
R112	1023.77	1073.00	4.81%	tl	1040.77	1059.70	1.79%	tl	1055.60	3558.99	1508.65	-1.65%	-0.39%
RC101	1708.30	1708.30		1.00	1708.30	1708.30		73.03	1708.90	144.04	130.49	0.04%	0.04%
RC102	1700.50	1700.50		3.00	1700.50	1700.50		112.40	1700.50	146.58	139.96	0.00%	0.00%
RC103	1696.80	1696.80		3.00	1696.80	1696.80		268.54	1697.40	149.67	132.36	0.04%	0.04%
RC104	1696.70	1696.70		3.00	1696.58	1696.70	0.01%	tl	1697.30	158.39	135.87	0.04%	0.04%
RC105	1700.10	1700.10		3.00	1700.10	1700.10		113.24	1700.10	143.17	127.86	0.00%	0.00%
RC106	1699.00	1699.00		3.00	1699.00	1699.00		109.36	1699.60	167.29	154.82	0.04%	0.04%
RC107	1698.60	1698.60		4.00	1698.60	1698.60		241.35	1699.20	156.67	139.11	0.04%	0.04%
RC108	1696.70	1696.70		4.00	1696.70	1696.70		691.80	1697.30	152.07	136.14	0.04%	0.04%
C201	1159.40	1159.40		465.00	1159.40	1159.40		114.47	1161.90	165.52	156.97	0.22%	0.22%
C202	1142.44	1156.90	1.27%	tl	1156.90	1156.90		205.56	1161.80	218.39	174.40	0.42%	0.42%
C203	1144.51	1159.20	1.28%	tl	1156.90	1156.90		3375.53	1160.80	509.22	314.56	0.14%	0.34%
C204	1137.05	1175.00	3.34%	tl	1156.90	1156.90		1602.00	1160.80	362.62	361.09	-1.22%	0.34%
C205	1154.91	1156.90	0.17%	tl	1156.90	1156.90		238.89	1160.80	387.75	268.03	0.34%	0.34%
C206	1149.33	1162.70	1.16%	tl	1156.90	1156.90		290.05	1160.80	438.08	360.80	-0.16%	0.34%
C207	1156.90	1156.90		2882.00	1156.90	1156.90		596.01	1160.80	423.61	361.04	0.34%	0.34%
C208	1148.45	1156.90	0.74%	tl	1156.90	1156.90		233.63	1160.80	362.41	360.93	0.34%	0.34%
R201	1107.70	1107.70		392.00	1107.70	1107.70		326.36	1122.00	428.51	356.05	1.27%	1.27%
R202	1080.20	1080.20		707.00	1080.20	1080.20		411.80	1097.10	802.67	542.43	1.54%	1.54%
R203	1045.60	1059.20	1.30%	tl	1051.94	1096.80	4.09%	tl	1064.40	2147.84	2040.20	0.49%	-3.04%
R204	1026.45	1068.60	4.11%	tl	1044.24	1053.50	0.88%	tl	1055.60	1918.53	656.53	-1.23%	0.20%
R205	1072.98	1093.10	1.88%	tl	1078.91	1099.70	1.89%	tl	1094.80	1337.96	386.56	0.16%	-0.45%
R206	1071.50	1071.50	0.00%	2874.00	1069.62	1073.90	0.40%	tl	1085.60	tl	1951.08	1.30%	1.08%
R207	1041.34	1077.50	3.47%	tl	1054.48	1077.00	2.09%	tl	1064.40	1875.52	1710.25	-1.23%	-1.18%
R208	1024.27	1125.00	9.83%	tl	1041.51	1067.60	2.44%	tl	1055.60	1889.48	948.30	-6.57%	-1.14%
R209	1039.90	1063.30	2.25%	tl	1054.50	1054.50		789.82	1055.80	700.34	484.11	-0.71%	0.12%
R210	1066.52	1072.90	0.60%	tl	1070.69	1072.90	0.21%	tl	1080.70	1057.87	393.28	0.72%	0.72%
R211	1027.38	1063.20	3.49%	tl	1041.38	1053.50	1.15%	tl	1055.60	3286.48	1172.31	-0.72%	0.20%
RC201	1708.30	1708.30		3.00	1708.30	1708.30		110.73	1708.90	142.48	131.28	0.04%	0.04%
RC202	1700.50	1700.50		6.00	1700.50	1700.50		174.49	1700.50	159.85	133.60	0.00%	0.00%
RC203	1696.80	1696.80		3.00	1696.80	1696.80		418.87	1697.40	150.73	132.08	0.04%	0.04%
RC204	1696.70	1696.70		3.00	1696.70	1696.70		3076.19	1697.30	179.52	150.58	0.04%	0.04%
RC205	1700.40	1700.40		2.00	1700.40	1700.40		93.59	1700.40	142.59	130.71	0.00%	0.00%
RC206	1699.00	1699.00		2.00	1699.00	1699.00		80.36	1699.60	160.68	156.68	0.04%	0.04%
RC207	1698.60	1698.60		3.00	1698.60	1698.60		200.48	1699.20	150.01	133.88	0.04%	0.04%
RC208	1696.70	1696.70		4.00	1696.70	1696.70		1778.10	1697.30	170.03	134.98	0.04%	0.04%
<i>Average</i>										784.59	486.75	-0.09%	0.09%
<i>Minimum</i>										37.27	31.89	-10.91%	-3.72%
<i>Maximum</i>										tl	2040.20	1.84%	1.84%

Table 4: Results of the state-of-the-art BC methods and the CG-based heuristic on the SDVRPTW instances with $n = 50$ and $Q = 100$.

<i>Instance</i>	Bianchessi and Irnich (2019)				Munari and Savelsbergh (2019)				CG-based heuristic				
	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>UB</i>	<i>T_{total}</i>	<i>T_{best}</i>	ΔUB_{BI}	ΔUB_{MS}
C101	587.50	587.50		9.00	587.50	587.50		28.91	587.80	324.71	322.86	0.05%	0.05%
C102	584.60	584.60		73.00	584.60	584.60		68.03	588.50	427.15	422.93	0.66%	0.66%
C103	582.10	582.10		909.00	579.49	602.50	3.82%	tl	583.30	458.98	442.63	0.21%	-3.29%
C104	578.80	578.80		1570.00	575.50	593.70	3.07%	tl	584.10	443.42	417.88	0.91%	-1.64%
C105	587.50	587.50		4.00	587.50	587.50		31.66	587.60	364.86	362.16	0.02%	0.02%
C106	587.50	587.50		9.00	587.50	587.50		29.68	589.10	361.77	359.85	0.27%	0.27%
C107	587.50	587.50		4.00	587.50	587.50		14.58	589.80	378.33	370.94	0.39%	0.39%
C108	584.00	584.00		42.00	584.00	584.00		598.74	587.60	392.95	384.99	0.61%	0.61%
C109	579.80	579.80		160.00	579.80	579.80		353.94	584.00	454.91	434.15	0.72%	0.72%
R101	1043.80	1043.80		0.00	1043.80	1043.80		1.79	1043.80	1.40	1.36	0.00%	0.00%
R102	842.78	913.20	8.36%	tl	862.43	948.20	9.05%	tl	913.20	1.86	1.75	0.00%	-3.83%
R103	740.52	804.70	9.09%	tl	747.19	840.50	11.10%	tl	807.60	84.29	75.36	0.36%	-4.07%
R104	688.04	714.80	3.89%	tl	687.00	724.60	5.19%	tl	710.30	737.08	566.43	-0.63%	-2.01%
R105	918.10	918.10		11.00	918.10	918.10		95.24	918.10	74.55	72.14	0.00%	0.00%
R106	774.57	824.70	6.47%	tl	779.54	827.20	5.76%	tl	821.50	91.90	87.87	-0.39%	-0.69%
R107	703.28	765.50	8.85%	tl	707.66	786.70	10.05%	tl	753.90	404.92	391.27	-1.54%	-4.35%
R108	678.00	732.40	8.02%	tl	686.81	716.90	4.20%	tl	706.30	1108.10	994.49	-3.70%	-1.50%
R109	764.48	810.40	6.01%	tl	767.56	810.50	5.30%	tl	804.10	94.98	83.05	-0.78%	-0.80%
R110	694.89	762.10	9.67%	tl	705.61	763.70	7.61%	tl	744.40	425.91	396.15	-2.38%	-2.59%
R111	706.35	758.40	7.37%	tl	713.17	803.30	11.22%	tl	752.70	331.76	310.84	-0.76%	-6.72%
R112	671.58	727.00	8.25%	tl	678.25	732.80	7.44%	tl	704.80	1394.77	1243.70	-3.15%	-3.97%
RC101	990.50	990.50		10.00	990.50	990.50		44.64	993.80	252.02	250.28	0.33%	0.33%
RC102	960.20	960.20		229.00	960.20	960.20		103.63	960.20	323.48	322.52	0.00%	0.00%
RC103	936.20	936.20		1876.00	936.20	936.20		695.31	936.20	379.66	362.39	0.00%	0.00%
RC104	915.90	915.90		2.00	915.90	915.90		115.10	915.90	363.34	361.26	0.00%	0.00%
RC105	957.40	957.40		203.00	957.40	957.40		80.45	957.40	362.73	362.21	0.00%	0.00%
RC106	936.40	936.40		13.00	936.40	936.40		44.63	936.40	285.03	284.38	0.00%	0.00%
RC107	915.10	915.10		2.00	915.10	915.10		64.36	915.10	280.26	279.44	0.00%	0.00%
RC108	911.90	911.90		1.00	911.90	911.90		58.00	911.90	362.62	360.77	0.00%	0.00%
C201	693.10	693.10		263.00	693.10	693.10		192.81	693.70	276.93	256.37	0.09%	0.09%
C202	686.20	686.20		447.00	686.20	686.20		731.44	686.20	287.54	274.71	0.00%	0.00%
C203	685.40	685.40		2338.00	685.40	685.40		346.49	685.40	378.91	373.90	0.00%	0.00%
C204	684.80	684.80		2759.00	684.80	684.80		341.46	686.10	602.61	476.07	0.19%	0.19%
C205	684.80	684.80		121.00	684.80	684.80		118.42	684.80	257.92	246.05	0.00%	0.00%
C206	684.80	684.80		439.00	684.80	684.80		126.37	685.20	316.00	307.07	0.06%	0.06%
C207	684.80	684.80		1117.00	684.80	684.80		140.92	685.70	478.24	438.90	0.13%	0.13%
C208	684.80	684.80		365.00	684.80	684.80		247.97	685.10	302.65	283.00	0.04%	0.04%
R201	843.00	843.00		80.00	843.00	843.00		43.19	843.00	113.18	101.68	0.00%	0.00%
R202	750.80	782.70	4.25%	tl	750.49	806.60	6.96%	tl	772.40	342.25	325.56	-1.33%	-4.43%
R203	697.78	736.70	5.58%	tl	708.24	722.90	2.03%	tl	719.80	404.72	353.88	-2.35%	-0.43%
R204	682.51	691.90	1.38%	tl	682.20	718.40	5.04%	tl	705.40	654.60	563.44	1.91%	-1.84%
R205	758.80	758.80		3286.00	758.80	758.80		1289.72	758.80	235.34	228.23	0.00%	0.00%
R206	719.59	728.10	1.18%	tl	720.71	759.10	5.06%	tl	730.30	508.72	483.25	0.30%	-3.94%
R207	691.47	712.00	2.97%	tl	691.98	734.30	5.76%	tl	715.40	633.47	492.64	0.48%	-2.64%
R208	673.82	691.90	2.68%	tl	681.82	701.10	2.75%	tl	703.50	1133.34	1112.60	1.65%	0.34%
R209	702.84	725.40	3.21%	tl	704.30	752.50	6.40%	tl	723.90	450.74	366.73	-0.21%	-3.95%
R210	722.45	745.70	3.22%	tl	718.25	798.90	10.10%	tl	750.80	816.89	683.52	0.68%	-6.41%
R211	674.39	715.80	6.14%	tl	676.92	721.60	6.19%	tl	701.70	655.83	539.44	-2.01%	-2.84%
RC201	966.20	966.20		2.00	966.20	966.20		40.82	977.30	333.97	332.95	1.14%	1.14%
RC202	946.50	946.50		64.00	946.50	946.50		80.10	946.90	365.35	365.12	0.04%	0.04%
RC203	926.40	926.40		22.00	926.40	926.40		143.03	926.40	371.34	362.23	0.00%	0.00%
RC204	915.90	915.90		1.00	915.90	915.90		53.81	915.90	370.32	362.51	0.00%	0.00%
RC205	946.70	946.70		4.00	946.70	946.70		44.07	946.70	363.11	362.37	0.00%	0.00%
RC206	940.80	940.80		1.00	940.80	940.80		14.88	940.80	337.24	335.58	0.00%	0.00%
RC207	924.10	924.10		8.00	924.10	924.10		75.35	924.10	365.81	361.30	0.00%	0.00%
RC208	911.90	911.90		1.00	911.90	911.90		70.16	911.90	362.70	361.89	0.00%	0.00%
<i>Average</i>										405.13	376.38	-0.14%	-1.02%
<i>Minimum</i>										1.40	1.36	-3.70%	-6.72%
<i>Maximum</i>										1394.77	1243.70	1.91%	1.14%

Table 5: Results of the state-of-the-art BC methods and the CG-based heuristic on the SDVRPTW instances with $n = 100$ and $Q = 50$.

<i>Instance</i>	Munari and Savelsbergh (2019)					CG-based heuristic				
	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>T_{best}</i>	<i>UB</i>	<i>T_{total}</i>	<i>T_{best}</i>	ΔUB_{MS}	Gap_{MS}
C101	2449.89	2460.50	0.43%	tl	3355.46	2468.70	478.69	401.52	0.33%	0.76%
C102	2443.04	2500.20	2.29%	tl	1531.52	2467.40	943.71	837.88	-1.33%	0.99%
C103	2442.48	2504.60	2.48%	tl	2107.81	2464.70	1114.88	1097.76	-1.62%	0.90%
C104	2441.25	2534.90	3.69%	tl	2238.88	2461.40	1310.31	776.09	-2.99%	0.82%
C105	2452.01	2459.20	0.29%	tl	1021.34	2468.00	589.24	568.38	0.36%	0.65%
C106	2448.64	2526.00	3.06%	tl	1172.73	2462.50	490.04	372.06	-2.58%	0.56%
C107	2448.99	2459.20	0.42%	tl	3494.93	2462.90	595.62	371.25	0.15%	0.56%
C108	2443.24	2495.00	2.07%	tl	2614.65	2458.50	477.29	460.75	-1.48%	0.62%
C109	2442.16	2497.30	2.21%	tl	2300.31	2456.50	397.18	375.12	-1.66%	0.58%
R101	1977.64	2066.70	4.31%	tl	1200.46	2064.10	tl	2629.84	-0.13%	4.19%
R102	1914.08	2127.70	10.04%	tl	326.95	1991.20	tl	442.05	-6.86%	3.87%
R103	1831.07	2158.20	15.16%	tl	667.03	1946.50	tl	438.49	-10.88%	5.93%
R104	1807.99	2084.10	13.25%	tl	970.83	1928.20	tl	1628.50	-8.09%	6.23%
R105	1897.18	2102.60	9.77%	tl	207.17	1977.60	tl	842.98	-6.32%	4.07%
R106	1859.72	2038.80	8.78%	tl	474.82	1947.40	tl	1074.58	-4.69%	4.50%
R107	1814.04	2016.20	10.03%	tl	815.87	1919.40	tl	442.97	-5.04%	5.49%
R108	1804.50	2114.00	14.64%	tl	938.25	1916.80	tl	1567.61	-10.29%	5.86%
R109	1829.81	2005.50	8.76%	tl	70.93	1932.10	tl	408.33	-3.80%	5.29%
R110	1808.52	2100.00	13.88%	tl	784.63	1904.70	tl	3944.87	-10.25%	5.05%
R111	1815.01	2121.30	14.44%	tl	756.27	1932.00	tl	471.52	-9.80%	6.06%
R112	1799.32	2046.00	12.06%	tl	111.33	1914.80	tl	1838.65	-6.85%	6.03%
RC101	2748.26	2922.30	5.96%	tl	1130.59	2835.70	tl	833.76	-3.05%	3.08%
RC102	2712.45	3075.60	11.81%	tl	549.27	2787.00	tl	3406.49	-10.36%	2.67%
RC103	2687.32	3114.70	13.72%	tl	663.11	2765.50	tl	tl	-12.63%	2.83%
RC104	2679.22	2895.60	7.47%	tl	984.43	2771.50	tl	484.59	-4.48%	3.33%
RC105	2726.69	2952.90	7.66%	tl	3251.64	2798.50	tl	3877.58	-5.52%	2.57%
RC106	2715.33	2890.00	6.04%	tl	2698.45	2782.10	tl	1493.91	-3.88%	2.40%
RC107	2689.13	3046.60	11.73%	tl	737.33	2769.50	tl	3896.09	-10.01%	2.90%
RC108	2679.84	3011.60	11.02%	tl	1006.06	2755.20	tl	556.66	-9.31%	2.74%
C201	2534.34	2605.80	2.74%	tl	773.46	2552.10	943.62	363.40	-2.10%	0.70%
C202	2521.45	2729.40	7.62%	tl	307.29	2549.90	tl	1245.48	-7.04%	1.12%
C203	2514.31	2865.40	12.25%	tl	653.06	2547.40	3636.76	362.15	-12.48%	1.30%
C204	2509.66	2852.10	12.01%	tl	882.47	2544.50	tl	3001.84	-12.09%	1.37%
C205	2527.39	2756.50	8.31%	tl	2208.39	2549.20	2034.86	1833.91	-8.13%	0.86%
C206	2525.54	2658.30	4.99%	tl	453.45	2546.70	tl	388.17	-4.38%	0.83%
C207	2524.59	2566.80	1.64%	tl	2390.29	2546.70	tl	1226.09	-0.79%	0.87%
C208	2525.62	2622.80	3.71%	tl	3047.55	2544.60	tl	552.91	-3.07%	0.75%
R201	1885.39	2044.30	7.77%	tl	2914.84	1968.00	tl	3634.10	-3.88%	4.20%
R202	1853.10	2084.40	11.10%	tl	502.74	1948.10	tl	423.24	-7.00%	4.88%
R203	1812.33	2092.10	13.37%	tl	901.64	1930.00	tl	362.97	-8.40%	6.10%
R204	1805.02	2093.40	13.78%	tl	1124.55	1914.10	tl	547.80	-9.37%	5.70%
R205	1836.17	2118.40	13.32%	tl	467.77	1938.90	tl	1237.91	-9.26%	5.30%
R206	1827.02	2099.40	12.97%	tl	680.03	1929.50	tl	474.93	-8.81%	5.31%
R207	1806.75	2050.70	11.90%	tl	950.14	1921.40	tl	3382.48	-6.73%	5.97%
R208	1798.29	2014.60	10.74%	tl	929.62	1905.90	tl	575.35	-5.70%	5.65%
R209	1812.84	2027.50	10.59%	tl	902.85	1902.50	tl	2022.90	-6.57%	4.71%
R210	1822.17	2124.40	14.23%	tl	824.48	1938.80	tl	1760.38	-9.57%	6.02%
R211	1797.75	2123.40	15.34%	tl	1026.85	1910.40	tl	1954.00	-11.15%	5.90%
RC201	2734.92	3007.70	9.07%	tl	410.51	2826.90	tl	936.88	-6.40%	3.25%
RC202	2697.39	3074.30	12.26%	tl	713.13	2776.30	tl	2106.82	-10.73%	2.84%
RC203	2691.29	2991.30	10.03%	tl	1104.36	2759.70	tl	2231.00	-8.39%	2.48%
RC204	2682.12	2971.30	9.73%	tl	1232.68	2769.90	tl	398.52	-7.27%	3.17%
RC205	2706.66	2991.80	9.53%	tl	483.93	2785.30	tl	947.29	-7.41%	2.82%
RC206	2706.05	3029.30	10.67%	tl	458.07	2783.40	tl	585.44	-8.83%	2.78%
RC207	2690.41	3354.50	19.80%	tl	698.21	2750.30	tl	3835.52	-21.97%	2.18%
RC208	2682.46	3455.10	22.36%	tl	1166.60	2749.70	tl	3417.36	-25.65%	2.45%
<i>Average</i>			9.27%	tl	1185.57		1084.35	1369.98	-6.90%	3.23%
<i>Minimum</i>			0.29%	tl	70.93		397.18	362.15	-25.65%	0.56%
<i>Maximum</i>			22.36%	tl	3494.93		tl	tl	0.36%	6.23%

Table 6: Results of the state-of-the-art BC methods and the CG-based heuristic on the SDVRPTW instances with $n = 100$ and $Q = 100$.

<i>Instance</i>	Munari and Savelsbergh (2019)					CG-based heuristic				
	<i>LB</i>	<i>UB</i>	<i>Gap</i>	<i>T_{total}</i>	<i>T_{best}</i>	<i>UB</i>	<i>T_{total}</i>	<i>T_{best}</i>	ΔUB_{MS}	Gap_{MS}
C101	1391.10	1391.10	0.00%	993.66	857.42	1395.50	747.53	649.28	0.32%	0.32%
C102	1374.30	1380.90	0.48%	tl	1744.15	1381.20	373.40	371.37	0.02%	0.50%
C103	1369.78	1513.40	9.49%	tl	350.92	1383.20	417.24	405.09	-9.41%	0.97%
C104	1364.74	1494.60	8.69%	tl	709.66	1387.80	474.50	461.53	-7.70%	1.66%
C105	1388.90	1389.30	0.03%	tl	909.75	1394.60	449.44	447.06	0.38%	0.41%
C106	1381.70	1381.70	0.00%	925.54	890.30	1392.10	403.76	390.12	0.75%	0.75%
C107	1382.30	1382.30	0.00%	602.73	542.40	1391.30	389.71	377.61	0.65%	0.65%
C108	1374.73	1417.70	3.03%	tl	639.01	1387.50	611.63	514.02	-2.18%	0.92%
C109	1369.98	1464.50	6.45%	tl	1712.43	1388.20	458.39	431.69	-5.50%	1.31%
R101	1638.40	1638.40	0.00%	67.33	56.30	1638.40	22.82	22.07	0.00%	0.00%
R102	1345.37	1580.80	14.89%	tl	300.79	1478.90	49.24	48.15	-6.89%	9.03%
R103	1179.77	1448.50	18.55%	tl	510.53	1281.50	tl	3582.72	-13.03%	7.94%
R104	1100.44	1359.30	19.04%	tl	597.59	1179.40	tl	1556.84	-15.25%	6.70%
R105	1334.03	1501.30	11.14%	tl	147.12	1400.50	2575.02	1879.56	-7.20%	4.75%
R106	1232.09	1513.20	18.58%	tl	434.53	1342.60	tl	3928.20	-12.71%	8.23%
R107	1135.02	1364.80	16.84%	tl	438.15	1241.70	tl	3672.17	-9.91%	8.59%
R108	1095.51	1294.60	15.38%	tl	700.17	1237.20	tl	2103.13	-4.64%	11.45%
R109	1166.50	1382.50	15.62%	tl	356.55	1312.40	tl	1044.38	-5.34%	11.12%
R110	1119.75	1397.90	19.90%	tl	428.55	1259.90	tl	1446.50	-10.95%	11.12%
R111	1129.97	1435.20	21.27%	tl	425.46	1244.70	tl	1116.10	-15.30%	9.22%
R112	1094.14	1319.10	17.05%	tl	1023.45	1245.20	tl	362.97	-5.93%	12.13%
RC101	1708.98	1827.60	6.49%	tl	1435.53	1791.80	tl	2777.39	-2.00%	4.62%
RC102	1580.26	1851.30	14.64%	tl	520.83	1707.50	tl	2831.06	-8.42%	7.45%
RC103	1536.97	1912.70	19.64%	tl	100.39	1686.60	tl	450.17	-13.41%	8.87%
RC104	1525.85	1733.20	11.96%	tl	2947.21	1589.80	645.45	573.48	-9.02%	4.02%
RC105	1618.47	1859.60	12.97%	tl	119.44	1737.00	tl	1398.75	-7.06%	6.82%
RC106	1576.75	1736.10	9.18%	tl	100.89	1672.10	tl	1646.74	-3.83%	5.70%
RC107	1529.38	1806.80	15.35%	tl	552.80	1613.90	tl	3926.24	-11.95%	5.24%
RC108	1516.11	1857.70	18.39%	tl	961.07	1602.10	tl	2035.90	-15.95%	5.37%
C201	1448.13	1586.30	8.71%	tl	1126.14	1469.90	tl	662.84	-7.92%	1.48%
C202	1435.56	1604.20	10.51%	tl	294.42	1472.50	1189.70	739.78	-8.94%	2.51%
C203	1425.23	1588.80	10.29%	tl	544.46	1464.40	852.05	507.91	-8.49%	2.67%
C204	1422.08	1549.30	8.21%	tl	217.86	1464.10	1221.79	1183.74	-5.82%	2.87%
C205	1432.99	1625.00	11.82%	tl	327.34	1461.40	1304.33	878.81	-11.19%	1.94%
C206	1430.41	1583.20	9.65%	tl	640.41	1456.50	987.59	876.56	-8.70%	1.79%
C207	1427.39	1648.70	13.42%	tl	2406.90	1457.10	935.15	868.77	-13.15%	2.04%
C208	1425.50	1581.20	9.85%	tl	555.53	1452.30	526.53	461.59	-8.88%	1.85%
R201	1269.46	1407.10	9.78%	tl	189.99	1345.00	tl	1227.32	-4.62%	5.62%
R202	1201.96	1449.80	17.09%	tl	423.26	1269.70	tl	3590.36	-14.18%	5.33%
R203	1119.19	1383.70	19.12%	tl	586.71	1258.70	tl	414.71	-9.93%	11.08%
R204	1097.29	1342.50	18.27%	tl	1054.61	1233.90	tl	393.46	-8.80%	11.07%
R205	1167.49	1337.90	12.74%	tl	255.16	1266.60	tl	tl	-5.63%	7.82%
R206	1142.47	1394.20	18.06%	tl	539.08	1254.30	tl	362.79	-11.15%	8.92%
R207	1108.55	1339.20	17.22%	tl	815.09	1210.80	tl	3636.34	-10.60%	8.44%
R208	1092.02	1304.50	16.29%	tl	1125.73	1228.70	tl	tl	-6.17%	11.12%
R209	1119.45	1345.80	16.82%	tl	533.63	1230.70	tl	3524.61	-9.35%	9.04%
R210	1132.62	1361.30	16.80%	tl	570.07	1241.20	tl	864.93	-9.68%	8.75%
R211	1090.88	1284.90	15.10%	tl	1318.22	1241.50	tl	1011.51	-3.50%	12.13%
RC201	1641.70	1828.00	10.19%	tl	3464.88	1732.40	tl	1060.18	-5.52%	5.24%
RC202	1552.00	1849.00	16.06%	tl	500.14	1661.30	tl	391.56	-11.30%	6.58%
RC203	1526.39	1898.70	19.61%	tl	101.90	1600.70	tl	2954.11	-18.62%	4.64%
RC204	1518.88	1674.70	9.30%	tl	1089.07	1577.40	tl	2188.54	-6.17%	3.71%
RC205	1582.38	1912.60	17.27%	tl	49.46	1697.00	tl	3940.83	-12.70%	6.75%
RC206	1581.18	1838.70	14.01%	tl	284.78	1664.30	tl	1612.87	-10.48%	4.99%
RC207	1537.29	1813.10	15.21%	tl	713.54	1644.70	tl	1297.90	-10.24%	6.53%
RC208	1518.16	1863.30	18.52%	tl	1361.23	1609.40	tl	894.71	-15.78%	5.67%
Average			12.59%	647.32	742.91		731.76	1407.35	-8.20%	5.65%
Minimum			0.00%	67.33	49.46		22.82	22.07	-18.62%	0.00%
Maximum			21.27%	tl	3464.88		tl	tl	0.75%	12.13%