universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master‘s Thesis

"Large neighborhood search for the vehicle routing Problem with time windows and deterministic and stochastic travel and service times."

verfasst von / submitted by

Marco Vaz

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2015 / Vienna 2015

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 920
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Quantitative Economics, Management, and Finance
Betreut von / Supervisor:	O.Univ.-Prof. Dipl.-Ing. Dr. Richard F. Hartl

Acknowledgements

I want to thank Professor Sophie Parragh for the guidance.

Abstract

In this work a large neighborhood search, LNS, will be applied to the vehicle routing problem with time windows (VRPTW). The LNS algorithm will be initialized with a route created by selecting the cheapest insertion (greedy algorithm). Afterwards, a destroy operator removes customers either by random or worst removal and a repair operator reinserts the customers either by using the greedy algorithm or the greedy algorithm with a regret factor. Different combinations of these destroy and repair operators will be used to analyze their performance with deterministic travel and service time. The best combination will then be tested and discussed using stochastic travel and service time.

Abstrakt

In dieser Arbeit wird eine 'Large Neighborhood Search' (LNS) auf das 'Vehicle Routing Problem' mit Zeitfenstern (VRPTW) angewandt. Der LNS-Algorithmus startet mit einer Route, die durch Auswahl der günstigsten Insertion (greedy Algorithmus) entsteht. Danach löscht ein Destroy-Operator entweder durch Random- oder durch Worst-Removal Kunden und ein Repair-Operator fügt diese Kunden wieder ein, entweder durch den Greedy-Algorithmus oder den Greedy-Algorithmus mit Regret-Faktor. Verschiedene Kombinationen der Destroy- und Repair-Operatoren werden verwendet um deren Performance mit deterministischen Weg- und Servicezeiten zu analysieren. Die beste Kombination wird dann mit stochastischen Weg- und Servicezeiten getestet und diskutiert.

Contents

Chapter 1

Introduction

The vehicle routing problem is an important and complex combinatorial problem where the objective is to build a route in which a certain number of customers is served at the minimum cost.

Applications from transportation logistics to internet networks can benefit from good and fast solutions. The vehicle routing problem with time windows (VRPTW) can be represented by a logistics fleet that needs to deliver packages to a certain number of customers that have specific opening hours. All the trucks start and end at the depot and their supply is limited. Each truck can only serve one customer after the other and each customer must be served by only one truck. The trucks are allowed to arrive before the customer time window is open but the unload service can only happen during the time window. In the relaxed version, the end of the time window can be crossed and the service can start after the time window end. In this case, a penalty for extra working hours will increase the costs.

In the stochastic case, two random variables, the travel time and the service time, are introduced. These two variables will change unpredictably. The randomness in the travel time can be interpreted as a delay during the route caused by an accident or traffic jam and in case of the service time, a delay could be caused by an unexpected problem during the service execution. This is a very important formulation of the problem, because it has several applications in real life. Traffic conditions, driving skills, mechanical failure, parking availability, parking distance, waiting in line to serve the customer and weather conditions influence the route calculations and are better modeled with the SVRPTW (stochastic vehicle routing problem with time windows).

Several papers have addressed the VRPTW problem with different algorithms. Tabu search proposed by Li et al [1], multiple ant colony system by Gambardella et al [6] and multi-objective genetic algorithms by Ombuki et al [5] for a multi objective VRPTW version. The stochastic problem in Li et al [1] with relaxed time windows is the same problem addressed in this work and the other two address the deterministic VRPTW with hard time windows. In all the above mentioned papers, the results are compared with the instances from Professor Solomon [20] like in this work and this will give an overview on how the different heuristics perform with those instances.

Even though some exact methods have been implemented with success [5] [7] for 100 customers, no algorithm has solved all VRPTW with 100 or more customers[5] efficiently. Because of this battle against the NP-hard nature of VRPTW, exact methods became impractical for bigger instances and heuristics methods have an important role in producing good solutions in reasonable time such as genetic algorithms [5] [8], evolution strategies [9], simulated annealing [10], tabu search [11] [1] or ant colony optimization [6].

Large Neighborhood Searches (LNS) have already been applied to VRPTW before. Pisinger et al [12] applied LNS like in this work and random removal, worst removal, greedy heuristic and regret heuristics are among the used algorithms. These algorithms are applied to the deterministic but not to the stochastic approach. Even though other variants can be found in the literature, 'A Branch-and-Price-Based Large Neighborhood Search Algorithm for the Vehicle Routing Problem with Time Windows' by Gagnon et al [13] and 'An Adaptive Large Neighborhood Search for a Vehicle Routing Problem with Multiple Trips' by Azi et al[14], few heuristics have been applied for the stochastic VRPTW, SVRPTW. Besides the work from Li et al [1], a tabu search proposed by Tas et al [15] and a hybrid Metaheuristics for the Vehicle Routing Problem but for Stochastic Demand by Bianchi et al [16].

For this the local search proposed in this work uses a large neighborhood search with nine different algorithms combinations to solve VRPTW and the stochastic VRPTW, SVRPTW.

I $\rightarrow$ random removal - worst removal - greedy - greedy regret

II $\rightarrow$ worst removal - greedy - greedy regret

III $\rightarrow$ random removal - greedy - greedy regret

IV $\rightarrow$ random removal - worst removal - greedy regret

V $\rightarrow$ worst removal - greedy regret

VI $\rightarrow$ random removal - greedy regret

VII $\rightarrow$ random removal - worst removal - greedy

VIII $\rightarrow$ worst removal - greedy

IX $\rightarrow$ random removal - greedy

Like the tabu search in Li et al [1], a local search is used, but without the memory structures. In this work, the local minimum will be avoided by allowing worse solutions [18] during the selection phase and in the random removal, a chance factor will destroy feasible routes making the route creation more diversify. These random processes will remove customers from a secure route position making a step back in an attempt to cross the local minimum and to continue in the direction of the optimum. In the greedy regret, a what-if analysis will predict the best customer insertion considering what if inserted in the second best position. It will be shown that VI random removal and greedy regret will perform the best due to this random destroy factor and what-if insertion. It will also be evident that combinations with worst removal and greedy perform the worst.

An analysis and comparison between the above algorithms and benchmarks with results available online will also be discussed.

Chapter 2

Mathematical formulation

The traditional version of the vehicle routing problem (VRP) can be defined as described in Li et al [1] and Laporte et al [3] on a complete graph $G = \{V, E\}$ where $V = \{0, 1, 2, .., n\}$ is the vertex set, E the edge set, 0 represents the depot and $1..n$ the customers. Each V_i is visited exactly once by only one vehicle and a fixed demand q_i has to be fulfilled within the service time δ_i. The total demand cannot exceed the vehicle maximum capacity Q. The edges (i, j) $i \neq j$ have non-negative distances between the point i and j and are associated with the matrix $C = (c_{ij})$ that represents the cost for traveling on the arc (i, j) obeying to the triangle inequality, $c_{ij} < c_{ik} + c_{kj}$ for all $i, j, k \in V$.

All routes start and end at V_0 and the duration of the route cannot exceed the maximum duration (a large penalty will be used).

In the VRPTW, the service can end after the end of the time window. There is only a penalty if the services starts after the end of the time window. The time window start will always be respected but the end can be crossed. This means that the service should start before the time window ends, but it can finish afterwards, the end of this service can cross the end of the time window without penalties.

If the vehicle arrives too late and the completion of the service is after the closing of the time window, the route cost will not be penalized but it creates a total route delay making the route less desirable. The contrary case where the service starts after the time window end will be penalized being this penalty proportional to crossed time.

The vehicles are also allowed to arrive before the time window starts. In this case the start of the service can only happen after the opening of the time window. This waiting time is allowed but not desirable, because it also delays the total route duration.

In some papers, the number of vehicles is also subjected to optimization but this is not the case here. The objective of this version of VRPTW and SVRPTW is to minimize the transportation costs and penalties.

The traveled distance is calculated using the normal Euclidean metric and the travel time based on the distance.

In the deterministic version, one unit of distance traveled is one unit of time elapsed and the route duration is calculated based on the traveled distance, waiting time and deterministic service duration. The difference for the stochastic version is that the travel and service times will be randomized and an expected value will be used. They will be calculated with random percent of the deterministic values. The total route duration will be the expected travel time, the waiting time and the expected service duration together.

For both cases, the total route duration should be shorter than the depot horizon.

In the deterministic version, all values are known. Travel and service time are realized beforehand, contrary to the stochastic. In the stochastic case, the traveled and service time will be an expected value and the routes will be created considering time fluctuations.

For the stochastic formulation, soft windows are necessary and in this case, a large penalty is added to the objective, if the service start crosses the time window end. The soft windows model is general and also includes the hard time windows case. It is possible to transform one into another by changing the penalties.

While conducting tests for this work, it was noticeable that in the deterministic case, high penalties created only feasible routes. These high penalties benefit the search for feasible routes, but in the real world there are too many unpredictable factors. It would be impossible to create routes that will always be feasible and for this reason, the stochastic VRPTW has an important role.

The advantage of SVRPTW is that it minimizes the distance considering the expected penalty. This may not create routes as short as in the deterministic due to the variability of the time, but these routes will handle the delays in the travel or service time better. There has to be a trade-off between traveled distance and expected costs.

The following notation used in the formulation of the problems.

m : number of vehicles used in the routes.

N : total number of customers.

K : set of total available vehicles. $K = \{1, 2, ..., m\}$.

n_k : number of customers being served by the vehicle k.

R : the set of routes of the solution. Each route is done by one vehicle and each customer belongs to one route. The route is represented by $R = \{R_k | R_k = \{r_0 = 0, r_1, r_2, \ldots, r_{n_k+1} = 0\}\}, k = 1, 2, \ldots, m$ and $r_j \in K\}$ where for the route R_i the j visited customer is r_j.

f : penalty for not serving a customer.

Q : maximum capacity of a vehicle that is equal for all vehicles.

q_i : customer $i \in V$ demand. The demand of the depot is 0.

c_{ij} : costs to travel between vertexes i and j for all $i, j \in V$. This cost is associated with the distance.

u_i : the deterministic time to serve customer i used for the deterministic case.

$[e_i, l_i]$: time window in which the vehicle should arrive. $i, l \in V$. e_i is the earliest and l_i the latest time the vehicle may arrive. Arriving after l_i will penalize the route. The service can be started as soon as in e_i and can cross l_i. $[e_0, l_0]$ at the depot is called the scheduling horizon.

λ_i : penalty for late arrivals per vehicle.

P_k : total penalties added to the route k, caused by late arrivals.

w_i : waiting time if the vehicle arrives before e_i else $w_i = 0$. The vehicle has to wait until e_i in order to start the service.

τ_{ij} : time to travel between vertexes i and j. The random variable for the stochastic version follows a uniform continuous distribution, i.e. $\tau_{ij} \sim \mathcal{U}(a_{ij}, b_{ij})$.

δ_i : the stochastic time to serve customer i used for the stochastic case. It is a random variable following an uniform continuous distribution, i.e. $\delta_i \sim \mathcal{U}(a_i, b_i)$.

s_{ik} : arrival time of vehicle k at customer i.

x_{ijk} : binary variable that defines whether or not i is connected to j for the vehicle k.

The mathematical formulation for the deterministic VRPTW:

$$\begin{aligned} \min \qquad & F_0(x) = f \cdot (N - \sum_{k \in K} n_k) + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} x_{ijk} \\ & + \cdot \sum_{i \in V} \sum_{k \in K} \lambda_i \cdot max(s_{ik} - l_i, 0) \end{aligned} \tag{2.1}$$

$$\text{s.t.} \quad \sum_{j \in V} \sum_{k \in K} x_{ijk} = 1 \qquad \forall i \in V \setminus 0 \tag{2.2}$$

$$\sum_{j \in V} x_{0jk} = 1 \qquad \forall k \in K \tag{2.3}$$

$$\sum_{i \in V} x_{i0k} = 1 \qquad \forall k \in K \tag{2.4}$$

$$\sum_{i \in V \setminus 0} x_{ijk} - \sum_{i \in V \setminus 0} x_{jik} = 0 \qquad \forall j \in V \setminus 0, k \in K \tag{2.5}$$

$$\sum_{i \in V \setminus 0} q_i \sum_{j \in V} x_{ijk} \leq Q \qquad \forall k \in K \tag{2.6}$$

$$x_{ijk} = 1 \implies max(e_i, s_{ik}) + u_i + c_{ij} = s_{jk} \qquad \forall i, j \in V, k \in K \tag{2.7}$$

$$x_{ijk} \in \{0, 1\} \qquad \forall i, j \in V, k \in K \tag{2.8}$$

The equation (2.1) is the objective function. The first part is done to ensure that all customers will be served. A penalty is added, if there are any customers that are not in any of the routes.

The second part represents the costs of traveling from i to j, if the arc is assigned to a route and the last part represents the penalty for late arrivals. This lateness is calculated with the difference between the arrival and the time window end, if the arrival is after the time window end. The equation (2.2) defines that in one vertice, there is only one arc in and one out, modeling the case where only one vehicle can serve a customer and all costumers must be served. The two following (2.3) and (2.4) ensure that from the starting and ending vertice, there is exactly one arc out and one in. This means that all vehicles start and end at the depot. The equation (2.5) ensures that there is a connected path in the route. If there is a connection from i to j there is a connection from j to i. (2.6) is the condition that limits the capacity of the vehicle i to the max capacity Q. The constraint (2.7) can be computed recursively and eliminates the subtours, making the route feasible by creating a connected path between the customers. Here the distance c_{ij} is used to represent the deterministic time.

The decision variable x_{ijk} (2.8) is binary variable that is one when for the vehicle k the vertice i is connected with j.

The formulation of the SVRPTW:

$$\min_x \qquad \left[f \cdot (N - \sum_{k \in K} n_k) + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} x_{ijk} + E(P(x, \delta, \tau)) \right] \tag{2.9}$$

$$\text{s.t. } (2.2) - (2.6) \text{ and } (2.8) \tag{2.10}$$

$$P(x, \delta, \tau) = min \sum_{k \in K} \sum_{i \in V} \lambda_i \max(s_{ik} - l_i, 0) \tag{2.11}$$

$$x_{ijk} = 1 \implies max(e_i, s_{ik}) + \delta_i + \tau_{ij} = s_{jk} \qquad \forall i, j \in V, k \in K \tag{2.12}$$

In the stochastic formulation (2.9) (2.11), the random variable P_k penalizes the late arrivals. It is calculated proportionally to the difference between vehicle arrival time after the time window end and the time window end.

The last equation (2.7) now is constructed with the random variables for service time δ_i and for the travel time τ_{ij}.

Chapter 3

Large neighborhood search proposal

The LNS algorithm was initially presented by Shaw et al [19] as a process of continual relaxation and re-optimization [19]. The customers are removed and the routing plan is then re-optimized by re-inserting the removed customers "using the full power of constraint programming via constraint propagation and heuristics" [19].

In the proposed large neighborhood search, LNS, we start with an initial solution that will be destroyed and repaired in a way to lead to an improvement of the objective. This is done with a fixed amount of iterations and each time, it is decided if the new solution is accepted or not. Better solutions will always be accepted and worse solutions will be accepted with a certain probability.

LNS algorithm:

1. Generate a starting solution s; $s \rightarrow s_{best}$
2. Repeat:
   (a) Apply the destroy operator d to s creating s'
   (b) Apply the repair operator r to s' creating s''
   (c) If s'' is better than s_{best} then s_{best}=s''. If not accept s'' as s_{best} with a certain probability.
3. until x iterations

4. Return s_{best}

The algorithms used for the destroy operator are random removal, worse removal and a combination of both and for the repair operator greedy, greedy regret and a combination of both.

I $\rightarrow$ random removal - worst removal - greedy - greedy regret

II $\rightarrow$ worst removal - greedy - greedy regret

III $\rightarrow$ random removal - greedy - greedy regret

IV $\rightarrow$ random removal - worst removal - greedy regret

V $\rightarrow$ worst removal - greedy regret

VI $\rightarrow$ random removal - greedy regret

VII $\rightarrow$ random removal - worst removal - greedy

VIII $\rightarrow$ worst removal - greedy

IX $\rightarrow$ random removal - greedy

After every iteration, the new solution costs will be evaluated. If the costs are smaller, the new solution will be accepted as the global best and if the costs are greater than the new solution, it will be accepted based on the simulated annealing method [18].

$$e^{-\left(\frac{-(f(s'')-f(s))}{\hat{t}}\right)} \tag{3.1}$$

This method (3.1) adapts the probability to how "good" a bad solution is and decreases the acceptance probability with the increase of iterations. It was used to input randomness and consequently help the algorithm to avoid local minimums in case it gets trapped.

It is defined based on the actual best solution $f(s)$ and the last solution $f(s'')$ that will be worse than $f(s)$.

The temperature parameter $\hat{t}$ [4] is calculated using the initial solution $f_c(s)$ and the parameter $\omega_{\hat{t}} = 0,05$.

$$\hat{t} = -\frac{\omega_{\hat{t}}}{\ln 0,5} \cdot f_c(s) \tag{3.2}$$

In every iteration, the temperature $\hat{t}$ is multiplied by a decrease parameter. In the test conducted the LNS performed best with the parameter equal to 0.997.

The values of the function $f(s)$, the objective function value, correspond to the costs CST of the route set RSET. In the used algorithm, these costs were calculated summing the total route distance plus a penalty factor times the arrival time minus time window end for late arrivals, plus penalty for customers in the pool of non-served customers P, plus penalty if demand exceeds the vehicle supply.

Destroy and repair algorithms:

- Destroy operators:
  - Worst removal
    * Repeat n times to remove n customers from the route set RSET.
    * Start with route set RSET and the associated cost CST.
    * for all the customers in the route set RSET.
      Get the costumer x that deteriorate CST the least if x is removed
    * Remove x from the route set RSET and add x to the pool of non-served customers P.
  - Random removal
    * Repeat n times to remove n customers from the route set RSET.

    * Start with route set RSET and the associated cost CST.
    * for all the customers in the route set RSET.
      Get one random customer x.
    * Remove x from the route set RSET and add x to the pool of non-served customers P.

- Repair operators:
  - Greedy heuristic
    * Repeat n times to insert n customers from the pool of non-served customers P.
    * Start with route set RSET, the associated cost CST and the pool of non-served customers P.
    * Get one costumer from P (in the inverse order that they were removed).
    * Insert x in all positions of RSET and get the position PID and route RID for which the cost CST decreases the most. (The costs CST will decrease with the insertion of a new customer due to the penalties for non-served customers. It is better to add the customer to the route than to keep it in P.)
    * Insert x in the route RID at the position PID.
  - Greedy regret heuristics
    * Repeat n times to insert n customers from the pool of non-served customers P.

* Start with route set RSET, with the associated cost CST and the pool of non-served customers P.

* Repeat for all customers x in P.
  Insert x in all positions of RSET and build a matrix M with the inserted positions PID, routes RID and the associated costs CST.
  Find in M the smallest costs CST1 (associated with a RID1) and the second smallest CST2 (associated with a RID2) where RID1 $\neq$ RID2.
  Calculate the difference DIFF from CST2 minus CST1.
* Get customer x* for which the value DIFF is the biggest.
* Insert x* in the associated route RID at the position PID.

Example of an execution:

The algorithm was initialized with greedy to first populate the route set RSET1 and the costs were stored in CST1. This initial RSET1 was then destroyed and repaired using one of the I-IX combinations. When two destroy operators were used, the selection between the two was done randomly with a probability of 0.5 for each, being the same principle used in the repair.

In the destroy phase, the number of removed customers was set randomly aswell. A number between 10 and 40 percent of the customers were removed. This 50% chance of algorithm selection and 10% to 40% customers to remove was implemented to give more randomness to the algorithm and by that to open the search space.

After this process, a new route set RSET2 was created with the new costs CST2. If the new CST2 is smaller than CST1, the new route set RSET2 becomes the best route. If CST2 is greater, the new route set RSET2 will be accepted with a probability. This was done generating a random number, RAND, between 0 and 1 and CST2 is accepted if RAND is less than $e^{-\left(\frac{-(f(RSET2)-f(RSET1))}{\hat{t}}\right)}$.

The destroy and repair process was repeated for 2500 iterations in the deterministic and 3000 in the stochastic case, for five seeds, for each instance. These five seeds calculation for each instance was used because of the several random parameters. The final cost for one instance is the average for the five seeds.

Chapter 4

Implementation

An implementation was done in c++ and executed in a single core at 2,2 Ghz. Instances R1, C1, RC1, R2, C2, RC2 from Professor Solomon[20] with 100 customers each were used. The instance R1 and R2 are randomly generated, C1 and C2 clustered and RC1 and RC2 a mix of random and clustered. The instances R1, C1 and RC1 (100's) have a shorter scheduling horizon creating smaller number of customers per route in opposite of R2, C2 and RC2 (200's) that allow more customers per route with a larger scheduling horizon. The solutions are described in the Appendix A and the results are from a two-stage algorithm, where first the number of vehicles is minimized and then the travel time.

In this work, only the distance is minimized and the minimization of the number of vehicles is not considered. Also, the maximum working hours for a driver was set large enough not to constrain the problem and allow to freely construct routes.

Three stages were implemented. In the first, a comparison between the destroy and repair combinations was done and the best combination was selected with the respective created routes. In the second stage, a stochastic version of the algorithm was used with the best destroy and repair combination from the stage one.

In the third and last stage, a comparison of performance for 1000 random instances was done for both the deterministic and the stochastic algorithms in order to compare how the created routes and expected costs perform under time variability.

The used algorithms combination:

I → random removal - worst removal - greedy - greedy regret

II → worst removal - greedy - greedy regret

III → random removal - greedy - greedy regret

IV → random removal - worst removal - greedy regret

V → worst removal - greedy regret

VI → random removal - greedy regret

VII → random removal - worst removal - greedy

VIII → worst removal - greedy

IX → random removal - greedy

In the first stage, to determine the best algorithm combination, the instances from Professor Solomon[20] were loaded without any change. The LNS was executed for all 56 instances and for each, 5 different seeds were used. This was implemented because the algorithm uses several random numbers and we wanted to have an average for the total route costs.

The implementation was done using penalties to force the algorithm to create, if possible, a feasible solution. To assign all customers a penalty of 1000000 was added to the costs for every customer that is not assigned to a route. The penalty used for the time window delay was also rather large, 100000, to force the program to create feasible solutions. This was done to compare the algorithms performance with the optimum solution found in the literature.

In all combinations I-IX, the calculated best routes are feasible and it will be shown in the results chapter that the combination VI is the best in average for all instances, deviating from the optimum in average 3.04%, see Table (4.1).

instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
avg all instances:	3,59%	24,32%	3,23%	3,41%	23,96%	3,04%	4,56%	28,69%	4,14%
Devided per group of problems									
100's	2,99%	23,26%	2,67%	2,30%	22,77%	2,03%	4,95%	28,10%	4,70%
200's	4,23%	25,47%	3,84%	4,60%	25,25%	4,12%	4,14%	29,33%	3,53%
R100	3,07%	24,61%	2,66%	2,41%	24,33%	2,14%	4,79%	30,61%	4,10%
C100	1,35%	16,23%	1,43%	0,61%	16,74%	0,56%	3,22%	17,04%	4,39%
RC100	4,71%	29,14%	4,08%	4,05%	27,22%	3,51%	7,13%	36,77%	5,93%
R200	4,95%	30,36%	4,17%	5,47%	31,05%	4,54%	4,88%	35,05%	3,74%
C200	2,69%	9,20%	2,14%	2,46%	8,05%	2,07%	2,36%	11,33%	2,52%
RC200	4,80%	35,01%	5,08%	5,55%	34,47%	5,60%	4,90%	39,47%	4,27%

Table 4.1: Deterministic: average traveled distance deviation from the optimum divided per groups of problems

In the second stage, a stochastic analysis took place for the combination VI with 2 new parameter matrices, one for the travel time TP and one for the service time SP. These parameter matrices were randomized and filled with values from 0.9 to 1.1. Each position is associated with each customer connection i-j in the case of TP and each customer i in the case of SP. This means that every time an access to the traveled time was required by the algorithm the associated entry in SP was multiplied by the traveled distance, simulating a random travel time. The same process happened for the service time, multiplying it by the parameter SP simulating the stochastic service time for a given customer i.

The stochastic algorithm was kept mostly equal to the deterministic version but with a new objective function. The main difference was in the route cost calculation.

Every time the algorithm requests a route cost, the function will simulate 10 new different scenarios with the 10 new matrices TP and 10 new matrices SP, creating 10 possible costs for the same route. These 10 scenarios will be the same for all the iterations being the outputted cost the average of these 10 scenarios. This implementation was done to include the stochastic nature of the problem.

The algorithm will run equally for 5 seeds for all the 56 initial instances, but for 3000 iterations and the penalty for time window delay was set to a lower value of 10. This was done to favor the arrival time after the window closing and be more flexible in the route creation. In this stage, the created routes were not all feasible; around 11% of the routes were unfeasible.

In the third stage, the routes calculated with the combination VI in the stage one and two were subjected to an expected cost evaluation. 1000 new time travel TP and 1000 new service time SP matrices were generated.

The best routes from the stage one and two were both loaded and a travel simulation for each of the 1000 scenarios completed. For each scenario, the trucks traveled through all the routes and the costs and penalties saved.

The expected cost for each instances of Professor Solomon[20] was calculated considering the average of the new 1000 calculated cost.

This stage was executed with a time window delay penalty of 10 and, as expected, several of the new 1000 instances make the calculated routes unfeasible, 52% for the deterministic and 28% for the stochastic algorithm.

Chapter 5

Results

The following Tables 5.1 to 5.7 show the results for an average of 5 seeds and the Tables 5.8 to 5.11 an average of 1000 scenarios. The description of the calculated best values from the literature, such as the authors and algorithms, can be found in the Appendix A and [17]. More information about the Authors abbreviations can also be found in the Appendix A.

In the Table 5.1, we can see the percentage of the difference (deviation) from the optimum. 5.2 show the traveled distance deviation against the best heuristic found in [20] (used only as a reference because the online heuristics algorithms minimize the NV first and therefor not directly comparable). Table 5.3 displays the average duration for one run of the algorithm (for one seed). This duration is influenced by how many processes were running on the computer and by the CPU usage. In 5.4, a minimum distance comparison (the minimum route distance calculated between all the 5 seeds) is shown instead of an average and was also used to benchmark the algorithms. In Table 5.5 and 5.6, we can see the number of used vehicles by the routes calculated. This number was never optimized and is just shown to serve as a reference. The heuristic results minimize the number of vehicles first and therefor are not directly comparable.

The percentage of waiting time in the whole route is presented in 5.7. This is an important value for the total route cost. Smaller waiting time indicates that the route is more compact making the travel time more efficient.

The last four Tables 5.8 - 5.11 show a comparison between the deterministic algorithm and the stochastic when testing the calculated route for 1000 new scenarios.

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	2,24%	13,01%	1,28%	1,14%	10,35%	1,24%	2,18%	22,15%	2,27%
2	R102-100	1,78%	13,75%	1,23%	1,20%	11,69%	1,72%	3,18%	16,12%	2,78%
3	R103-100	3,25%	26,58%	1,82%	1,96%	27,12%	1,78%	3,98%	28,26%	3,73%
4	R104-100	4,25%	25,57%	3,41%	3,57%	27,92%	2,34%	5,71%	34,85%	4,16%
5	R105-100	2,06%	23,54%	2,54%	0,90%	20,83%	1,96%	5,27%	26,06%	3,02%
6	R106-100	2,49%	22,81%	2,39%	2,31%	22,05%	1,22%	4,20%	24,98%	3,95%
7	R107-100	2,68%	23,62%	2,52%	2,24%	25,22%	1,85%	5,03%	26,92%	3,50%
8	R108-100	4,34%	28,30%	3,77%	3,99%	25,71%	3,67%	5,28%	37,46%	4,86%
9	R109-100	3,38%	28,75%	2,71%	1,73%	26,15%	1,44%	4,93%	33,85%	5,15%
10	R110-100	4,72%	29,87%	4,25%	3,28%	32,20%	3,26%	5,88%	38,91%	5,10%
11	R111-100	2,16%	29,85%	1,26%	2,51%	30,29%	1,95%	5,22%	34,85%	4,94%
12	R112-100	3,48%	29,65%	4,69%	4,12%	32,40%	3,23%	6,61%	42,90%	5,78%
13	C101-100	0,20%	0,20%	1,57%	0,20%	0,20%	0,20%	1,53%	0,20%	2,09%
14	C102-100	2,01%	21,26%	0,20%	0,77%	24,41%	0,91%	1,59%	20,51%	6,79%
15	C103-100	2,36%	44,11%	4,22%	0,21%	39,61%	2,20%	2,91%	45,81%	6,33%
16	C104-100	0,95%	26,08%	2,67%	0,34%	31,33%	0,23%	6,97%	26,07%	6,59%
17	C105-100	1,02%	11,50%	0,20%	2,37%	11,10%	0,20%	2,81%	13,60%	4,92%
18	C106-100	0,81%	23,12%	0,75%	0,99%	22,28%	0,75%	0,68%	24,38%	3,00%
19	C107-100	1,86%	3,07%	1,75%	0,20%	3,07%	0,20%	4,80%	3,07%	2,92%
20	C108-100	0,22%	8,74%	0,57%	0,20%	11,26%	0,20%	1,67%	11,78%	3,05%
21	C109-100	2,73%	7,99%	0,95%	0,20%	7,43%	0,20%	6,00%	7,99%	3,85%
22	RC101-100	2,45%	14,87%	2,67%	2,93%	14,95%	2,74%	3,97%	23,67%	3,46%
23	RC102-100	2,85%	22,11%	2,32%	2,71%	21,54%	2,21%	4,72%	33,43%	3,62%
24	RC103-100	6,58%	38,71%	6,97%	6,24%	32,61%	4,98%	8,46%	45,69%	7,37%
25	RC104-100	5,20%	29,91%	4,43%	3,50%	27,62%	3,60%	7,19%	42,82%	6,73%
26	RC105-100	4,92%	21,62%	3,38%	4,98%	18,99%	3,81%	5,97%	25,81%	3,91%
27	RC106-100	4,15%	30,90%	3,25%	3,06%	29,62%	2,51%	5,42%	36,43%	4,08%
28	RC107-100	6,33%	40,08%	5,77%	4,74%	38,87%	4,26%	9,55%	43,16%	7,86%
29	RC108-100	5,16%	34,89%	3,86%	4,23%	33,57%	3,94%	11,77%	43,20%	10,42%
30	R201-100	3,73%	29,79%	3,61%	4,04%	28,01%	3,63%	4,55%	34,63%	2,89%
31	R202-100	4,77%	35,92%	3,59%	7,01%	34,08%	4,97%	3,95%	40,13%	2,82%
32	R203-100	4,31%	25,67%	4,92%	4,68%	24,88%	4,19%	2,86%	30,27%	3,72%
33	R204-100	3,27%	22,75%	2,09%	5,08%	20,67%	3,07%	4,05%	32,07%	3,07%
34	R205-100	2,73%	20,79%	2,76%	4,77%	27,52%	4,86%	4,14%	23,53%	4,00%
35	R206-100	5,87%	29,34%	5,06%	7,46%	29,22%	5,69%	5,07%	33,70%	4,13%
36	R207-100	8,64%	29,97%	6,78%	6,66%	37,47%	5,43%	6,27%	34,85%	4,30%
37	R208-100	5,40%	25,79%	4,38%	5,58%	25,42%	5,06%	5,33%	34,81%	4,31%
38	R209-100	3,84%	25,65%	4,32%	4,37%	27,65%	5,34%	6,06%	29,28%	3,93%
39	R210-100	4,10%	36,98%	2,71%	3,41%	38,67%	2,02%	4,13%	39,33%	3,24%
40	R211-100	7,80%	51,37%	5,63%	7,06%	47,93%	5,74%	7,22%	52,97%	4,69%
41	C201-100	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%
42	C202-100	0,42%	6,38%	0,42%	0,42%	5,29%	0,42%	0,42%	6,38%	0,42%
43	C203-100	7,53%	26,21%	3,98%	5,36%	15,49%	5,00%	2,96%	26,67%	1,98%
44	C204-100	2,00%	11,89%	2,59%	1,73%	11,52%	2,52%	3,60%	20,43%	4,48%
45	C205-100	3,66%	13,85%	2,19%	3,97%	14,02%	0,42%	3,65%	14,12%	3,47%
46	C206-100	6,59%	9,04%	6,64%	6,00%	10,34%	6,00%	6,96%	16,68%	4,64%
47	C207-100	0,43%	4,81%	0,43%	0,42%	5,37%	0,42%	0,43%	4,81%	1,71%
48	C208-100	0,43%	0,99%	0,43%	1,35%	1,95%	1,35%	0,43%	1,12%	3,01%
49	RC201-100	3,65%	53,99%	2,86%	5,31%	48,67%	3,98%	4,65%	59,08%	3,72%
50	RC202-100	4,90%	33,42%	6,14%	4,09%	37,10%	7,00%	2,40%	37,80%	2,77%
51	RC203-100	6,55%	34,48%	5,48%	5,56%	41,36%	5,23%	6,23%	37,68%	4,86%
52	RC204-100	4,66%	25,19%	6,63%	6,93%	23,71%	8,18%	3,72%	28,85%	3,82%
53	RC205-100	3,93%	27,53%	3,05%	3,74%	25,68%	2,87%	4,69%	34,49%	4,16%
54	RC206-100	3,78%	30,41%	4,03%	4,49%	33,92%	3,70%	5,15%	30,53%	4,07%
55	RC207-100	4,09%	23,29%	5,72%	4,29%	21,73%	4,63%	5,48%	29,34%	4,83%
56	RC208-100	6,85%	51,75%	6,74%	10,04%	43,57%	9,26%	6,85%	58,02%	5,93%
	avg all instances:	3,59%	24,32%	3,23%	3,41%	23,96%	3,04%	4,56%	28,69%	4,14%
	Devided per group of problems									
	100's	2,99%	23,26%	2,67%	2,30%	22,77%	2,03%	4,95%	28,10%	4,70%
	200's	4,23%	25,47%	3,84%	4,60%	25,25%	4,12%	4,14%	29,33%	3,53%
	R100	3,07%	24,61%	2,66%	2,41%	24,33%	2,14%	4,79%	30,61%	4,10%
	C100	1,35%	16,23%	1,43%	0,61%	16,74%	0,56%	3,22%	17,04%	4,39%
	RC100	4,71%	29,14%	4,08%	4,05%	27,22%	3,51%	7,13%	36,77%	5,93%
	R200	4,95%	30,36%	4,17%	5,47%	31,05%	4,54%	4,88%	35,05%	3,74%
	C200	2,69%	9,20%	2,14%	2,46%	8,05%	2,07%	2,36%	11,33%	2,52%
	RC200	4,80%	35,01%	5,08%	5,55%	34,47%	5,60%	4,90%	39,47%	4,27%

Table 5.1: Deterministic: average traveled distance deviation from optimum

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	1,73 %	12,45 %	0,78 %	0,64 %	9,80 %	0,74 %	1,67 %	21,55 %	1,76 %
2	R102-100	0,44 %	12,25 %	-0,10 %	-0,12 %	10,23 %	0,38 %	1,82 %	14,60 %	1,43 %
3	R103-100	-3,46 %	18,36 %	-4,80 %	-4,66 %	18,86 %	-4,83 %	-2,77 %	19,93 %	-3,01 %
4	R104-100	0,56 %	21,11 %	-0,26 %	-0,10 %	23,38 %	-1,29 %	1,96 %	30,07 %	0,46 %
5	R105-100	0,44 %	21,58 %	0,91 %	-0,70 %	18,91 %	0,34 %	3,60 %	24,06 %	1,39 %
6	R106-100	1,07 %	21,11 %	0,96 %	0,89 %	20,36 %	-0,18 %	2,75 %	23,24 %	2,51 %
7	R107-100	-1,04 %	19,13 %	-1,20 %	-1,47 %	20,68 %	-1,85 %	1,22 %	22,32 %	-0,25 %
8	R108-100	1,21 %	24,46 %	0,67 %	0,87 %	21,94 %	0,56 %	2,12 %	33,35 %	1,72 %
9	R109-100	-0,76 %	23,60 %	-1,40 %	-2,34 %	21,10 %	-2,62 %	0,73 %	28,49 %	0,94 %
10	R110-100	-0,01 %	24,00 %	-0,46 %	-1,39 %	26,22 %	-1,41 %	1,09 %	32,63 %	0,35 %
11	R111-100	-2,32 %	24,17 %	-3,17 %	-1,98 %	24,58 %	-2,51 %	0,62 %	28,94 %	0,34 %
12	R112-100	-0,06 %	25,22 %	1,11 %	0,56 %	27,88 %	-0,29 %	2,96 %	38,02 %	2,16 %
13	C101-100	0,00 %	0,00 %	1,37 %	0,00 %	0,00 %	0,00 %	1,33 %	0,00 %	1,88 %
14	C102-100	1,80 %	21,02 %	0,00 %	0,57 %	24,16 %	0,71 %	1,39 %	20,27 %	6,58 %
15	C103-100	2,15 %	43,80 %	4,00 %	0,00 %	39,32 %	1,98 %	2,69 %	45,50 %	6,11 %
16	C104-100	0,72 %	25,79 %	2,43 %	0,11 %	31,03 %	0,00 %	6,72 %	25,78 %	6,35 %
17	C105-100	0,82 %	11,28 %	0,00 %	2,17 %	10,88 %	0,00 %	2,61 %	13,37 %	4,71 %
18	C106-100	0,61 %	22,88 %	0,55 %	0,79 %	22,04 %	0,55 %	0,48 %	24,13 %	2,79 %
19	C107-100	1,66 %	2,87 %	1,55 %	0,00 %	2,87 %	0,00 %	4,59 %	2,87 %	2,71 %
20	C108-100	0,02 %	8,53 %	0,37 %	0,00 %	11,04 %	0,00 %	1,46 %	11,56 %	2,85 %
21	C109-100	2,52 %	7,77 %	0,75 %	0,00 %	7,22 %	0,00 %	5,79 %	7,77 %	3,65 %
22	RC101-100	-2,21 %	9,65 %	-1,99 %	-1,75 %	9,72 %	-1,93 %	-0,76 %	18,05 %	-1,24 %
23	RC102-100	-3,59 %	14,46 %	-4,09 %	-3,72 %	13,93 %	-4,19 %	-1,84 %	25,07 %	-2,87 %
24	RC103-100	6,27 %	38,31 %	6,66 %	5,93 %	32,22 %	4,68 %	8,15 %	45,27 %	7,06 %
25	RC104-100	4,91 %	29,54 %	4,13 %	3,21 %	27,26 %	3,31 %	6,89 %	42,42 %	6,43 %
26	RC105-100	-2,53 %	12,99 %	-3,96 %	-2,48 %	10,54 %	-3,56 %	-1,55 %	16,87 %	-3,47 %
27	RC106-100	0,35 %	26,12 %	-0,52 %	-0,71 %	24,88 %	-1,23 %	1,57 %	31,45 %	0,27 %
28	RC107-100	4,37 %	37,50 %	3,82 %	2,81 %	36,31 %	2,33 %	7,53 %	40,52 %	5,87 %
29	RC108-100	2,80 %	31,85 %	1,53 %	1,89 %	30,57 %	1,61 %	9,26 %	39,98 %	7,93 %
30	R201-100	-5,32 %	18,48 %	-5,42 %	-5,03 %	16,85 %	-5,41 %	-4,56 %	22,90 %	-6,08 %
31	R202-100	-9,48 %	17,43 %	-10,50 %	-7,55 %	15,84 %	-9,31 %	-10,19 %	21,07 %	-11,17 %
32	R203-100	-3,32 %	16,47 %	-2,76 %	-2,98 %	15,74 %	-3,44 %	-4,66 %	20,74 %	-3,87 %
33	R204-100	-8,51 %	8,74 %	-9,56 %	-6,91 %	6,90 %	-8,70 %	-7,83 %	16,99 %	-8,69 %
34	R205-100	-1,88 %	15,37 %	-1,85 %	0,07 %	21,80 %	0,16 %	-0,53 %	17,99 %	-0,67 %
35	R206-100	2,33 %	25,02 %	1,56 %	3,87 %	24,91 %	2,16 %	1,56 %	29,24 %	0,65 %
36	R207-100	-3,44 %	15,52 %	-5,09 %	-5,20 %	22,18 %	-6,29 %	-5,55 %	19,85 %	-7,30 %
37	R208-100	1,69 %	21,36 %	0,71 %	1,86 %	21,01 %	1,37 %	1,63 %	30,07 %	0,64 %
38	R209-100	-2,37 %	18,14 %	-1,92 %	-1,87 %	20,02 %	-0,96 %	-0,28 %	21,55 %	-2,28 %
39	R210-100	-0,20 %	31,31 %	-1,53 %	-0,87 %	32,94 %	-2,20 %	-0,18 %	33,57 %	-1,03 %
40	R211-100	-9,83 %	26,62 %	-11,65 %	-10,45 %	23,73 %	-11,56 %	-10,31 %	27,95 %	-12,43 %
41	C201-100	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %
42	C202-100	0,00 %	5,93 %	0,00 %	0,00 %	4,86 %	0,00 %	0,00 %	5,93 %	0,00 %
43	C203-100	7,08 %	25,69 %	3,54 %	4,92 %	15,01 %	4,57 %	2,53 %	26,14 %	1,55 %
44	C204-100	1,57 %	11,41 %	2,15 %	1,30 %	11,05 %	2,08 %	3,16 %	19,92 %	4,04 %
45	C205-100	3,22 %	13,37 %	1,76 %	3,53 %	13,54 %	0,00 %	3,21 %	13,64 %	3,04 %
46	C206-100	6,14 %	8,58 %	6,19 %	5,55 %	9,87 %	5,55 %	6,51 %	16,19 %	4,20 %
47	C207-100	0,00 %	4,37 %	0,00 %	0,00 %	4,92 %	0,00 %	0,00 %	4,37 %	1,28 %
48	C208-100	0,00 %	0,56 %	0,00 %	0,92 %	1,51 %	0,92 %	0,00 %	0,68 %	2,57 %
49	RC201-100	-7,04 %	38,11 %	-7,75 %	-5,55 %	33,34 %	-6,75 %	-6,14 %	42,67 %	-6,97 %
50	RC202-100	-16,19 %	6,60 %	-15,19 %	-16,83 %	9,54 %	-14,51 %	-18,19 %	10,10 %	-17,89 %
51	RC203-100	-6,23 %	18,35 %	-7,18 %	-7,11 %	24,40 %	-7,40 %	-6,52 %	21,16 %	-7,72 %
52	RC204-100	2,70 %	22,85 %	4,64 %	4,93 %	21,40 %	6,16 %	1,79 %	26,45 %	1,88 %
53	RC205-100	-7,55 %	13,45 %	-8,32 %	-7,71 %	11,81 %	-8,48 %	-6,86 %	19,64 %	-7,34 %
54	RC206-100	-4,84 %	19,58 %	-4,61 %	-4,19 %	22,80 %	-4,91 %	-3,58 %	19,68 %	-4,58 %
55	RC207-100	-5,55 %	11,87 %	-4,07 %	-5,37 %	10,46 %	-5,05 %	-4,28 %	17,37 %	-4,88 %
56	RC208-100	0,13 %	42,22 %	0,03 %	3,12 %	34,55 %	2,39 %	0,13 %	48,09 %	-0,72 %
	avg all instances:	-0,86 %	18,74 %	-1,20 %	-1,05 %	18,37 %	-1,40 %	0,09 %	22,89 %	-0,29 %
	100's	0,64 %	20,41 %	0,33 %	-0,03 %	19,93 %	-0,30 %	2,55 %	25,11 %	2,32 %
	200's	-2,48 %	16,94 %	-2,84 %	-2,13 %	16,70 %	-2,58 %	-2,56 %	20,52 %	-3,10 %
	R100	-0,18 %	20,62 %	-0,58 %	-0,82 %	20,33 %	-1,08 %	1,48 %	26,43 %	0,82 %
	C100	1,15 %	15,99 %	1,22 %	0,40 %	16,51 %	0,36 %	3,01 %	16,81 %	4,18 %
	RC100	1,30 %	25,05 %	0,70 %	0,65 %	23,18 %	0,13 %	3,66 %	32,45 %	2,50 %
	R200	-3,67 %	19,50 %	-4,37 %	-3,19 %	20,17 %	-4,02 %	-3,72 %	23,81 %	-4,75 %
	C200	2,25 %	8,74 %	1,71 %	2,03 %	7,59 %	1,64 %	1,93 %	10,86 %	2,08 %
	RC200	-5,6 %	21,6 %	-5,3 %	-4,8 %	21,0 %	-4,8 %	-5,5 %	25,6 %	-6,0 %

Table 5.2: Deterministic: average traveled distance deviation from heuristics (used only as a reference because the online heuristics algorithms minimize the NV first).

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	112,64	54,10	168,45	216,96	100,05	335,40	6,61	5,89	6,68
2	R102-100	108,83	51,71	164,06	212,90	95,70	326,14	6,42	5,64	6,31
3	R103-100	105,65	50,07	158,89	205,05	95,04	273,87	5,75	5,34	5,70
4	R104-100	103,78	49,23	153,49	186,45	94,39	244,00	5,12	4,69	5,05
5	R105-100	105,60	50,87	151,88	162,63	94,98	252,86	5,84	5,30	5,96
6	R106-100	104,70	51,67	123,72	161,20	94,55	248,24	5,60	5,11	5,60
7	R107-100	102,36	49,98	121,68	159,26	93,78	246,77	5,26	4,80	5,18
8	R108-100	86,30	48,94	119,30	156,87	90,69	242,94	5,08	4,66	4,86
9	R109-100	82,64	49,92	126,61	161,17	78,85	250,74	5,64	5,11	5,58
10	R110-100	81,76	49,55	125,83	158,58	74,52	246,27	5,30	4,92	5,31
11	R111-100	81,24	49,50	123,54	160,02	74,09	245,23	5,25	4,81	5,28
12	R112-100	80,45	49,48	123,56	158,56	74,19	242,20	5,09	4,74	5,17
13	C101-100	79,40	48,56	118,46	156,96	72,58	238,33	4,69	4,24	4,71
14	C102-100	80,65	48,82	121,67	156,52	72,79	240,05	4,69	4,38	4,87
15	C103-100	79,70	47,40	121,31	155,88	73,00	239,59	4,69	4,34	4,76
16	C104-100	78,78	43,23	118,12	154,71	72,38	237,72	4,56	4,16	4,65
17	C105-100	80,58	39,03	120,77	156,84	72,76	240,42	4,71	4,29	4,73
18	C106-100	80,04	38,58	124,41	157,59	73,58	241,81	4,71	4,32	4,84
19	C107-100	77,93	38,21	120,33	157,68	72,94	241,37	4,66	4,19	4,63
20	C108-100	79,17	38,28	120,30	155,10	73,17	241,15	4,62	4,40	4,76
21	C109-100	79,01	38,38	122,33	155,49	73,02	265,69	4,57	4,22	4,65
22	RC101-100	83,93	40,34	125,69	164,12	75,77	301,48	5,72	5,41	6,00
23	RC102-100	82,68	40,01	124,03	160,48	74,79	320,47	5,54	5,00	5,67
24	RC103-100	83,30	39,60	121,94	160,16	73,75	285,39	5,13	4,72	5,31
25	RC104-100	80,23	38,91	121,25	157,45	73,81	298,43	4,91	4,58	5,10
26	RC105-100	84,24	40,29	126,45	162,82	75,18	309,21	5,66	5,26	5,86
27	RC106-100	82,63	39,61	126,46	162,90	74,68	417,22	5,36	4,96	5,54
28	RC107-100	82,81	39,86	125,19	158,91	74,05	251,45	5,15	4,90	5,42
29	RC108-100	79,60	39,60	124,07	158,61	74,07	252,99	4,98	4,62	5,20
30	R201-100	76,92	38,13	120,54	156,90	72,61	232,65	4,34	3,90	4,32
31	R202-100	78,60	38,31	116,98	152,79	72,70	228,29	4,20	3,90	4,20
32	R203-100	77,29	38,02	117,05	168,56	72,08	228,38	4,14	3,91	4,11
33	R204-100	77,33	38,01	119,08	175,73	72,53	263,24	4,08	3,88	4,05
34	R205-100	77,26	38,11	117,81	181,45	72,09	295,62	4,14	3,83	4,16
35	R206-100	78,59	38,12	114,55	193,79	71,99	285,79	4,07	3,86	4,00
36	R207-100	77,40	38,13	116,70	189,54	73,01	270,13	4,02	3,96	3,94
37	R208-100	76,81	37,77	115,20	170,61	72,54	271,17	4,05	3,83	3,91
38	R209-100	79,33	37,88	115,69	178,63	72,28	287,58	4,10	3,80	4,07
39	R210-100	77,31	38,00	116,86	190,02	71,89	259,70	4,11	3,88	4,06
40	R211-100	77,75	37,85	116,02	182,31	71,62	208,40	3,97	3,78	4,02
41	C201-100	76,65	38,20	120,39	210,49	71,86	224,55	4,02	3,76	3,87
42	C202-100	77,70	38,10	133,44	283,02	71,78	224,24	3,99	3,76	3,97
43	C203-100	77,58	37,72	132,34	147,92	71,97	245,66	4,01	3,75	3,99
44	C204-100	76,45	37,71	130,40	155,34	71,99	231,04	4,00	3,68	3,96
45	C205-100	78,44	38,26	150,81	156,31	72,14	243,34	4,02	3,75	3,97
46	C206-100	77,40	37,84	144,78	154,50	71,92	259,93	4,04	3,82	4,01
47	C207-100	76,59	38,04	148,65	148,62	71,96	257,51	4,05	3,84	3,95
48	C208-100	76,32	38,02	133,69	146,06	72,17	264,40	4,07	3,75	4,04
49	RC201-100	78,97	38,15	136,65	150,14	72,51	267,83	4,37	3,90	4,49
50	RC202-100	78,96	38,23	141,87	148,73	72,67	259,76	4,30	3,87	4,23
51	RC203-100	76,32	37,78	145,27	155,62	72,06	251,16	4,11	3,82	4,16
52	RC204-100	76,29	38,13	144,13	187,78	72,47	257,12	3,95	3,85	4,03
53	RC205-100	77,97	38,17	140,84	188,16	72,32	240,04	4,33	4,07	4,29
54	RC206-100	79,21	37,95	226,73	196,14	72,15	216,02	4,05	3,85	4,07
55	RC207-100	76,58	37,92	180,68	183,73	72,28	282,94	4,04	3,88	4,12
56	RC208-100	76,27	37,81	114,35	173,91	71,87	285,54	3,96	3,79	4,02
	avg all instances:	83	42	131	170	76	261	5	4	5
	per iteration in seconds	0,03302	0,01664	0,05254	0,06799	0,03045	0,10442	0,00187	0,00173	0,00188
	per iteration in miliseconds	33,02079857	16,64355286	52,53770571	67,99058286	30,44708143	104,4248029	1,8704	1,733322857	1,881444286
	devided per group of problems in seconds									
	R100	96,33	50,42	138,42	174,97	88,40	262,89	5,58	5,08	5,56
	C100	79,47	42,28	120,85	156,31	72,91	242,90	4,66	4,28	4,73
	RC100	82,43	39,78	124,39	160,68	74,51	304,58	5,31	4,93	5,51
	R200	77,69	38,03	116,95	176,39	72,30	257,36	4,11	3,87	4,08
	C200	77,14	37,99	136,81	175,28	71,97	243,83	4,02	3,76	3,97
	RC200	77,57	38,02	153,81	173,03	72,29	257,55	4,14	3,88	4,18
	Devided per group of problems in minutes:									
	R100	1,61	0,84	2,31	2,92	1,47	4,38	0,09	0,08	0,09
	C100	1,32	0,70	2,01	2,61	1,22	4,05	0,08	0,07	0,08
	RC100	1,37	0,66	2,07	2,68	1,24	5,08	0,09	0,08	0,09
	R200	1,29	0,63	1,95	2,94	1,21	4,29	0,07	0,06	0,07
	C200	1,29	0,63	2,28	2,92	1,20	4,06	0,07	0,06	0,07
	RC200	1,29	0,63	2,56	2,88	1,20	4,29	0,07	0,06	0,07

Table 5.3: Deterministic: average execution duration in seconds

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	1,31%	11,67%	0,89%	0,75%	9,42%	0,96%	0,76%	20,23%	0,83%
2	R102-100	0,83%	12,17%	0,77%	0,58%	10,37%	1,27%	1,19%	15,97%	2,16%
3	R103-100	2,34%	22,21%	1,25%	1,04%	20,42%	1,11%	3,19%	27,34%	1,58%
4	R104-100	2,68%	15,89%	2,41%	2,01%	22,34%	1,72%	4,28%	24,15%	2,71%
5	R105-100	0,49%	18,02%	0,73%	0,73%	18,08%	1,00%	3,54%	18,37%	1,92%
6	R106-100	1,23%	20,46%	1,08%	1,41%	21,55%	0,39%	2,82%	23,00%	2,83%
7	R107-100	1,18%	21,96%	1,37%	1,26%	23,23%	1,27%	3,40%	24,22%	2,43%
8	R108-100	2,91%	23,62%	2,41%	2,61%	22,23%	2,92%	4,45%	29,68%	3,53%
9	R109-100	2,08%	25,24%	1,28%	1,35%	20,36%	0,66%	2,53%	27,89%	2,03%
10	R110-100	3,18%	20,24%	3,34%	2,02%	25,86%	1,63%	3,17%	34,20%	3,79%
11	R111-100	0,66%	23,29%	0,85%	0,70%	23,30%	0,80%	4,11%	32,76%	4,12%
12	R112-100	2,78%	23,23%	3,76%	1,29%	21,32%	1,35%	3,90%	35,39%	2,32%
13	C101-100	0,20%	0,20%	0,20%	0,20%	0,20%	0,20%	0,20%	0,20%	0,20%
14	C102-100	0,20%	20,51%	0,20%	0,20%	22,15%	0,20%	0,20%	20,51%	0,20%
15	C103-100	0,21%	41,33%	0,21%	0,21%	35,52%	0,21%	0,21%	43,54%	3,36%
16	C104-100	0,23%	18,55%	0,23%	0,23%	29,70%	0,23%	3,27%	17,27%	4,30%
17	C105-100	0,20%	9,88%	0,20%	0,20%	10,13%	0,20%	0,20%	10,75%	0,20%
18	C106-100	0,20%	19,73%	0,20%	0,20%	22,28%	0,20%	0,20%	24,07%	0,20%
19	C107-100	0,20%	3,07%	0,20%	0,20%	3,07%	0,20%	0,20%	3,07%	0,20%
20	C108-100	0,20%	7,02%	0,20%	0,20%	7,88%	0,20%	0,20%	11,78%	0,20%
21	C109-100	0,20%	7,99%	0,20%	0,20%	7,43%	0,20%	3,37%	7,99%	0,20%
22	RC101-100	1,49%	13,81%	2,38%	2,09%	14,00%	2,35%	2,73%	21,62%	2,45%
23	RC102-100	1,87%	19,42%	1,73%	2,14%	17,07%	2,00%	2,36%	27,15%	2,86%
24	RC103-100	5,15%	27,85%	5,97%	5,32%	26,77%	3,25%	7,00%	39,77%	5,15%
25	RC104-100	3,14%	26,92%	2,31%	1,22%	22,05%	2,78%	4,14%	37,84%	4,97%
26	RC105-100	2,82%	19,90%	2,01%	3,82%	16,07%	1,93%	3,72%	19,01%	2,16%
27	RC106-100	3,44%	24,69%	2,33%	1,12%	23,42%	1,03%	3,91%	32,09%	1,56%
28	RC107-100	4,92%	26,22%	4,15%	3,95%	28,70%	3,81%	5,42%	29,08%	6,76%
29	RC108-100	3,13%	28,82%	2,08%	2,61%	31,24%	0,45%	8,60%	37,76%	4,72%
30	R201-100	2,14%	27,87%	2,42%	2,61%	27,84%	2,94%	2,42%	33,56%	1,88%
31	R202-100	1,08%	33,54%	1,45%	4,13%	28,16%	3,85%	3,05%	35,30%	0,91%
32	R203-100	1,22%	23,72%	1,29%	2,05%	24,27%	3,45%	1,19%	26,04%	1,72%
33	R204-100	1,08%	19,40%	1,32%	3,44%	19,02%	2,80%	3,20%	30,48%	1,46%
34	R205-100	1,17%	18,73%	0,71%	3,40%	22,88%	2,10%	2,09%	18,73%	1,69%
35	R206-100	4,42%	26,54%	3,04%	6,28%	25,31%	4,01%	3,51%	27,50%	2,51%
36	R207-100	5,07%	28,75%	5,14%	3,20%	36,65%	3,45%	5,05%	33,49%	2,81%
37	R208-100	4,66%	23,65%	3,11%	2,93%	17,41%	4,19%	4,71%	29,57%	1,19%
38	R209-100	3,16%	21,65%	3,14%	1,34%	22,76%	2,86%	4,26%	24,68%	2,92%
39	R210-100	3,22%	31,65%	0,91%	2,29%	37,16%	1,33%	2,93%	38,68%	2,75%
40	R211-100	4,23%	43,15%	3,91%	1,92%	42,26%	3,89%	3,09%	45,37%	2,73%
41	C201-100	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%	0,42%
42	C202-100	0,42%	5,29%	0,42%	0,42%	5,29%	0,42%	0,42%	5,29%	0,42%
43	C203-100	0,42%	22,72%	0,42%	1,95%	14,94%	1,95%	0,42%	25,45%	0,42%
44	C204-100	0,42%	5,35%	0,42%	1,44%	11,14%	1,44%	0,42%	17,17%	0,42%
45	C205-100	0,42%	13,05%	0,42%	0,42%	13,05%	0,42%	0,42%	13,05%	0,42%
46	C206-100	0,43%	9,04%	0,43%	0,43%	9,04%	0,43%	0,43%	16,68%	0,43%
47	C207-100	0,42%	4,81%	0,42%	0,42%	5,37%	0,42%	0,42%	4,81%	0,42%
48	C208-100	0,43%	0,95%	0,43%	0,43%	0,95%	0,43%	0,43%	1,00%	0,43%
49	RC201-100	2,09%	39,51%	1,36%	3,69%	32,64%	2,55%	2,40%	57,99%	2,63%
50	RC202-100	2,35%	30,45%	3,86%	2,48%	34,99%	5,04%	1,11%	31,80%	1,37%
51	RC203-100	3,74%	29,26%	3,62%	3,24%	38,86%	2,10%	5,40%	30,22%	3,68%
52	RC204-100	2,53%	20,47%	2,55%	4,70%	22,47%	2,80%	0,59%	26,38%	2,68%
53	RC205-100	1,64%	23,41%	1,25%	1,11%	21,67%	1,80%	1,60%	29,14%	0,47%
54	RC206-100	2,52%	28,34%	3,67%	4,31%	31,56%	2,59%	4,19%	27,99%	2,93%
55	RC207-100	3,14%	21,96%	1,05%	3,37%	21,45%	3,99%	3,93%	21,55%	3,37%
56	RC208-100	2,40%	47,87%	2,63%	7,83%	33,38%	4,13%	3,81%	51,25%	4,46%
	avg all instances:	1,87%	20,63%	1,69%	1,97%	20,66%	1,79%	2,59%	25,00%	2,10%
	Devided per group of problems									
	100's	1,71%	19,10%	1,55%	1,37%	19,18%	1,19%	2,87%	24,02%	2,41%
	200's	2,05%	22,28%	1,85%	2,60%	22,26%	2,44%	2,29%	26,06%	1,76%
	R100	1,81%	19,83%	1,68%	1,31%	19,87%	1,26%	3,11%	26,10%	2,52%
	C100	0,20%	14,25%	0,20%	0,20%	15,37%	0,20%	0,89%	15,46%	1,01%
	RC100	3,25%	23,46%	2,87%	2,78%	22,42%	2,20%	4,74%	30,54%	3,83%
	R200	2,86%	27,15%	2,40%	3,05%	27,61%	3,17%	3,23%	31,22%	2,05%
	C200	0,42%	7,70%	0,42%	0,74%	7,53%	0,74%	0,42%	10,48%	0,42%
	RC200	2,6%	30,2%	2,5%	3,8%	29,6%	3,1%	2,9%	34,5%	2,7%

Table 5.4: Deterministic: deviation from the optimum of the minimum route distance calculated between all the 5 seeds

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	-1,0 %	9,0 %	-2,0 %	-2,0 %	10,0 %	-4,0 %	-1,0 %	18,0 %	0,0 %
2	R102-100	1,1 %	10,0 %	-1,1 %	1,1 %	7,8 %	0,0 %	1,1 %	11,1 %	1,1 %
3	R103-100	4,3 %	14,3 %	1,4 %	0,0 %	15,7 %	5,7 %	7,1 %	21,4 %	5,7 %
4	R104-100	0,0 %	16,4 %	3,6 %	0,0 %	10,9 %	0,0 %	5,5 %	20,0 %	5,5 %
5	R105-100	1,3 %	17,3 %	1,3 %	0,0 %	13,3 %	-1,3 %	4,0 %	21,3 %	4,0 %
6	R106-100	1,5 %	23,1 %	3,1 %	6,2 %	23,1 %	1,5 %	7,7 %	23,1 %	7,7 %
7	R107-100	5,5 %	18,2 %	5,5 %	1,8 %	25,5 %	5,5 %	10,9 %	21,8 %	9,1 %
8	R108-100									
9	R109-100	0,0 %	12,3 %	0,0 %	0,0 %	13,8 %	0,0 %	4,6 %	20,0 %	3,1 %
10	R110-100	3,3 %	13,3 %	0,0 %	0,0 %	13,3 %	-1,7 %	3,3 %	16,7 %	1,7 %
11	R111-100	1,7 %	15,0 %	0,0 %	0,0 %	11,7 %	0,0 %	5,0 %	16,7 %	3,3 %
12	R112-100									
13	C101-100	0,0 %	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	2,0 %	0,0 %	2,0 %
14	C102-100	2,0 %	10,0 %	0,0 %	0,0 %	10,0 %	2,0 %	2,0 %	10,0 %	8,0 %
15	C103-100	2,0 %	18,0 %	2,0 %	0,0 %	10,0 %	0,0 %	0,0 %	18,0 %	2,0 %
16	C104-100	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	4,0 %	4,0 %
17	C105-100	0,0 %	10,0 %	0,0 %	0,0 %	10,0 %	0,0 %	4,0 %	12,0 %	6,0 %
18	C106-100	0,0 %	18,0 %	2,0 %	2,0 %	20,0 %	0,0 %	0,0 %	20,0 %	2,0 %
19	C107-100	0,0 %	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	4,0 %	0,0 %	0,0 %
20	C108-100	0,0 %	12,0 %	0,0 %	0,0 %	12,0 %	0,0 %	2,0 %	10,0 %	2,0 %
21	C109-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
22	RC101-100	6,7 %	21,3 %	8,0 %	8,0 %	21,3 %	10,7 %	8,0 %	29,3 %	8,0 %
23	RC102-100	7,1 %	15,7 %	5,7 %	4,3 %	15,7 %	5,7 %	8,6 %	15,7 %	7,1 %
24	RC103-100	10,9 %	30,9 %	16,4 %	10,9 %	21,8 %	10,9 %	14,5 %	34,5 %	14,5 %
25	RC104-100									
26	RC105-100	4,0 %	18,7 %	5,3 %	2,7 %	14,7 %	2,7 %	8,0 %	22,7 %	4,0 %
27	RC106-100									
28	RC107-100	6,7 %	18,3 %	1,7 %	0,0 %	18,3 %	3,3 %	10,0 %	31,7 %	8,3 %
29	RC108-100	3,6 %	16,4 %	3,6 %	1,8 %	18,2 %	0,0 %	7,3 %	20,0 %	12,7 %
30	R201-100	-12,5 %	-25,0 %	-7,5 %	-12,5 %	-25,0 %	-15,0 %	-7,5 %	-25,0 %	-10,0 %
31	R202-100									
32	R203-100									
33	R204-100									
34	R205-100									
35	R206-100									
36	R207-100									
37	R208-100									
38	R209-100									
39	R210-100									
40	R211-100									
41	C201-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
42	C202-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
43	C203-100	26,7 %	40,0 %	13,3 %	13,3 %	33,3 %	13,3 %	6,7 %	40,0 %	6,7 %
44	C204-100	6,7 %	0,0 %	6,7 %	0,0 %	0,0 %	6,7 %	13,3 %	0,0 %	26,7 %
45	C205-100	13,3 %	33,3 %	6,7 %	20,0 %	33,3 %	0,0 %	13,3 %	33,3 %	13,3 %
46	C206-100	26,7 %	33,3 %	26,7 %	20,0 %	33,3 %	20,0 %	26,7 %	33,3 %	20,0 %
47	C207-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	6,7 %
48	C208-100	0,0 %	0,0 %	0,0 %	6,7 %	6,7 %	6,7 %	0,0 %	0,0 %	13,3 %
49	RC201-100	-6,7 %	-28,9 %	-11,1 %	-13,3 %	-33,3 %	-8,9 %	-15,6 %	-24,4 %	-8,9 %
50	RC202-100	-15,0 %	-27,5 %	-10,0 %	-12,5 %	-27,5 %	-22,5 %	-12,5 %	-32,5 %	-12,5 %
51	RC203-100									
52	RC204-100									
53	RC205-100	5,7 %	0,0 %	0,0 %	0,0 %	2,9 %	5,7 %	0,0 %	0,0 %	2,9 %
54	RC206-100									
55	RC207-100									
56	RC208-100									
	avg all instances:	2,9 %	9,9 %	2,3 %	1,6 %	9,2 %	1,3 %	3,9 %	12,0 %	4,9 %
	Devided per group of problems									
	100's	2 %	14 %	2 %	1 %	13 %	2 %	5 %	17 %	5 %
	200's	4 %	2 %	2 %	2 %	2 %	0 %	2 %	2 %	5 %
	R100	1,77 %	14,89 %	1,18 %	0,71 %	14,51 %	0,57 %	4,83 %	19,01 %	4,11 %
	C100	0,44 %	7,78 %	0,89 %	0,22 %	6,89 %	0,22 %	1,56 %	8,22 %	2,89 %
	RC100	6,50 %	20,22 %	6,79 %	4,61 %	18,34 %	5,55 %	9,40 %	25,65 %	9,12 %
	R200	-12,50 %	-25,00 %	-7,50 %	-12,50 %	-25,00 %	-15,00 %	-7,50 %	-25,00 %	-10,00 %
	C200	9,17 %	13,33 %	6,67 %	7,50 %	13,33 %	5,83 %	7,50 %	13,33 %	10,83 %
	RC200	-5,3 %	-18,8 %	-7,0 %	-8,6 %	-19,3 %	-8,6 %	-9,4 %	-19,0 %	-6,2 %

Table 5.5: Deterministic: deviation from the optimum of the average number of used vehicles.

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	4,2 %	14,7 %	3,2 %	3,2 %	15,8 %	1,1 %	4,2 %	24,2 %	5,3 %
2	R102-100	7,1 %	16,5 %	4,7 %	7,1 %	14,1 %	5,9 %	7,1 %	17,6 %	7,1 %
3	R103-100	12,3 %	23,1 %	9,2 %	7,7 %	24,6 %	13,8 %	15,4 %	30,8 %	13,8 %
4	R104-100	22,2 %	42,2 %	26,7 %	22,2 %	35,6 %	22,2 %	28,9 %	46,7 %	28,9 %
5	R105-100	8,6 %	25,7 %	8,6 %	7,1 %	21,4 %	5,7 %	11,4 %	30,0 %	11,4 %
6	R106-100	10,0 %	33,3 %	11,7 %	15,0 %	33,3 %	10,0 %	16,7 %	33,3 %	16,7 %
7	R107-100	16,0 %	30,0 %	16,0 %	12,0 %	38,0 %	16,0 %	22,0 %	34,0 %	20,0 %
8	R108-100	20,0 %	28,9 %	17,8 %	17,8 %	31,1 %	17,8 %	22,2 %	40,0 %	20,0 %
9	R109-100	18,2 %	32,7 %	18,2 %	18,2 %	34,5 %	18,2 %	23,6 %	41,8 %	21,8 %
10	R110-100	24,0 %	36,0 %	20,0 %	20,0 %	36,0 %	18,0 %	24,0 %	40,0 %	22,0 %
11	R111-100	22,0 %	38,0 %	20,0 %	20,0 %	34,0 %	20,0 %	26,0 %	40,0 %	24,0 %
12	R112-100	20,0 %	37,8 %	24,4 %	20,0 %	42,2 %	15,6 %	28,9 %	46,7 %	26,7 %
13	C101-100	0,0 %	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	2,0 %	0,0 %	2,0 %
14	C102-100	2,0 %	10,0 %	0,0 %	0,0 %	10,0 %	2,0 %	2,0 %	10,0 %	8,0 %
15	C103-100	2,0 %	18,0 %	2,0 %	0,0 %	10,0 %	0,0 %	0,0 %	18,0 %	2,0 %
16	C104-100	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	4,0 %	4,0 %
17	C105-100	0,0 %	10,0 %	0,0 %	0,0 %	10,0 %	0,0 %	4,0 %	12,0 %	6,0 %
18	C106-100	0,0 %	18,0 %	2,0 %	2,0 %	20,0 %	0,0 %	0,0 %	20,0 %	2,0 %
19	C107-100	0,0 %	0,0 %	2,0 %	0,0 %	0,0 %	0,0 %	4,0 %	0,0 %	0,0 %
20	C108-100	0,0 %	12,0 %	0,0 %	0,0 %	12,0 %	0,0 %	2,0 %	10,0 %	2,0 %
21	C109-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
22	RC101-100	14,3 %	30,0 %	15,7 %	15,7 %	30,0 %	18,6 %	15,7 %	38,6 %	15,7 %
23	RC102-100	25,0 %	35,0 %	23,3 %	21,7 %	35,0 %	23,3 %	26,7 %	35,0 %	25,0 %
24	RC103-100	10,9 %	30,9 %	16,4 %	10,9 %	21,8 %	10,9 %	14,5 %	34,5 %	14,5 %
25	RC104-100	10,0 %	24,0 %	12,0 %	8,0 %	20,0 %	10,0 %	12,0 %	32,0 %	18,0 %
26	RC105-100	20,0 %	36,9 %	21,5 %	18,5 %	32,3 %	18,5 %	24,6 %	41,5 %	20,0 %
27	RC106-100	23,6 %	45,5 %	25,5 %	25,5 %	40,0 %	23,6 %	29,1 %	47,3 %	25,5 %
28	RC107-100	16,4 %	29,1 %	10,9 %	9,1 %	29,1 %	12,7 %	20,0 %	43,6 %	18,2 %
29	RC108-100	14,0 %	28,0 %	14,0 %	12,0 %	30,0 %	10,0 %	18,0 %	32,0 %	24,0 %
30	R201-100	75,0 %	50,0 %	85,0 %	75,0 %	50,0 %	70,0 %	85,0 %	50,0 %	80,0 %
31	R202-100	93,3 %	93,3 %	106,7 %	93,3 %	86,7 %	100,0 %	100,0 %	66,7 %	120,0 %
32	R203-100	86,7 %	66,7 %	73,3 %	80,0 %	66,7 %	100,0 %	86,7 %	66,7 %	80,0 %
33	R204-100	140,0 %	110,0 %	150,0 %	140,0 %	150,0 %	140,0 %	120,0 %	100,0 %	130,0 %
34	R205-100	73,3 %	66,7 %	73,3 %	86,7 %	60,0 %	73,3 %	80,0 %	60,0 %	80,0 %
35	R206-100	40,0 %	33,3 %	60,0 %	40,0 %	33,3 %	53,3 %	60,0 %	33,3 %	46,7 %
36	R207-100	110,0 %	100,0 %	110,0 %	120,0 %	90,0 %	110,0 %	110,0 %	100,0 %	100,0 %
37	R208-100	70,0 %	50,0 %	70,0 %	80,0 %	50,0 %	60,0 %	80,0 %	50,0 %	80,0 %
38	R209-100	60,0 %	33,3 %	60,0 %	46,7 %	53,3 %	66,7 %	53,3 %	40,0 %	66,7 %
39	R210-100	100,0 %	66,7 %	93,3 %	93,3 %	66,7 %	100,0 %	80,0 %	66,7 %	100,0 %
40	R211-100	100,0 %	100,0 %	110,0 %	100,0 %	100,0 %	100,0 %	100,0 %	100,0 %	110,0 %
41	C201-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
42	C202-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %
43	C203-100	26,7 %	40,0 %	13,3 %	13,3 %	33,3 %	13,3 %	6,7 %	40,0 %	6,7 %
44	C204-100	6,7 %	0,0 %	6,7 %	0,0 %	0,0 %	6,7 %	13,3 %	0,0 %	26,7 %
45	C205-100	13,3 %	33,3 %	6,7 %	20,0 %	33,3 %	0,0 %	13,3 %	33,3 %	13,3 %
46	C206-100	26,7 %	33,3 %	26,7 %	20,0 %	33,3 %	20,0 %	26,7 %	33,3 %	20,0 %
47	C207-100	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	0,0 %	6,7 %
48	C208-100	0,0 %	0,0 %	0,0 %	6,7 %	6,7 %	6,7 %	0,0 %	0,0 %	13,3 %
49	RC201-100	110,0 %	60,0 %	100,0 %	95,0 %	50,0 %	105,0 %	90,0 %	70,0 %	105,0 %
50	RC202-100	126,7 %	93,3 %	140,0 %	133,3 %	93,3 %	106,7 %	133,3 %	80,0 %	133,3 %
51	RC203-100	66,7 %	66,7 %	86,7 %	66,7 %	66,7 %	73,3 %	80,0 %	73,3 %	80,0 %
52	RC204-100	33,3 %	33,3 %	33,3 %	33,3 %	33,3 %	33,3 %	26,7 %	33,3 %	33,3 %
53	RC205-100	85,0 %	75,0 %	75,0 %	75,0 %	80,0 %	85,0 %	75,0 %	75,0 %	80,0 %
54	RC206-100	73,3 %	66,7 %	73,3 %	80,0 %	66,7 %	73,3 %	73,3 %	66,7 %	66,7 %
55	RC207-100	66,7 %	66,7 %	86,7 %	80,0 %	66,7 %	73,3 %	73,3 %	73,3 %	86,7 %
56	RC208-100	33,3 %	33,3 %	33,3 %	33,3 %	33,3 %	33,3 %	40,0 %	33,3 %	33,3 %
	avg all instances:	34,6 %	36,8 %	35,7 %	34,0 %	36,9 %	33,9 %	35,9 %	38,4 %	37,6 %
	Devided per group of problems									
	100's	11 %	24 %	11 %	10 %	23 %	10 %	14 %	28 %	14 %
	200's	60 %	51 %	62 %	60 %	52 %	59 %	60 %	50 %	63 %
	R100	15,38 %	29,91 %	15,03 %	14,19 %	30,06 %	13,69 %	19,20 %	35,43 %	18,14 %
	C100	0,44 %	7,78 %	0,89 %	0,22 %	6,89 %	0,22 %	1,56 %	8,22 %	2,89 %
	RC100	16,77 %	32,42 %	17,41 %	15,16 %	29,78 %	15,95 %	20,08 %	38,07 %	20,11 %
	R200	86,21 %	70,00 %	90,15 %	86,82 %	73,33 %	88,48 %	86,82 %	66,67 %	90,30 %
	C200	9,17 %	13,33 %	6,67 %	7,50 %	13,33 %	5,83 %	7,50 %	13,33 %	10,83 %
	RC200	74,4 %	61,9 %	78,5 %	74,6 %	61,3 %	72,9 %	74,0 %	63,1 %	77,3 %

Table 5.6: Deterministic: deviation from heuristics of the average number of used vehicles (used only as a reference because the online heuristics algorithms minimize the NV first)

num instances	instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
1	R101-100	27 %	30 %	26 %	27 %	31 %	25 %	27 %	30 %	27 %
2	R102-100	27 %	30 %	27 %	28 %	29 %	27 %	27 %	29 %	28 %
3	R103-100	23 %	20 %	23 %	21 %	21 %	25 %	25 %	22 %	24 %
4	R104-100	13 %	16 %	15 %	15 %	13 %	13 %	14 %	14 %	16 %
5	R105-100	15 %	17 %	14 %	14 %	16 %	13 %	16 %	18 %	16 %
6	R106-100	13 %	16 %	13 %	14 %	16 %	13 %	14 %	15 %	15 %
7	R107-100	11 %	12 %	12 %	10 %	15 %	12 %	14 %	12 %	15 %
8	R108-100	12 %	10 %	11 %	10 %	12 %	10 %	10 %	12 %	10 %
9	R109-100	11 %	10 %	10 %	11 %	11 %	11 %	12 %	12 %	11 %
10	R110-100	11 %	8 %	9 %	11 %	8 %	10 %	12 %	8 %	11 %
11	R111-100	14 %	12 %	14 %	13 %	10 %	14 %	14 %	10 %	14 %
12	R112-100	9 %	9 %	10 %	9 %	11 %	8 %	10 %	10 %	9 %
13	C101-100	0 %	0 %	1 %	0 %	0 %	0 %	2 %	0 %	1 %
14	C102-100	1 %	6 %	0 %	0 %	6 %	1 %	2 %	5 %	3 %
15	C103-100	4 %	11 %	2 %	1 %	10 %	1 %	2 %	8 %	4 %
16	C104-100	3 %	7 %	3 %	2 %	7 %	2 %	4 %	6 %	6 %
17	C105-100	0 %	5 %	0 %	0 %	6 %	0 %	2 %	6 %	2 %
18	C106-100	0 %	9 %	1 %	1 %	11 %	0 %	0 %	10 %	1 %
19	C107-100	0 %	1 %	0 %	0 %	1 %	0 %	3 %	1 %	0 %
20	C108-100	0 %	4 %	0 %	0 %	4 %	0 %	0 %	1 %	1 %
21	C109-100	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %
22	RC101-100	16 %	18 %	16 %	16 %	18 %	17 %	16 %	18 %	15 %
23	RC102-100	17 %	15 %	17 %	16 %	15 %	16 %	17 %	13 %	18 %
24	RC103-100	12 %	7 %	13 %	10 %	7 %	13 %	12 %	8 %	14 %
25	RC104-100	6 %	7 %	8 %	7 %	7 %	8 %	7 %	8 %	7 %
26	RC105-100	16 %	18 %	18 %	16 %	17 %	16 %	20 %	18 %	18 %
27	RC106-100	10 %	10 %	12 %	12 %	8 %	12 %	13 %	9 %	11 %
28	RC107-100	12 %	8 %	10 %	9 %	8 %	10 %	11 %	11 %	11 %
29	RC108-100	9 %	6 %	10 %	8 %	7 %	7 %	9 %	5 %	10 %
30	R201-100	62 %	51 %	63 %	61 %	51 %	61 %	63 %	48 %	62 %
31	R202-100	56 %	45 %	59 %	54 %	43 %	56 %	58 %	38 %	61 %
32	R203-100	60 %	48 %	57 %	57 %	48 %	61 %	59 %	49 %	58 %
33	R204-100	52 %	48 %	53 %	50 %	53 %	52 %	48 %	45 %	51 %
34	R205-100	50 %	44 %	50 %	53 %	39 %	49 %	52 %	41 %	51 %
35	R206-100	41 %	32 %	47 %	41 %	33 %	45 %	48 %	31 %	43 %
36	R207-100	42 %	32 %	42 %	45 %	33 %	42 %	41 %	38 %	40 %
37	R208-100	29 %	16 %	32 %	33 %	18 %	27 %	31 %	16 %	33 %
38	R209-100	45 %	32 %	44 %	41 %	36 %	44 %	42 %	32 %	46 %
39	R210-100	59 %	43 %	59 %	58 %	43 %	59 %	55 %	43 %	60 %
40	R211-100	34 %	26 %	37 %	35 %	26 %	34 %	35 %	25 %	37 %
41	C201-100	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %
42	C202-100	0 %	1 %	0 %	0 %	1 %	0 %	0 %	1 %	0 %
43	C203-100	10 %	19 %	5 %	5 %	17 %	6 %	3 %	20 %	3 %
44	C204-100	4 %	1 %	4 %	0 %	1 %	3 %	6 %	0 %	7 %
45	C205-100	6 %	21 %	4 %	7 %	21 %	0 %	8 %	21 %	9 %
46	C206-100	16 %	21 %	16 %	11 %	18 %	11 %	16 %	13 %	11 %
47	C207-100	0 %	2 %	1 %	1 %	2 %	1 %	1 %	2 %	5 %
48	C208-100	2 %	2 %	2 %	3 %	3 %	2 %	2 %	2 %	9 %
49	RC201-100	63 %	40 %	61 %	60 %	39 %	62 %	58 %	39 %	62 %
50	RC202-100	56 %	45 %	59 %	57 %	44 %	53 %	58 %	40 %	59 %
51	RC203-100	52 %	45 %	56 %	53 %	47 %	54 %	54 %	46 %	53 %
52	RC204-100	40 %	36 %	40 %	39 %	35 %	38 %	37 %	35 %	39 %
53	RC205-100	62 %	55 %	61 %	60 %	56 %	63 %	59 %	55 %	61 %
54	RC206-100	45 %	37 %	44 %	44 %	35 %	44 %	45 %	37 %	44 %
55	RC207-100	42 %	39 %	46 %	45 %	39 %	45 %	42 %	39 %	46 %
56	RC208-100	32 %	23 %	33 %	32 %	24 %	33 %	35 %	23 %	33 %
	avg all instances:	22 %	20 %	23 %	22 %	20 %	22 %	23 %	20 %	23 %
	Devided per group of problems									
	100's	10 %	11 %	10 %	10 %	11 %	10 %	11 %	11 %	11 %
	200's	36 %	30 %	36 %	35 %	30 %	35 %	35 %	29 %	36 %
	R100	16 %	16 %	15 %	15 %	16 %	15 %	16 %	16 %	16 %
	C100	1 %	5 %	1 %	0 %	5 %	0 %	2 %	4 %	2 %
	RC100	12 %	11 %	13 %	12 %	11 %	12 %	13 %	11 %	13 %
	R200	48 %	38 %	49 %	48 %	39 %	48 %	48 %	37 %	49 %
	C200	5 %	8 %	4 %	3 %	8 %	3 %	4 %	7 %	5 %
	RC200	49 %	40 %	50 %	49 %	40 %	49 %	49 %	39 %	50 %

Table 5.7: Deterministic: percentage of waiting time in the total route duration

							deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
instance name	OPTIMAL DISTANCE	NV	Authors	BEST HEURISTICS DISTANCE	NV	Authors	avg distance	avg distance	avg deviation from optimum	avg deviation from optimum	avg deviation from Best Heuristics	avg deviation from Best Heuristics
R101-100	1637,7	20	KDMSS	1645,79	19	H	1658,040112	1630,382129	1,24%	-0,45%	0,74%	-0,94%
R102-100	1466,6	18	KDMSS	1486,12	17	RT	1491,752905	1435,526123	1,72%	-2,12%	0,38%	-3,40%
R103-100	1208,7	14	CR+L	1292,68	13	LLH	1230,252979	1229,491601	1,78%	1,72%	-4,83%	-4,89%
R104-100	971,5	11	EIV	1007,24	9	M	994,200415	1008,702991	2,34%	3,83%	-1,29%	0,15%
R105-100	1355,3	15	KDMSS	1377,11	14	RT	1381,819629	1374,668213	1,96%	1,43%	0,34%	-0,18%
R106-100	1234,6	13	CR+KLM	1251,98	12	M	1249,691284	1257,194312	1,22%	1,83%	-0,18%	0,42%
R107-100	1064,6	11	CR+KLM	1104,66	10	S97	1084,27124	1095,70813	1,85%	2,92%	-1,85%	-0,81%
R108-100	932,1	NA	see [17]	960,88	9	BBB	966,2671508	971,2764038	3,67%	4,20%	0,56%	1,08%
R109-100	1146,9	13	CR+KLM	1194,73	11	HG	1163,382593	1174,817749	1,44%	2,43%	-2,62%	-1,67%
R110-100	1068	12	CR+KLM	1118,59	10	M	1102,817456	1112,544727	3,26%	4,17%	-1,41%	-0,54%
R111-100	1048,7	12	CR+KLM	1096,72	10	RGP	1069,14104	1076,673633	1,95%	2,67%	-2,51%	-1,83%
R112-100	948,6	NA	see [17]	982,14	9	GTA	979,260193	981,8409054	3,23%	3,50%	-0,29%	-0,03%
C101-100	827,3	10	KDMSS	828,94	10	RT	828,93689	881,1570922	0,20%	6,51%	0,00%	6,30%
C102-100	827,3	10	KDMSS	828,94	10	RT	834,8117068	866,8434448	0,91%	4,78%	0,71%	4,57%
C103-100	826,3	10	KDMSS	828,06	10	RT	844,4716306	855,0189574	2,20%	3,48%	1,98%	3,26%
C104-100	822,9	10	KDMSS	824,78	10	RT	824,776733	837,1177492	0,23%	1,73%	0,00%	1,50%
C105-100	827,3	10	KDMSS	828,94	10	RT	828,93689	849,640686	0,20%	2,70%	0,00%	2,50%
C106-100	827,3	10	KDMSS	828,94	10	RT	833,5339846	857,640686	0,75%	3,67%	0,55%	3,46%
C107-100	827,3	10	KDMSS	828,94	10	RT	828,93689	848,5901734	0,20%	2,57%	0,00%	2,37%
C108-100	827,3	10	KDMSS	828,94	10	RT	828,93689	846,4891482	0,20%	2,32%	0,00%	2,12%
C109-100	827,3	10	KDMSS	828,94	10	RT	828,93689	844,388123	0,20%	2,07%	0,00%	1,86%
RC101-100	1619,8	15	KDMSS	1696,94	14	TBGGP	1664,204395	1667,842822	2,74%	2,97%	-1,93%	-1,71%
RC102-100	1457,4	14	CR+KLM	1554,75	12	TBGGP	1489,654785	1489,249609	2,21%	2,19%	-4,19%	-4,21%
RC103-100	1258	11	CR+KLM	1261,67	11	S98	1320,686426	1346,85127	4,98%	7,06%	4,68%	6,75%
RC104-100	1132,3	NA	see [17]	1135,48	10	CLM	1173,077832	1180,166553	3,60%	4,23%	3,31%	3,94%
RC105-100	1513,7	15	KDMSS	1629,44	13	BBB	1571,35188	1549,774194	3,81%	2,38%	-3,56%	-4,89%
RC106-100	1372,7	NA	see [17]	1424,73	11	BBB	1407,162353	1408,366016	2,51%	2,60%	-1,23%	-1,15%
RC107-100	1207,8	12	EIV	1230,48	11	S97	1259,203003	1261,465405	4,26%	4,44%	2,33%	2,52%
RC108-100	1114,2	11	EIV	1139,82	10	TBGGP	1158,142505	1161,816626	3,94%	4,27%	1,61%	1,93%
R201-100	1143,2	8	KLM	1252,37	4	HG	1184,661401	1184,86958	3,63%	3,64%	-5,41%	-5,39%
R202-100	1029,6	NA	see [17]	1191,7	3	RGP	1080,730469	1068,554102	4,97%	3,78%	-9,31%	-10,33%
R203-100	870,8	NA	see [17]	939,54	3	M	907,2571288	902,4806396	4,19%	3,64%	-3,44%	-3,94%
R204-100	731,3	NA	see [17]	825,52	2	BVH	753,7207274	757,568945	3,07%	3,59%	-8,70%	-8,23%
R205-100	949,8	NA	see [17]	994,42	3	RGP	995,988379	974,7100098	4,86%	2,62%	0,16%	-1,98%
R206-100	875,9	NA	see [17]	906,14	3	SSSD	925,7012328	928,4628174	5,69%	6,00%	2,16%	2,46%
R207-100	794	NA	see [17]	893,33	2	BVH	837,113391	841,2376466	5,43%	5,95%	-6,29%	-5,83%
R208-100	701,2	NA	see [17]	726,75	2	M	736,7035032	732,6227172	5,06%	4,48%	1,37%	0,81%
R209-100	854,8	NA	see [17]	909,16	3	H	900,4270386	903,4842898	5,34%	5,70%	-0,96%	-0,62%
R210-100	900,5	NA	see [17]	939,34	3	M	918,6826292	934,633789	2,02%	3,79%	-2,20%	-0,50%
R211-100	746,7	NA	see [17]	892,71	2	BVH	789,5497314	803,9983644	5,74%	7,67%	-11,56%	-9,94%
C201-100	589,1	3	CR+KLM	591,56	3	RT	591,55658	607,935547	0,42%	3,20%	0,00%	2,77%
C202-100	589,1	3	CR+KLM	591,56	3	RT	591,55658	607,643188	0,42%	3,15%	0,00%	2,72%
C203-100	588,7	3	EKLM	591,17	3	RT	618,1573366	623,0604128	5,00%	5,84%	4,57%	5,39%
C204-100	588,1	3	EIV	590,6	3	RT	602,910022	601,3870238	2,52%	2,26%	2,08%	1,83%
C205-100	586,4	3	CR+KLM	588,88	3	RT	588,875977	642,8301392	0,42%	9,62%	0,00%	9,16%
C206-100	586	3	CR+KLM	588,49	3	RT	621,139429	619,1669798	6,00%	5,66%	5,55%	5,21%
C207-100	585,8	3	CR+KLM	588,29	3	RT	588,286377	614,7764162	0,42%	4,95%	0,00%	4,50%
C208-100	585,8	3	EKLM	588,32	3	RT	593,7285526	600,878784	1,35%	2,57%	0,92%	2,13%
RC201-100	1261,8	9	KLM	1406,91	4	M	1311,987256	1309,890552	3,98%	3,81%	-6,75%	-6,90%
RC202-100	1092,3	8	IV+C	1367,09	3	CC	1168,732544	1165,750317	7,00%	6,72%	-14,51%	-14,73%
RC203-100	923,7	NA	see [17]	1049,62	3	CC	971,969458	988,0774658	5,23%	6,97%	-7,40%	-5,86%
RC204-100	783,5	NA	see [17]	798,41	3	M	847,555017	830,473999	8,18%	6,00%	6,16%	4,02%
RC205-100	1154	7	IV+C	1297,19	4	M	1187,140137	1239,116772	2,87%	7,38%	-8,48%	-4,48%
RC206-100	1051,1	NA	see [17]	1146,32	3	H	1089,991138	1106,402441	3,70%	5,26%	-4,91%	-3,48%
RC207-100	962,9	NA	see [17]	1061,14	3	BVH	1007,502746	1025,972949	4,63%	6,55%	-5,05%	-3,31%
RC208-100	776,1	NA	see [17]	828,14	3	IKMUY	847,9465942	863,3763916	9,26%	11,25%	2,39%	4,25%
							deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
							avg distance	avg distance	avg deviation from optimum	avg deviation from optimum	avg deviation from Best Heuristics	avg deviation from Best Heuristics
						AVERAGE ALL INSTANCES:	1002,789822	1010,367996	3,04%	4,07%	-1,40%	-0,39%
						100's	1134,367541	1141,422258	2,03%	2,97%	-0,30%	0,64%
						200's	861,4656065	869,6060104	4,12%	5,26%	-2,58%	-1,49%
						R100	1197,57475	1195,735576	2,14%	2,18%	-1,08%	-1,05%
						C100	831,3642783	854,0984511	0,56%	3,31%	0,36%	3,10%
						RC100	1380,435397	1383,191562	3,51%	3,77%	0,13%	0,40%
						R200	911,8668756	912,0566273	4,54%	4,62%	-4,02%	-3,95%
						C200	599,5263568	614,7098114	2,07%	4,66%	1,64%	4,21%
						RC200	1054,103111	1066,132611	5,60%	6,74%	-4,82%	-3,81%

Table 5.8: Deterministic vs Stochastic: Average distance for 5 seeds.

							deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
instance name	OPTIMAL DISTANCE	NV	Authors	BEST HEURISTICS DISTANCE	NV	Authors	NV	NV	NV avg deviation from optimum	NV avg deviation from optimum	NV avg deviation from Best Heuristics	NV avg deviation from Best Heuristics
R101-100	1637,7	20	KDMSS	1645,79	19	H	19,2	18,2	-4,00%	-9,00%	1,05%	-4,21%
R102-100	1466,6	18	KDMSS	1486,12	17	RT	18	16,8	0,00%	-6,67%	5,88%	-1,18%
R103-100	1208,7	14	CR+L	1292,68	13	LLH	14,8	14,4	5,71%	2,86%	13,85%	10,77%
R104-100	971,5	11	EIV	1007,24	9	M	11	11,2	0,00%	1,82%	22,22%	24,44%
R105-100	1355,3	15	KDMSS	1377,11	14	RT	14,8	15	-1,33%	0,00%	5,71%	7,14%
R106-100	1234,6	13	CR+KLM	1251,98	12	M	13,2	13,2	1,54%	1,54%	10,00%	10,00%
R107-100	1064,6	11	CR+KLM	1104,66	10	S97	11,6	11,8	5,45%	7,27%	16,00%	18,00%
R108-100	932,1	NA	see [17]	960,88	9	BBB	10,6	10,4			17,78%	15,56%
R109-100	1146,9	13	CR+KLM	1194,73	11	HG	13	13	0,00%	0,00%	18,18%	18,18%
R110-100	1068	12	CR+KLM	1118,59	10	M	11,8	11,8	-1,67%	-1,67%	18,00%	18,00%
R111-100	1048,7	12	CR+KLM	1096,72	10	RGP	12	12	0,00%	0,00%	20,00%	20,00%
R112-100	948,6	NA	see [17]	982,14	9	GTA	10,4	11			15,56%	22,22%
C101-100	827,3	10	KDMSS	828,94	10	RT	10	10,6	0,00%	6,00%	0,00%	6,00%
C102-100	827,3	10	KDMSS	828,94	10	RT	10,2	10,2	2,00%	2,00%	2,00%	2,00%
C103-100	826,3	10	KDMSS	828,06	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
C104-100	822,9	10	KDMSS	824,78	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
C105-100	827,3	10	KDMSS	828,94	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
C106-100	827,3	10	KDMSS	828,94	10	RT	10	10,4	0,00%	4,00%	0,00%	4,00%
C107-100	827,3	10	KDMSS	828,94	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
C108-100	827,3	10	KDMSS	828,94	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
C109-100	827,3	10	KDMSS	828,94	10	RT	10	10	0,00%	0,00%	0,00%	0,00%
RC101-100	1619,8	15	KDMSS	1696,94	14	TBGGP	16,6	15,8	10,67%	5,33%	18,57%	12,86%
RC102-100	1457,4	14	CR+KLM	1554,75	12	TBGGP	14,8	14,8	5,71%	5,71%	23,33%	23,33%
RC103-100	1258	11	CR+KLM	1261,67	11	S98	12,2	12,4	10,91%	12,73%	10,91%	12,73%
RC104-100	1132,3	NA	see [17]	1135,48	10	CLM	11	11			10,00%	10,00%
RC105-100	1513,7	15	KDMSS	1629,44	13	BBB	15,4	15	2,67%	0,00%	18,46%	15,38%
RC106-100	1372,7	NA	see [17]	1424,73	11	BBB	13,6	13,6			23,64%	23,64%
RC107-100	1207,8	12	EIV	1230,48	11	S97	12,4	12,4	3,33%	3,33%	12,73%	12,73%
RC108-100	1114,2	11	EIV	1139,82	10	TBGGP	11	11,4	0,00%	3,64%	10,00%	14,00%
R201-100	1143,2	8	KLM	1252,37	4	HG	6,8	6,8	-15,00%	-15,00%	70,00%	70,00%
R202-100	1029,6	NA	see [17]	1191,7	3	RGP	6	6,2			100,00%	106,67%
R203-100	870,8	NA	see [17]	939,54	3	M	6	5,6			100,00%	86,67%
R204-100	731,3	NA	see [17]	825,52	2	BVH	4,8	4,2			140,00%	110,00%
R205-100	949,8	NA	see [17]	994,42	3	RGP	5,2	5,2			73,33%	73,33%
R206-100	875,9	NA	see [17]	906,14	3	SSSD	4,6	4,4			53,33%	46,67%
R207-100	794	NA	see [17]	893,33	2	BVH	4,2	3,8			110,00%	90,00%
R208-100	701,2	NA	see [17]	726,75	2	M	3,2	3,6			60,00%	80,00%
R209-100	854,8	NA	see [17]	909,16	3	H	5	5,2			66,67%	73,33%
R210-100	900,5	NA	see [17]	939,34	3	M	6	5,4			100,00%	80,00%
R211-100	746,7	NA	see [17]	892,71	2	BVH	4	3,8			100,00%	90,00%
C201-100	589,1	3	CR+KLM	591,56	3	RT	3	4	0,00%	33,33%	0,00%	33,33%
C202-100	589,1	3	CR+KLM	591,56	3	RT	3	3	0,00%	0,00%	0,00%	0,00%
C203-100	588,7	3	EKLM	591,17	3	RT	3,4	3,8	13,33%	26,67%	13,33%	26,67%
C204-100	588,1	3	EIV	590,6	3	RT	3,2	3	6,67%	0,00%	6,67%	0,00%
C205-100	586,4	3	CR+KLM	588,88	3	RT	3	4	0,00%	33,33%	0,00%	33,33%
C206-100	586	3	CR+KLM	588,49	3	RT	3,6	3,4	20,00%	13,33%	20,00%	13,33%
C207-100	585,8	3	CR+KLM	588,29	3	RT	3	3,4	0,00%	13,33%	0,00%	13,33%
C208-100	585,8	3	EKLM	588,32	3	RT	3,2	3	6,67%	0,00%	6,67%	0,00%
RC201-100	1261,8	9	KLM	1406,91	4	M	8,2	7,8	-8,89%	-13,33%	105,00%	95,00%
RC202-100	1092,3	8	IV+C	1367,09	3	CC	6,2	6,2	-22,50%	-22,50%	106,67%	106,67%
RC203-100	923,7	NA	see [17]	1049,62	3	CC	5,2	5,2			73,33%	73,33%
RC204-100	783,5	NA	see [17]	798,41	3	M	4	4			33,33%	33,33%
RC205-100	1154	7	IV+C	1297,19	4	M	7,4	7	5,71%	0,00%	85,00%	75,00%
RC206-100	1051,1	NA	see [17]	1146,32	3	H	5,2	6,4			73,33%	113,33%
RC207-100	962,9	NA	see [17]	1061,14	3	BVH	5,2	5,8			73,33%	93,33%
RC208-100	776,1	NA	see [17]	828,14	3	IKMUY	4	4			33,33%	33,33%
							deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
							NV	NV	NV avg deviation from optimum	NV avg deviation from optimum	NV avg deviation from Best Heuristics	NV avg deviation from Best Heuristics
						AVERAGE ALL INSTANCES:	8,646428571	8,653571429	1,27%	2,92%	33,88%	34,74%
						100's	12,33103448	12,28965517	1,64%	1,56%	10,13%	10,19%
						200's	4,688888889	4,748148148	0,50%	5,76%	59,38%	61,11%
						R100	13,36666667	13,23333333	0,57%	-0,38%	13,69%	13,24%
						C100	10,02222222	10,13333333	0,22%	1,33%	0,22%	1,33%
						RC100	13,375	13,3	5,55%	5,12%	15,95%	15,58%
						R200	5,072727273	4,927272727	-15,00%	-15,00%	88,48%	82,42%
						C200	3,175	3,45	5,83%	15,00%	5,83%	15,00%
						RC200	5,675	5,8	-8,56%	-11,94%	72,92%	77,92%

Table 5.9: Deterministic vs Stochastic: Expected number of vehicles for 5 seeds (the comparison with the best heuristics is used only as a reference because the online algorithms minimize the NV first).

	deterministic	stochastic	deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
instance name	expected NV to start service after tw	expected NV to start service after tw	expected time crossed after tw end	expected time crossed after tw end	expected costs 1000 instances (penalty 10)	expected costs 1000 instances (penalty 10)	avg n. of scenarios where at least one tw is not respected	avg n. of scenarios where at least one tw is not respected
R101-100	2,3568316	3,4021784	2,1709994	3,6389732	1663,333911	1650,629566	91,3%	98,78%
R102-100	2,7435646	4,8251488	2,4905734	5,7786154	1501,888501	1479,099219	93,3%	99,98%
R103-100	2,8833664	1,059406	2,5202866	1,4865808	1243,274707	1232,183203	93,4%	59,3%
R104-100	4,1257428	1,2902972	5,422548	1,3411224	1038,581616	1012,12605	97,4%	65,92%
R105-100	1,607723	1,3271286	1,6603472	1,1990306	1384,742749	1373,047876	83,4%	77,48%
R106-100	2,2726734	0,9233664	3,0057836	1,093804	1267,376709	1255,685767	90,6%	50,76%
R107-100	1,340396	0,3790098	1,7318126	0,3365796	1090,853271	1088,226855	76,7%	28,76%
R108-100	1,5942574	0,2732676	2,415959	0,293165	980,8622682	964,590686	67,1%	20,24%
R109-100	2,149703	0,6833662	2,1082812	0,5904474	1172,946728	1169,09187	89,6%	45,26%
R110-100	3,4813862	0,939802	4,404193	1,2406452	1135,940381	1113,935034	94,5%	62,7%
R111-100	2,7162378	1,4233664	3,6961132	1,6113268	1095,516748	1082,127856	87,2%	81,3%
R112-100	1,3669306	0,2346534	2,1226618	0,227427	990,7925294	974,3881348	66,6%	19,62%
C101-100	2,181188	0,9617822	20,156563	5,009494	1022,295715	922,5265624	72,4%	43,4%
C102-100	1,6883168	0,5807922	17,0262828	2,6386998	996,8092526	884,6485474	68,0%	33,52%
C103-100	1,0483168	0,3366336	10,5254394	1,5035386	941,3628906	861,5870364	53,7%	25,18%
C104-100	1,0506932	0,1817822	13,756567	0,8028702	954,1722778	836,8607546	57,2%	18,06%
C105-100	0,909901	0,00297	13,249454	0,009743	953,224487	841,324585	50,9%	0,4%
C106-100	1,7542572	0,595446	18,3456664	3,1159444	1008,738391	880,3071292	71,3%	30,94%
C107-100	0,906931	0,0679208	13,23971	0,5815726	953,127014	846,0017702	50,8%	4,16%
C108-100	0,910495	0,0653466	13,2427694	0,4566624	953,1575558	842,669556	50,8%	5,62%
C109-100	0,906931	0	13,2346094	0	953,0760008	836,01947	50,8%	0%
RC101-100	2,242178	1,770297	2,3568048	2,785305	1671,295801	1679,183984	90,4%	71,36%
RC102-100	2,1302972	0,9920792	2,5796474	0,951163	1500,703027	1484,016748	84,2%	62,32%
RC103-100	2,5362376	0,5764356	3,2725542	0,6252546	1340,336084	1339,76936	89,3%	41,74%
RC104-100	1,3742576	0,3724752	2,3267514	0,4744842	1184,732226	1173,228711	73,6%	28,74%
RC105-100	2,219604	1,520396	3,3259056	2,3153836	1589,051636	1557,58789	86,5%	73,32%
RC106-100	1,629307	1,0728712	2,087499	1,3857658	1414,104321	1408,282373	77,7%	65,26%
RC107-100	1,5748516	0,5429702	2,3818324	0,53314	1270,555615	1254,302319	76,8%	39,52%
RC108-100	1,3245542	0,4423762	2,2602472	0,6165646	1169,277515	1156,479175	72,5%	32,34%
R201-100	0,171485	0,3158414	0,1832826	0,5009474	1174,762305	1178,142627	16,3%	32,12%
R202-100	0,26099	0,829505	0,2057274	1,0300912	1072,086499	1068,274463	23,3%	61,8%
R203-100	0,3342572	0,070693	0,3162114	0,035623	901,4343874	893,9009032	31,7%	6,12%
R204-100	0,1380198	0,0172278	0,2259822	0,0182184	748,5170532	750,2477662	13,8%	1,62%
R205-100	0,227921	0,2005938	0,193894	0,1417872	988,0667602	966,4696532	22,4%	20,1%
R206-100	0,2007918	0,0475248	0,2806106	0,0272204	919,3468994	919,542102	18,0%	3,98%
R207-100	0,0930694	0,0584158	0,1337248	0,0626594	830,1615478	833,5307374	9,4%	6,1%
R208-100	0,001584	0,048713	0,0006876	0,0680394	729,4197022	726,0490356	0,2%	4,98%
R209-100	0,2279208	0,3736634	0,222805	0,3281286	893,738098	897,8176514	22,7%	29,62%
R210-100	0,339208	0,1463366	0,2414906	0,087706	912,0028198	926,253723	30,3%	13,48%
R211-100	0,1920792	0,2772278	0,2699916	0,5199496	784,4300536	801,2354738	18,0%	23,06%
C201-100	0,977228	0,042574	22,54978	0,498906	811,199829	606,907349	50,0%	1,62%
C202-100	0,969307	0,037624	22,451336	0,437406	810,215393	606,003235	50,0%	1,32%
C203-100	0,1176238	0,028317	1,2027992	0,3395758	624,0659912	620,2887818	5,9%	1,76%
C204-100	0,0176238	0,0174258	0,2152416	0,1931166	599,0978026	597,3671142	1,1%	0,86%
C205-100	0,857426	0,00099	19,547939	0,002223	778,524292	636,4875366	46,2%	0,2%
C206-100	1,0415844	0,0075246	23,7594456	0,1052226	852,5822148	614,0915286	56,0%	0,44%
C207-100	1,289109	0,0976234	25,518568	1,2085246	837,64978	620,774463	58,3%	5,4%
C208-100	0,855446	0,00297	19,542166	0,01159	783,2696288	595,042114	46,2%	0,1%
RC201-100	0,3247524	0	0,3175672	0	1302,16897	1296,92644	28,2%	0%
RC202-100	0,1859408	0,0647522	0,2738026	0,0643964	1159,898633	1154,847461	16,8%	6%
RC203-100	0,2665346	0,2849506	0,251136	0,2614956	964,8591552	980,9104246	23,7%	26,74%
RC204-100	0,2485148	0,0526734	0,446669	0,0462192	843,632971	822,709949	18,9%	5,1%
RC205-100	0,4443566	0,0744556	0,5728476	0,117369	1181,115112	1228,021192	32,5%	7,84%
RC206-100	0,308515	0,0457424	0,4766796	0,0137624	1083,969971	1095,585425	28,6%	3,82%
RC207-100	0,7324754	0,3237622	0,6066654	0,2142144	1003,593885	1017,953809	58,1%	29,68%
RC208-100	0,3851486	0,0053466	0,6529276	0,006234	846,0796506	854,890979	29,6%	0,46%
	deterministic	stochastic	deterministic	stochastic	deterministic	stochastic	deterministic	stochastic
	expected NV to start service after tw	expected NV to start service after tw	expecteTimeCrossedAfterTwEnd	expecteTimeCrossedAfterTwEnd	expected costs 1000 instances (penalty 10)	expected costs 1000 instances (penalty 10)	avg n. of scenarios where at least one tw is not respected	avg n. of scenarios where at least one tw is not respected
AVERAGE ALL INSTANCES:	1,182786471	0,541357843	5,853175725	0,874713025	1051,393202	1009,110715	52,93%	28,21%
100's	1,897487241	0,925640172	6,452340083	1,470458731	1187,659653	1144,825107	76,13%	44,34%
200's	0,415144904	0,128610156	5,209628822	0,234838007	905,0329409	863,3434051	28,00%	10,9%
R100	2,386567733	1,396749233	2,81246325	1,569809783	1213,84251	1199,594343	85,93%	59,17%
C100	1,261892222	0,310297067	14,75300682	1,568725	970,6626205	861,3272679	58,42%	17,92%
RC100	1,8789109	0,911237575	2,57390525	1,2108826	1392,507028	1381,60632	81,36%	51,82%
R200	0,198847836	0,216885673	0,206764345	0,256397327	904,9060114	905,5876487	18,73%	18,45%
C200	0,7656685	0,0293811	16,84840943	0,349570575	762,0756164	612,1202653	39,22%	1,46%
RC200	0,362029775	0,106460375	0,449786875	0,090461375	1048,164793	1056,48071	29,54%	9,95%

Table 5.10: Deterministic vs Stochastic: Expected costs and others results.

	deterministic	stochastic	deterministic	stochastic
instance name	avg waiting time for tw to open	avg waiting time for tw to open	avg execution duration (minutes)	avg execution duration (minutes)
R101-100	907,785327	817,915674	5,59	45,16110667
R102-100	909,001245	819,849707	5,435606667	37,04233
R103-100	745,2980956	629,6452638	4,56449	39,24308333
R104-100	298,2549562	322,3919312	4,06672	38,72688
R105-100	342,284326	382,55918	4,214353333	41,35375333
R106-100	323,9885986	284,2815672	4,137276667	34,45409667
R107-100	297,0380374	341,0774168	4,112776667	28,20755333
R108-100	224,4270388	221,2867798	4,049046667	27,46372333
R109-100	262,6502196	260,939819	4,178953333	40,27808
R110-100	223,935913	229,3763668	4,104526667	36,27235333
R111-100	335,66521	334,2199706	4,087166667	34,35839333
R112-100	164,8046996	183,77887	4,03663	30,99727
C101-100	-0,000367	163,4089234	3,97223	38,09954667
C102-100	63,6654416	362,8522582	4,000806667	38,06970667
C103-100	145,5088382	302,5497926	3,993113333	38,16800333
C104-100	174,0992434	320,934399	3,96201	35,97449333
C105-100	-0,000367	-6,10E-05	4,006956667	31,41814
C106-100	12,3785154	29,3577514	4,030173333	38,47688333
C107-100	-0,000367	16,99635	4,022793333	38,08101333
C108-100	-0,000367	-0,0002812	4,019183333	34,78852667
C109-100	-0,000367	-0,0002326	4,428126667	32,57835667
RC101-100	537,2480956	447,8470218	5,02471	42,65994667
RC102-100	473,5934086	500,1065918	5,341146667	36,36765667
RC103-100	347,349756	299,4999512	4,756533333	35,89709333
RC104-100	182,0942382	135,296924	4,973836667	27,69797333
RC105-100	502,6447024	500,4723878	5,153436667	42,40374
RC106-100	323,0769532	310,7323242	6,953713333	32,14323
RC107-100	251,8267332	247,8664304	4,19087	39,96024667
RC108-100	172,6935304	181,1856202	4,21647	38,93475
R201-100	3448,37981	3239,419287	3,877456667	33,88149333
R202-100	2734,803516	2971,644531	3,80484	34,97171333
R203-100	2935,871289	2863,172974	3,80636	34,70176
R204-100	1900,524341	1519,912061	4,387273333	33,70098333
R205-100	1936,715478	1927,672314	4,927023333	30,23901333
R206-100	1610,404676	1540,216821	4,763243333	29,27163333
R207-100	1361,081336	1222,767529	4,50224	33,21803667
R208-100	667,1875124	1024,258191	4,519423333	33,14892333
R209-100	1552,321057	1722,88861	4,79301	34,54340667
R210-100	2743,440125	2416,743506	4,328346667	30,46712667
R211-100	929,418335	868,4454832	3,473313333	28,98728
C201-100	6,10E-05	10,09082	3,742556667	33,44318
C202-100	6,10E-05	12,362671	3,737326667	32,84281667
C203-100	710,8678588	1678,6349	4,09438	34,4856
C204-100	305,1669312	19,0838748	3,850603333	33,09843
C205-100	0	1874,405212	4,055686667	25,40355333
C206-100	1335,701196	898,2570436	4,33211	34,30652667
C207-100	72,11792	815,526123	4,291873333	34,12726333
C208-100	213,7765256	53,24231	4,40668	33,66011667
RC201-100	3729,384521	3560,19646	4,46389	32,66967333
RC202-100	2554,775513	2540,582495	4,329373333	34,77847333
RC203-100	2314,923218	2216,603736	4,18603	34,31953
RC204-100	1118,250012	1181,317896	4,28533	33,21661667
RC205-100	3687,42959	3192,470728	4,000686667	35,01995333
RC206-100	1658,280005	2303,658056	3,600383333	34,94640333
RC207-100	1624,525623	1811,806396	4,715713333	30,45473333
RC208-100	0,333947475	0,333947475	4,759063333	28,98492333
	deterministic	stochastic	deterministic	stochastic
	avg waiting time for tw to open	avg waiting time for tw to open	avg execution duration min	avg execution duration min
AVERAGE ALL INSTANCES:	881,553424	930,9311192	4,351033452	34,61012667
100's	283,4934927	298,1527137	4,469781264	36,38889414
200's	1523,914091	1610,581999	4,223489506	32,69959864
R100	419,5944722	402,2768789	4,381462222	36,12988528
C100	43,96113373	132,8998778	4,048377037	36,18385222
RC100	348,8159272	327,8759064	5,076339583	37,00807958
R200	1983,64977	1937,921937	4,289320909	32,46648818
C200	329,7038192	670,2003693	4,063902083	32,67093583
RC200	2085,987804	2100,871214	4,29255875	33,04878833

Table 5.11: Deterministic vs Stochastic: Expected waiting time* for 1000 instances and execution duration. *Negative values represent the expected time that customers wait to be served.

Chapter 6

Conclusion

In this work, it was studied how the routes calculated with large neighborhood search, LNS, in a deterministic and stochastic version would compare between them when applied to the vehicle routing problem with time windows and stochastic travel and service times.

In this version of the vehicle routing problem, the time windows are soft and the vehicles only get penalized, if the vehicle arrives after the customer time window end.

The algorithms were executed for 5 seeds for each of the 56 instances from Professor Solomon[20] and an average cost calculated. Random customer instances R1 and R2, Clustered C1 and C2 and a mix of both RC1 and RC2 were used.

Three stages were implemented. In the first stage, the deterministic phase, a selection of algorithms combination for the LNS was done for the vehicle routing problem with time windows. Several combinations of destroy and repair algorithms were used with deterministic instances to find which combination creates the best solutions when comparing with results available in the literature.

I $\rightarrow$ random removal - worst removal - greedy - greedy regret

II $\rightarrow$ worst removal - greedy - greedy regret

III $\rightarrow$ random removal - greedy - greedy regret

instance name	Combination 1	Combination 2	Combination 3	Combination 4	Combination 5	Combination 6	Combination 7	Combination 8	Combination 9
avg all instances:	3,59%	24,32%	3,23%	3,41%	23,96%	3,04%	4,56%	28,69%	4,14%
Devided per group of problems									
100's	2,99%	23,26%	2,67%	2,30%	22,77%	2,03%	4,95%	28,10%	4,70%
200's	4,23%	25,47%	3,84%	4,60%	25,25%	4,12%	4,14%	29,33%	3,53%
R100	3,07%	24,61%	2,66%	2,41%	24,33%	2,14%	4,79%	30,61%	4,10%
C100	1,35%	16,23%	1,43%	0,61%	16,74%	0,56%	3,22%	17,04%	4,39%
RC100	4,71%	29,14%	4,08%	4,05%	27,22%	3,51%	7,13%	36,77%	5,93%
R200	4,95%	30,36%	4,17%	5,47%	31,05%	4,54%	4,88%	35,05%	3,74%
C200	2,69%	9,20%	2,14%	2,46%	8,05%	2,07%	2,36%	11,33%	2,52%
RC200	4,80%	35,01%	5,08%	5,55%	34,47%	5,60%	4,90%	39,47%	4,27%

Table 6.1: Deterministic: average traveled distance deviation from the optimum divided per groups of problems

IV $\rightarrow$ random removal - worst removal - greedy regret

V $\rightarrow$ worst removal - greedy regret

VI $\rightarrow$ random removal - greedy regret

VII $\rightarrow$ random removal - worst removal - greedy

VIII $\rightarrow$ worst removal - greedy

IX $\rightarrow$ random removal - greedy

In the algorithm selection, the combination VI (random removal - greedy regret) was in average the best with a deviation of 3.04% from the optimum, performing particularly well for the clustered instances problems C's being the best of all 9 algorithms.

The combination III (random removal - greedy - greedy regret) was the second best with a deviation of 3.2% from the optimum.

From all the combinations, VI took by far the most time to execute, with more than 4 minutes per instance per seed. Because the execution duration is also relevant for a very large number of instances, the combination VI might not be desirable if computational power is limited. The combination III calculated similar results in half of the time, being more convenient possibly.

In relation to the number of vehicles, VI created the routes with the smallest number due to more compact routes. The waiting time is similar for all the combinations with an average of 20% of the total route duration.

The only test for which VI did not perform so well was in the shortest route calculation. The shortest route calculated between the 5 seeds was selected and compared with the optimum. Even though in average the route from the combination VI was the best, the shortest routes were calculated by III deviating in average 1.69% from the optimum and 1.79% for VI.

In the second stage, the stochastic phase, the LNS was implemented with an expected cost included in the objective function. The best algorithm VI from the first phase was modified to calculate expected costs for 10 random scenarios each time the cost function was called. These intermediate scenarios were created randomizing the deterministic travel and service time with a +-10% parameter.

In the third and last phase, the comparison phase, the routes calculated from phase one and two were tested with 1000 new stochastic scenarios to determine how the stochastic algorithm performs against the deterministic.

In this comparison between the deterministic and the stochastic, it can be concluded that the stochastic produces routes with practically the same traveled distance as the deterministic with a difference of around 0.7%, the deterministic being shorter.

The noticeable advantage of the stochastic algorithm is that it can absorb more randomness than the deterministic.

For the 1000 scenarios, the expected number of vehicles to start the service after the time window end in the stochastic is half of the deterministic, the expected time crossed between the time window end and the service start is more than 6 times smaller for the stochastic, the number of scenarios where at least one time window is not respected is almost half in the stochastic and with the proposed configuration of penalties equal to 10 the expected cost is 4% smaller than the deterministic.

The results show that the deterministic solution is not good enough for some real life problems. The intermediate stochastic calculation added to the algorithm helped to create better routes that deal better with unpredictable situations. In average, the stochastic algorithm creates less penalized routes than the deterministic when stochastic travel and service times are used.

Appendix A

Sources

The following legends can be found in [20].

Legend: OPTIMAL R

C - A. Chabrier, “Vehicle Routing Problem with Elementary Shortest Path based Column Generation.” Forthcoming in: Computers and Operations Research (2005).

CR - W. Cook and J. L. Rich, ”A parallel cutting plane algorithm for the vehicle routing problem with time windows,” Working Paper, Computational and Applied Mathematics, Rice University, Houston, TX, 1999.

DLP - E. Danna and C. Le Pape, “Accelerating branch-and-price with local search: A case study on the vehicle routing problem with time windows,” In: Column Generation, G. Desaulniers, J. Desrosiers, and M. M. Solomon (eds.), 99-130, Kluwer Academic Publishers (2005).

IV - S. Irnich and D. Villeneuve, “The shortest path problem with k-cycle elimination ($k \geq 3$): Improving a branch-and-price algorithm for the VRPTW.” Forthcoming in: INFORMS Journal of Computing (2005).

KDMSS - N. Kohl, J. Desrosiers, O. B. G. Madsen, M. M. Solomon, and F. Soumis, ”2-Path Cuts for the Vehicle Routing Problem with Time Windows,” Transportation Science, Vol. 33 (1), 101-116 (1999).

KLM - B. Kallehauge, J. Larsen, and O.B.G. Madsen. "Lagrangean duality and non-differentiable optimization applied on routing with time windows - experimental results." Internal report IMM-REP-2000-8, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 2000.

L - J. Larsen. "Parallelization of the vehicle routing problem with time windows." Ph.D. Thesis IMM-PHD-1999-62, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 1999.

Legend: OPTIMAL C

C - A. Chabrier, "Vehicle Routing Problem with Elementary Shortest Path based Column Generation." Forthcoming in:Computers and Operations Research (2005).

CR - W. Cook and J. L. Rich, "A parallel cutting plane algorithm for the vehicle routing problem with time windows," Working Paper, Computational and Applied Mathematics, Rice University, Houston, TX, 1999.

DLP - E. Danna and C. Le Pape, "Accelerating branch-and-price with local search: A case study on the vehicle routing problem with time windows," In: Column Generation, G. Desaulniers, J. Desrosiers, and M. M. Solomon (eds.), 99-130, Kluwer Academic Publishers (2005).

IV - S. Irnich and D. Villeneuve, "The shortest path problem with k-cycle elimination ($k \geq 3$): Improving a branch-and-price algorithm for the VRPTW." Forthcoming in: INFORMS Journal of Computing (2005).

KDMSS - N. Kohl, J. Desrosiers, O. B. G. Madsen, M. M. Solomon, and F. Soumis, "2-Path Cuts for the Vehicle Routing Problem with Time Windows," Transportation Science, Vol. 33 (1), 101-116 (1999).

KLM - B. Kallehauge, J. Larsen, and O.B.G. Madsen. "Lagrangean duality and non-differentiable optimization applied on routing with time windows - experimental results." Internal report IMM-REP-2000-8, Department of MathematicalModelling, Technical University of Denmark, Lyngby, Denmark, 2000.

L - J. Larsen. "Parallelization of the vehicle routing problem with time windows." Ph.D. Thesis IMM-PHD-1999-62, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 1999.

Legend: OPTIMAL RC

C - A. Chabrier, “Vehicle Routing Problem with Elementary Shortest Path based Column Generation.” Forthcoming in:Computers and Operations Research (2005).

CR - W. Cook and J. L. Rich, "A parallel cutting plane algorithm for the vehicle routing problem with time windows," Working Paper, Computational and Applied Mathematics, Rice University, Houston, TX, 1999.

DLP - E. Danna and C. Le Pape, “Accelerating branch-and-price with local search: A case study on the vehicle routing problem with time windows,” In: Column Generation, G. Desaulniers, J. Desrosiers, and M. M. Solomon (eds.), 99-130, Kluwer Academic Publishers (2005).

IV - S. Irnich and D. Villeneuve, “The shortest path problem with k-cycle elimination ($k \geq 3$): Improving a branch-and-price algorithm for the VRPTW.” Forthcoming in: INFORMS Journal of Computing (2005).

KDMSS - N. Kohl, J. Desrosiers, O. B. G. Madsen, M. M. Solomon, and F. Soumis, "2-Path Cuts for the Vehicle Routing Problem with Time Windows," Transportation Science, Vol. 33 (1), 101-116 (1999).

KLM - B. Kallehauge, J. Larsen, and O.B.G. Madsen. "Lagrangean duality and non-differentiable optimization applied on routing with time windows - experimental results.” Internal report IMM-REP-2000-8, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 2000.

L - J. Larsen. "Parallelization of the vehicle routing problem with time windows." Ph.D. Thesis IMM-PHD-1999-62, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 1999.

Legend: BEST HEURISTICS
BVH - R. Bent and P. Van Hentenryck, "A Two-Stage Hybrid Local Search for the Vehicle Routing Problem with Time Windows," Technical Report CS-01-06, Department of Computer Science, Brown University, 2001.

BBB - J. Berger, M. Barkaoui and O. Bräysy, "A Parallel Hybrid Genetic Algorithm for the Vehicle Routing Problem with Time Windows," Working paper, Defense Research Establishment Valcartier, Canada, 2001.

CC - Z. J. Czech and P. Czarnas, "A Parallel Simulated Annealing for the Vehicle Routing Problem with Time Windows," Proc. 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing, Canary Islands, Spain, (January 9–11, 2002), 376–383.

CLM - J.-F. Cordeau, G. Laporte, and A. Mercier, "A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows," Working Paper CRT-00-03, Centre for Research on Transportation, Montreal, Canada, 2000.

GTA - L. M. Gambardella, E. Taillard, and G. Agazzi, "MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows," in New Ideas in Optimization, D. Corne, M. Dorigo and F. Glover (eds), 63-76, McGraw-Hill, London, 1999.

HG - J. Homberger and H. Gehring, "Two Evolutionary Metaheuristics for the Vehicle Routing Problem with Time Windows," INFOR, VOL. 37, 297-318, (1999).

H - J. Homberger, "Verteilt-parallele Metaheuristiken zur Tourenplanung," Gaber, Wiesbaden (2000).

IKMUY - T. Ibaraki, M. Kubo, T. Masuda, T. Uno and M. Yagiura, "Effective Local Search Algorithms for the Vehicle Routing Problem with General Time Windows," Working Paper, Department of Applied Mathematics and Physics, Kyoto University, Japan, 2001.

LLH - H. Li, A. Lim, and J. Huang, "Local Search with Annealing-like Restarts to Solve the VRPTW," Working Paper, Department of Computer

Science, National University of Singapore, 2001.

M - D. Mester, "An Evolutionary Strategies Algorithm for Large Scale Vehicle Routing Problem with Capacitate and Time Windows Restrictions," Working Paper, Institute of Evolution, University of Haifa, Israel (2002).

RT - Y. Rochat and E.D. Taillard, "Probabilistic Diversification and Intensification in Local Search for Vehicle Routing," Journal of Heuristics 1, 147-167, (1995).

RGP - L.M. Rousseau, M. Gendreau and G. Pesant, "Using Constraint-Based Operators to Solve the Vehicle Routing Problem with Time Windows," Journal of Heuristics, forthcoming.

SSSD - G. Schrimpf, J. Schneider, H. Stamm-Wilbrandt and G. Dueck, "Record Breaking Optimization Results Using the Ruin and Recreate Principle," Journal of Computational Physics 159, 139-171, (2000).

S97 - P. Shaw, "A New Local Search Algorithm Providing High Quality Solutions to Vehicle Routing Problems," Working Paper, University of Strathclyde, Glasgow, Scotland, 1997.

S98 - P. Shaw, "Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems," in Principles and Practice of Constraint Programming - CP98, Lecture Notes in Computer Science, 417-431, M. Maher and J.-F. Puget (eds), Springer-Verlag, New York, 1998.

TBGGP - E. Taillard, P. Badeau, M. Gendreau, F. Geurtin, and J.Y. Potvin, "A Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows," Transportation Science, 31, 170-186, (1997).

Bibliography

[1] Xiangyong Li, Peng Tian, Stephen C.H. Leung.

"Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm." International Journal of Production Economics Volume 125, Issue 1, May 2010, Pages 137–145.

[2] Attila A. Kovacs, Sophie N. Parragh, Karl F. Doerner, Richard F. Hartl.

"Adaptive large neighborhood search for service technician routing and scheduling problems." Journal of Scheduling October 2012, Volume 15, Issue 5, pp 579-600.

[3] Gilbert Laporte

"The Vehicle Routing Problem: An overview of exact and approximate algorithms." The Vehicle Routing Problem: An overview of exact and approximate algorithms. European Journal of Operational Research Volume 59, Issue 3, 25 June 1992, Pages 345–358.

[4] Ropke, S., Pisinger, D.

"An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows." Transportation Science, 40, Published Online: November 1, 2006 Page Range: 455 - 472.

[5] Beatrice Ombuki, Brian J. Ross and Franklin Hanshar.

"Multi-Objective Genetic Algorithms for Vehicle Routing Problem with Time Windows." Springer - Applied Intelligence February 2006, Volume 24, Issue 1, pp 17-30.

[6] L. M, Gambardella, E. Taillard and G. Agazzi.

“MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows.” New Ideas in Optimization, pp.63-76, McGraw-Hill, London, 1999.

[7] Niklas Kohl.

“Exact Methods for Time Constrained Routing and Related Scheduling Problems.” PhD Thesis, Department of Mathematical Modeling, Technical University of Denmark, 1995. pp. 234.

[8] Kenny Q. Zhu.

“A Diversity-controlling Adaptive Genetic Algorithm for the Vehicle Routing Problem with Time Windows.” Proceedings of the 15th IEEE International Conference on Tools for Artificial Intelligence (ICTAI 2003), pp.176-183, 2003.

[9] J. Homberger and H. Gehring.

“Two Evolutionary Meta-heuristics for the vehicle routing problem with time windows.” INFORMS Journal on Computing, 37(3): 297-318, 1999.

[10] W. C. Chiang and Russell.

“Simulated Annealing Metaheuristic for the Vehicle Routing Problem with Time Windows.” Annals of Operations Research, Volume 63, Issue 1, pp 3-27, 1996.

[11] W. C. Chiang and Russell.

“A Reactive Tabu Search Metaheuristic for the Vehicle Routing Problem with Time Windows.” INFORMS Journal on Computing, 9:417-430, 1997.

[12] David Pisinger, Stefan Ropke.

"A general heuristic for vehicle routing problems." Computers & Operations Research Volume 34, Issue 8, August 2007, Pages 2403–2435.

[13] Eric Prescott-Gagnon, Guy Desaulniers, Louis-Martin Rousseau.

"A Branch-and-Price-Based Large Neighborhood Search Algorithm for the Vehicle Routing Problem with Time Windows." Wiley - Networks, An International Journal. Special Issue: Route 2007 Volume 54, Issue 4, pages 190–204, December 2009.

[14] Nabila Azi, Michel Gendreau, Jean-Yves Potvin.

"An Adaptive Large Neighborhood Search for a Vehicle Routing Problem with Multiple Trips." Elsevier Science - Journal Computers and Operations Research Volume 41, January, 2014 Pages 167-173.

[15] Duygu Tas, Nico Dellaert, Tom van Woensel, Ton de Kok.

"Vehicle routing problem with stochastic travel times including soft time windows and service costs." Computers & Operations Research Volume 40, Issue 1, January 2013, Pages 214–224.

[16] Leonora Bianchi, Mauro Birattari, Marco Chiarandini, Max Manfrin, Monaldo Mastrolilli, Luis Paquete, Olivia Rossi-Doria, Tommaso Schiavinotto.

"Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands." Springer - Journal of Mathematical Modelling and Algorithms April 2006, Volume 5, Issue 1, pp 91-110.

[17] Roberto Baldacci, Aristide Mingozzi, and Roberto Roberti.

"New Route Relaxation and Pricing Strategies for the Vehicle Routing Problem." Informs - Published Online: October 1, 2011 Page Range: 1269 - 1283.

[18] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi.

"Optimization by Simulated Annealing" Science 13 May 1983: Vol. 220 no. 4598 pp. 671-680 DOI: 10.1126/science.220.4598.671.

[19] Paul Shaw.

"Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems." Springer - Principles and Practice of Constraint Programming — CP98 Lecture Notes in Computer Science Volume 1520, 1998, pp 417-431.

[20] Marius M. Solomon.

http://w.cba.neu.edu/ msolomon/problems.htm