Heuristic Approaches for the Robust Vehicle Routing Problem

Elyn L. Solano-Charris[1,2], Christian Prins[1], and Andréa Cynthia Santos[1](✉)

[1] ICD-LOSI, Université de Technologie de Troyes, 12, rue Marie Curie, CS 42060, 10004 Troyes CEDEX, France
[2] Universidad de La Sabana, Campus del Puente del Común, Km. 7, Chía, Cundinamarca, Colombia
{elyn.solano_charris,christian.prins,andrea.duhamel}@utt.fr

Abstract. In this article, the Robust Vehicle Routing Problem (RVRP) with uncertain traveling costs is studied. It covers a number of important applications in urban transportation and large scale bio-terrorism emergency. The uncertain data are defined as a bounded set of discrete scenarios associated with each arc of the transportation network. The objective is to determine a set of vehicle routes minimizing the worst total cost over all scenarios. A mixed integer linear program is proposed to model the problem. Then, we adapt some classical VRP heuristics to the RVRP, such as Clarke and Wright, randomized Clarke and Wright, Sequential Best Insertion, Parallel Best Insertion and the Pilot versions of the Best Insertion heuristics. In addition, a local search is developed to improve the obtained solutions and be integrated in a Greedy Randomized Adaptive Search Procedure (GRASP). Computational results are presented for both the mathematical formulation and the proposed heuristics.

Keywords: Vehicle routing · Robust optimization · *Min-max* objective · Heuristic · Local search · Metaheuristic

1 Introduction

The Vehicle Routing Problem (VRP) is a NP-hard problem which aims at defining routes for a fleet of vehicles, such that each vehicle starts and ends its tour at a depot node, each customer is visited once, and vehicle loads comply with vehicle capacity [1]. Introduced by Dantzig and Ramser [2], the VRP is one of the most studied problems in combinatorial optimization. One of its main assumptions is that the parameters and the data are assumed to be deterministic and known in advance [3–5]. Therefore, a perturbation on the input data could result in suboptimal or even infeasible solutions [6]. This assumption simplifies the problem but makes it less realistic since uncertainties occur in most real life contexts. Thus, a new and important trend consists in investigating extensions of the VRP with uncertain data, both in terms of theoretical and practical issues.

P. Fouilhoux et al. (Eds.): ISCO 2014, LNCS 8596, pp. 384–395, 2014.
DOI: 10.1007/978-3-319-09174-7_33

In the last years, VRP problems with parameters affected by uncertainties have been treated by using stochastic approaches, which models uncertainties through random variables with known probability distribution [7–10]. The robust optimization approach is an alternative to stochastic programming, designed as a mean to protect solutions against undesirable impacts due to incomplete or imprecise information on the data. It has been introduced in [11] and applied to a number of applications such as portfolio optimization [12], transportation [13], supply chain management [14] and network design problems [15].

The Robust Vehicle Routing Problem (RVRP) usually refers to uncertain data in the given instances: time windows, traveling costs, demands etc. This study considers the RVRP where each arc is weighted by an uncertain traveling cost or time. This version has important applications in urban transportation and evacuation problems such as large scale bio-terrorism emergency. The RVRP considered in this work is defined on a connected and directed graph $G = (V, A)$ with a set $V = \{0, 1, 2...n\}$ of n vertices (customers), including the depot (0), and a set $A = \{(i, j)|i, j \in V, i \neq j\}$ of arcs. Uncertain data are modeled here as a set of p discrete scenarios $S = \{1, 2, ...p\}$, where each scenario $k \in S$ specifies one cost $c_{ij}^{k} \in \mathbb{R}$ to each arc $(i, j) \in A$. Moreover, a demand d_i is associated with each customer $i \in V$ and a fleet of identical vehicles $F = \{1, 2, ..m\}$, located at the depot, is available. Each vehicle has capacity equal to Q. A solution is a set of vehicle routes starting and ending at the depot, visiting each customer once and respecting vehicles capacity. Its cost is the total cost of traversed arcs. We consider a *min-max* objective: the worst cost of the solution over all scenarios must be minimized.

This work brings the following contributions. We handle uncertain data as a bounded set of discrete scenarios for the costs of the arcs in a directed network (the VRP literature considers undirected graphs with symmetric costs). This situation reflects for instance transit problems in urban networks. A simple mathematical formulation is introduced for this RVRP. Then, we propose several constructive heuristics such as the Clarke and Wright (CW), Randomized CW (RCW), Parallel Best Insertion (PBI), Sequential Best Insertion (SBI), Pilot Parallel Best Insertion (PPBI) and Pilot Sequential Best Insertion (PSBI). In addition, more sophisticated strategies, such as local search and Greedy Randomized Adaptive Search Procedure (GRASP) are elaborated. To the best of our knowledge, no heuristic has been published in the literature to solve the RVRP investigated in this study.

The remaining of this work is organized as follows: a bibliographical review is introduced in Sect. 2, followed by a description of a mathematical formulation in Sect. 3. Then, the proposed heuristics are detailed in Sect. 4. Finally, the computational experiments and concluding remarks are respectively given in Sects. 5 and 6.

2 Related Works

Some works in the literature deal with the RVRP, mainly with uncertain data associated with time windows, travel times, travel costs or demands. We present in this section the main works which either apply robust optimization techniques and some entry points for research applying stochastic programming for the RVRP. The pioneer work [16] addresses the RVRP with uncertain demands and time windows. Analytical results on cluster-first route-second heuristics are given for large scale RVRPs. A survey which outlines the RVRP models with uncertainties related to demands, travel times and cost coefficients can be found in [17]. Some issues on applying stochastic programming and robust optimization are also discussed. Reference [18] provides a more extended review on the RVRP.

The RVRP with uncertain demands is probably the most investigated case. For instance, a Branch-and-Bound (B&B) algorithm is proposed by [19] which considers the *min-max* optimization criterion. The authors analyze the trade-off between robust solutions and deterministic solutions. The computational results show that the robust solution can protect from unmet demand while incurring a small additional cost over the deterministic optimal routes. Furthermore, a Particle Swarm Optimization (PSO) strategy integrating a local search is proposed in [6]. The PSO results are compared with the B&B proposed by [19], and performs well when costs are affected by small perturbations. The authors in [20] adapted the two-index and three-index VRP formulations and another one using the Miller, Tucker and Zemlin (MTZ) subtour elimination constraints. A Branch-and-Cut method is applied. The results demonstrate the computational advantages of the robust rounded capacity inequality cuts for the RVRP and the robust two-index vehicle flow formulation. Moreover, the price of robustness using different level of uncertainties is also analyzed.

More recently, the Open Vehicle Routing Problem (OVRP) with uncertain demands has been investigated by [21]. The OVRP differs from the VRP since vehicles do not return to the depot. In order to trade off the unmet demands, four heuristics strategies are considered to obtain the optimal solution when demands are disturbed, and a differential evolutionary algorithm is proposed. Instances with up to 199 customers have been tested.

Concerning the RVRP with uncertainties on time windows, a cutting-plane algorithm is embedded in a B&B method and in a column generation approach based on path inequalities and resource inequalities in [22]. Computational results are presented using the budget uncertainty polytopes and the results show that the path inequalities are almost as easy to separate as in the deterministic VRP case.

Uncertain travel times are handled via stochastic programming in [23]. The authors consider a two-stage recourse stochastic programming solved by a B&B. Two strategies "here and now" and "wait and see" are investigated. For the scenarios, the authors restrict the number of times an uncertain travel time can have the worst value. The results show that the approach obtains good solutions when the penalization over the objective function is small. As far as we know, the only work dealing with robust optimization strategies for the RVRP with uncertain travel costs is [24]. The authors considers the RVRP with uncertain travel costs modeled as interval data. An ant colony algorithm is introduced, where perturbations are performed on the objective coefficients towards the upper bounds of the interval data. This work differs from the RVRP focused here, since interval data are considered instead of discrete scenarios. Moreover, we consider in this paper a directed network with asymmetric costs.

Finally, the work [25] deals with the RVRP with uncertain data in travel times and demands. The authors also consider delays. Thus, the number of acceptable delayed segments are provided in order to determine a supported robustness. A multiflow formulation solved by Dantzig-Wolfe decomposition scheme are presented. The solutions are compared with the solutions obtained with the Monte-Carlo simulation and show that a robust solution can be improved with a small penalty in the optimal value.

3 Mathematical Formulation

A Mixed Integer Linear Programming (MILP) formulation for the *min-max* RVRP is given from (1) to (10). It makes use of binary variables x_{ij} which defines if an arc (i, j) belongs to the solution ($x_{ij} = 1$) or not ($x_{ij} = 0$). Variables t_i specify the vehicle load when leaving each node $i \in V$. Thus, the variable t_0 associated with the depot is considered only when vehicles leave the depot, as a consequence $t_0 = 0$.

$$\min \quad Z = \delta \qquad \text{subject to:} \tag{1}$$

$$\sum_{(i,j)\in A} c_{ij}^{k} x_{ij} \leq \delta \qquad \forall k \in S \tag{2}$$

$$\sum_{i\in V} x_{ij} = 1 \qquad \forall j \in V\backslash\{0\} \tag{3}$$

$$\sum_{j\in V} x_{ij} = 1 \qquad \forall i \in V\backslash\{0\} \tag{4}$$

$$\sum_{i\in V} x_{0i} = m \tag{5}$$

$$t_j \geq t_i + d_j - Q(1 - x_{ij}) \qquad \forall (i,j) \in A,\ i, j \neq 0 \tag{6}$$

$$d_i \leq t_i \leq Q \qquad \forall i \in V \tag{7}$$

$$x_{ij} \in \{0, 1\} \qquad \forall (i,j) \in A \tag{8}$$

$$\delta \geq 0 \tag{9}$$

$$t_i \geq 0 \qquad \forall i \in V \tag{10}$$

The objective function (1) together with constraints (2) ensure that the worst total cost is minimized. Equalities (3) and (4) are the classical flow conservation constraints, which guarantee that only one vehicle arrives at each customer i and leaves it. Constraints (5) specify that m vehicles leave the depot and return to it, due to the flow conservation restrictions. Constraints (6) and (7) generalize the classical MTZ constraints for the TSP [26]. Here they are based on vehicle loads: if a vehicle visits i then j, its load increases by d_j. These constraints prevent subtours and ensure that vehicle capacity is respected. Finally, variables are defined from (8) to (10).

4 Heuristic Methods

Two Clarke and Wright-based heuristics, two insertion-based heuristics, and two pilot insertion-based heuristics are described below, as well as a local search and a GRASP. The main differences between the proposed RVRP heuristics and the similar versions for the VRP found in the literature are mainly the use of scenarios, the asymmetric arc costs, and a lexicographic approach to compare decisions in a greedy heuristic and evaluate moves in the local search.

A solution (complete or being constructed) is defined by a set of feasible routes, the total cost for each scenario k and the worst cost (maximum of these costs). We first tried implementations that do not degrade the worst cost, but the results were mitigated. Indeed, several decisions can lead to the same variation of the worst cost and cannot be distinguished. Moreover, a single move in the local search is in general not enough to decrease the worst cost: a sequence of moves is required. The lexicographic approach consists in sorting the costs of a solution (one per scenario) in non-increasing order, giving what we call the *lexicographic vector* of a solution. Then, a solution is said to be better than another solution if its vector is lexicographically smaller. This strategy is quite fruitful for instance, in the local search, a sequence of moves can improve progressively the lexicographic vector until the first component (the worst case) decreases. The price to pay is a multiplication of complexity expressions by $O(p \log p)$ to sort the costs and get the *lexicographic vector*. However, this extra cost is acceptable if the number of scenarios is relatively small compared with the number of customers. Compared to classical VRP heuristics, the algorithms are also complicated by the directed network. For instance, the cost of a sequence of customers changes when reversed, contrary to the undirected case.

4.1 Constructive Heuristics

Clarke and Wright-Based Heuristics. The CW heuristic or savings method is a well-known constructive heuristic for the VRP [27]. Its general idea consists of concatenating two routes such that the cost saving is maximized. The original CW heuristic considers symmetric costs associated with the arcs. As mentioned above, costs are asymmetric. Thus, there are more ways to concatenate two routes than in the original CW for each scenario. Furthermore, since the optimization criterion considered is a *min-max* one, edges are sorted in increasing order of savings instead of decreasing order in the CW.

A randomized version of the CW, referred as RCW is also proposed. The RCW is based on the CW, but when evaluating the merger of two routes, the resulting solution cost is increased by a random percentage in the range $[0, \theta]$. Thus, instead of selecting the best savings at each iteration, good moves, not necessarily the best, can be done. The best concatenation at each iteration is determined in $O(n^2)$, multiplied by the complexity of the sorting algorithm in $O(p \log p)$. Hence, CW runs in $O(n^3 p \log p)$, where p is the number of scenarios and n is the number of customers.

Insertion-Based Heuristics. The insertion heuristics have been proposed by [28]. The best insertion heuristics build a set of feasible routes by selecting seed customers and inserting them in one of the partial routes already created. At each iteration, the heuristic expands the current route by inserting the best unserviced customer, such that the vehicle capacity is ensured. SBI and PBI heuristics are introduced below.

SBI begins with a single route reduced to a loop on the depot. Best insertions are performed in the current route and it stops whenever all customers are attended, or if the new cannot be done because it exceeds the vehicle capacity. SBI tends to assign few customers to the last vehicle, thus routes are not balanced considering the number of customers.

PBI employs all vehicles available and fills m routes in parallel, which are initially empty. Then, at each iteration, the heuristic evaluates all feasible insertions of unrouted customers for every available routes. Since demands cannot be splitted, PBIH can fail to use m vehicles. In this case, one extra route is created and the heuristic performs similar steps as the SBIH.

The SBI and PBI mainly differ on the way the routes are built. In the SBI, a client is inserted at a time, and the routes are filled one after the other. While in the PBI heuristics, a set of routes are initially available and the customers are inserted in parallel to each route, i.e. the first customer is assigned for each route, only then, the second customers is set to the routes, etc. The resulting solutions can be found in $O(n^2 p \log p)$ by the SBI and for the PBI heuristics in $O(mn^2 p \log p)$.

Pilot Insertion-Based Heuristics. In the pilot method [29], a main heuristic calls an auxiliary heuristic (the pilot heuristic) to guide its decisions. We derived two pilot heuristics from SBI and PBI, called respectively PSBI and PPBI. It requires partial solutions, generated by the insertion heuristic, and with some customers already attended. Starting from the depot, the Pilot heuristic tests at each iteration all possible ways of extending the emerging route, adding to a route, one customer not visited at a time, following the PSBI. For both the PSBI and PPBI, a copy of the solution under construction is taken to insert the customer tested in incumbent partial solution, and the heuristic iterates by calling the pilot heuristic. The best insertion is performed. The difference between the PSBI and the PPBI is the way customers are included in the partial solution, and follows the insertions strategies previously described for the SBI and PBI. Since the pilot version of a heuristic consist in calling it as a subroutine at each iteration, the complexity of the non-pilot version is squared, making the pilot approach time-consuming.

4.2 Local Search

Relocation, *Interchanges*, and *2-opt* moves are applied in the local search procedure for the RVRP. Each iteration of the local search examines all ordered pairs of distinct routes (T, U) and evaluates the moves on one route if $T = U$ or otherwise on two routes $T \neq U$. The cost of the solution obtained if the move were performed is computed for each scenario. The current iteration stops as soon as a move improving the *lexicographic vector* of the current solution is detected (first improvement local search), or if no such move exists.

Interchanges exchanges two chains which may have 1 or 2 customers each. The lengths of the two swapped chains can be different.

Relocations move one or two adjacent customers to a different position.

2-opt moves intra-routes try to improve a solution by inverting a subsequence between two customers i and j (included, j must be after i). When $T \neq U$, vehicles capacity must be checked before computing the cost variations. A variant of *2-opt* is also proposed considering the asymmetric cost when a route is inverted. For this variant, the chains of nodes before i and after j are inverted.

Concerning complexity, the number of *interchanges* moves is $O(n^2)$ since they are applied to each pair of customers. There are also $O(n^2)$ *relocations*, because the n customers can be inserted in $O(n)$ different positions. There are also $O(n^2)$ *2-opt* moves. Each move traditionally evaluated in $O(1)$ is now checked in $O(p \log p)$, due to the construction of the *lexicographic vector* of the solution obtained by each move.

4.3 Greedy Randomized Adaptive Search Procedure

The GRASP is a multi-start metaheuristic proposed by [30]. It basically consists in building at each iteration, an initial solution using a randomized constructive heuristic, then in improving it by a local search. The best solution found is kept. GRASP is especially interesting as it only requires two components (a randomized heuristic and a local search) and very few parameters like the number of iterations. The GRASP for the RVRP makes use of the RCW heuristic and the local search presented in Sect. 4.2. The stopping criteria is the number of iterations which can be fine-tuned.

5 Computational Experiments

The tests were performed on a Dell Precision M6600 with a 2.2 GHz Intel Core i7-2720QM, 16 GB of RAM and Windows Professional. The proposed heuristics and metaheuristics were developed in Delphi XE, and the mathematical formulation was tested using GLPK (GNU Linear Programming Kit) under default parameters. All experiments for the mathematical formulation were carried out with a runtime limit of four hours. The goals of the experiments are to analyze the heuristics performance and the impact of using discrete scenarios for the RVRP.

Table 1. Results for the constructive and greedy heuristics

Instance name	d	Q	GLPK					*Gap'*(%)					
			LR	LB	UB	T(s)	*Gap*	SBI	PBI	PSBI	PPBI	CW	RCW
n10-m2-p10	264	150	195.5	217	*217	8.2	0.0	36.4	36.4	5.5	13.4	21.2	4.6
n10-m2-p20	264	175	250.6	284	*284	32.2	0.0	18.0	18.0	15.1	12.0	7.4	2.8
n10-m2-p30	264	200	272.1	301	*301	35.8	0.0	23.9	23.9	16.6	12.3	8.6	4.7
n15-m2-p10	346	200	320.2	346	*346	313.2	0.0	37.9	37.9	24.9	24.3	19.1	9.5
n15-m2-p20	346	230	339.3	373	*373	6,560	0.0	32.4	32.4	21.7	19.3	12.6	6.2
n15-m2-p30	346	260	368.7	398	404	-	1.5	49.0	49.0	25.1	21.4	21.4	9.5
n20-m2-p10	441	250	402.3	419	423	-	1.0	49.6	40.3	29.6	27.0	24.8	16.2
n20-m2-p20	441	300	443.3	460	470	-	2.2	48.9	37.8	26.1	24.3	16.3	9.3
n20-m2-p30	441	350	463.7	481	501	-	4.2	44.3	44.3	25.4	23.9	20.6	14.6
n10-m3-p10	264	95	227.0	255	*255	10.4	0.0	30.6	17.2	6.3	8.2	19.2	4.7
n10-m3-p20	264	105	279.5	316	*316	24.7	0.0	16.8	16.8	12.7	6.6	12.7	0.6
n10-m3-p30	264	115	302.0	337	*337	38.9	0.0	28.8	28.8	10.4	9.8	8.9	0.0
n15-m3-p10	346	120	349.1	381	*381	2,200	0.0	38.6	38.6	13.9	17.3	15.2	1.0
n15-m3-p20	346	140	365.2	399	*399	6,871	0.0	26.1	26.1	18.0	14.5	13.8	4.3
n15-m3-p30	346	160	394.4	426	433	-	1.6	36.9	36.9	22.8	16.4	11.7	5.9
n20-m3-p10	441	160	427.3	443	448	-	1.1	43.1	51.5	32.1	30.0	26.9	17.2
n20-m3-p20	441	175	466.3	481	497	-	3.3	50.3	34.1	23.3	24.9	20.6	15.0
n20-m3-p30	441	190	489.6	508	528	-	3.9	54.3	53.1	23.8	22.8	17.1	16.9
Average							1.0	37.0	34.6	19.6	18.3	16.6	8.4

For the purpose of these experiments, random instances were generated as follows. The travel cost of each arc and the demand per client is randomly chosen in [1,50]. The number of vehicles is either 2 or 3. The capacity of vehicles is selected to ensure a slack of $0.2Q$ to $0.8Q$ between the total demand d and fleet capacity mQ. The number of scenarios is either 10, 20 or 30. The file name format of each instance is $n\rho$-$m\beta$-$p\gamma$, where ρ, β and γ stand respectively for the numerical values of n, m and p.

Tables 1 and 2 summarize the results for the MILP, the greedy heuristics, the local search and the GRASP. The MILP results in Table 1 corresponds to the five columns LR (linear relaxation), LB (best lower bound), UB (best upper bound (UB), *Gap* (percentage deviation of the optimum or the best upper bound to LB), and $T(s)$ (computational time in seconds) except for the instances where the solver has attained the time limit. In this case, it is referred as "–". The last six columns indicate the percentage gap *Gap'* between the upper bound produced by each proposed heuristic and the lower bound achieved by GLPK. The RCW heuristic is run 50 times with $\theta = 8\,\%$. Due to lack of space, running times of greedy heuristics are not reported but here after they are commented. In Table 2, the results produced by the GLPK are recalled, followed by the solution values and running times for the local search and the GRASP. The local search is applied to the solution produced by the CW heuristic. The GRASP performs $ncalls = 500$ iterations, each of them calling the RCW heuristic with $\theta = 8\,\%$ and the local search. Optimal values are identified by asterisks "$*$".

Results in Tables 1 and 2 demonstrate that in spite of its simple definition, the RVRP is a very hard problem to solve. In fact, the heuristics which work

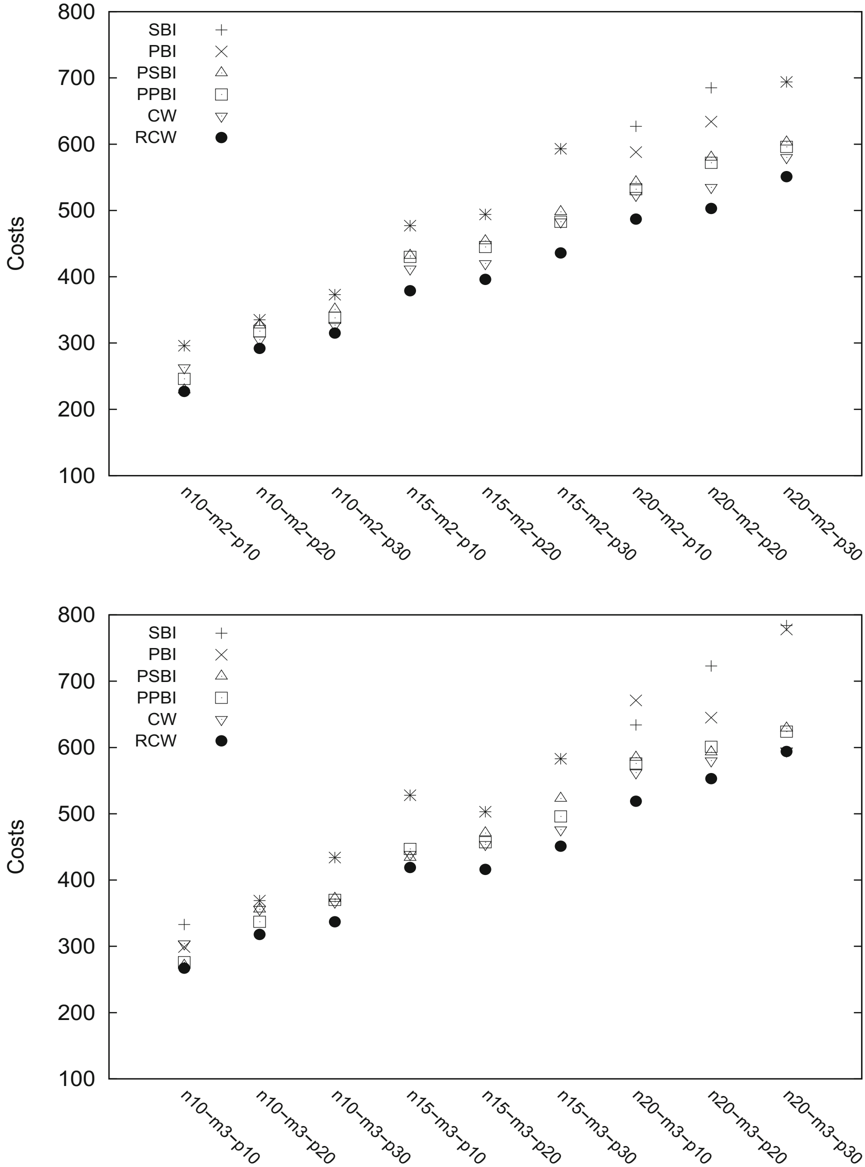

Fig. 1. Heuristic results respectively for the instances with $m = 2$ and $m = 3$.

well for the VRP are not able to find a good approximation. The two insertion heuristics are the fastest (less than 1 ms on average) but lead to very poor average gaps, 37.0 % for SBI and 34.6 % for PBI.

The pilot heuristics are able to find better solutions, with average gaps of 19.6 % for the PSBI and 18.3 % for the PPBI, and the running time reaches

Table 2. Results for the heuristic coupled with the local search and the GRASP

Instance name	GLPK					CW+ LS			GRASP		
	LR	LB	UB	T(s)	*Gap*	Cost	*Gap*'	T(s)	Cost	*Gap*'	T(s)
n10-m2-p10	195.5	217	*217	8.2	0.0	247	13.8	0.00	*217	0.0	0.28
n10-m2-p20	250.6	284	*284	32.2	0.0	296	4.2	0.00	*284	0.0	0.65
n10-m2-p30	272.1	301	*301	35.8	0.0	307	2.0	0.00	*301	0.0	1.11
n15-m2-p10	320.2	346	*346	313.2	0.0	385	11.3	0.00	347	0.3	1.02
n15-m2-p20	339.3	373	*373	6,560	0.0	410	9.9	0.01	376	0.8	2.39
n15-m2-p30	368.7	398	404	-	1.5	426	7.0	0.01	406	2.0	4.21
n20-m2-p10	402.3	419	423	-	1.0	458	9.3	0.01	436	4.1	2.58
n20-m2-p20	443.3	460	470	-	2.2	508	10.4	0.01	484	5.2	6.22
n20-m2-p30	463.7	481	501	-	4.2	536	11.4	0.02	513	6.7	10.48
n10-m3-p10	227.0	255	*255	10.4	0.0	279	9.4	0.00	*255	0.0	0.22
n10-m3-p20	279.5	316	*316	24.7	0.0	333	5.4	0.00	*316	0.0	0.56
n10-m3-p30	302.0	337	*337	38.9	0.0	348	3.3	0.00	*337	0.0	0.98
n15-m3-p10	349.1	381	*381	2,200.0	0.0	396	3.9	0.00	*381	0.0	0.93
n15-m3-p20	365.2	399	*399	6,871.0	0.0	411	3.0	0.00	*399	0.0	2.25
n15-m3-p30	394.4	426	433	-	1.6	476	11.7	0.01	435	2.1	3.98
n20-m3-p10	427.3	443	448	-	1.1	514	16.0	0.01	464	4.7	2.45
n20-m3-p20	466.3	481	497	-	3.3	540	12.3	0.01	511	6.2	6.01
n20-m3-p30	489.6	508	528	-	3.9	573	12.8	0.02	541	6.5	10.16
Average				9,350.4	1.0		8.7	0.01		2.1	3.14

50 ms on average. In fact, the heuristics with best performances are the CW and the RCW, as can be seen in Table 2 and in Fig. 1. Indeed, they find solutions with average gaps of 16.6 % and 8.4 %, respectively. CW is quite fast (5 ms on average) but the price to pay for RCW and its 50 iterations is an augmented average duration (around 0.3 s).

Starting from the solution returned by CW, the average gap is lowered from 16.6 % to 8.7 % by the local search, in less than 1 s. It is noticed that the best solutions for the RVRP were obtained with the GRASP procedure, which is able to retrieve 8 optima out of the 10 found by GLPK, with a small average gap of 2.1 % in 3.14 s, while GLPK achieves an averaged gap of 1.0 % in 9,350.4 s. This results show the advantages of using our local search and the GRASP to achieve small gaps and retrieve most proven optima in competitive computational time. This good performance probably comes from the combination of a constructive heuristic, a local search with moves that work well on the RVRP, and the random sampling of the local optima done by the GRASP in the solution space.

6 Conclusions

This article considers the RVRP with uncertain data associated with the costs or travel times. In this case, the variation of travel costs in the transportation network are considered. Constructive and greedy heuristics, a local search and a GRASP procedure are proposed. Such strategies have been addressed to deal with asymmetric costs and also with a set of discrete scenarios.

The experimental results show the RVRP is a hard problem in spite of its rather simple statement. Among the proposed heuristics, the CW-based heuristics outperform the others, and the local search manages to reduce their solution gaps. The GRASP is able to find even better solutions, including most proven optima, in reasonable running times.

Regarding future work, the current mathematical formulation can be strengthened and other formulations can be explored. We are currently investigating other metaheuristics and hybridizations for the RVRP. In addition, other optimization criteria can be studied such as *min-max* regret and lexicographical criterion. Finally, we are working on the design of transportation urban networks for which uncertainties are modeled as scenarios, to take into account the delays produced by traffic jams.

Acknowledgements. This research is part of the project METHODI which is funded by the Champagne-Ardenne Region.

References

1. Toth, P., Vigo, D.: Exact solution of the vehicle routing problem. In: Fleet Management and Logistics, pp. 1–31 (1998)
2. Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. Manage. Sci. **6**, 80–91 (1959)
3. Achutan, N.R., Caccetta, L., Hill, S.P.: A new subtour elimination constraint for the vehicle routing problem. Eur. J. Oper. Res. **91**, 573–586 (1996)
4. Cordeau, J., Gendreau, M., Laporte, G., Potvin, J., Semet, F.: A guide to vehicle routing heuristics. J. Oper. Res. Soc. **53**, 512–522 (2002)
5. Brandão, J.: A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. Eur. J. Oper. Res. **19**, 716–728 (2009)
6. Moghaddam, B.F., Ruiz, R., Sadjadi, S.: Vehicle routing problem with uncertain demands: an advanced particle swarm algorithm. Comput. Ind. Eng. **62**, 306–317 (2012)
7. Golden, B.L., Yee, J.R.: A framework for probabilistic vehicle routing. AIIE Trans. **11**, 109–112 (1979)
8. Gendreau, M., Laporte, G., Seguin, R.: A tabu search heuristic for the vehicle routing problem with stochastic demands and customers. Oper. Res. **44**, 469–477 (1996)
9. Jula, H., Dessouky, M.M., Ioannou, P.: Truck route planning in non-stationary stochastic networks with time-windows at customer locations. IEEE Trans. Intell. Transp. Syst. **37**, 51–63 (2006)
10. Ai, J., Kachitvichyanukul, V.: A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. Comput. Oper. Res. **36**, 1693–1702 (2009)
11. Ben-Tal, A., Nemirovski, A.: Robust convex optimization. Math. Oper. Res. **23**, 769–805 (1998)
12. El-Ghaoui, L., Oks, M., Oustry, F.: Worst-case value-at-risk and robust portfolio optimization: a conic programming approach. Oper. Res. **51**, 543–556 (2003)

13. Erera, A.L., Morales, J.C., Savelsbergh, M.: Robust optimization for empty repositioning problems. Oper. Res. **57**, 468–483 (2009)
14. Tal, A.B., Golany, B., Nemirovski, A., Vial, J.: Supplier-retailer exible commitments contracts: a robust optimization approach. Manuf. Serv. Oper. Manage. **7**, 248–273 (2005)
15. Atamturk, A., Zhang, M.: Two-stage robust network flow and design under demand uncertainty. Oper. Res. **55**, 662–673 (2007)
16. Bertsimas, D., Simchi-Levi, D.: A new generation of vehicle routing research: robust algorithms, addressing uncertainty. Oper. Res. **44**, 286–304 (1996)
17. Ordóñez, F.: Robust vehicle routing. INFORMS TutORials Oper. Res. **7**, 153–178 (2010)
18. Solano, E., Santos, A.C., Prins, C.: An overview on solving robust vehicle routing problem. In: 14ème congrès de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF) 2 p. (2013)
19. Sungur, I., Ordóñez, F., Dessouky, M.: A robust optimization approach for the capacitated vehicle routing problem with demand uncertainty. IIE Trans. **40**, 509–523 (2008)
20. Gounaris, C., Wiesemann, W., Floudas, C.A.: The robust capacitated vehicle routing problem under demand uncertainty. Oper. Res. **61**, 677–693 (2013)
21. Erbao, C., Mingyong, L., Hongming, Y.: Open vehicle routing problem with demand uncertainty and its robust strategies. Expert Syst. Appl. **41**, 3569–3575 (2014)
22. Agra, A., Hvattum, L.M., Christiansen, M., Figuereido, R., Poss, M., Requejo, C.: The robust vehicle routing problem with time windows. Comput. Oper. Res. **40**, 856–866 (2013)
23. Han, J., Lee, C., Park, S.: A robust scenario approach for the vehicle routing problem with uncertain travel times. Transp. Sci. **63**, 1294–1306 (2013)
24. Toklu, N., Montemanni, R., Gambardella, L.M.: An ant colony system for the capacitated vehicle routing problem with uncertain travel costs, pp. 32–39 (2013)
25. Lee, C., Lee, K., Park, S.: Robust vehicle routing problem with deadlines and travel time/demand uncertainty. J. Oper. Res. Soc. **63**, 1294–1306 (2012)
26. Miller, C., Tucker, A., Zemlin, R.: Integer programming formulations and traveling salesman problems. J. ACM **7**, 326–329 (1960)
27. Clarke, G., Wright, J.: Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. **12**, 568–581 (1964)
28. Solomon, M.M.: Algorithms for the vehicle routing and scheduling problem with time window constraints. Oper. Res. **35**, 254–265 (1987)
29. Duis, C.W., Voss, S.: The pilot method: a strategy for heuristic repetition with application to the steiner problem in graphs. Networks **34**, 181–191 (1999)
30. Feo, T., Resende, M.: A probabilistic heuristic for a computationally difficult set covering problem. Oper. Res. Lett. **8**, 67–71 (1989)