A ring-based diversification scheme for routing problems

Rafael E. Aleman* and Xinhui Zhang

BIE Department,
Wright State University,
207 Russ Engineering Center,
3640 Colonel Glenn Hwy,
Dayton, OH 45435, USA
E-mail: rafael.enrique.aleman@gmail.com
E-mail: xinhui.zhang@wright.edu
*Corresponding author

Raymond R. Hill

Department of Operational Sciences,
2950 Hobson Way, Bldg 641, Wright-Patterson AFB,
OH 45433, USA
E-mail: rayrhill@gmail.com

Abstract: The split delivery vehicle routing problem (SDVRP) relaxes the classical vehicle routing problem (VRP) by allowing multiple vehicles to supply the demand of individual customers thereby potentially reducing costs. This article provides an up-to-date review of the SDVRP literature and presents a new solution diversification scheme based on concentric rings centred at the depot that partitions the original problem. The resulting subproblems are then solved using a constructive approach. Different ring settings produce varied partitions and thus different solutions to the original problem are obtained and improved via a variable neighbourhood descent. Computational results on available test problems demonstrate the effectiveness of the proposed algorithm and present new best solutions to some of the tested problems.

Keywords: RAC; rings-based diversification scheme; route angle control; split delivery; vehicle routing.

Reference to this paper should be made as follows: Aleman, R.E., Zhang, X. and Hill, R.R. (2009) ‘A ring-based diversification scheme for routing problems’, *Int. J. Mathematics in Operational Research*, Vol. 1, Nos. 1/2, pp.163–190.

Biographical notes: Rafael Enrique Aleman is a PhD candidate at the Department of Biomedical, Industrial and Human Factors Engineering at Wright State University, Dayton, OH. He received his BS in Electrical Engineering from Pontificia Universidad Javeriana, Colombia, in 2001 and his MS in engineering from WSU in 2005. His research interests are in the areas of optimisation, operations research and robotics, especially unmanned aerial vehicles, using mathematical models, heuristic search algorithms and applying optimisation theories of linear and integer programming. He is a member of the Institute of Industrial Engineers (IIE) and INFORMS.

Xinhui Zhang is an Assistant Professor of industrial engineering at Wright State University. He received his PhD in Operations Research and Industrial Engineering from the University of Texas at Austin. His research interests are in mathematical programming and optimisation, especially solving large problems in the field of manufacturing, logistics, transportation and supply chain management. He has participated and lead research projects such as airline crew recovery, equipment scheduling for mail processing facilities and has published in *IIE Transaction*, *IEEE Transaction*, *Journal of the Operational Research Society*, *Computer and Operations Research* and *European Journal of Operations Research*.

Raymond R. Hill is a Professor of Operations Research with the Air Force Institute of Technology. He has a PhD in Industrial and Systems Engineering from the Ohio State University. His research interests include applications of simulation, agent-based modelling, mathematical modelling and applied statistics. He is a member of the INFORMS, the Military Applications Society of INFORMS, the Military Operations Research Society (MORS) and the Institute of Industrial Engineering (IIE). He is an Associate Editor for *Military Operations Research*, *Journal of Defense Modeling and Simulation*, *Journal of Simulation* and *Int. J. Mathematics in Operational Research*.

1 Introduction

The vehicle routing problem (VRP) was introduced almost 50 years ago and is still under active investigation by practitioners and researchers. In its classical version, the problem is to effectively design routes that a fleet of homogeneous vehicles will follow to supply the demand of geographically scattered customers without exceeding the vehicle capacity and considering that customers can be visited by exactly one vehicle. Traditionally, solution techniques for the VRP have been classified as exact approaches, classical heuristic algorithms (i.e. constructive, saving, improvement, sweep, petal and matching algorithms) and metaheuristic algorithms (i.e. tabu search, genetic algorithms, simulated annealing, etc.). See Bodin and Golden (1981), Laporte et al. (2000), Toth and Vigo (2002), Cordeau et al. (2002, 2005) and Laporte (2007) for a full complete survey and description of these techniques. The split delivery vehicle routing problem (SDVRP) is a variant of the VRP where individual customer demands can be supplied by multiple vehicles. In contrast to the VRP, there is a limited number of heuristic solution techniques to solve the SDVRP including local (Dror and Trudeau, 1989) and tabu search (Archetti, Hertz and Speranza, 2006). A scatter search method (Mota, Campos and Corberan, 2007), a hybrid approach (Chen, Golden and Wasil, 2007), a memetic algorithm (Boudia, Prins and Reghioui, 2007) and a column generation approach (Jin, Liu and Eksioglu, 2008) were recently developed to effectively solve benchmark problems.

Most of the existing techniques to solve the SDVRP perform an aggressive search in certain regions of the solution space, but do not employ diversification strategies to make a better exploration of the space. Diversification methods are usually used within heuristic search methods to increase the effectiveness of the search procedure particularly on hard problems. The exploration of different regions of the solution space helps to overcome any local optimum and increase the chance of finding a global optimal solution. When the search appears to have stagnated, it is useful to examine ways to move the search process into other areas of the search space that may not have been explored. If some predefined criteria are met, the algorithm moves to a 'diversified' solution whose attributes differ from those of the already evaluated solutions. Resuming the search from a diversified solution is intended to explore new regions of the search space. Although the exploration of different regions in the solution space can help to find better solutions, the cost in processing time may be high and hence it is sometimes unattractive to diversify the search. We examine this search process tradeoff.

This article presents a new diversification scheme for routing problems applied to the SDVRP. This scheme is based on a geographical division of the problem by means of concentric rings centred at the depot that temporarily exclude a subset of customers. A partial solution to the original problem is created and the excluded customers are then incorporated into the solution by means of a constructive approach until a complete solution is obtained. Different ring settings produce varied partitions and thus different solutions to the original problem are obtained. The search is restarted from those solutions and improved via a variable neighbourhood descent (VND). The diversification scheme created is used with the constructive approach and iterative constructive approach (ICA) with route angle control (RAC) and the VND described in Aleman, Zhang and Hill (2007) to obtain SDVRP solutions. The remainder of this article is organised as follows. Section 2 provides an up-to-date literature review of the SDVRP and diversification methods applied to VRPs and SDVRPs. The proposed diversification method is described in Section 3 and the solution approach is given in Section 3.2. Computational results are presented in Section 4 with conclusions presented in Section 5.

2 Background

In this section, an up-to-date literature review of the SDVRP, a review of the constructive approach, ICA and VND approaches of Aleman, Zhang and Hill (2007), and representative diversification methods applied to VRPs and SDVRPs is presented. The review of diversification strategies is limited to how the solutions are generated and does not cover how they evolve during any subsequent search. The studies discussed address the classical VRP and the literature available for the generation of multiple solutions in the context of split deliveries. The number of studies on SDVRPs is limited and there are a limited number of solution methods for this combinatorial problem. These solution methods include some exact approaches for small-sized problems and local search, tabu search and hybrid methods for larger problems. As discussed below, a couple of publications present population-based solution methodologies including scatter search and memetic algorithms. Simulated annealing has been recently used to find solutions to the capacited VRP with heterogeneous fixed fleet and split services, but no computational results are reported on benchmark problem instances.

2.1 *Solving the split delivery vehicle routing problem*

The SDVRP is a relaxation of the classical VRP. SDVRP was first introduced by Dror and Trudeau (1989, 1990) as a variant of the classical VRP where the demand of a customer can be supplied by one or more vehicles. In the VRP vehicles with the same capacity depart from a central depot and follow designated routes to visit and fully supply the demands of geographically scattered customers. The combined demand of the customers visited by each vehicle cannot exceed the vehicle's capacity. After supplying the customer demands, all vehicles return to the central depot. The goal is to effectively design the vehicle routes to minimise the total travelled distance.

Mathematically, the SDVRP is defined on an undirected, fully connected graph $G=(V,E)$ where $V=\{0,1,\ldots,n\}$ is the set of $n+1$ nodes of the graph, and $E=\{(i,j): i,j \in V, i<j\}$ is the set of edges connecting the nodes. Node 0 represents a depot where a fleet $M=\{1,\ldots,m\}$ of identical vehicles with capacity Q are stationed, while the remaining node set $N=\{1,\ldots,n\}$ represents the customers. A non-negative cost, usually the inter-node distance, c_{ij}, is associated with every edge $(i,j)\in E$. Each customer $i \in N$ has a demand of q_i units and is located at a point (x_i, y_j) in the 2D space with respect to the depot location, (x_0, y_0). The SDVRP potentially allows reducing the operational cost of the fleet, especially when the average customer demand exceeds 10% of the vehicle capacity (as stated by Dror and Trudeau (1989). In their worst-case analysis of the SDVRP, Archetti, Savelsbergh and Speranza (2006) show that the reduction in delivery costs that can be obtained by allowing split deliveries is at most 50%, and this reduction bound is tight. Archetti, Savelsbergh and Speranza (2008) suggest that the benefits are mainly due to the reduction in the number of vehicles required to supply the customer demands. Their mathematical analysis proved that the maximum reduction in the number of vehicles is 50% and the largest reduction occurs when the mean customer demand is between 50 and 70% of the vehicle capacity and the demand variances are relatively small.

Dror and Trudeau (1989) propose a local search which uses an initial VRP solution and then uses a *k*-split interchange and route addition operators to introduce split deliveries if reductions in the objective function value are possible with the split delivery. Dror and Trudeau (1990) present some properties and valid inequalities for the SDVRP. Frizzell and Giffin (1992) use grid network distances in the problem and present a constructive approach to cluster the customers and a blocking mechanism to assign the demand of clustered customers to available vehicles. In an apparent first attempt to incorporate uncertainty into the SDVRP, Bouzaiene-Ayari, Dror and Laporte (1993) adapt the Clarke and Wright algorithm to solve the problem with stochastic demands. Dror, Laporte and Trudeau (1994) describe a branch-and-bound approach using valid inequalities and exactly solve instances with up to 20 customers. In a second paper, Frizzell and Giffin (1995) introduce time windows into the problem and use these time windows as a criteria in the constructive approach to assign the customer demands. Mullaseril, Dror and Leung (1997) adapt the local search of Dror and Trudeau (1989) to model a feed distribution problem on a cattle ranch as a SDVRP with time windows. Sierksma and Tijssen (1998) formulate a set-covering problem and a column generation method to schedule helicopters in the North Sea for crew exchange purposes.

Belenguer, Martinez and Mota (2000) study the SDVRP and estimate lower bounds using a cutting plane algorithm. They generate 14 random instances each having the same distribution of customers, but with different ranges of customer demands. Song, Lee and Kim (2002) model a distribution problem in Korea as a SDVRP to route vehicles and deliver newspapers from a central facility to different distribution centres at different times. Ho and Haugland (2004) present a tabu search to solve the SDVRP with time windows, adapt existing multiple-routes operators to the context of split deliveries and introduce the relocate split operator which changes the customer being split among two routes. Nowak (2005) study the pickup and delivery routing problem with split loads and present a heuristic approach to solve a real problem. Liu (2005) present a two-stage algorithm and a branch-and-price (B&P) approach to solve some of the problems previously solved by Belenguer, Martinez and Mota (2000). Archetti, Hertz and Speranza (2006) describe a tabu search approach to solve the SDVRP and solve seven benchmark and 42 newly generated test problem instances using random customer demands. Their results have been used in recent studies for empirical comparison of algorithm performance. Lee et al. (2006) present a shortest path approach to exactly solve the SDVRP with up to seven customers.

Belfiore and Yoshizaki (2006) study the implementation of a scatter search algorithm in an actual problem to supply 519 customers in 12 states in Brazil. The problem involves heterogeneous vehicles, time windows, accessibility constraints and split deliveries. Yu, Chen and Chu (2006) propose an approximate linear model with subtour elimination constraints, lagrangian relaxation and a heuristic method to solve the inventory routing problem with split deliveries. Jin, Liu and Bowden (2007) propose a cutting plane algorithm to optimally solve the SDVRP dividing the original problem into clustering and travelling salesman subproblems. The distances of the travelling salesman subproblems are repeatedly added as bounds to the clustering subproblems to find better solutions. Ambrosino and Sciomachen (2007) model an actual problem in Italy to plan the country-wide distribution of fresh/dry and frozen food. Mota, Campos and Corberan (2007) present a scatter search procedure that uses the minimum fleet size and produces good results on instances previously solved in the literature, particularly on problems with average customer demands less than half of the vehicle capacity. Wilck and Cavalier (2007) consider an objective function involving total distance travelled and vehicle loads. They use a constructive approach to find good solutions to small-sized problems. Chen, Golden and Wasil (2007) developed a hybrid approach combining a mixed-integer program and a record-to-record travel algorithm that produces high-quality solutions compared to the existing literature. Tavakkoli-Moghaddam et al. (2007) present a mixed-integer linear and a simulated annealing method for solving the SDVRP with heterogeneous vehicles. Boudia, Prins and Reghioui (2007) implemented a memetic algorithm with population management and produced high-quality solutions on the problems of Belenguer, Martinez and Mota (2000), Archetti, Hertz and Speranza (2006). Jin, Liu and Eksioglu (2008) present a column generation approach to estimate bounds for the SDVRP with large customer demands. The algorithm improves some of the bounds found by Belenguer, Martinez and Mota (2000).

2.2 The constructive approach, iterative constructive approach and variable neighbourhood descent solution approaches

2.2.1 Constructive approach

The parallel constructive approach of Aleman, Zhang and Hill (2007) uses an ordered list L of customers based on the distances from the depot and then inserts them into the solution under construction to initiate new routes or modify existing ones. The farthest customer from the depot is assigned the first position in L whereas the closest customer to the depot is assigned the last position. Once L is designed, customers are sequentially inserted into the routes until all customer demands are satisfied. A customer demand can be split when that demand exceeds the capacity left on the selected vehicle. In this case, any remaining demand is assigned to either an empty vehicle or the best vehicle available. The characteristic that differentiates the constructive approach from existing constructive approaches is that it uses a novel RAC mechanism to avoid the design of spatial spread routes. The intuition of RAC is to avoid overlapping routes among the vehicles. The RAC mechanism utilises the angle of a route, defined by the customers assigned to the route, to penalise insertions into far routes and favour closer routes. The polar angle of customer *i* relative to the depot, denoted θ_i, is defined as:

$$\theta_i = \arctan\frac{y_i - y_0}{x_i - x_0} \tag{1}$$

where (x_i, y_i) represents the location of customer *i* and customer 0 represents the depot. The angle of route *R* is then defined as $\theta_R = \max\{|\theta_i - \theta_j|; \forall i, j \in N \cap R\}$. The constructive approach can obtain solutions of good quality at a very low computational effort using the minimum number of vehicles $m = \left\lceil \sum_{i \in N} q_i / Q \right\rceil$, where $\lceil x \rceil$ denotes the smallest integer greater than or equal to *x*.

2.2.2 Iterative constructive approach

A limitation of the constructive approach is that the customers closest to the depot inserted later in routes due to their position in *L*, tend to deteriorate the quality of the final solution as there are a limited number of route alternatives available at the moment of their insertion. In Aleman, Zhang and Hill (2007), ICA is an iterative approach that applies adaptive memory and dynamic modification of the list *L*. The resulting ICA executes the constructive approach iteratively, but modifies *L* each iteration. Customers that cause the widest spread of routes are assigned an earlier position in *L* to ensure their insertion into more adequate routes. This is done as follows. First, the customer i^* producing the widest route is identified. Second, the closest route r^* to i^* in the current solution is selected. Third, the customer i_a spending the last resources of r^* needed to fully supply customer i^* is determined. Finally, customer i^* is relocated in *L* so that it will be inserted into the solution right before i_a. This guarantees a full service and a less expensive delivery for i^*.

2.2.3 Variable neighbourhood descent

The VND presented in Aleman, Zhang and Hill (2007) looks to improve the SDVRP solutions found with ICA. The VND uses three neighbourhoods. The first and second neighbourhoods are based on the standard customer shift and customer swap to move customers among routes. In the case of the customer shift, if the customer to be shifted is currently using split delivery, it is simply removed from the origin route and its quantity is increased in the destination route (the split is eliminated). In the case of the customer swap, if a customer to be swapped is currently using split delivery, it is also removed from the origin route and its quantity is increased in the destination route. The other customer is then moved from one route to the other. The third neighbourhood is a new operator that moves a customer among routes when the destination route does not have enough capacity to cover the demand of the moved customer. In order to make the move feasible, the load of the destination route is released by reducing its delivery to any other customer in the route with enough demand (i.e. larger than the demand of the moved customer). The unserved demand is then served by the original route. At the end, the customer shift is feasible and a new split is introduced by sharing a customer. A detailed illustration of these operators is found in Aleman, Zhang and Hill (2007) along with an empirical evaluation of constructive approach, ICA and VND performance on all available problem sets.

2.3 Solution diversification techniques

Despite using a variety of local search and local improvement methods, search heuristics can still have problems finding really good solutions to hard problems. A diversification scheme aims to move the search process into new, hopefully unvisited, regions of the solution space. Once in those new regions, the search process resumes.

One of the first studies for generating diversified solutions is by Rochat and Taillard (1995) who partition large problems into independent subproblems, each defined by sectors and regions centred at the depot, and then optimise each subproblem independently. Their diversified solutions are generated with a local search by considering various initial partitions of the problem. Tarantilis and Kiranoudis (2002) present a population-based heuristic called BoneRoute that extracts sequences of nodes, or bones, from the pool of solutions to compose partial solutions. They complete these partial solutions with a constructive approach. The diversified solutions forming the initial pool are generated with the savings algorithm of Paessens (1988). Berger and Barkaoui (2003) propose a hybrid genetic algorithm to evolve two populations using selection, recombination, mutation and migration operators. The generation of initial solutions is based on a random construction of feasible solutions. A solution is rapidly constructed through a sequential insertion heuristic which inserts customers into randomly chosen positions within routes. Customer insertion order is randomly modified to ensure unbiased solution generation.

Reimann, Doerner and Hartl (2004) present the D-Ants algorithm that uses the SavingsAnts system of Doerner et al. (2002) as the mechanism to generate a pool of solutions. In the SavingsAnts approach, solutions are generated using attractiveness values balancing the savings values of the classical Clarke and Wright algorithm and the pheromone information from previous iterations. The D-Ants approach is effective solving small- and large-scale benchmark instances as well as real world-sized problems.

In his evolutionary algorithm, Prins (2004) proposes a population of solutions initialised using three heuristic methods (Clarke andWright, 1964; Gillett and Miller, 1974; Mole and Jameson, 1976) and utilising random permutations of customers to produce a complete population. Chromosomes represent solutions in the form of giant tours formed with the ordered sequence of routes. In genetic algorithms for solving routing problems, each bit in the chromosome usually represents a customer and multiple copies of the depot are used to separate the routes. Instead of using copies of the depot, Prins utilises an optimal splitting procedure to determine the best way to separate the routes in the chromosome. The routes and the fitness value of each solution are determined by solving a min-cost path problem on an auxiliary graph.

Tarantilis (2005) employes the method of Glover (1998) to generate a collection of diversified solutions and initiate an adaptive memory solution procedure. This methodology systematically generates different permutations, or sequences, of customers and then successively assigns customers to routes to produce a VRP solution using a generalised assignment process. These diversified solutions are then improved with a tabu search and combined using elite parts of the routes to produce new solutions that update the adaptive memory components.

Only a few studies have used an initial set of solutions as a means to solve the SDVRP. Belfiore and Yoshizaki (2006) study the implementation of a scatter search for a routing problem with split deliveries, heterogeneous vehicles, time windows and accessibility constraints applied to a retail market in Brazil. Their first attempt is to solve the split deliveries with those particular side constraints. The initial solutions for the scatter search are generated using the constructive heuristic of Dullaert et al. (2002) for the problem with heterogeneous vehicles and time windows. Random elements are used to diversify the solutions.

Mota, Campos and Corberan (2007) propose a scatter search that generates a population of feasible solutions based on a giant travelling salesman problem (TSP) tour visiting all customers. The construction of a solution commences by selecting a starting customer in the giant tour and sequentially cutting that tour into individual routes where the demand of the first and last customer of each route is split when that demand does not fit the first vehicle serving it. The selection of non-consecutive starting customers helps obtain different solutions. Mota, Campos and Corberan (2007) also adapt the algorithm of Clarke andWright for split deliveries to find another fraction of the population of solutions. This adaptation of the Clarke and Wright algorithm does not guarantee feasibility, but produces diversified solutions by statistically prohibiting half of the savings used in the construction of previous solutions. Boudia, Prins and Reghioui (2007) solve the SDVRP using a memetic algorithm with population management and create the initial population both heuristically and randomly; two solutions are constructed heuristically by the algorithms of Clarke and Wright (1964) and Gillett and Miller (1974) whereas the rest of the population is generated with a random permutation of the customers. This method to generate the initial population is similar to that of Prins (2004), the only difference is the number of solutions created heuristically. The memetic algorithm produces new best SDVRP solutions for benchmark problems of Christofides and Eilon (1969) and Christofides, Mingozzi and Toth (1979) involving 75 and 120 customers with original demands and improves the state of the art algorithms (Archetti, Hertz and Speranza, 2006; Chen, Golden and Wasil, 2007) in some of the other tested problems.

3 An aggressive diversification-based search algorithm

3.1 A new diversification method

Constructive approach quickly finds SDVRP solutions whose objective function value is usually within 10% of the best existing solutions. However, those solutions display a common pattern; customers closer to the depot considerably deteriorate the quality of the overall solution. To mitigate this problem, the ICA changes the sequence of customers in L to improve upon constructive approach solutions. The VND further improves the search process using a set of local search moves. Although each method is effective and can avoid getting trapped in a local optimum region, these procedures do not guarantee a thorough exploration of the solution space. Diversification schemes are explicitly designed to improve solution space exploration.

A diversification scheme is proposed and tested based on a geographical division of the customers using rings, or spatial bands, centred at the depot. The geographic space of the problem is marked with rings of varying circumferences used to group the customers. The original problem is partially solved with the customers located inside certain rings. These selected customers are assigned to routes using the constructive approach. Subsequently, the remaining customers, belonging to the other rings, are inserted into the partial solution to yield a complete solution to the original problem.

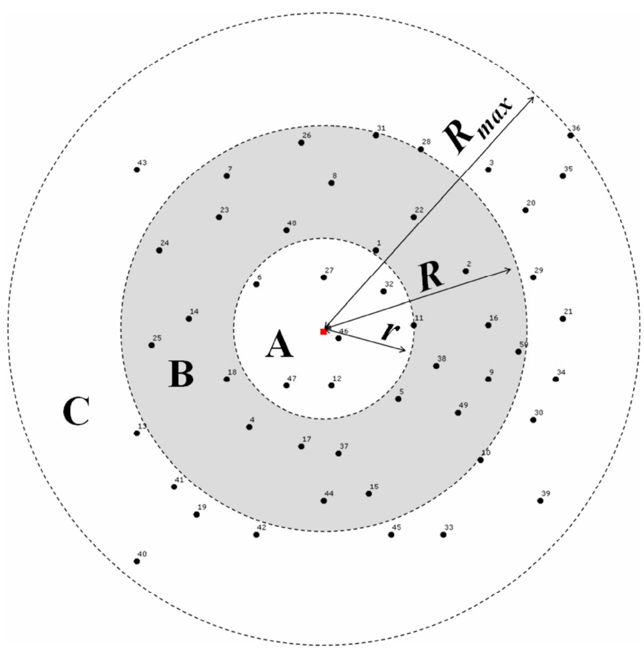
A ring is defined by an inner and outer radius, r_{in} and r_{out}, measured outward from the depot. The customers inside a ring are those whose distance from the depot is greater than r_{in} and less than or equal to $r_{\text{out}}, r_{\text{in}} < c_{0j} \le r_{\text{out}}$. Although any number of rings can be used, the proposed scheme uses three non-overlapping rings encompassing all customers. Figure 1 illustrates the geographical division used in the scheme applied to a problem involving 50 customers. In the figure, there are three rings of radius r, R and $R_{\max}$, respectively. These define the rings A, B and C. For example, ring A is defined by $r_{\text{in}} = 0$ and $r_{\text{out}} = r$, ring B is defined by $r_{\text{in}} = r$ and $r_{\text{out}} = R$, and ring C is defined by $r_{\text{in}} = R$ and $r_{\text{out}} = R_{\max}$.

The original problem includes the n customers in the set $N = \{1, \ldots, n\}$. Rings A, B, and C partition N into three subsets N_A, N_B and N_C such that:

1 $N_A \cap N_B = N_A \cap N_C = N_B \cap N_C = \phi$

2 $N = N_A \cup N_B \cup N_C$.

Figure 1 Geographical depiction of ring-based partition of 50 customers problem (see online version for colours)



A partial solution is then obtained using the constructive approach with the customers in N_A and N_C while the customers in N_B are temporarily excluded. The complete solution is then obtained when the customers in N_B are inserted into the partial solution also using the constructive approach. Varying values of r and R varies the size of B, and subsequently N_B, the exclusion set in the method, yielding a variety of solutions. This approach yields a much more aggressive diversification strategy than obtained just using local improvement methods such as found with ICA.

The range of diversified solutions based on varied sizes of B is next examined. Figure 2 shows the solutions to Problem 1 and Figure 3 shows solutions to Problem 6 from Archetti, Hertz and Speranza (2006) obtained with the constructive approach and the diversification scheme using different settings for ring B. In each figure, solution values are shown as a percentage deviation from the objective function value obtained using the basic constructive approach. A negative deviation indicates that the diversification directly provides a better solution whereas a positive deviation indicates the diversified solution is not as good. The width and location of ring B varies as a function of $R_{\max}$. The inner radius r varies in the range $[0.1R_{\max}, 0.9R_{\max}]$ in steps of 0.1 $R_{\max}$ whereas the outer radius R varies in the range $[0.2R_{\max}, 1.0R_{\max}]$ also in steps of $0.1R_{\max}$. The ring settings r and R are shown in each figure in the form r–R on the horizontal axis. Note that different settings for the inner and outer radius can be used to generate a larger number of diversified solutions. This diversification scheme can be applied to any routing problem. Also, note that since this scheme generates diverse solutions, the initial solution quality is not a primary concern; local improvement is ultimately applied to the diverse solutions in the computational procedure.

Figure 2 Diversified solutions for Problem 1 involving 50 customers with the original demands (see online version for colours)

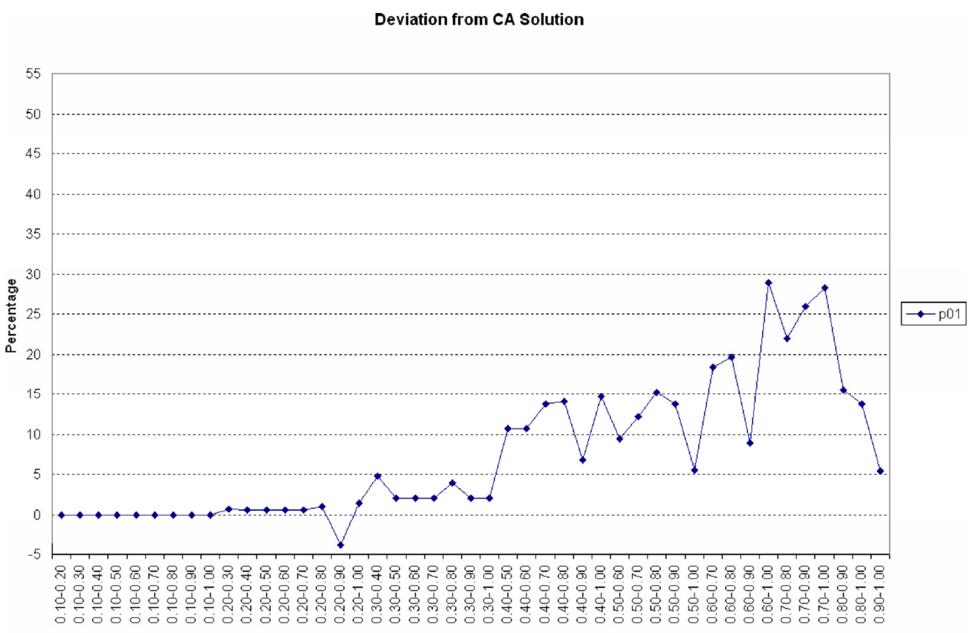
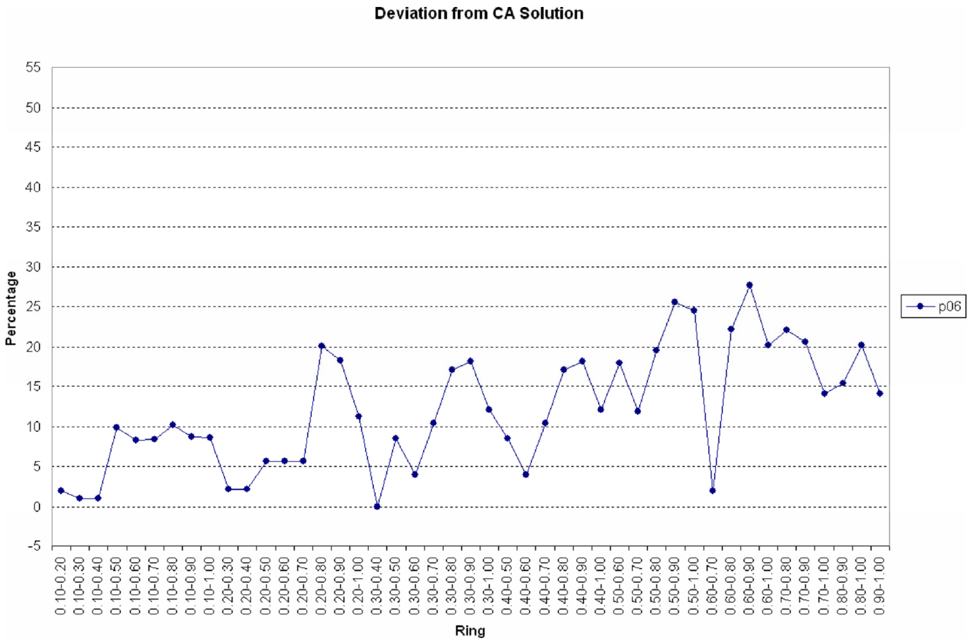


Figure 3 Diversified solutions for Problem 6 involving 120 customers with the original demands (see online version for colours)



Figures 4–7 illustrate four solutions taken from the generated set shown in Figure 2. Figure 4 illustrates the solution found with the basic constructive approach while Figure 5 illustrates the solution in the diversified set which is the most different from the basic constructive approach solution shown in Figure 4. In this case, the number of edges appearing in one solution, but not in the other are counted to measure the difference between two solutions. Figure 6 illustrates the solution in the diversified set with the lowest objective function value z, and finally Figure 7 illustrates the solution in the diversified set with the highest objective function value z. These figures reinforce our confidence that the ring-based diversification process does in fact generate a variety of solutions.

3.2 The iterative constructive approach + variable neighbourhood descent with diversification (iVNDiv) solution approach

The proposed solution approach couples the algorithms presented in Aleman, Zhang and Hill (2007) with the new diversification methodology. The idea is to solve the problem using the ICA and VND of Aleman, Zhang and Hill (2007) and then restart the search from different points in the solution space when the diversification phase commences. The result is a multi-start algorithm for the SDVRP.

The details of iVNDiv are given in Algorithms 1 and 2. The set of solutions is generated using Algorithm 1. This algorithm utilises various ring settings to partition the original problem in a variety of ways and produce different solutions. The algorithm constructs solutions with the constructive approach of Aleman, Zhang and Hill (2007). Once a complete solution is generated, it is added to the full set of solutions. Before adding solutions, their objective function values are verified to guarantee unique elements in the set of diverse solutions.

Figure 4 Illustration of the basic constructive approach solution for Problem 1 involving 50 customers with the original demands (z = 578.83) (see online version for colours)

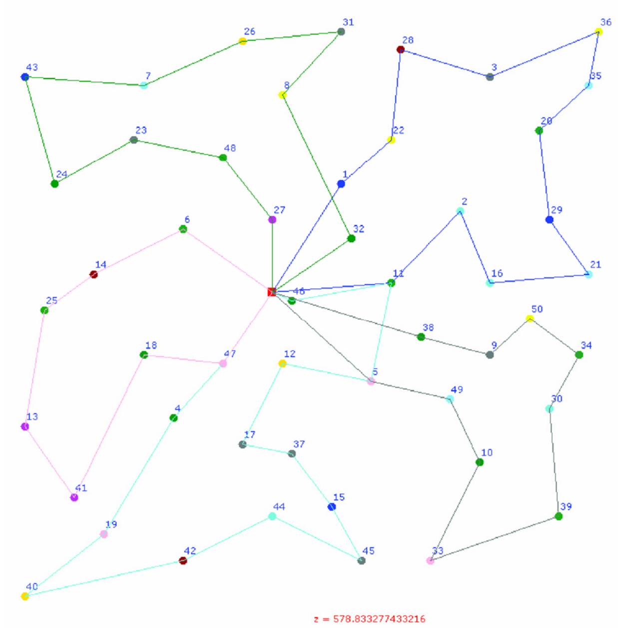


Figure 5 Illustration of a solution taken from the set of solutions for Problem 1 that most differs from the basic constructive approach solution (z = 640.58 and ring settings 0.40–0.50) (see online version for colours)

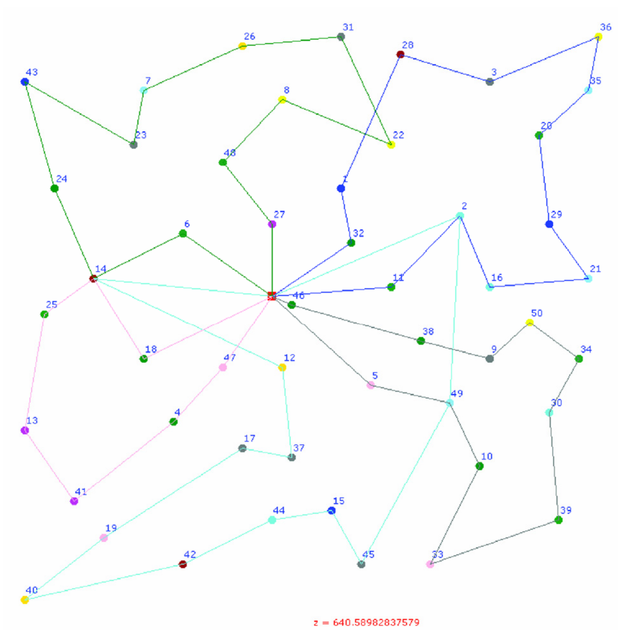


Figure 6 Illustration of a solution taken from the set of solutions for Problem 1 with the lowest objective function value (z = 556.56 and ring settings 0.20–0.90) (see online version for colours)

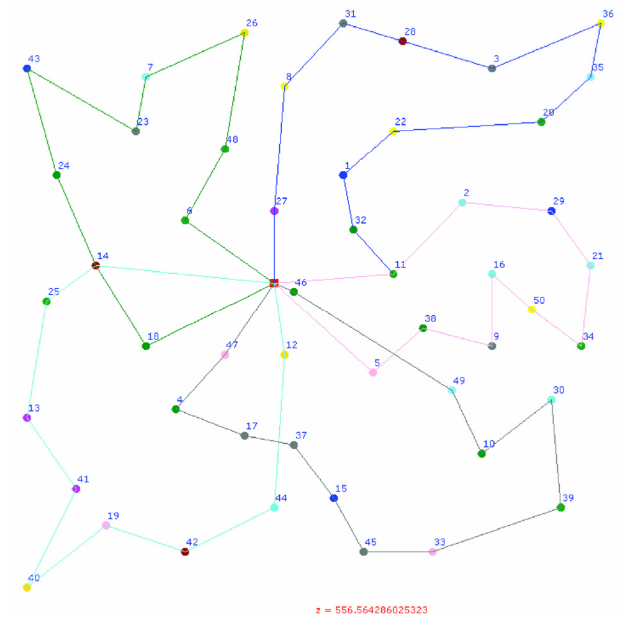
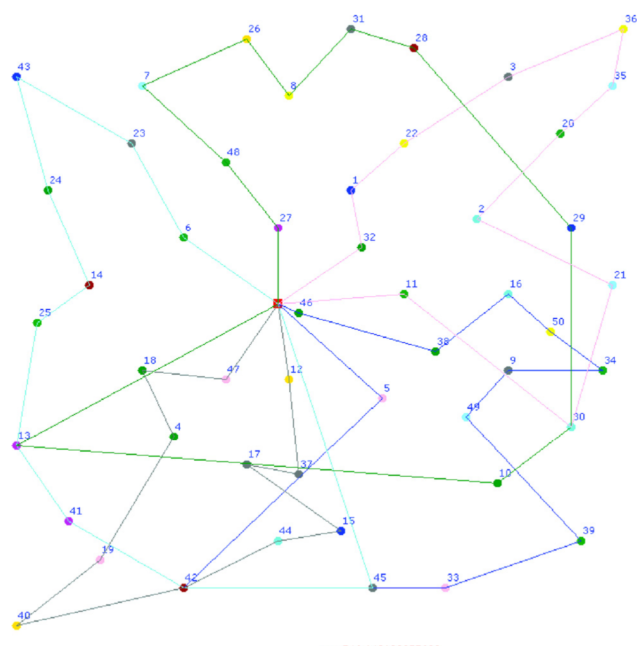


Figure 7 Illustration of a solution taken from the set of solutions for Problem 1 with the highest objective function value (z = 746.14 and ring settings 0.60–1.00) (see online version for colours)



With the full set of diversified solutions, the ICA and VND of Aleman, Zhang and Hill (2007) are used to improve the solutions in the diversified set. The number of solutions from the set used as starting points in the solution space varies. The more solutions used, the higher the computational cost. A maximum of five starting solutions are used in the proposed iVNDiv to balance the quality of the solution and their running times. These five solutions are the best solutions in the set of diversified solutions. The iVNDiv is presented in Algorithm 2.

Algorithm 1 - Generation of set S of solutions

```
Set S = ∅.
Let N = {1, ..., n} be the set of customers in the original problem.
Let N_A ⊂ N be the subset of customers in accordance with c_0i ≤ r for i ∈ N.
Let N_B ⊂ N be the subset of customers in accordance with r < c_0i ≤ R.
Let N_C ⊂ N be the subset of customers in accordance with R < c_0i ≤ R_max for i ∈ N.
width = 0.10
for r = 0 ; r < 1.00 ; r = r + width do
  for R = r + width ; R ≤ 1.00 ; R = R + width do
    Design a list L_1 with all customers i ∈ N_A ∪ N_C.
    Order the customers in L_1 by nonincreasing distance from the depot.
    With L_1, use the CA to find a partial solution s_1 to the problem.
    Define a list L_2 with all customers i ∈ N_B.
    Order the customers in L_2 by nonincreasing distance from the depot.
    Using the CA, insert the customers in L_2 sequentially into s_1 to produce a complete solution s_2.
    Set S = S ∪ {s_2}.
  end for
end for
return S
```

Algorithm 2 - ICA+VND With Diversification (iVNDiv)

```
Execute Algorithm 1 to generate set S of solutions.
Set x_b as the best solution with solution value f(x_b) = ∞
jumpCounter = 0.
maxJumps = 5.
while jumpCounter < maxJumps do
  Select the solution x from set S with the lowest objective function value and remove it from S.
  Execute the ICA+VND approach of Aleman et al. (2007) to improve x and obtain x'.
  if f(x') < f(x_b) then
    x_b = x'
  end if
  jumpCounter = jumpCounter + 1.
end while
return x_b
```

4 Computational results

The iVNDiv was implemented in C# and experiments were carried out using a Pentium 4, 2.8 GHz, 512MB of RAM. Our iVNDiv algorithm was tested on problem sets available in the literature including Belenguer, Martinez and Mota (2000), Archetti, Hertz and Speranza (2006), Chen, Golden and Wasil (2007) and Jin, Liu and Eksioglu (2008). These sets have been solved with existing approaches which are compared to iVNDiv in our empirical analysis. The tested instances are identified using the form p-aaa-nnn. The first field, p, is an alphabetical character to identify the publication where the problem is identified. The second field, aaa, is a string of variable length corresponding to the name of the instance adopted in the publication, whereas the third field is a three-digit integer denoting the number of customers excluding the depot. The first field, p, takes the following values: a, b, c, j (Archetti, Hertz and Speranza, 2006; Belenguer, Martinez and Mota, 2000; Chen, Golden and Wasil, 2007; Jin, Liu and Bowden, 2007, respectively).

The instances used by Archetti, Hertz and Speranza (2006) are the same Problems 1–5, 11 and 12 given in Christofides and Eilon (1969) and Christofides, Mingozzi and Toth (1979) involving 50–199 customers in addition to the depot. In problems 1–5, customers are randomly distributed in the plane, while they are clustered in problems 11 and 12. From those seven problems, Archetti, Hertz and Speranza (2006) generated 42 more by randomly modifying the customer demands at different intervals. These random problems are unavailable. However, Mota, Campos and Corberan (2007) used the same algorithm of Archetti, Hertz and Speranza (2006) to generate their problems. We obtained those from Boudia, Prins and Reghioui (2007). In our analysis, the problems with random customer demands have the exact same demand values as in Mota et al. and Boudia et al. Belenguer, Martinez and Mota (2000) used a total of 25 problems: 11 TSPLIB problems involving 21–100 customers and 14 randomly generated problems from the TSPLIB (eil51, eil76 and eil101). The same vehicle capacity $Q = 160$ is used in each problem and the customer demands are randomly generated within six intervals expressed as a funtion of Q, as in Dror and Trudeau (1989) and Archetti, Hertz and Speranza (2006). Chen, Golden and Wasil (2007) recently generated a new set of 21 problems involving 8–288 customers. Each problem has a geometric symmetry with customers located in concentric circles around the depot. Jin, Liu and Bowden (2007) used a TSPLIB instance with 21 customers and generated four problems involving 18–22 customers.

Computational results for the Archetti et al. instances are presented in Tables 1–3. The solution values from the existing algorithms are reproduced from the corresponding references. The existing algorithms are the VND (ICA + VND) of Aleman, Zhang and Hill (2007), the scatter search (SS) of Mota, Campos and Corberan (2007), the memetic algorithm with population management (MA|PM) of Boudia, Prins and Reghioui (2007), the three tabu searches (Splitabu, Splitabu-DT and Fast-Splitabu) of Archetti, Hertz and Speranza (2006) and the hybrid algorithm (EMIP + VRTR) of Chen, Golden and Wasil (2007). In Table 1, columns with header m contain the number of vehicles in the final iVNDiv, ICA + VND, scatter search and MA|PM solutions, which is always the minimum possible, whereas columns with header m' contain the number of vehicles in the final solutions found with the tabu searches of Archetti, Hertz and Speranza (2006). Results in Table 1 show that the ICA + VND algorithm is clearly dominated by its counterpart with the proposed diversification scheme, i.e. iVNDiv. Compared to scatter search, our iVNDiv provides better solutions specially in problems with large customer demands in the ranges [0.10–0.90], [0.30–0.70] and [0.70–0.90], where the largest cost reduction can occur, as shown in Dror and Trudeau (1989) and Archetti, Savelsbergh and Speranza (2008). Although we find solutions of similar quality and perform better in one problem, the MA|PM clearly dominates our iVNDiv in this problem set.

Table 1 Computational results of iVNDiv on instances

			iVNDiv(a)	ICA+VND(a)		SS(a)		MA|PM(a)			Splitabu(b)		Splitabu-DT(b)		Fast-Splitabu(b)		EMIP+VRTR(c)	
Problem	Demand	m	z	z	IMP	z	IMP	z	IMP	m'	z	IMP	z	IMP	z	IMP	z	IMP
a-01-050		5	524.61	540.82	**-3.09**	531.02	**-1.22**	524.61	0.00	5	530.06	**-1.04**	533.55	**-1.70**	533.55	**-1.70**	524.61	0.00
a-02-075		10	851.24	880.28	**-3.41**	839.75	1.35	823.89	3.21	10	851.67	**-0.05**	849.54	0.20	849.54	0.20	840.18	1.30
a-03-100		8	852.74	854.13	**-0.16**	835.82	1.98	829.44	2.73	8	846.18	0.77	835.62	2.01	835.74	1.99	-	-
a-04-150		12	1074.11	1088.91	**-1.38**	1056.92	1.60	1042.37	2.95	12	1062.20	1.11	1069.84	0.40	1088.31	**-1.32**	1041.99	2.99
a-05-199		16	1368.67	1390.55	**-1.60**	1340.44	2.06	1311.59	4.17	16	1367.82	0.06	1342.85	1.89	1346.39	1.63	1307.40	4.48
a-06-120		7	1201.83	1223.28	**-1.79**	1042.97	13.22	1041.20	13.37	7	1084.73	9.74	1056.01	12.13	1056.01	12.13	1043.18	13.20
a-07-100		10	824.78	824.82	0.00	820.92	0.47	819.56	0.63		822.60	0.26	825.32	**-0.07**	825.32	**-0.07**	819.56	0.63
Average					-1.63		2.78		3.87			1.55		2.12		1.84		3.77
a-01-050	[0.01-0.10]	3	471.92	473.22	**-0.28**	460.79	2.36	460.79	2.36		463.85	1.71	463.76	1.73	463.76	1.73	457.21	3.12
a-02-075	[0.01-0.10]	4	597.46	617.65	**-3.38**	602.67	**-0.87**	600.06	**-0.44**		607.66	**-1.71**	605.24	**-1.30**	605.24	**-1.30**	598.25	**-0.13**
a-03-100	[0.01-0.10]	5	745.35	789.16	**-5.88**	729.67	2.10	726.81	2.49		772.79	**-3.68**	752.20	**-0.92**	752.20	**-0.92**	-	-
a-04-150	[0.01-0.10]	8	891.98	893.49	**-0.17**	883.05	1.00	875.61	1.83		894.98	**-0.34**	890.95	0.11	893.66	**-0.19**	875.16	1.89
a-05-199	[0.01-0.10]	10	1073.55	1079.04	**-0.51**	1039.51	3.17	1018.71	5.11		1073.60	0.00	1056.27	1.61	1068.70	0.45	1040.20	3.11
a-06-120	[0.01-0.10]	6	1087.80	1101.14	**-1.23**	979.57	9.95	976.57	10.23		1095.75	**-0.73**	1084.70	0.29	1084.70	0.29	985.17	9.43
a-07-100	[0.01-0.10]	5	673.54	673.54	0.00	633.80	5.90	649.73	3.53		662.80	1.59	648.74	3.68	653.70	2.95	651.44	3.28
Average					-1.63		3.37		3.59			-0.45		0.74		0.43		3.45
a-01-050	[0.10-0.30]	**10**	766.19	777.75	**-1.51**	769.60	**-0.45**	751.41	1.93	11	764.40	0.23	761.40	0.62	761.40	0.62	723.57	5.56
a-02-075	[0.10-0.30]	**15**	1099.47	1099.47	0.00	1074.01	2.32	1074.46	2.27	16	1099.03	0.04	1095.32	0.38	1095.32	0.38	1081.10	1.67
a-03-100	[0.10-0.30]	**20**	1425.90	1452.52	**-1.87**	1416.48	0.66	1392.85	2.32	22	1428.87	**-0.21**	1424.81	0.08	1426.18	**-0.02**	-	-
a-04-150	[0.10-0.30]	**29**	1978.01	1978.01	0.00	1974.70	0.17	1878.71	5.02	32	1940.67	1.89	1918.25	3.02	1924.67	2.70	1844.96	6.73
a-05-199	[0.10-0.30]	**38**	2464.65	2502.54	**-1.54**	2435.08	1.20	2340.14	5.05	41	2419.98	1.81	2384.15	3.27	2393.82	2.87	2258.66	8.36
a-06-120	[0.10-0.30]	**23**	2806.92	2806.92	0.00	2783.10	0.85	2720.38	3.08	26	2900.99	**-3.35**	2918.71	**-3.98**	2921.30	**-4.08**	2568.90	8.48
a-07-100	[0.10-0.30]	20	1428.27	1428.27	0.00	1423.49	0.34	1417.28	0.77		1470.96	**-2.99**	1462.01	**-2.36**	1467.32	**-2.73**	1414.33	0.98
Average					-0.70		0.73		2.92			-0.37		0.15		-0.04		5.30
a-01-050	[0.10-0.50]	**15**	1039.89	1045.93	**-0.58**	1025.91	1.34	988.31	4.96	16	1007.68	3.10	1008.67	3.00	1008.67	3.00	943.86	9.23
a-02-075	[0.10-0.50]	**22**	1478.67	1503.02	**-1.65**	1484.62	**-0.40**	1413.80	4.39	24	1450.11	1.93	1443.62	2.37	1449.74	1.96	1393.53	5.76
a-03-100	[0.10-0.50]	**29**	1956.13	1957.55	**-0.07**	1926.15	1.53	1845.30	5.67	33	1887.83	3.49	1894.72	3.14	1897.12	3.02	-	-
a-04-150	[0.10-0.50]	**43**	2671.62	2685.33	**-0.51**	2649.97	0.81	2561.65	4.12	49	2634.09	1.40	2632.71	1.46	2650.61	0.79	2532.93	5.19
a-05-199	[0.10-0.50]	**56**	3411.38	3450.84	**-1.16**	3310.71	2.95	3191.25	6.45	63	3298.19	3.32	3284.47	3.72	3314.00	2.85	3202.57	6.12
a-06-120	[0.10-0.50]	**34**	4026.53	4085.36	**-1.46**	3996.29	0.75	3934.39	2.29	40	4166.78	**-3.48**	4206.12	**-4.46**	4261.66	**-5.84**	3687.06	8.43
a-07-100	[0.10-0.50]	29	2007.11	2046.15	**-1.95**	2022.30	**-0.76**	1994.59	0.62		2030.04	**-1.14**	2029.99	**-1.14**	2053.86	**-2.33**	1973.34	1.68
Average					-1.05		0.89		4.07			1.23		1.16		0.49		6.07
a-01-050	[0.10-0.90]	**25**	1522.43	1547.32	**-1.64**	1580.77	**-3.83**	1467.06	3.64	26	1493.92	1.87	1469.92	3.45	1470.39	3.42	1408.34	7.49
a-02-075	[0.10-0.90]	**37**	2200.51	2212.93	**-0.56**	2233.08	**-1.48**	2102.58	4.45	41	2121.28	3.60	2124.43	3.46	2142.32	2.64	2056.54	6.54
a-03-100	[0.10-0.90]	**48**	2865.86	2925.13	**-2.07**	2932.34	**-2.32**	2780.95	2.96	56	2826.61	1.37	2794.08	2.50	2837.78	0.98	-	-
a-04-150	[0.10-0.90]	**73**	4165.18	4192.50	**-0.66**	4185.68	**-0.49**	4045.87	2.86	84	4006.28	3.82	3909.72	6.13	3987.62	4.26	3945.38	5.28
a-05-199	[0.10-0.90]	**93**	5184.57	5192.06	**-0.14**	5085.64	1.91	4941.22	4.69	107	5039.65	2.80	4853.83	6.38	5044.72	2.70	5094.61	1.74
a-06-120	[0.10-0.90]	**56**	6364.87	6483.06	**-1.86**	6361.46	0.05	6318.37	0.73	67	6560.33	**-3.07**	6583.97	**-3.44**	6730.80	**-5.75**	6079.14	4.49
a-07-100	[0.10-0.90]	48	3156.31	3178.28	**-0.70**	3187.44	**-0.99**	3113.72	1.35		3158.09	**-0.06**	3101.53	1.74	3109.04	1.50	3162.22	**-0.19**
Average					-1.09		-1.02		2.96			1.48		2.89		1.39		4.23
a-01-050	[0.30-0.70]	**25**	1540.39	1557.52	**-1.11**	1568.04	**-1.80**	1477.01	4.11	26	1509.31	2.02	1496.90	2.82	1496.95	2.82	1408.68	8.55
a-02-075	[0.30-0.70]	**37**	2238.98	2241.59	**-0.12**	2228.90	0.45	2132.16	4.77	39	2175.99	2.81	2160.51	3.51	2164.22	3.34	2112.61	5.64
a-03-100	[0.30-0.70]	**49**	2941.64	2945.19	**-0.12**	2986.33	**-1.52**	2858.87	2.81	53	2914.79	0.91	2870.50	2.42	2921.95	0.67	-	-
a-04-150	[0.30-0.70]	**73**	4165.18	4192.50	**-0.66**	4185.68	**-0.49**	4045.87	2.86	80	4146.75	0.44	4039.70	3.01	4139.06	0.63	4011.74	3.68
a-05-199	[0.30-0.70]	**96**	5363.65	5366.06	**-0.04**	5265.01	1.84	5155.36	3.88	103	5368.92	**-0.10**	5102.84	4.86	5312.67	0.95	5088.08	5.14
a-06-120	[0.30-0.70]	**58**	6545.50	6591.40	**-0.70**	6481.09	0.98	6424.71	1.85	65	6866.34	**-4.90**	6639.55	**-1.44**	6829.05	**-4.33**	6123.96	6.44
a-07-100	[0.30-0.70]	49	3225.63	3318.08	**-2.87**	3248.76	**-0.72**	3155.69	2.17		3222.03	0.11	3038.02	5.82	3187.92	1.17	3134.56	2.82
Average					-0.80		-0.18		3.21			0.19		3.00		0.75		5.38
a-01-050	[0.70-0.90]	**40**	2215.34	2215.34	0.00	2312.48	**-4.38**	2154.35	2.75	42	2176.39	1.76	2165.21	2.26	2167.93	2.14	2056.01	7.19
a-02-075	[0.70-0.90]	**60**	3304.24	3341.26	**-1.12**	3387.86	**-2.53**	3200.35	3.14	61	3229.46	2.26	3180.64	3.74	3241.77	1.89	3067.19	7.17
a-03-100	[0.70-0.90]	**80**	4429.21	4455.14	**-0.59**	4580.98	**-3.43**	4312.95	2.62	82	4368.77	1.36	4302.31	2.86	4418.83	0.23	-	-
a-04-150	[0.70-0.90]	**119**	6482.11	6513.36	**-0.48**	6479.46	0.04	6267.48	3.31	123	6354.21	1.97	6196.36	4.41	6513.36	**-0.48**	5950.35	8.20
a-05-199	[0.70-0.90]	**158**	8329.55	8368.35	**-0.47**	8323.72	0.07	8081.58	2.98	162	8343.95	**-0.17**	7944.63	4.62	8586.55	**-3.09**	7207.04	13.48
a-06-120	[0.70-0.90]	**95**	10302.16	10302.16	0.00	10158.32	1.40	10063.47	2.32	99	10550.57	**-2.41**	10304.08	**-0.02**	11007.73	**-6.85**	8941.79	13.20
a-07-100	[0.70-0.90]	80	5028.78	5058.76	**-0.60**	5065.26	**-0.73**	4919.48	2.17		4960.75	1.35	4867.79	3.20	4965.94	1.25	4779.13	4.96
Average					-0.46		-1.37		2.76			0.88		3.01		-0.70		9.04

Notes: *z* denotes objective function value obtained; IMP denotes percentage objective function reduction over iVNDiv; *m* denotes number of vehicles in final iVNDiv, ICA + VND scatter search and MA|PM final solutions; *m′* denotes number of vehicles in final Splitabu, Splitabu-DT and Fast-Splitabu solutions; [a]tested instances were generated by Mota, Campos and Corberan (2007); [b]tested instances were generated by Archetti, Savelsbergh and Speranza (2006), iVNDiv run using similar problem generator; [c]tested instances were generated by Chen, Golden and Wasil (2007).

Table 2 Comparison of iVNDiv to best known solutions for instances

		Best Known			iVNDiv		% above Best	
Problem	Demand	z	m	Source	z	m	z	m
a-01-050		521.00	5	Belenguer et al. (2000)	524.61	5	0.69	0.00
a-02-075		823.89	10	Boudia et al. (2007)	851.24	10	3.32	0.00
a-03-100		829.44	8	Boudia et al. (2007)	852.74	8	2.81	0.00
a-04-150		1041.99		Chen et al. (2007)	1074.11	12	3.08	
a-05-199		1307.40		Chen et al. (2007)	1368.67	16	4.69	
a-06-120		1041.20	7	Boudia et al. (2007)	1201.83	7	15.43	0.00
a-07-100		819.56	10	Boudia et al. (2007)	824.78	10	0.64	0.00
Average							4.38	0.00
a-01-050	[0.01-0.10]	457.21		Chen et al. (2007)	471.92	3	3.22	
a-02-075	[0.01-0.10]	598.25		Chen et al. (2007)	597.46	4	**-0.13**	
a-03-100	[0.01-0.10]	726.81	5	Boudia et al. (2007)	745.35	5	2.55	0.00
a-04-150	[0.01-0.10]	875.16		Chen et al. (2007)	891.98	8	1.92	
a-05-199	[0.01-0.10]	1018.71	10	Boudia et al. (2007)	1073.55	10	5.38	0.00
a-06-120	[0.01-0.10]	976.57	6	Boudia et al. (2007)	1087.80	6	11.39	0.00
a-07-100	[0.01-0.10]	633.80	5	Mota et al. (2007)	673.54	5	6.27	0.00
Average							4.37	0.00
a-01-050	[0.10-0.30]	723.57		Chen et al. (2007)	766.19	10	5.89	
a-02-075	[0.10-0.30]	1074.01	15	Mota et al. (2007)	1099.47	15	2.37	0.00
a-03-100	[0.10-0.30]	1392.85	20	Boudia et al. (2007)	1425.90	20	2.37	0.00
a-04-150	[0.10-0.30]	1844.96		Chen et al. (2007)	1978.01	29	7.21	
a-05-199	[0.10-0.30]	2258.66		Chen et al. (2007)	2464.65	38	9.12	
a-06-120	[0.10-0.30]	2568.90		Chen et al. (2007)	2806.92	23	9.27	
a-07-100	[0.10-0.30]	1414.33		Chen et al. (2007)	1428.27	20	0.99	
Average							5.32	0.00
a-01-050	[0.10-0.50]	943.86		Chen et al. (2007)	1039.89	15	10.17	
a-02-075	[0.10-0.50]	1393.53		Chen et al. (2007)	1478.67	22	6.11	
a-03-100	[0.10-0.50]	1845.30	29	Boudia et al. (2007)	1956.13	29	6.01	0.00
a-04-150	[0.10-0.50]	2532.93		Chen et al. (2007)	2671.62	43	5.48	
a-05-199	[0.10-0.50]	3191.25	56	Boudia et al. (2007)	3411.38	56	6.90	0.00
a-06-120	[0.10-0.50]	3687.06		Chen et al. (2007)	4026.53	34	9.21	
a-07-100	[0.10-0.50]	1973.34		Chen et al. (2007)	2007.11	29	1.71	
Average							6.51	0.00
a-01-050	[0.10-0.90]	1408.34		Chen et al. (2007)	1522.43	25	8.10	
a-02-075	[0.10-0.90]	2056.54		Chen et al. (2007)	2200.51	37	7.00	
a-03-100	[0.10-0.90]	2746.75	56	Archetti et al. (2006)	2865.86	48	4.34	-14.29
a-04-150	[0.10-0.90]	3849.73	84	Archetti et al. (2006)	4165.18	73	8.19	-13.10
a-05-199	[0.10-0.90]	4737.47	107	Archetti et al. (2006)	5184.57	93	9.44	-13.08
a-06-120	[0.10-0.90]	6079.14		Chen et al. (2007)	6364.87	56	4.70	
a-07-100	[0.10-0.90]	3010.50		Archetti et al. (2006)	3156.31	48	4.84	
Average							6.66	-13.49
a-01-050	[0.30-0.70]	1408.68		Chen et al. (2007)	1540.39	25	9.35	
a-02-075	[0.30-0.70]	2112.61		Chen et al. (2007)	2238.98	37	5.98	
a-03-100	[0.30-0.70]	2764.25	53	Archetti et al. (2006)	2941.64	49	6.42	-7.55
a-04-150	[0.30-0.70]	3967.11	80	Archetti et al. (2006)	4165.18	73	4.99	-8.75
a-05-199	[0.30-0.70]	5001.45	103	Archetti et al. (2006)	5363.65	96	7.24	-6.80
a-06-120	[0.30-0.70]	6123.96		Chen et al. (2007)	6545.50	58	6.88	
a-07-100	[0.30-0.70]	2882.12		Archetti et al. (2006)	3225.63	49	11.92	
Average							7.54	-7.70
a-01-050	[0.70-0.90]	2056.01		Chen et al. (2007)	2215.34	40	7.75	
a-02-075	[0.70-0.90]	3067.19		Chen et al. (2007)	3304.24	60	7.73	
a-03-100	[0.70-0.90]	4278.83	82	Archetti et al. (2006)	4429.21	80	3.51	-2.44
a-04-150	[0.70-0.90]	5950.35		Chen et al. (2007)	6482.11	119	8.94	
a-05-199	[0.70-0.90]	7207.04		Chen et al. (2007)	8329.55	158	15.58	
a-06-120	[0.70-0.90]	8941.79		Chen et al. (2007)	10302.16	95	15.21	
a-07-100	[0.70-0.90]	4773.59		Archetti et al. (2006)	5028.78	80	5.35	
Average							9.15	-2.44
Overall							6.28	-3.30

Notes: *z* denotes objective function value obtained; *m* denotes number of vehicles in final solution.

Table 3 Running times of iVNDiv on instances

Problem	Demand	iVNDiv[(a)]	ICA+VND[(a)]	SS[(b)]	MA|PM[(c)]	Splitabu[(d)]	Splitabu-DT[(d)]	EMIP+VTR[(e)]
a-01-050		54.9063	10.89	24.80	8.53	17.00	13.20	1.80
a-02-075		83.2813	9.81	61.66	35.72	63.60	35.80	4.00
a-03-100		319.3281	43.50	108.80	34.59	59.60	57.60	-
a-04-150		1361.1563	129.23	261.28	103.69	439.60	389.00	10.00
a-05-199		3284.6406	534.83	352.31	353.84	1900.40	386.40	18.10
a-06-120		3414.4063	257.30	131.34	50.92	40.00	38.40	5.60
a-07-100		126.0781	21.02	108.41	42.89	86.40	49.00	3.70
a-01-050	[0.01-0.10]	33.7031	4.52	26.86	12.38	9.00	4.80	1.90
a-02-075	[0.01-0.10]	303.7656	51.28	68.80	18.75	42.40	13.00	25.80
a-03-100	[0.01-0.10]	2194.2344	415.47	125.06	37.12	58.60	31.20	-
a-04-150	[0.01-0.10]	3461.4375	666.20	352.09	100.27	258.00	172.80	107.80
a-05-199	[0.01-0.10]	15505.2183	3750.44	963.84	356.22	753.80	525.80	413.40
a-06-120	[0.01-0.10]	3952.6719	341.59	163.28	72.98	60.60	42.40	36.40
a-07-100	[0.01-0.10]	1207.4219	222.42	80.56	34.97	71.20	57.80	53.90
a-01-050	[0.10-0.30]	19.7656	1.59	26.31	10.22	27.20	21.80	3.40
a-02-075	[0.10-0.30]	73.0469	13.19	86.02	34.14	78.00	45.40	57.00
a-03-100	[0.10-0.30]	190.5313	34.09	98.00	78.06	121.60	95.80	-
a-04-150	[0.10-0.30]	878.5469	164.19	10.06	147.89	544.80	393.20	308.00
a-05-199	[0.10-0.30]	1457.1563	248.83	19.11	347.14	1224.40	754.80	618.50
a-06-120	[0.10-0.30]	558.5625	54.25	11.33	144.19	516.00	142.60	136.40
a-07-100	[0.10-0.30]	123	22.56	151.25	43.27	85.20	146.00	126.50
a-01-050	[0.10-0.50]	18.1563	2.81	3.84	12.49	55.60	28.20	14.70
a-02-075	[0.10-0.50]	67.7969	11.25	6.09	37.38	71.00	123.20	214.00
a-03-100	[0.10-0.50]	154.4688	25.16	7.55	28.39	205.60	136.20	-
a-04-150	[0.10-0.50]	625.8281	111.66	16.17	224.89	563.80	739.20	630.50
a-05-199	[0.10-0.50]	2173.8438	339.36	20.64	436.20	3810.60	2668.00	1775.70
a-06-120	[0.10-0.50]	358.5625	40.53	63.80	163.14	259.00	268.00	220.70
a-07-100	[0.10-0.50]	107.4688	11.92	41.23	51.31	188.20	292.80	287.60
a-01-050	[0.10-0.90]	16.3594	2.83	3.91	21.42	34.00	60.80	55.40
a-02-075	[0.10-0.90]	71.1094	10.80	6.64	46.11	311.20	193.40	401.10
a-03-100	[0.10-0.90]	126.5156	19.00	9.16	84.38	412.20	648.60	-
a-04-150	[0.10-0.90]	671.3594	141.27	25.03	244.91	1822.40	2278.00	2220.00
a-05-199	[0.10-0.90]	3650.5938	662.77	71.09	725.69	2598.40	3297.20	3038.10
a-06-120	[0.10-0.90]	458.9063	42.00	15.86	196.14	1037.00	877.80	722.80
a-07-100	[0.10-0.90]	96.9844	12.89	9.08	52.13	523.40	259.60	251.20
a-01-050	[0.30-0.70]	15.3281	2.20	4.25	24.53	51.80	48.60	47.90
a-02-075	[0.30-0.70]	80.2969	11.28	7.14	51.78	184.40	128.60	509.60
a-03-100	[0.30-0.70]	103.9375	15.14	10.36	100.16	453.80	810.20	-
a-04-150	[0.30-0.70]	675.3906	143.05	19.38	244.86	1512.40	3008.00	3028.30
a-05-199	[0.30-0.70]	3026.2188	349.97	120.28	749.94	2279.40	3565.60	3035.70
a-06-120	[0.30-0.70]	469.1719	59.20	17.16	271.39	476.60	658.60	605.40
a-07-100	[0.30-0.70]	110.0469	13.69	9.73	91.31	411.00	777.80	716.50
a-01-050	[0.70-0.90]	18.7031	2.59	4.13	22.91	159.80	106.40	135.40
a-02-075	[0.70-0.90]	58.0469	10.25	7.66	27.48	436.60	869.20	811.00
a-03-100	[0.70-0.90]	94.9844	14.31	12.06	55.75	1891.40	1398.40	-
a-04-150	[0.70-0.90]	584.8438	93.78	131.91	401.62	8782.80	10223.20	10038.80
a-05-199	[0.70-0.90]	2124.6563	460.89	165.28	571.70	11346.80	21849.20	12542.30
a-06-120	[0.70-0.90]	636.7188	59.28	20.17	298.08	2032.60	1825.60	725.40
a-07-100	[0.70-0.90]	178.1875	20.70	9.19	180.11	1865.00	1004.40	1024.30

Notes: [a]P4, 512MB, 2.8 GHz; [b]P4, 1.0GB, 2.4 GHz; [c]PC 3.0 GHz; [d]P4, 256MB, 2.4 GHz; [e]P4, 512MB, 1.7 GHz.

The comparison with the tabu searches and EMIP + VRTR on the problems with random customer demands is not straightforward. First, there is a potential discrepancy regarding the actual customer demand values used by Archetti, Hertz and Speranza (2006). Second, the values reproduced from Chen, Golden and Wasil (2007) correspond to the median values from 30 solution instances for each random problem. Across the board, there is no apparent dominance of iVNDiv over the tabu searches, but the number of vehicles in the final iVNDiv solutions is generally lower than in the tabu solutions. In some cases, the tabu searches use up to 14 more vehicles than iVNDiv (see for example problem a-05–199 with demands in [0.10–0.90]), which can lead to solutions with lower objective function values, but possibly higher operational costs in actual problems. Chen, Golden and Wasil (2007) do not provide the number of vehicles used in their final EMIP + VRTR solutions, but the fleet size is apparently a decision variable. Our iVNDiv is able to improve EMIP + VRTR in only one problem.

Table 2 shows the best known SDVRP solutions available in the literature for the instances of Archetti et al. and a comparison with the iVNDiv solutions. Note that EMIP + VRTR values are median values from 30 instances. The best known solutions are presented with their solution values z, the number m of vehicles and the publication where they are reproduced from. Problem a-01–050 with original demands is optimally solved by Belenguer, Martinez and Mota (2000) and its solution value was calculated using integer inter-node distances. The tables also provide the percentage improvement of the iVNDiv solutions over the best ones. Out of the 49 problems, 26 best solutions have been found with the EMIP + VRTR hybrid approach of Chen, Golden and Wasil (2007), 10 with the MA|PM memetic algorithm, 10 with the tabu searches and 2 with the scatter search. This table is also important to recall the effect of the number of vehicles in the final solutions and that iVNDiv, scatter search and MA|PM utilise the smallest fleet possible. The iVNDiv improves the best known solution to problem a-02–075 in the range [0.01–0.10] and generally uses less vehicles than other approaches.

The running times in seconds are provided in Table 3. The characteristics of the machines where the different approaches were run are listed at the bottom of the table. By comparing the results for iVNDiv and ICA + VND in the table, we notice the impact of the diversification scheme on the running time. However, the diversification procedure does not deteriorate the running time considerably as the average customer demands get larger, which is not the case for the tabu searches and EMIP + VRTR. With the tabu searches, one reason for the increase in the computational effort may be the neighbourhood structure used in the search. For each customer, Archetti, Hertz and Speranza (2006) evaluate removals from the visiting vehicles and/or insertions into other vehicles to find the best way to move to a neighbour solution. This operator may be particularly expensive when the number of vehicles is large, as in problems with larger customer demands. In the case of EMIP + VRTR, the number of endpoints increases with the number of routes as one or two endpoints and the closest neighbours to each endpoint are considered for each route. For larger average customer demands, the resulting mixed-integer program can be considerable in size and more difficult to solve. Figure 8 shows the average running times of the existing algorithms on the instances of Archetti, Hertz and Speranza (2006) grouped by demand range. Note how the time increases with the customer demand for the tabu searches and EMIP + VRTR. Because both the constructive approach and VND of Aleman, Zhang and Hill (2007) evaluate the cheapest insertion position, a larger number of customers per route increases the complexity of iVNDiv. The large average running time of iVNDiv on problems in the range [0.01–0.10] is caused by a large number of stops per route. We found our running times relate to the ratio n/m (see Table 4), which is an estimation of the expected stops per route. This dependency is illustrated in Figure 9.

Table 4 Expected stops per route for problems

Demand range	*Stops per route (n/m)*	*Running time (sec)*
[0.70–0.90]	1.25	528.02
[0.30–0.70]	2.04	640.06
[0.10–0.90]	2.08	727.40
[0.10–0.50]	3.46	500.88
[0.10–0.30]	5.09	471.52
Original	11.73	1234.83
[0.01–0.10]	19.15	3808.35

Figure 8 Average running times vs. demand range for instances (see online version for colours)

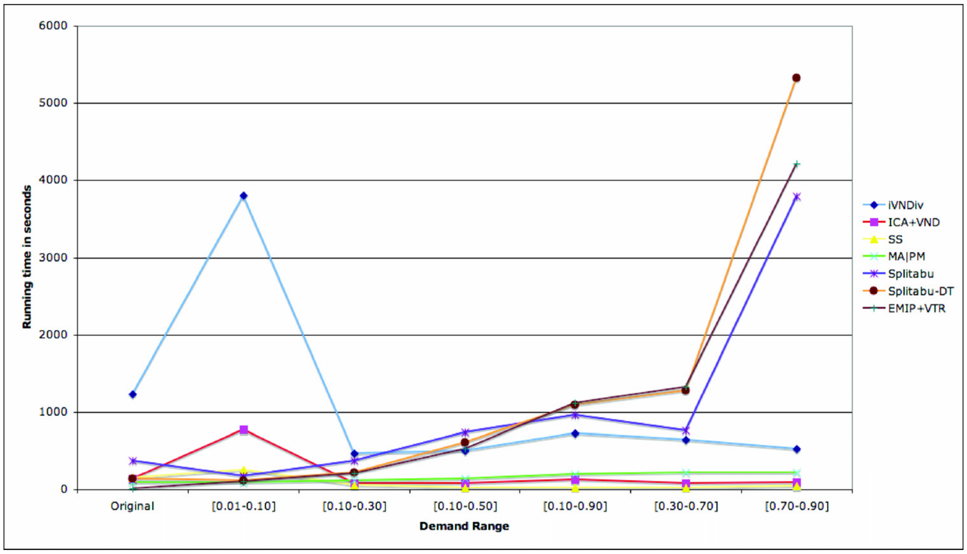
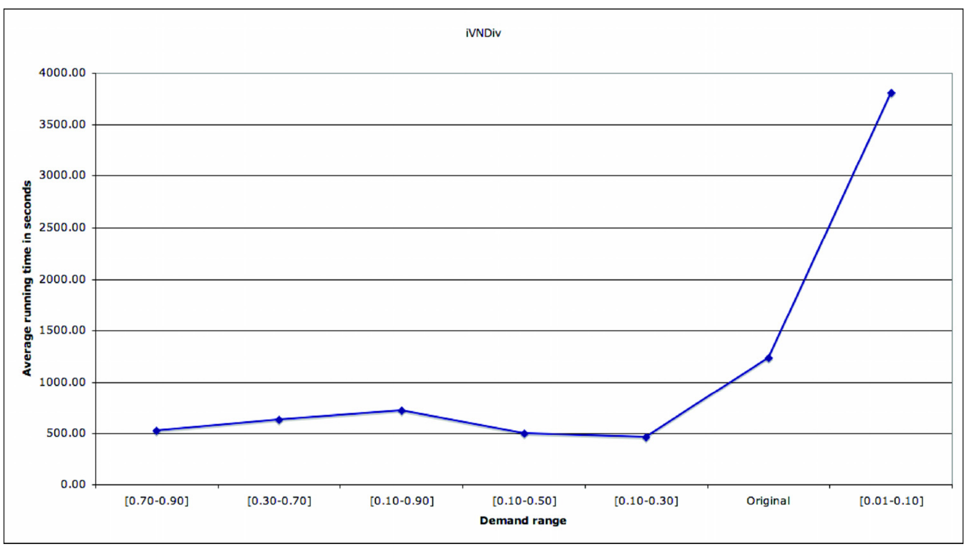


Figure 9 Average running times of iVNDiv vs. demand range for instances (see online version for colours)



Tables 5 and 6 show the computational results on the 25 problems given in Belenguer, Martinez and Mota (2000). The iVNDiv solution values are compared with the bounds found by Belenguer et al. with a cutting plane and a heuristic approach, the solution values found with MA|PM, the bounds obtained with the B&P approach of Liu (2005), the solution values found with EMIP + VRTR, and the bounds produced by the column generation approach of Jin, Liu and Eksioglu (2008). Unpublished values are omitted from the tables. Bold type indicate iVNDiv providing a better feasible solution. Note that upper bounds and solution values obtained by Belenguer et al. and MA|PM are calculated by rounding inter-node distance values to the nearest integer.

Table 5 Computational results of iVNDiv on some TSPLIB instances

	iVNDiv		*Belenguer, Martinez and Mota (2000)*			*MA|PM*		
Problem	z^a	CPU^b	UB^a	*LB*	*% ALB*	Z^a	CPU^c	*IMP*
b-eil22–021	375	4.19	375	375	0.00	375	4.11	0.00
b-eil23–022	570	3.42	569	569	0.18	569	5.47	0.18
b-eil30–029	510	14.47	510	508	0.39	503	5.70	1.37
b-eil33–032	851	14.03	835	833	2.12	835	5.19	1.88
b-eil51–050	521	54.91	521	511.57	1.81	521	7.28	0.00
b-eilA76–075	847	83.28	832	782.7	7.59	828	35.94	2.24
b-eilB76–075	1,055	79.00	1,023	937.47	11.14	1,019	13.09	3.41
b-eilC76–075	746	148.20	735	706.01	5.36	738	14.75	1.07
b-eilD76–075	695	140.83	683	659.43	5.12	682	23.12	1.87
b-eilA101–100	843	319.33	817	793.48	5.87	818	25.25	2.97
b-eilB101–100	1,122	185.84	1,077	1,005.85	10.35	1,082	21.81	3.57

	iVNDiv		*B&P*			*Jin, Liu and Eksioglu (2008)*			
Problem	*Z*	CPU^b	*UB*	*LB*	*% ALB*	*UB*	*LB*	CPU^d	*% ALB*
b-eil22–021	375.28	4.19	**376.00**	373.60	0.45	–	–	–	–
b-eil23–022	569.75	3.42	**608.00**	564.30	0.96	–	–	–	–
b-eil30–029	512.72	14.47	**515.30**	507.20	1.08	–	–	–	–
b-eil33–032	853.10	14.03	**873.40**	830.20	2.68	–	–	–	–
b-eil51–050	524.61	54.91	**558.50**	507.60	3.24	–	–	–	–
b-eilA76–075	851.24	83.28	**900.70**	800.30	5.98	–	–	–	–
b-eilB76–075	1059.57	79.00	**1163.10**	965.70	8.86	**1063.75**	981.14	45084.00	7.40
b-eilC76–075	753.29	148.20	**809.30**	711.20	5.59	–	–	–	–
b-eilD76–075	699.35	140.83	**768.80**	652.30	6.73	–	–	–	–
b-eilA101–100	852.74	319.33	**910.20**	797.50	6.48	–	–	–	–
b-eilB101–100	1139.27	185.84	**1174.10**	1013.90	11.00	–	–	–	–

Notes: *z* denotes objective function value obtained; CPU denotes running time in seconds; IMP denotes percentage objective function reduction over iVNDiv; % ALB denotes percent iVNDiv above lower bound; MA|PM uses one more vehicle than the minimum fleet size on instance b-eil30–029; [a]Objective function value obtained with euclidean distances truncated to the nearest integer; [b]P4, 512MB, 2.8 GHz; [c]PC 3.0 GHz; [d]P4, 2GB, 2.8 GHz.

Table 7 shows the computational results on the new 21 problems generated by Chen et al. The table contains solution values *z*, running times in seconds and the percentage improvements of iVNDiv over the EMIP + VRTR hybrid approach. Bold text is used to indicate the new best solutions found so far for this new problem set. As far as we know, this is the first time this problem set is used after Aleman, Zhang and Hill (2007) and Chen, Golden and Wasil (2007). Out of the 21 problems, the iVNDiv improves the solution values in 15 cases and equals the EMIP + VRTR solution in the problem with eight customers. The iVNDiv is computationally faster than EMIP + VRTR in all the cases. ICA + VND performs similar to iVNDiv in terms of solution values, although iVNDiv performs better in seven cases. It is difficult to see how the running time of EMIP + VRTR is affected by the problem size as it uses the maximum amount of computing time to solve the endpoint mixed-integer program for problems with at least 24 customers (Chen, Golden and Wasil, 2007).

Table 6 Computational results of iVNDiv on the random problems

	iVNDiv		*Belenguer, Martinez and Mota (2000)*			*MA|PM*		
Problem	Z^a	CPU^b	UB^a	*LB*	*% ALB*	z^a	CPU^c	*IMP*
b-S51D1–050	466	40.53	458	454	2.58	458	8.77	1.72
b-S51D2–050	725	28.34	**726**	676.63	6.67	707	7.44	2.48
b-S51D3–050	994	14.70	972	905.22	8.93	945	7.84	4.93
b-S51D4–050	1,672	16.53	**1,677**	1520.67	9.05	1,578	11.98	5.62
b-S51D5–050	1,385	13.94	**1,440**	1272.86	8.10	1,351	16.72	2.45
b-S51D6–050	2,211	16.83	**2,327**	2113.03	4.43	2,182	9.92	1.31
b-S76D1–075	600	476.27	594	584.87	2.52	592	15.23	1.33
b-S76D2–075	1,138	46.94	**1,147**	1020.32	10.34	1,089	30.5	4.31
b-S76D3–075	1,485	53.34	1,474	1346.29	9.34	1,427	12.89	3.91
b-S76D4–075	2,160	51.84	**2,257**	2011.64	6.87	2,117	8.76	1.99
b-S101D1–100	740	2125.58	716	700.56	5.33	717	49.75	3.11
b-S101D2–100	1,426	217.91	1,393	†1270.97	10.87	1,372	31.72	3.79
b-S101D3–100	1,974	146.61	**1,975**	†1739.66	11.87	1,891	33.98	4.20
b-S101D5–100	2,970	104.05	2,915	†2630.43	11.43	2,854	18.66	3.91

Table 6 Computational results of iVNDiv on the random problems (continued)

	iVNDiv		*B&P*			*EMIP + VRTR*			Jin, Liu and *Eksioglu (2008)*			
Problem	*Z*	*CPU*[b]	*UB*	*LB*	*% ALB*	*Z*	*CPU*[d]	*IMP*	*UB*	*LB*	*CPU(e)*	*% ALB*
b-S51D1–050	471.92	40.53	**513.90**	449.90	4.67	–	–	–	–	–	–	–
b-S51D2–050	731.01	28.34	**1296.50**	556.70	23.85	–	–	–	723.37	694.98	5,978	4.93
b-S51D3–050	1001.22	14.70	986.00	956.00	4.52	–	–	–	968.85	922.72	607	7.84
b-S51D4–050	1680.66	16.53	1654.00	1623.00	3.43	1586.50	201.74	5.60	1657.61	1505.35	260	10.43
b-S51D5–050	1389.40	13.94	**1434.00**	1416.00	–1.91	1355.50	201.62	2.44	**1439.92**	1297.46	46	6.62
b-S51D6–050	2218.23	16.83	**2316.00**	2270.00	–2.33	2197.80	301.90	0.92	**2300.21**	2108.59	243	4.94
b-S76D1–075	606.47	476.27	–	–	–	–	–	–	–	–	–	–
b-S76D2–075	1143.36	46.94	–	–	–	–	–	–	**1185.72**	1066.17	12,806	6.75
b-S76D3–075	1490.08	53.34	–	–	–	–	–	–	**1504.94**	1397.43	2,030	6.22
b-S76D4–075	2173.61	51.84	**2205.00**	2178.00	–0.20	2136.40	601.92	1.71	**2219.07**	2019.91	1,813	7.07
b-S101D1–100	749.19	2125.58	–	–	–	–	–	–	–	–	–	–
b-S101D2–100	1443.44	217.91	–	–	–	–	–	–	**1474.51**	1349.77	47,658	6.49
b-S101D3–100	1988.78	146.61	–	–	–	–	–	–	**2012.86**	1837.33	7,959	7.62
b-S101D5–100	2984.48	104.05	–	–	–	2846.20	645.99	4.63	2954.96	2725.5	847	8.68

Notes: [†]earlier termination of the algorithm due to memory overflow (Belenguer, Martinez and Mota, 2000); *z* denotes objective function value obtained; CPU denotes running time in seconds; IMP denotes percentage objective function reduction over iVNDiv; % ALB denotes percent iVNDiv above lower bound; [a]objective function value obtained with euclidean distances truncated to the nearest integer; [b]P4, 512MB, 2.8 GHz; [c]PC 3.0 GHz; [d]P4, 512MB, 1.7 GHz; [e]P4, 2GB, 2.8 GHz.

Table 7 Computational results of iVNDiv on instances

	EMIP + VRTR		*ICA + VND*		*iVNDiv*		
Problem	*z*	*CPU*[a]	*z*	*CPU*[b]	*z*	*CPU*[b]	*m*
c-SD01–008	**228.28**	0.7	228.28	0.06	228.28	0.19	6
c-SD02–016	714.40	54.4	**708.28**	0.22	**708.28**	1.48	12
c-SD03–016	430.61	67.3	**430.58**	0.17	**430.58**	0.58	12
c-SD04–024	**631.06**	400	635.84	0.55	635.84	2.31	18
c-SD05–032	1408.12	402.7	**1390.57**	0.69	**1390.57**	5.55	24
c-SD06–032	**831.21**	408.3	831.24	0.94	831.24	2.95	24
c-SD07–040	3714.40	403.2	**3640.00**	1.03	**3640.00**	8.13	30
c-SD08–048	5200.00	404.1	**5068.28**	1.75	**5068.28**	11.91	36
c-SD09–048	**2059.84**	404.3	2071.03	2.91	2071.03	19.73	36
c-SD10–064	2749.11	400	**2747.83**	3.58	**2742.84**	33.27	48
c-SD11–080	13612.12	400.1	**13280.00**	3.97	**13280.00**	35.16	60
c-SD12–080	7399.06	408.3	7279.97	4.00	**7265.70**	43.13	60
c-SD13–096	10367.06	404.5	**10110.58**	5.80	**10110.58**	50.97	72
c-SD14–120	11023.00	5021.7	10893.50	15.49	**10829.25**	141.77	90
c-SD15–144	15271.77	5042.3	**15168.28**	18.33	**15168.28**	191.66	108
c-SD16–144	**3449.05**	5014.7	3635.27	39.71	3580.07	2120.14	108
c-SD17–160	26665.76	5023.6	26559.93	17.42	**26556.13**	179.61	120
c-SD18–160	14546.58	5028.6	14440.59	40.38	**14372.80**	366.14	120
c-SD19–192	20559.21	5034.2	20191.19	27.64	**20188.62**	330.06	144
c-SD20–240	40408.22	5,053	39813.49	63.18	**39803.13**	633.33	180
c-SD21–288	**11491.67**	5,051	11799.60	738.49	11682.09	9387.55	216

Notes: *z* denotes objective function value obtained; CPU denotes running time in seconds.; *m* denotes the number of vehicles in the final iVNDiv and ICA + VND solutions; [a]P4, 512MB, 1.7 GHz; [b]P4, 512MB, 2.8 GHz.

Finally, Table 8 shows the solution values and running times in seconds of iVNDiv with respect to the optimal solutions found with the exact approach of Jin, Liu and Bowden (2007). iVNDiv generates high-quality solutions that are 1.19% above optimality on average for the tested problems. We note how difficult it is to exactly solve small-sized problems (up to 21 customers) while iVNDiv finds the optimal solution in three of the tested problems and is quite close to optimality in another problem at a low computational effort.

Table 8 Computational results of iVNDiv versus optimality on instances

	TSVI		*iVNDiv*		*% Above*
Problem	*z**	*CPU[a](hour)*	*z*	*CPU[b]*	*Optimality*
j-Eil22–021	375.28	17	375.28	4.19	**0.00**
j-J1–018	127.39	13	127.49	1.73	0.08
j-J2–021	388.44	84	388.44	4.59	**0.00**
j-J3–022	367.93	17	389.54	3.73	5.87
j-J4–022	372.2	13	372.2	5.64	**0.00**

Notes: *z** denotes optimal objective function value; *z* denotes objective function value obtained; CPU denotes running time: [a]—; [b]P4, 512MB, 2.8 GHz.

5 Conclusions

Diversification of a local search process can be computationally expensive, but of benefit on harder optimisation problems. This research presents a new diversification method for routing problems based on a novel use of spatially varied concentric rings around the routing depot. A set of diversified solutions are used to restart the VND search process of Aleman, Zhang and Hill (2007). A comprehensive empirical test of this new diversification method was conducted and the reported results show the utility of this new diversification scheme. The proposed diversification strategy can be used to solve any variant of the VRP as long as the constructive approach considers the corresponding side constraints. Although our diversification scheme is based on a geographical division of the problem by means of concentric rings centred at the depot, this geographical division can be modified. For example, instead of excluding all the customers in a complete ring, it may be divided into sectors to exclude only the customers in those regions of the ring.

There are a couple future avenues of research. For instance, we employ an aggressive diversification scheme focusing on the best solutions. Future studies might consider examining the worse solutions as a means of potentially maximising the distance between a current solution and a new search area. Another avenue would be to use the ring-based diversification method as a vocabulary building mechanism to construct either high-quality solutions, or to diversify, solutions whose components are selected based on low frequency of use. These avenues are currently under investigation.

Acknowledgements

The authors are grateful to the referees for their suggestions and comments on our article. We want to thank Claudia Archetti for providing us the computational results of the tabu searches and Reghioui Mohamed for facilitating the instances with random demands used in their computational experiments.

References

Aleman, R.E., Zhang, X. and Hill, R.R. (2007) 'An adaptive memory algorithm for the split delivery vehicle routing problem', *Submitted to the Special Issue on Advances in Metaheuristics of the Journal of Heuristics*, November.

Ambrosino, D. and Sciomachen, A. (2007) 'A food distribution network problem: a case study', *IMA Journal of Management Mathematics*, Vol. 18, pp.33–53, January.

Archetti, C., Hertz, A. and Speranza, M.G. (2006) 'A tabu search algorithm for the split delivery vehicle routing problem', *Transportation Science*, Vol. 40, pp.64–73, February.

Archetti, C., Savelsbergh, M.W.P. and Speranza, M.G. (2006) 'Worst-case analysis for split delivery vehicle routing problems', *Transportation Science*, Vol. 40, pp.226–234, May.

Archetti, C., Savelsbergh, M.W.P. and Speranza, M.G. (2008) 'To split or not to split: that is the question', *Transportation Research. Part E*, Vol. 44, pp.114–123.

Belenguer, J., Martinez, M. and Mota, E. (2000) 'A lower bound for the split delivery vehicle routing problem', *Operations Research*, Vol. 48, pp.801–810, September/October.

Belfiore, P.P. and Yoshizaki, H.T.Y. (2006) 'Scatter search for heterogeneous fleet vehicle routing problems with time windows and split deliveries', *Produção: Uma Publicação da Associação Brasileira de Engenharia de Produção*, Vol. 16, pp.455–469, September/December.

Berger, J. and Barkaoui, M. (2003) 'A new hybrid genetic algorithm for the capacitated vehicle routing problem', *Journal of the Operational Research Society*, Vol. 54, pp.1254–1262, December.

Bodin, L. and Golden, B. (1981) 'Classification in vehicle routing and scheduling', *Networks*, Vol. 11, pp.97–108.

Boudia, M., Prins, C. and Reghioui, M. (2007) 'An effective memetic algorithm with population management for the split delivery vehicle routing problem', *Hybrid Metaheuristics*, Vol. 4771, Lecture Notes in Computer Science, pp.16–30. Springer Berlin/Heidelberg, September.

Bouzaiene-Ayari, B., Dror, M. and Laporte, G. (1993) 'Vehicle routing with stochastic demands and split deliveries', *Foundations of Computing and Decision Sciences*, Vol. 18, pp.63–69.

Chen, S., Golden, B. and Wasil, E. (2007) 'The split delivery vehicle routing problem: applications, algorithms, test problems, and computational results', *Networks*, Vol. 49, pp.318–329, July.

Christofides, N. and Eilon, S. (1969) 'An algorithm for the vehicle-dispatching problem', *Operational Research Quarterly*, Vol. 20, pp.309–318, September.

Christofides, N., Mingozzi, A. and Toth, P. (1979) 'The vehicle routing problem', In N. Christofides, A. Mingozzi, P. Toth and C. Sandi (Eds), *Combinatorial Optimization*. Chichester, England, UK: John Wiley & Sons Ltd, Vol. 11, pp.315–338.

Clarke, G. and Wright, J. (1964) 'Scheduling of vehicles from a central depot to a number of delivery points', *Operations Research*, Vol. 12, pp.568–581, July.

Cordeau, J-F., Gendreau, M., Hertz, A., Laporte, G. and Sormany, J-S. (2005) 'New heuristics for the vehicle routing problem', in A. Langevin and D. Riopel (Eds), *Logistics Systems: Design and Optimization*. New York, NY: Springer Science + Business Media, Vol. 9, pp.279–297.

Cordeau, J-F., Gendreau, M., Laporte, G., Potvin, J-Y. and Semet, F. (2002) 'A guide to vehicle routing heuristics', *Journal of the Operational Research Society*, Vol. 53, pp.512–522, May.

Doerner, K., Gronalt, M., Hartl, R.F., Reimann, M., Strauss, C. and Stummer, M. (2002) 'Savingsants for the vehicle routing problem', in S. Cagnoni, J. Gottlieb, E. Hart, M. Middendorf and G.R. Raidl (Eds), *Applications of Evolutionary Computing: Proceedings EvoWorkshops 2002*, Vol. 2279, Lecture Notes in Computer Science, Chapter 2, pp.11–20. Kinsale, Ireland: Springer Berlin/Heidelberg, April.

Dror, M. and Trudeau, P. (1989) 'Savings by split delivery routing', *Transportation Science*, Vol. 23, pp.141–145, May.

Dror, M. and Trudeau, P. (1990) 'Split delivery routing', *Naval Research Logistics*, Vol. 37, pp.383–402, June.

Dror, M., Laporte, G. and Trudeau, P. (1994) 'Vehicle routing with split deliveries', *Discrete Applied Mathematics*, Vol. 50, pp.239–254, May.

Dullaert, W., Janssens, G.K., Sörensen, K. and Vernimmen, B. (2002) 'New heuristics for the fleet size and mix vehicle routing problem with time windows', *Journal of the Operational Research Society*, Vol. 53, pp.1232–1238, November.

Frizzell, P. and Giffin, J. (1992) 'The bounded split delivery vehicle routing problem with grid network distances', *Asia-Pacific Journal of Operational Research*, Vol. 9, pp.101–116.

Frizzell, P. and Giffin, J. (1995) 'The split delivery vehicle scheduling problem with time windows and grid network distances', *Computers and Operations Research*, Vol. 22, pp.655–667, July.

Gillett, B. and Miller, L. (1974) 'A heuristic algorithm for the vehicle-dispatch problem', *Operations Research*, Vol. 22, pp.340–349, March.

Glover, F. (1998) 'A template for scatter search and path relinking', in J-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, D. Snyers (Eds), *Artificial Evolution, Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, pp.13–54.

Ho, S. and Haugland, D. (2004) 'A tabu search heuristic for the vehicle routing problem with time windows and split deliveries', *Computers and Operations Research*, Vol. 31, pp.1947–1964, October.

Jin, M., Liu, K. and Bowden, R.O. (2007) 'A two-stage algorithm with valid inequalities for the split delivery vehicle routing problem', *Int. J. Production Economics*, Vol. 105, pp.228–242, January.

Jin, M., Liu, K. and Eksioglu, B. (2008) 'A column generation algorithm for the vehicle routing problem with split delivery', *Operations Research Letters*, Vol. 36, pp.265–270, March.

Laporte, G. (2007) 'What you should know about the vehicle routing problem', *Naval Research Logistics*, Vol. 54, pp.811–819, September.

Laporte, G., Gendreau, M., Potvin, J-Y. and Semet, F.B. (2000) 'Classical and modern heuristics for the vehicle routing', *International Transactions in Operational Research*, Vol. 7, pp.285–300, July.

Lee, C-G., Epelman, M.A., White, C.C. and Bozer, Y. (2006) 'A shortest path approach to the multiple-vehicle routing problem with split pick-ups', *Transportation Research Part B*, Vol. 40, pp.265–284, May.

Liu, K. (2005) 'A study on the split delivery vehicle routing problem', *PhD Thesis*, Mississippi State University, December.

Mole, R.H. and Jameson, S.R. (1976) 'A sequential route-building algorithm employing a generalized savings criterion', *Operational Research Quarterly*, Vol. 27, pp.503–511.

Mota, E., Campos, V. and Corberan, A. (2007) 'A new metaheuristic for the vehicle routing problem with split demands', in *Evolutionary Computation in Combinatorial Optimization*, Vol. 4446, Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, April, pp.121–129.

Mullaseril, P.A., Dror, M. and Leung, J. (1997) 'Split-delivery routing heuristics in livestock feed distribution', *Journal of the Operational Research Society*, Vol. 48, pp.107–116, February.

Nowak, M. (2005) 'The pickup and delivery problem with split loads', *PhD Thesis*, Georgia Institute of Technology, August.

Paessens, H. (1988) 'The savings algorithm for the vehicle routing problem', *European Journal of Operational Research*, Vol. 34, pp.336–344, March.

Prins, C. (2004) 'A simple and effective evolutionary algorithm for the vehicle routing problem', *Computers and Operations Research*, Vol. 31, pp.1985–2002, October.

Reimann, M., Doerner, K. and Hartl, R.F. (2004) 'D-ants: savings based ants divide and conquer the vehicle routing problem', *Computers and Operations Research*, Vol. 31, pp.563–591, April.

Rochat, Y. and Taillard, E. (1995) 'Probabilistic diversification and intensification in local search for vehicle routing', *Journal of Heuristics*, Vol. 1, pp.147–167.

Sierksma, G. and Tijssen, G. (1998) 'Routing helicopters for crew exchanges on off-shores locations', *Annals of Operations Research*, Vol. 76, pp.261–286.

Song, S.H., Lee, K.S. and Kim, G.S. (2002) 'A practical approach to solving a newspaper logistics problem using a digital map', *Computers and Industrial Engineering*, Vol. 43, pp.315–330, July.

Tarantilis, C.D. (2005) 'Solving the vehicle routing problem with adaptive memory programming methodology', *Computers and Operations Research*, Vol. 32, pp.2309–2327, September.

Tarantilis, C.D. and Kiranoudis, C.T. (2002) 'Boneroute: an adaptive memory-based method for effective fleet management', *Annals of Operations Research*, Vol. 115, pp.227–241, September.

Tavakkoli-Moghaddam, R., Safaei, N., Kah, M. and Rabbani, M. (2007) 'A new capacitated vehicle routing problem with split service for minimizing fleet cost by simulated annealing', *Journal of the Franklin Institute*, Vol. 344, pp.406–425, August.

Toth, P. and Vigo, D. (Eds) (2002) *The Vehicle Routing Problem*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Wilck, J. and Cavalier, T. (2007) 'Solving the split delivery vehicle routing problem with a loaded travel cost objective', in G. Bayraksan, W. Lin, Y. Son and R. Wysk (Eds), Paper presented in the Proceedings the of *2007 Industrial Engineering Research Conference*, Nashville, TN, Institute of Industrial Engineers, May.

Yu, Y., Chen, H. and Chu, F. (2006) 'Large scale inventory routing problem with split delivery: a new model and lagrangian relaxation approach', *Int. J. Services Operations and Informatics*, Vol. 1, pp.304–320.