Rieck, Julia; Zimmermann, Jürgen

**Article**

# Exact Solutions to the Symmetric and Asymmetric Vehicle Routing Problem with Simultaneous Delivery and Pick-Up

BuR - Business Research

**Provided in Cooperation with:**
VHB - Verband der Hochschullehrer für Betriebswirtschaft, German Academic Association of Business Research

*Suggested Citation:* Rieck, Julia; Zimmermann, Jürgen (2013) : Exact Solutions to the Symmetric and Asymmetric Vehicle Routing Problem with Simultaneous Delivery and Pick-Up, BuR - Business Research, ISSN 1866-8658, VHB - Verband der Hochschullehrer für Betriebswirtschaft, German Academic Association of Business Research, Göttingen, Vol. 6, Iss. 1, pp. 77-92, https://doi.org/10.1007/BF03342743

This Version is available at:
https://hdl.handle.net/10419/103720

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

# *Exact Solutions to the Symmetric and Asymmetric Vehicle Routing Problem with Simultaneous Delivery and Pick-Up*

*Julia Rieck*, *Institute of Management and Economics, Operations Research Group, Clausthal University of Technology, Germany, E-mail: julia.rieck@tu-clausthal.de*
*Jürgen Zimmermann*, *Institute of Management and Economics, Operations Research Group, Clausthal University of Technology, Germany, E-mail: juergen.zimmermann@tu-clausthal.de*

*Abstract*
*In reverse logistics networks, products (e.g., bottles or containers) have to be transported from a depot to customer locations and, after use, from customer locations back to the depot. In order to operate economically beneficial, companies prefer a simultaneous delivery and pick-up service. The resulting Vehicle Routing Problem with Simultaneous Delivery and Pick-up (VRPSDP) is an operational problem, which has to be solved daily by many companies. We present two mixed-integer linear model formulations for the VRPSDP, namely a vehicle-flow and a commodity-flow model. In order to strengthen the models, domain-reducing preprocessing techniques, and effective cutting planes are outlined. Symmetric benchmark instances known from the literature as well as new asymmetric instances derived from real-world problems are solved to optimality using CPLEX 12.1.*

*JEL classification: R41, C61, M11*

*Keywords: reverse logistics, vehicle routing, simultaneous delivery and pick-up, mixed-integer linear programming*

*Manuscript received May 10, 2012, accepted by Karl Inderfurth (Operations and Information Systems) January 30, 2013.*

## 1 Introduction

The vehicle routing problem with simultaneous delivery and pick-up is an extension of the Capacitated Vehicle Routing Problem (CVRP) where products have to be transported from the depot to customer locations and other products have to be taken from customer locations back to the depot. Each customer requires two types of service – a delivery and a pick-up – and both activities have to be carried out simultaneously by the same vehicle. Traditionally, the objective is to find a set of routes that minimizes the transportation costs or the distances traveled, where the customer demands are satisfied and vehicle load capacities are not exceeded at any time.
The VRPSDP is an operational problem which typically occurs in *reverse logistics*. In addition to the distribution process to customers, used or obsolete products have to be transported in the opposite direction. Depending on the type of product and its previous usage, the reverse material flow may enter the supply chain of the original manufacturer or a different supply chain. In *closed-loop supply chains*, companies reintegrate returned products into their own processes in order to reap the maximum economic benefit from end-of-use products and to utilize all materials already integrated in the value added chain. Additionally, a reduction of waste and a decrease of natural resource extraction may be achieved.
In order to handle the arising flow of used products, companies need to set up new logistics infrastructures, e.g., sorting and disassembly centers. Therefore, many authors consider network design and architecture issues. Less attention has currently been paid to tactical and operational aspects concerning routing and handling in reverse logistics networks. For that reason, this paper considers

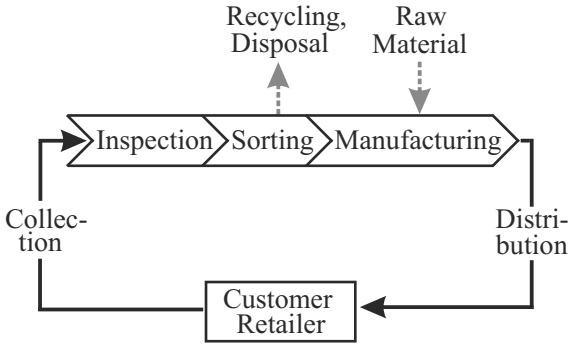
BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

the VRPSDP, specifies an application area, and provides exact solutions to symmetric and asymmetric benchmark instances.
The remainder of the paper is organized as follows: In section 2 we investigate an interesting practical example of the VRPSDP that occurs in closed-loop supply chains. Section 3 is devoted to a literature review on simultaneous delivery and pick-up problems. In section 4 two mathematical formulations for the VRPSDP are presented which can be posted to a standard solver (e.g., CPLEX). Based on the formulations, we proceed to describe methods for improving the quality of the models in terms of computation time and solution gap. The results of a comprehensive performance analysis, where symmetric and asymmetric instances are considered, are given in section 5. Finally, conclusions are presented in section 6.

## 2 Practical application and closed-loop supply chains

Vehicle routing with simultaneous delivery and pick-up typically arises in closed-loop supply chains. Figure 1 shows the structure of a general closed-loop supply chain, where raw materials enter the system at the manufacturing facility. After the production process, finished goods are delivered to customers or retailers (forward flow). From customer/retailer locations, a flow of used or obsolete products (e.g., spare parts or commodities which are unsold) runs to the place of recovery (reverse flow). There, inspection and sorting activities are carried out in order to differentiate between components that may be part of new production processes as well as recyclable materials and disposal.
A closed-loop supply chain for security containers may be found in the field of *data destruction*. The constant evolution of data processing and the increase in volume of sensitive data has brought about the need for responsible service providers to destroy data in compliance with international standards. Documents (or disks, magnetic tapes, videos, and microfilms) must be deleted once the legal retention period is met. In order to ensure that data goes directly and unseen into the destruction plant, customers appoint security containers that provide protection from unauthorized access. Standard container capacities are 240 liters and 350 liters. Special vehicles are required to deliver empty containers to customers (e.g., companies, public bodies, or private households) and to pick-up full containers at customers. The filled containers are depleted in a safe place (i.e., at the depot location). The documents are sorted into various paper types and then shredded as well as baled to reduce the amount of space required. The paper bales are temporarily stored in the depot and then transported to paper mills, where they are recycled into new paper products. The security containers themselves are cleaned after the depletion process and then reused directly at new customer locations. The material flows to and from customers are usually lower than the vehicle capacity, and the number of delivery and pick-up containers at customer locations generally differ from each other. Therefore, a VRPSDP has to be solved in order to generate a feasible set of routes for the vehicles. Since a data destruction service provider usually operates on a regional level (i.e., within a 100 km radius), varying numbers of urban and country customers have to be considered during the planning phase.

**Figure 1: General closed-loop supply chain**




## 3 Literature review

In the last twenty years a number of articles appeared in the domains of reverse logistics and closed-loop supply chains (see e.g., the survey of Bostel, Dejax, and Lu 2005). However, most papers deal with strategic and tactical problems, since they directly affect the companies' profitability. For a comprehensive review on strategic problems, we refer to Akçalı, Çetinkaya, and Üster (2009). Tactical applications can be found in Sheu, Chou, and Hu (2005), Gu and Ji (2008) or Gomes Salema, Barbosa Póvoa, and Novais (2009 and 2010).
Operational problems in closed-loop supply chains

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

deal with vehicle routing and scheduling decisions. If customers require both a delivery and a pick-up service, companies have to decide whether to simultaneously pick-up returns and distribute new products. The trade-off occurs between the exploitation of existing vehicle capacities and the reduction of flexibility due to mixed pick-up and delivery operations (De Koster, De Brito, and Van De Vendel 2002). A survey on different variants of pick-up and delivery problems was presented in Parragh, Doerner, and Hartl (2008).

For many companies (particularly for data destruction service providers), it is economically beneficial to simultaneously deliver and pick-up. Therefore, in what follows we will concentrate on articles that are related to the VRPSDP. As an extension of the CVRP, the VRPSDP is $\mathcal{NP}$-hard in the strong sense (Garey and Johnson 1979; Toth and Vigo 2002). This fact and the necessity to solve real-world instances have pushed researchers to develop heuristic algorithms. In contrast, exact algorithms have received little attention in the literature.

The VRPSDP is introduced by a case study that deals with a public library delivery and pick-up system in Ohio. To solve the problem, Min (1989) presented a solution approach which is analogous to a "cluster-first route-second" method, i.e., the customers are initially clustered into groups and afterwards a traveling salesman problem is solved for each group. Similarly, Halse (1990) devised an improved "cluster-first route-second" method. Dethloff (2001) used the "cheapest-insertion" concept to solve the VRPSDP. Based on a chosen single customer route, customers are inserted into the current route according to three criteria; extra travel distance, capacity remaining on the route, and distance to the depot and back. Nagy and Salhi (2005) introduced the idea of "weakly feasible" and "strongly feasible" routes. If the length of a route does not exceed a maximum distance, the route is called weakly feasible. If, in addition to the weak feasibility, the total load on the routes is below the capacity of the vehicles, the route is called strongly feasible. The algorithm relies on a route construction procedure to provide an initial (weakly feasible) solution and on a variety of routines to improve the routes and to reach strong feasibility. Mitra (2005) considered the VRPSDP, where the delivery and pick-up quantities at customer locations are not limited to the capacity of the vehicles. Therefore, it may be possible that a customer must be visited more than once to fulfill the requirements. As a result, the next visits can be made by the same vehicle or by other ones. The author developed a mixed-integer linear program and a route construction heuristic.

Most of the metaheuristics developed for the VRPSDP are based on pure or hybrid versions of the tabu search method. Montané and Galvão (2006) presented a tabu search algorithm where initial solutions are found using constructive heuristic procedures. The neighborhood is built on the moves relocation, interchange, crossover, and 2-opt. Chen and Wu (2006) introduced a hybrid heuristic based on record-to-record travel, tabu lists, and route improvement procedures such as 2-exchange, swap, 2-opt, and or-opt. An "insertion-based" algorithm is used to generate initial solutions. Privé, Renaud, Boctor, and Laporte (2006) studied a problem that arises from the distribution of soft drinks and collection of recyclable containers of a Quebec-based company. The problem is modeled as a multi-product vehicle routing problem with a heterogeneous vehicle fleet, time windows, and simultaneous delivery and pick-up at customer locations, where the weight of the material collected is always less than the weight of the material delivered. The objective is the minimization of routing costs, lower revenues resulting from the sale of recyclable material. Three construction heuristics, one inspired from the "nearest-neighbor" concept and two derived from petal algorithms (see, e.g., Ryan, Hjorring, and Glover 1993), are developed for the problem. A subsequent improvement method uses a combination of 3-opt and 2-interchange moves as well as route merges. Bianchessi and Righini (2007) presented and compared several heuristic algorithms for the VRPSDP, in particular greedy insertion, local search, and tabu search procedures. Wassan, Wassan, and Nagy (2008) designed a reactive tabu search metaheuristic where an initial solution is constructed using a modification of the sweep algorithm (Gillett and Miller 1974). The core of the algorithm is a fast feasibility check for neighboring moves (e.g., shift, swap, and reverse). Zachariadis, Tarantilis, and Kiranoudis (2009) proposed a hybrid framework based on two metaheuristics, namely tabu search and guided local search. The algorithm is aimed at achieving a vast exploration of the search space by escaping from local optima

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

and intensifying at promising solution areas. To guarantee the robustness of the method, the number of search parameters is kept to a minimum. Subramanian, Drummond, Bentes, Ochi, and Farias (2010) presented a multi-start heuristic that consecutively generates a solution by means of a greedy procedure. In order to improve this solution, a local search phase as well as some perturbation mechanisms are used.

Recently, some population based metaheuristics were developed for the VRPSDP. Ai and Kachitvichyanukul (2009) proposed a particle swarm optimization procedure based on a solution representation which consists of a priority list of customers and its preferred vehicle. The particle is converted into the problem specific solution through a decoding procedure. Gajpal and Abad (2009) presented an ant colony system algorithm that uses a construction phase as well as three local search schemes based upon interchange and insertion operations as well as an exchange of sub-paths. An initial solution is constructed using the nearest neighbor heuristic. Zachariadis, Tarantilis, and Kiranoudis (2010) introduced an adaptive memory programming methodology for the VRPSDP. The proposed adaptive memory collects solution characteristics obtained through the search process. These characteristics are combined to produce new solutions that are subsequently improved by a tabu search method.

Most of the proposed metaheuristics are tested on instances with up to 400 customers. Since the authors usually neglected the consideration of lower bounds, no solution gap values are determined.

Exact solution procedures presented for the VRPSDP may be separated in:

- set-partitioning or set-covering formulations containing a binary variable for every potential vehicle route as well as
- two-index or three-index formulations containing binary variables indicating whether an arc in the underlying digraph is selected (by a specified vehicle) or not.

Angelelli and Mansini (2002) considered the VRPSDP, where the delivery and pick-up at customer locations must be carried out within given time windows. They implemented a branch-and-price approach based on a set-covering formulation for the master problem. A relaxation of the elementary shortest path problem with capacity constraints and time windows is used as pricing problem. Dell'Amico, Righini, and Salani (2006) examined the application of branch-and-price techniques, too. Thereby, the pricing problem is solved using dynamic programming. In order to improve the effectiveness of the pricing algorithm, the authors considered two new strategies: bidirectional search and bounded number of steps. Furthermore, different branching strategies are implemented, namely branching on arcs, branching on resources, and branching on cycles.

Both Mitra (2005) as well as Chen and Wu (2006) used two-/three-index formulations in combination with the standard solver CPLEX to validate results obtained by heuristic algorithms. Subramanian, Uchoa, and Ochi (2010) presented an undirected and a directed two-commodity-flow formulation that make use of variables indicating the pick-up, the delivery, as well as the simultaneous pick-up and delivery flows. The models are embedded into a branch-and-cut approach containing the "CVRPSEP package" (Lysgaard 2003). Furthermore, Subramanian, Uchoa, Pessoa, and Ochi (2011) proposed a branch-and-cut method over a symmetric formulation with only edge variables.

All exact methods introduced in this section are constructed to solve symmetric problem instances. For the asymmetric case, only the one-commodity-flow model presented by Dell'Amico, Righini, and Salani (2006) and the directed two-commodity-flow model presented by Subramanian, Uchoa, and Ochi (2010) can be considered. Due to the lack of exact solution algorithms designed for asymmetric instances, we propose below exact solution procedures for both the symmetric and the asymmetric VRPSDP based on branch-and-cut.

## 4 Model formulations

In this section, all relevant notation and terminology used throughout the paper are introduced (subsection 4.1). Furthermore, we present two model formulations for the VRPSDP, namely a vehicle-flow formulation (subsection 4.2) and a commodity-flow formulation (subsection 4.4), which can be posted to a standard solver. Subsections 4.3, 4.5, and 4.6 are devoted to efficient modeling techniques that can help to ensure better

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

performance results.

### 4.1 General model components

The VRPSDP is defined on a complete digraph $G = (V, A)$, where $V = \{0, \ldots, n\} = C \cup \{0\}$ is the node set and $A = \{\langle i,j\rangle \mid i,j \in V\}$ is the arc set. Each node $i \in C$ represents a customer, while node 0 corresponds to the depot. A non-negative weight $c_{ij}$ is associated with each arc $\langle i,j\rangle \in A$ that represents the transportation costs between nodes $i$ and $j$. We assume that the cost matrix satisfies the triangle inequality; this condition is always fulfilled in practical situations. A set of $K$ identical vehicles, each with capacity $Q$, is available at the depot. The capacity of the vehicles cannot be exceeded in any route. Each customer $i \in C$ is associated with a demand $d_i \geq 0$ that should be delivered and a demand $p_i \geq 0$ that should be picked-up, where $d_i + p_i > 0$ and $d_i, p_i \leq Q$. The demands of customers are measured in abstract transport units, which can be calculated from the dimension or weight of the individual product. For the depot, we set $d_0 := p_0 := 0$.
Both the vehicle-flow and the commodity-flow formulation contain binary variables indicating whether an arc in the underlying digraph $G$ is selected or not, i.e., they make use of decision variables

$$(1)\quad x_{ij} := \begin{cases} 1, \text{if arc } \langle i,j\rangle \text{ belongs to the solution} \\ 0, \text{otherwise} \end{cases}$$

with $i,j \in V$.

### 4.2 Vehicle-flow model

A *vehicle-flow model* describes the vehicle's motion and filling level in the road network. In order to construct a suitable vehicle-flow model, we therefore use the idea of "Resource Extension Functions" that determine the updates of resources along an arc in the underlying network (Irnich 2008). With binary variables $x_{ij}$ (1) and auxiliary variables:

$l_i \geq 0$ amount of load after visiting customer $i \in C$

$ld_i \geq 0$ amount of load that has to be delivered to node $i \in V$ and to all other following nodes

the vehicle-flow model for the VRPSDP has the form:

$$(2)\quad \text{Minimize} \quad \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

subject to

$$(3)\quad \sum_{\substack{i \in V \\ i \neq j}} x_{ij} = 1 \qquad j \in C$$

$$(4)\quad \sum_{\substack{i \in V \\ i \neq j}} x_{ji} = 1 \qquad j \in C$$

$$(5)\quad \sum_{i \in C} x_{0i} \leq K$$

$$(6)\quad ld_i \geq ld_j + d_i - \mathrm{M}_1(1 - x_{ij}) \qquad i \in V, j \in C$$

$$(7)\quad l_j \geq ld_j - d_j + p_j \qquad j \in C$$

$$(8)\quad l_j \geq l_i - d_j + p_j - \mathrm{M}_2(1 - x_{ij}) \qquad i,j \in C$$

$$(9)\quad d_i \leq ld_i \leq Q \qquad i \in V$$

$$(10)\quad p_i \leq l_i \leq Q \qquad i \in C$$

$$(11)\quad x_{ij} \in \{0, 1\} \qquad i,j \in V.$$

Objective function (2) represents the transportation costs which are to be minimized. The indegree and outdegree constraints (3) and (4) ensure that each customer is visited exactly once by a vehicle. Inequality (5) states that no more than $K$ routes are created. With constraints (6), we specify the delivery quantity that has to be loaded at the depot. Additionally, constraints (6) force an order for customer visits in the routes, which ensure that no subtours without the depot are generated. Inequalities (7) and (8) indicate the amount of load in the vehicles after the visit of the first customer and the other customers in the routes, respectively. Constraints (6) and (8) are disjunctive constraints that are linearized by using large multipliers ("big-M values"). In order to create valid inequalities, we set $\mathrm{M}_1 = \mathrm{M}_2 := Q$. Constraints (9) and (10) guarantee that the capacity of the vehicles cannot be exceeded.

### 4.3 Efficient modeling techniques for the vehicle-flow model

The vehicle-flow model contains $|V|^2$ binary as well as $|C| + |V|$ real auxiliary variables. Furthermore, the model has no special structural characteristics and, therefore, as in generic binary models, only instances with a small, or medium, number of customers can be solved to optimality using a standard solver (e.g., CPLEX). Since

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

the run times are quite large for medium-scale instances, we want to give the solver supplementary knowledge such as preprocessing information and additional constraints. Under preliminary tests, we have realized that the following modeling techniques result in significant improvements in terms of computation time and memory space.
In order to facilitate the solution process, it is appropriate to restrict the *domains* of auxiliary variables. Considering model (2)–(11), the bounds of auxiliary variables $l_i$ and $ld_i, i \in C$, can be strengthened by investigating the differences between delivery and pick-up quantities.

**Figure 2: Vehicle route with two customers *i* and *j***

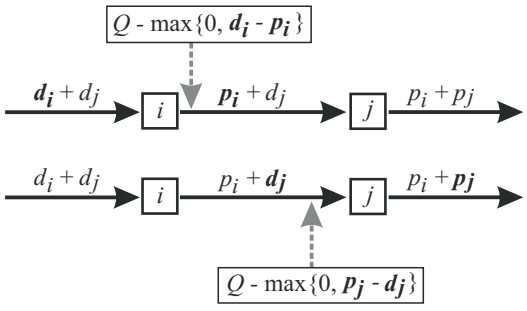



Figure 2 shows a vehicle route with two customers $i$ and $j$. The solid arcs are weighted with the quantities that flow along the corresponding road segment. The quantities given in the boxes indicate the maximum amount of load after visiting customer $i$ and before visiting customer $j$. The latter quantity is always larger than, or equal to, the quantity that has to be delivered to $j$ and to all other following nodes. Using these information, we obtain the conditions:

$$(12) \quad l_i \le Q - \max\{0, d_i - p_i\} \quad i \in C$$
$$(13) \quad ld_i \le Q - \max\{0, p_i - d_i\} \quad i \in C.$$

The vehicle-flow model involves two *big-M values*, which have to be chosen smallest possible while maintaining feasibility of all potentially optimal solutions. We therefore replace the general "big-M values" ($M_1$ and $M_2$) by specified $M_1^{ij}$ and $M_2^{ij}$ for the corresponding node pairs $\{i,j\}$.
In order to ensure that inequalities (6) are redundant, if $x_{ij} = 0$, $M_1^{ij}$ must be larger than, or equal to, variable $ld_j$. Thus, we are able to utilize conditions (13) and replace the value $M_1$ by

$$M_1^{ij} := Q - \max\{0, p_j - d_j\} \quad i \in V, j \in C.$$

Constraints (8) are always fulfilled, if we choose $M_2^{ij}$ as larger than, or equal to, the difference between $l_i$ and $d_j$. Using conditions (12), the "big-M value" $M_2$ can then be replaced by

$$M_2^{ij} := Q - \max\{d_j, d_i - p_i + d_j\} \quad i,j \in C.$$

In addition to the auxiliary variables used in the vehicle-flow model, information on pick-up loadings can be collected and exploited. To that end, we introduce new auxiliary variables

$lp_i \ge 0$ amount of pick-up load after visiting node $i \in V$.

that are used to formulate the following *additional constraints*:

$$(14) \quad lp_j \ge lp_i + p_j - M_3(1 - x_{ij}) \quad i \in C, j \in V$$
$$(15) \quad p_i \le lp_i \le Q \quad i \in V$$
$$(16) \quad l_i = lp_i + ld_i - d_i \quad i \in C.$$

With inequalities (14), we determine the pick-up quantity that has to be offloaded at the depot. Constraints (15) guarantee that the capacity of the vehicles is not exceeded, and constraints (16) are structural equalities that help to correlate the load variables with each other.
In general case, the "big-M value" in disjunctive constraints (14) can be set to $M_3 := Q$. However, to choose "big-M" as small as possible, constants $M_3^{ij}$ are introduced that are larger than, or equal to, variable $lp_i$. Since the amount of pick-up load after visiting customer $i$ is always larger than, or equal to, the total quantity of products after visiting customer $i$, conditions (12) can be used to formulate:

$$M_3^{ij} := Q - \max\{0, d_i - p_i\} \quad i \in C, j \in V.$$

### 4.4 Commodity-flow model

In addition to binary variables $x_{ij}$ (1), the *commodity-flow model* makes use of two additional sets of continuous variables which represent the amounts of demand that flow along the associated arcs (Dell'Amico, Righini, and Salani 2006). We identify the decision variable $y_{ij} \ge 0$ with the amount of delivery load carried along arc $\langle i,j \rangle$ and the decision variable $z_{ij} \ge 0$ with the amount of collected load carried along $\langle i,j \rangle$, $i,j \in V$. Then,

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

the mixed-integer linear program for the VRPSDP can be formulated as follows:

$$\text{(17) Minimize} \quad \sum_{i\in V}\sum_{j\in V} c_{ij}x_{ij}$$

subject to

$$\text{(18)} \quad \sum_{\substack{i\in V\\ i\neq j}} x_{ij} = 1 \qquad j \in C$$

$$\text{(19)} \quad \sum_{\substack{i\in V\\ i\neq j}} x_{ij} - \sum_{\substack{i\in V\\ i\neq j}} x_{ji} = 0 \qquad j \in V$$

$$\text{(20)} \quad \sum_{i\in C} x_{0i} \leq K$$

$$\text{(21)} \quad \sum_{\substack{i\in V\\ i\neq j}} y_{ij} - \sum_{\substack{i\in C\\ i\neq j}} y_{ji} = d_j \qquad j \in C$$

$$\text{(22)} \quad \sum_{\substack{i\in V\\ i\neq j}} z_{ji} - \sum_{\substack{i\in C\\ i\neq j}} z_{ij} = p_j \qquad j \in C$$

$$\text{(23)} \quad y_{ij} + z_{ij} \leq \mathrm{M}_4\, x_{ij} \qquad i,j \in V, i \neq j$$

$$\text{(24)} \quad 0 \leq y_{ij} \leq Q,\ 0 \leq z_{ij} \leq Q \qquad i,j \in V$$

$$\text{(25)} \quad x_{ij} \in \{0,1\} \qquad i,j \in V.$$

As previously reported, objective function (17) expresses the transportation costs. Constraints (18) ensure that each customer is reached precisely once. Constraints (19) characterize the flow on the path to be followed by a vehicle. Inequality (20) guarantees that no more than $K$ routes are created. Constraints (21) and (22) are the balance equations to satisfy the delivery and pick-up demand at customer $j$, respectively. It is easy to see that inequalities (21) and (22) eliminate subtours, since they sequence customers in the routes beginning and ending at the depot (source and sink of products). Disjunctive constraints (23) guarantee that the capacity of the vehicles is not exceeded in any route; the "big-M value" can be set to $\mathrm{M}_4 := Q$.

### 4.5 Efficient modeling techniques for the commodity-flow model

The commodity-flow formulation requires $|V|^2$ binary variables as well as $2|V|^2$ continuous variables. In comparison with the vehicle-flow model, the number of continuous variables is relatively high. However, the formulation uses only one set of disjunctive ("big-M") constraints. Generally, the reduction or elimination of big-M constraints improves the linear programming lower bounds of an integer program. We will investigate this statement in subsection 5.2.

In order to restrict the *domains* of continues variables in the model, Subramanian, Uchoa, and Ochi (2010) replaced constraints (24) by the following conditions:

$$\text{(26)} \quad d_j x_{ij} \leq y_{ij} \leq (Q - d_i)\, x_{ij} \qquad i,j \in V$$

$$\text{(27)} \quad p_i x_{ij} \leq z_{ij} \leq (Q - p_j)\, x_{ij} \qquad i,j \in V.$$

Constraints (26) ensure that the quantity on a used road segment $\langle i,j\rangle$ is at least equal to the delivery demand of customer $j$. Since node $i$ is visited before $j$, the maximum quantity that flows along the arc is less than, or equal to, the difference between capacity $Q$ and $d_i$. Considering the pick-up demands of nodes $i$ and $j$, constraints (27) can be specified in an analogous manner.

Disjunctive constraints (23) are linearized by using a *big-M value*. The general value $\mathrm{M}_4$ can be replaced by specified $\mathrm{M}_4^{ij}$ if the differences between delivery and pick-up quantities are considered (see also Fig. 2). We then obtain

$$\mathrm{M}_4^{ij} := Q - \max\{0, d_i - p_i, p_j - d_j\} \quad i,j \in V.$$

Under preliminary tests, we have determined that the consideration of *additional constraints* improves the solver solution process. The following inequalities:

$$\text{(28)} \quad \sum_{i\in V} y_{ij} \geq d_j \qquad j \in C$$

$$\text{(29)} \quad \sum_{j\in V} z_{ij} \geq p_i \qquad i \in C$$

set lower bounds on the total quantities of products entering and leaving a customer location.

### 4.6 Valid inequalities for both models

In our computational experiments, each VRPSDP problem instance is solved using CPLEX 12.1 and the Concert interface for communications with the solver. CPLEX uses a branch-and-cut approach, and the features of CPLEX allow general cuts to be generated during optimization. We identified clique, implied-bound, and flow-cover cuts as particularly suitable for the vehicle-flow model. For the commodity-flow model, the integration of implied-bound, flow-cover, and flow-path cuts improves the results significantly. Furthermore, we

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

allow a priority branching on special ordered sets of type one. Models and cutting planes are implemented using C++ code compiled with MS Visual Studio 2008.

In order to further strengthen both models, we include rounded capacity cuts which are special inequalities for the symmetric CVRP. To insert rounded capacity cuts, we use an adjusted variant of the "CVRPSEP package" implemented by Lysgaard (2003). If we transfer the linear programming solution $x$ to the separation algorithm, we substitute the sum $(x_{ij}+x_{ji})$ for the decision variable $x_{(i,j)}$ (specifying if edge $(i,j)$ is in the solution). The rounded capacity inequalities

$$\sum_{i\in S}\sum_{j\in V\setminus S} x_{ij} + x_{ji} \geq 2\ r(S) \qquad S \subseteq V \setminus \{0\}, S \neq \emptyset$$

impose the vehicle capacity restrictions, where $r(S)$ is the minimum number of vehicles needed to serve a customer set $S \subseteq C$. For the VRPSDP, we obtain

$$r(S) = \left\lceil \max\{\textstyle\sum_{i\in S} d_i, \sum_{i\in S} p_i\}/Q \right\rceil.$$

The separation problem of rounded capacity cuts is known to be $\mathcal{NP}$-hard. For the computational tractability, we only add rounded capacity cuts at the first 400 nodes of the branch-and-bound tree. Since all decision and auxiliary variables are bounded, the convex hull $\mathcal{H}$ of the set of feasible solutions to the VRPSDP has either the form of a convex polytope or it is empty. This implies that there always exists an optimal solution if $\mathcal{H} \neq \emptyset$ is satisfied.

# 5 Computational results

The computational tests have been performed on benchmark instances known from the literature as well as on instances that are derived from real-world data (cf. the application in section 2). In total, we used 206 symmetric and asymmetric test instances to evaluate the performance of our two models.

During a series of preliminary tests, we have investigated the effectiveness of the directed two-commodity-flow model presented by Subramanian, Uchoa, and Ochi (2010). A comparison of run times showed that the model formulation is indeed able to cope with small-scale asymmetric instances that are, in general, easy to solve. For more difficult instances, our computing environment was not able to complete the enumeration within the time limit. In the remaining part of this paper, we therefore exclude the two-commodity-flow model from the computational study.

We start off by describing the composition and generation of the problem instances used for testing the models (subsection 5.1). In subsection 5.2 a comprehensive performance study is presented.

## 5.1 Benchmark instances

In order to investigate the increases in computation time caused by different problem structures, we subdivided the benchmark set in different classes. Particularly, class 1 contains asymmetric real-world instances generated in accordance with our cooperation partner (a data destruction service provider). Classes 2 to 5 cover symmetric instances that were proposed in the literature. Finally, classes 6 and 7 incorporate asymmetric instances constructed on the basis of benchmark sets from the literature, which are usually considered to evaluate heuristic solution methods.

Class 1 is the largest test set and it includes instances that are generated using real-world data. Thereby, the customers are positioned in a special geographic area in the North of Germany (region around Hannover with a radius of 100 km). Each instance contains urban customers in and near Hannover and country customers usually placed in clusters. Transportation costs are determined by the Euclidean distance (rounded downward to the second decimal place) multiplied by a route factor for the different route segments (motorway, highway, and street). Delivery and pick-up quantities are measured in abstract transport units, where the different filling levels of security containers are considered, i.e., we set $d_i \in \{0,\ldots,20\}$ and $p_i \in \{0,\ldots,66\}$ for all customers $i \in C$. Furthermore, each instance involves two to six vehicles with capacity $Q \in \{100, 120\}$.

In the second class, a subset of problem instances introduced by Mitra (2005) is considered, where $c_{ij} \in \{10,\ldots,28\}$ for all $i,j \in V$, $c_{ii} = \infty$, $d_i, p_i \in \{1, 5\}$ for all $i \in C$, and $Q = 10$ transport units.

Class 3 contains the benchmark of Chen and Wu (2006) that is derived from the extended Solomon instances R121, R141, R161, R181, and R1101 (Homberger and Gehring 2002). The benchmark is composed of 25 instances with 15, 17, and 20 customers. Transportation costs are determined by the Euclidean distance rounded to

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

the third decimal place. In order to incorporate simultaneous deliveries and pick-ups, a ratio $r_i := \min\{(x_i/y_i), (y_i/x_i)\}$ is calculated for each customer $i \in C$; $x_i, y_i$ are the $x$- and $y$-coordinate values of $i$. The integer delivery demand of customer $i$ is equal to the rounded solution of $d_i := r_i\,\delta_i$, where $\delta_i$ is the original demand of the underlying problem instance and $r_i$ is rounded to the second decimal place. The pick-up demand $p_i$ is set to $p_i := \delta_i - d_i$. Classes 4 and 5 are introduced by Dell'Amico, Righini, and Salani (2006). Both benchmarks consist in equal parts of instances with 20 and 40 customers. Transportation costs are given by the Euclidean distance rounded downward to the next integer value. The numbers of vehicles are assumed to be fixed; the numbers come from the solution of a multiple bin packing problem. Class 4 contains instances that are derived from the Solomon (1987) instances C101, R101, and RC101, where the delivery quantity $d_i$ equals the original demand $\delta_i$ of customer $i \in C$. The pick-up quantity $p_i$ is calculated by $p_i := \lfloor(1-\gamma)d_i\rfloor$ if $i \in C$ is even, and by $p_i := \lfloor(1+\gamma)d_i\rfloor$ if $i \in C$ is odd; $\gamma \in \{0.2, 0.8\}$. Class 5 is obtained from the LIBrary of capacitated Vehicle Routing Problems (VRPLIB)[1]. The original demands in the instances are considered as delivery demands. Furthermore, the pick-up demand of each customer $i$ is computed as $p_i = \lfloor(0.5+r)d_i\rfloor$, where $r$ is taken from an uniform distribution in the interval $[0, 1]$.

Classes 6 and 7 contain instances with 50 customers that were introduced by Dethloff (2001) as well as Salhi and Nagy (1999). Previous heuristic results for these instances are based on non-rounded input data. Since the benchmarks are constructed using well-conceived rules, we used them as a basis for generating asymmetric instances. In order to avoid numerical problems during presolve and cut generation, we decided to use a rounding scheme for the transportation costs, the vehicle capacity, as well as the pick-up and delivery quantities. Due to the rounding scheme, the route composition obtained by aforementioned heuristics and our solution procedure varies widely. For example, quantities of $9.\bar{6}$, $5.\bar{6}$, and $4.\bar{6}$ units can be combined in a vehicle with a capacity of 20 units, whereas rounded quantities $9.\bar{6} \approx 9.7$, $5.\bar{6} \approx 5.7$, and $4.\bar{6} \approx 4.7$ cannot be combined. Further, we introduced a matrix $\Lambda = (\lambda_{ij}), i,j \in V$, to modify all instances of classes 6 and 7; $\lambda_{ij} \in [1.0, 1.5]$ for $i = 0, j \in C$; $i,j \in C, i < j$ and $\lambda_{ij} := 1.0$ otherwise. Then, an asymmetric cost matrix $\Gamma = (\gamma_{ij}), i,j \in V$, is obtained by $\gamma_{ij} := c'_{ij}\lambda_{ij}$, where $c'_{ij}$ is the Euclidean distance between pairs of nodes rounded to the fourth decimal place. The costs that do not satisfy the triangle inequality are appropriately decreased by using the Floyd-Warshall algorithm. We calculate the delivery and pick-up demands with a similar technique as used in class 3, i.e., $d_i := r_i\,\delta_i$, $p_i := \delta_i - d_i$ and both values as well as the capacity of the vehicles are rounded to the second decimal place.

It should be pointed out that the specification of the exact number of digits after the decimal point is crucial in order to use the results for experimental performance comparison of different solution procedures. If the number of digits is not specified, rounding differences in the objective function values as well as different route compositions or customer sequences can occur and may lead to a misinterpretation of the optimality. The consideration of a rounding scheme is non-critical for real-world VRPSDP instances, because the number of digits can individually be adjusted to the underlying problem.

### 5.2 Performance study

The following section will give an overview of and a discussion on the results obtained for our two model formulations. All tests are performed on an Intel 6-core processor, 3.46 GHz, 24 GB RAM, running Windows 7.

We begin our analysis with the first test set which consists of real-world asymmetric instances. In order to show the effects of modeling techniques described in subsections 4.3, 4.5, and 4.6, we solved each instance with 20 and 30 customers six times:

- test run $m_1$-I: using the vehicle-flow model ($m_1$).
- test run $m_1$-II: using the vehicle-flow model ($m_1$) and valid inequalities given in subsection 4.6.
- test run $m_1$-III: using the vehicle-flow model ($m_1$), valid inequalities, and efficient modeling techniques given in subsection 4.3.
- test run $m_2$-I: using the commodity-flow model ($m_2$).

[1] The VRPLIB is available at http://elib.zib.de/pub/Packages/mp-testdata/vehicle-rout/vrplib/index.html.

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

**Table 1: Computational results for small-scale practical (asymmetric) instances**

| class 1 | | | test run $m_1$-I | | | test run $m_1$-II | test run $m_1$-III |
|---|---|---|---|---|---|---|---|
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | $t_{cpu}$ | $t_{cpu}$ |
| 20 | 2/120 | 10 | 9 | 23.3 | 1,874.81 | 0.43 | 0.42 |
| 20 | 3/100 | 10 | 5 | 5.1 | 5,751.64 | 0.73 | 0.69 |
| 30 | 2/120 | 10 | 3 | 11.3 | 3,888.43 | 2.14 | 2.86 |
| 30 | 3/100 | 10 | 1 | 22.1 | 986.52 | 273.65 | 313.09 |
| class 1 | | | test run $m_2$-I | | | test run $m_2$-II | test run $m_2$-III |
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | $t_{cpu}$ | $t_{cpu}$ |
| 20 | 2/120 | 10 | 10 | 0.0 | 0.70 | 0.47 | 0.47 |
| 20 | 3/100 | 10 | 10 | 0.0 | 2.64 | 0.68 | 0.65 |
| 30 | 2/120 | 10 | 10 | 0.0 | 9.46 | 2.73 | 2.79 |
| 30 | 3/100 | 10 | 9 | 7.1 | 2,303.87 | 54.26 | 30.79 |

- test run $m_2$-II: using the commodity-flow model ($m_2$) and valid inequalities.
- test run $m_2$-III: using the commodity-flow model ($m_2$), valid inequalities, and efficient modeling techniques given in subsection 4.5.

Table 1 shows the average computation times for small-scale instances with up to 30 customers. In column "$K/Q$" the number of vehicles used with respective capacity and in column "$no$" the total number of instances with the same $|C|/K/Q$ values is given.
Since the considered VRPSDP is an $\mathcal{NP}$-hard optimization problem, we cannot expect that a branch-and-cut approach will terminate within a reasonable time limit. That is why we allow a maximum computation time of twelve hours after which the best solution found is returned. In column "#opt" the number of optimal solutions found and in column "gap" the gap (in percentage) between the best integer objective and the current lower bound of all active nodes is given. Column "$t_{cpu}$" displays the average computation time (in seconds) for all optimally solved instances.
The results highlight that the average computation times depend on the number of customers and on the number of vehicles. Furthermore, the consideration of valid inequalities is highly important for the performance of the models. Without valid inequalities, only 18 (39) out of 40 instances could be solved to optimality using the vehicle-flow (commodity-flow) model. In addition, test runs $m_i$-II and $m_i$-III, $i = 1, 2$, are efficient. All instances are solved to optimality and column "$t_{cpu}$" shows the average CPU time over the respective ten instances. The resulting average run times are lower than six minutes, but it is apparent that the commodity-flow model outperforms the vehicle-flow model. Considering the computation time differences between related test runs ($m_1$-II and $m_1$-III as well as $m_2$-II and $m_2$-III), the additional value of modeling techniques cannot be clearly realized. However, this result will change by analyzing medium-scale instances with more than 30 customers.
In order to improve the bounding accuracy and to speed up the solver, particularly for medium-scale instances, an upper bound on the objective function can be supplied. We used the solution obtained after passing 10,000 iterations of a simplified version of the multi-start procedure introduced in Rieck and Zimmermann (2009) to construct an upper bound and a start solution. Then, to investigate the effects of modeling techniques in combination with start solutions, we solved each instance with 40, 50, and 60 customers eight times. Thereby, we used the test runs: $m_i$-I, $m_i$-II, $m_i$-II-start, and $m_i$-III-start, $i = 1, 2$, where the suffix "start" imply that a start solution is inserted.
Table 2 summarizes the results for the different test runs. As expected, the solution process without valid inequalities is characterized by many instances that could not be solved to optimality. In particular, the vehicle-flow model is never able to terminate the enumeration. If we concentrate on test runs $m_i$-III-start, $i = 1, 2$, the commodity-flow model produces the best results in terms of computation time and solution gap. All instances with 40 customers and most of the larger instances are solved to optimality. Additionally, the positive performance effect of the described modeling techniques (subsections 4.3 and 4.5) can now be realized in its whole extent. In test run $m_2$-II-start, for example, only 46 instances are solved

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

**Table 2: Computational results for medium-scale practical (asymmetric) instances**

| class 1 | | | test run $m_1$-I | | | test run $m_1$-II | | |
|---|---|---|---|---|---|---|---|---|
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | #opt | gap | $t_{cpu}$ |
| 40 | 4/120 | 10 | 0 | 21.5 | — | 10 | 0.0 | 184.02 |
| 40 | 5/100 | 10 | 0 | 34.7 | — | 8 | 0.4 | 4,925.61 |
| 50 | 4/120 | 10 | 0 | 24.9 | — | 8 | 1.2 | 281.54 |
| 50 | 5/100 | 10 | 0 | 32.2 | — | 7 | 1.1 | 7,810.35 |
| 60 | 5/120 | 10 | 0 | 27.2 | — | 5 | 0.6 | 224.02 |
| 60 | 6/100 | 10 | 0 | 33.5 | — | 1 | 1.1 | 739.99 |
| class 1 | | | test run $m_1$-II-start | | | test run $m_1$-III-start | | |
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | #opt | gap | $t_{cpu}$ |
| 40 | 4/120 | 10 | 10 | 0.0 | 133.94 | 10 | 0.0 | 70.03 |
| 40 | 5/100 | 10 | 8 | 0.3 | 3,924.23 | 8 | 0.3 | 898.57 |
| 50 | 4/120 | 10 | 8 | 1.0 | 235.89 | 8 | 0.8 | 233.55 |
| 50 | 5/100 | 10 | 7 | 1.1 | 6,350.80 | 7 | 1.1 | 6,103.00 |
| 60 | 5/120 | 10 | 6 | 0.7 | 2,443.32 | 7 | 0.9 | 7,618.27 |
| 60 | 6/100 | 10 | 1 | 0.7 | 768.41 | 2 | 0.7 | 21,687.49 |
| class 1 | | | test run $m_2$-I | | | test run $m_2$-II | | |
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | #opt | gap | $t_{cpu}$ |
| 40 | 4/120 | 10 | 5 | 3.0 | 3,774.27 | 10 | 0.0 | 4,109.53 |
| 40 | 5/100 | 10 | 3 | 2.4 | 13,252.45 | 10 | 0.0 | 2,832.78 |
| 50 | 4/120 | 10 | 3 | 4.3 | 1,290.99 | 8 | 0.8 | 93.85 |
| 50 | 5/100 | 10 | 1 | 3.6 | 8,737.76 | 7 | 1.1 | 755.85 |
| 60 | 5/120 | 10 | 2 | 2.7 | 10,480.56 | 8 | 1.1 | 2,370.62 |
| 60 | 6/100 | 10 | 0 | 6.9 | — | 3 | 0.5 | 15,378.48 |
| class 1 | | | test run $m_2$-II-start | | | test run $m_2$-III-start | | |
| $\|C\|$ | $K/Q$ | $no$ | #opt | gap | $t_{cpu}$ | #opt | gap | $t_{cpu}$ |
| 40 | 4/120 | 10 | 10 | 0.0 | 636.19 | 10 | 0.0 | 30.00 |
| 40 | 5/100 | 10 | 10 | 0.0 | 2,562.53 | 10 | 0.0 | 841.33 |
| 50 | 4/120 | 10 | 8 | 0.8 | 87.47 | 9 | 0.8 | 3,148.62 |
| 50 | 5/100 | 10 | 7 | 0.9 | 639.51 | 8 | 1.0 | 3,471.31 |
| 60 | 5/120 | 10 | 8 | 1.1 | 2,139.99 | 8 | 1.1 | 1,194.80 |
| 60 | 6/100 | 10 | 3 | 0.5 | 10,214.19 | 6 | 0.4 | 17,057.30 |

to optimality, whereas in test run $m_2$-III-start 51 optimal solutions are generated. With additional constraints, the value of the linear programming relaxation at the root node increases on average by 0.1% for the vehicle-flow model and by 0.2% for the commodity-flow model.

Since the results of test runs $m_i$-I, $m_i$-II, and $m_i$-II-start, $i = 1, 2$, are sobering, we skip pursuing further details for them.

We continue our analysis using small-scale symmetric problem instances with up to 22 customers. Table 3 shows the optimal solutions and run times. The instances are denoted by the same descriptors that are used in the literature. In column "costs" the lowest transportation costs (or distances traveled) and in column "$t_{cpu}$" the corresponding CPU time (in seconds) are given. With column "model" we demonstrate the model formulation that was used to create the best run time; $m_1$ is taken for the vehicle-flow model (test run $m_1$-III-start) and $m_2$ for the commodity-flow model (test run $m_2$-III-start). If the costs are written in bold type, the underlying instance is solved to optimality and the optimality is proven for the first time. The optimal solution for the Min (1989) problem instance was found and verified by Montané and Galvão (2006) within two hours. Thereby, the authors compared the heuristic solution to a lower bound obtained by CPLEX. With our approach, the optimality could be proven in two seconds.

Class 2 contains instances that are generated using deterministic rules. All instances are solved to optimality with low computational effort. Instances that are derived from the Solomon benchmark (here classes 3 and 4) stand out with the composition of geographical data. In group "C" the customers are placed in clusters, in group "R" the geographical data is randomly generated, and group "RC" contains a mix of clustered and randomly generated customer locations. For these

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

**Table 3: Computational results for small-scale symmetric instances**

| | $\|C\|$ | $K/Q$ | costs | $t_{cpu}$ | model | | $\|C\|$ | $K/Q$ | costs | $t_{cpu}$ | model |
|---|---|---|---|---|---|---|---|---|---|---|---|
| class 2 | | | | | | | | | | | |
| Mitra_1_1 | 19 | 2/10 | **210** | 0.09 | $m_1$ | Mitra_2_1 | 19 | 2/10 | **244** | 0.64 | $m_1$ |
| Mitra_1_2 | 19 | 10/10 | **290** | 0.92 | $m_1$ | Mitra_2_2 | 19 | 10/10 | **461** | 1.12 | $m_1$ |
| Mitra_1_6 | 19 | 10/10 | **290** | 1.22 | $m_2$ | Mitra_2_6 | 19 | 10/10 | **461** | 1.04 | $m_1$ |
| Mitra_1_16 | 19 | 10/10 | **290** | 0.72 | $m_2$ | Mitra_2_16 | 19 | 10/10 | **461** | 1.03 | $m_2$ |
| class 3 | | | | | | | | | | | |
| R121 | 15 | 3/80 | 610.80 | 0.36 | $m_2$ | R121 | 17 | 2/120 | 564.39 | 0.34 | $m_1$ |
| R141 | 15 | 3/80 | 750.06 | 0.20 | $m_1$ | R141 | 17 | 2/120 | 757.74 | 0.33 | $m_1$ |
| R161 | 15 | 2/80 | 1,166.82 | 0.15 | $m_1$ | R161 | 17 | 2/120 | 1,192.99 | 0.35 | $m_2$ |
| R181 | 15 | 2/80 | 1,968.38 | 3.33 | $m_1$ | R181 | 17 | 2/120 | 1,786.75 | 0.30 | $m_1$ |
| R1101 | 15 | 2/80 | 2,033.88 | 0.61 | $m_1$ | R1101 | 17 | 2/120 | 2,052.03 | 0.25 | $m_1$ |
| R121 | 15 | 2/120 | 542.20 | 0.40 | $m_1$ | R121 | 20 | 2/120 | **623.67** | 1.33 | $m_1$ |
| R141 | 15 | 2/120 | 669.72 | 0.13 | $m_1$ | R141 | 20 | 2/120 | 798.39 | 0.47 | $m_1$ |
| R161 | 15 | 2/120 | 1,162.58 | 1.62 | $m_1$ | R161 | 20 | 2/120 | 1,279.51 | 0.64 | $m_1$ |
| R181 | 15 | 2/120 | 1,755.95 | 0.07 | $m_1$ | R181 | 20 | 2/120 | 1,865.87 | 1.05 | $m_1$ |
| R1101 | 15 | 2/120 | 1,809.69 | 0.30 | $m_1$ | R1101 | 20 | 2/120 | **2,119.54** | 0.70 | $m_2$ |
| R121 | 17 | 4/80 | **726.06** | 6.03 | $m_1$ | | | | | | |
| R141 | 17 | 3/80 | 791.10 | 0.17 | $m_2$ | | | | | | |
| R161 | 17 | 3/80 | 1,211.22 | 0.24 | $m_1$ | | | | | | |
| R181 | 17 | 3/80 | **1,991.59** | 3.45 | $m_2$ | | | | | | |
| R1101 | 17 | 2/80 | **2,295.88** | 5.55 | $m_2$ | | | | | | |
| class 4 | | | | | | | | | | | |
| C101_20_02 | 20 | 4/100 | 272 | 2.66 | $m_1$ | C101_20_08 | 20 | 4/100 | 279 | 3.06 | $m_1$ |
| R101_20_02 | 20 | 3/100 | 329 | 1.32 | $m_1$ | R101_20_08 | 20 | 3/100 | 342 | 1.62 | $m_1$ |
| RC101_20_02 | 20 | 5/100 | 428 | 0.74 | $m_2$ | RC101_20_08 | 20 | 5/100 | 458 | 1.63 | $m_2$ |
| class 5 | | | | | | | | | | | |
| 3C_20_50_1 | 20 | 11/150 | 12,720 | 0.81 | $m_1$ | 3C_20_80_1 | 20 | *11*/150 | 12,802 | 2.54 | $m_1$ |
| 3C_20_50_2 | 20 | 7/200 | *10,461* | 26.61 | $m_1$ | 3C_20_80_2 | 20 | 8/200 | 10,087 | 2.43 | $m_1$ |
| 3C_20_50_3 | 20 | 6/300 | 8,387 | 2.91 | $m_1$ | 3C_20_80_3 | 20 | 5/300 | 8,317 | 9.99 | $m_1$ |
| 3C_20_66_1 | 20 | 12/150 | 14,578 | 2.12 | $m_1$ | | | | | | |
| 3C_20_66_2 | 20 | 8/200 | 11,176 | 687.18 | $m_2$ | | | | | | |
| 3C_20_66_3 | 20 | 5/300 | 8,160 | 12.79 | $m_1$ | | | | | | |
| Min | 22 | 2/10,500 | 88 | 1.36 | $m_1$ | | | | | | |

instances the vehicle-flow model performs reasonably well. We are able to solve five problem instances to optimality for the first time. Class 5 contains vehicle routing instances from real-world projects of the University of Cologne. In contrast to the other classes, each instance considers a large number of vehicles with high capacity. All instances with 20 customers are solved to optimality, but for 3C_20_80_1 the number $K$ of vehicles used has to be increased to 11 to obtain the result indicated in the literature. Additionally, we discover that the optimal objective function value of instance 3C_20_50_2 (written in italic type) is lower than those published in the literature. The deviation of 10.5% might occur from insufficient rounding during the optimization process in Dell'Amico, Righini, and Salani (2006). The Min (1989) problem is obtained from a real-world data set. It is aimed at studying the effects and capabilities of the solution procedure in a more realistic way. Here, the procedure based on the vehicle-flow model provides the fastest run time.

The best results for instances considered in Table 3 are created with the vehicle-flow model $m_1$. In spite of the large number of "big-M constraints", $m_1$ seems to be the best choice to solve small-scale symmetric instances of the VRPSDP.

Tables 4 and 5 show the results concerning run times and solution gap for the remaining 57 instances with 40 customers (classes 4 and 5) and with 50 customers (classes 6 and 7). If the optimal solution is not reached within the time limit, column "costs" shows the lowest transportation costs of a feasible solution (best integer solution) and

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

**Table 4: Computational results for medium-scale symmetric instances**

| | $K/Q$ | costs | gap | $t_{cpu}$ | model |
|---|---|---|---|---|---|
| class 4 | | | | | |
| C101_40_02 | 8/100 | 551 | 0.0 | 904.50 | $m_2$ |
| R101_40_02 | 6/100 | *596* | 0.0 | 380.31 | $m_2$ |
| RC101_40_02 | 9/100 | 886 | 0.0 | 23.24 | $m_2$ |
| C101_40_08 | 8/100 | 569 | 0.0 | 1,897.99 | $m_2$ |
| R101_40_08 | 6/100 | 646 | 5.5 | — | $m_2$ |
| RC101_40_08 | 9/100 | 926 | 0.0 | 2,264.59 | $m_2$ |
| class 5 | | | | | |
| 3C_40_50_1 | 22/150 | *27,245* | 0.8 | — | $m_2$ |
| 3C_40_50_2 | 16/200 | 21,773 | 2.8 | — | $m_2$ |
| 3C_40_50_3 | 10/300 | *15,523* | 3,7 | — | $m_2$ |
| 3C_40_66_1 | 22/150 | 25,981 | 0.0 | 1,083.30 | $m_2$ |
| 3C_40_66_2 | 15/200 | 21,366 | 2.3 | — | $m_2$ |
| 3C_40_66_3 | 10/300 | 15,293 | 2.1 | — | $m_2$ |
| 3C_40_80_1 | 21/150 | 26,617 | 0.0 | 2,261.65 | $m_2$ |
| 3C_40_80_2 | 16/200 | 20,652 | 0.0 | 13,039.67 | $m_2$ |
| 3C_40_80_3 | 10/300 | 15,365 | 0.8 | — | $m_2$ |

**Table 5: Computational results for medium-scale asymmetric instances**

| name | $K/Q$ | costs | gap | $t_{cpu}$ | model |
|---|---|---|---|---|---|
| class 6 | | | | | |
| asym-SCA3-0 | 4/823.69 | 677.35 | 0.0 | 4,474.26 | $m_1$ |
| asym-SCA3-1 | 4/772.50 | 758.90 | 0.0 | 506.35 | $m_1$ |
| asym-SCA3-2 | 4/832.67 | 735.18 | 0.0 | 41.39 | $m_1$ |
| asym-SCA3-3 | 4/859.51 | 735.79 | 0.0 | 63.88 | $m_1$ |
| asym-SCA3-4 | 4/986.34 | 741.75 | 0.0 | 503.77 | $m_1$ |
| asym-SCA3-5 | 4/813.17 | 702.45 | 0.0 | 48.39 | $m_1$ |
| asym-SCA3-6 | 4/806.68 | 707.72 | 0.0 | 58.62 | $m_1$ |
| asym-SCA3-7 | 4/833.51 | 708.24 | 0.0 | 64.49 | $m_1$ |
| asym-SCA3-8 | 4/827.01 | 771.94 | 0.0 | 18,631.57 | $m_1$ |
| asym-SCA3-9 | 4/753.84 | 726.77 | 0.0 | 52.42 | $m_1$ |
| asym-SCA8-0 | 9/308.88 | 1,026.79 | 3.9 | — | $m_1$ |
| asym-SCA8-1 | 9/289.69 | 1,127.41 | 3.3 | — | $m_1$ |
| asym-SCA8-2 | 9/312.25 | 1,126.12 | 2.2 | — | $m_1$ |
| asym-SCA8-3 | 9/322.31 | 1,062.99 | 3.0 | — | $m_2$ |
| asym-SCA8-4 | 9/369.88 | 1,114.12 | 1.2 | — | $m_1$ |
| asym-SCA8-5 | 9/304.94 | 1,085.96 | 1.4 | — | $m_1$ |
| asym-SCA8-6 | 9/302.50 | 1,038.59 | 1.3 | — | $m_1$ |
| asym-SCA8-7 | 9/312.57 | 1,114.17 | 3.1 | — | $m_1$ |
| asym-SCA8-8 | 9/310.13 | 1,165.08 | 4.4 | — | $m_1$ |
| asym-SCA8-9 | 9/282.69 | 1,145.71 | 3.2 | — | $m_1$ |
| asym-CON3-0 | 4/808.10 | 667.46 | 0.0 | 120.87 | $m_1$ |
| asym-CON3-1 | 4/926.26 | 590.82 | 0.0 | 657.28 | $m_1$ |
| asym-CON3-2 | 4/854.49 | 558.89 | 0.0 | 24,496.66 | $m_2$ |
| asym-CON3-3 | 4/849.40 | 634.93 | 0.0 | 49.73 | $m_1$ |
| asym-CON3-4 | 4/877.64 | 627.95 | 0.0 | 419.08 | $m_1$ |
| asym-CON3-5 | 4/772.54 | 603.56 | 0.0 | 17,746.84 | $m_1$ |
| asym-CON3-6 | 4/734.11 | 539.58 | 0.0 | 28,863.43 | $m_1$ |
| asym-CON3-7 | 4/829.02 | 627.05 | 0.0 | 206.48 | $m_1$ |
| asym-CON3-8 | 4/823.92 | 561.65 | 0.0 | 171.10 | $m_1$ |
| asym-CON3-9 | 4/718.82 | 619.95 | 0.0 | 28,816.44 | $m_1$ |
| asym-CON8-0 | 9/303.04 | 918.21 | 1.8 | — | $m_1$ |
| asym-CON8-1 | 9/347.35 | 772.44 | 1.0 | — | $m_2$ |
| asym-CON8-2 | 9/320.44 | 738.99 | 0.6 | — | $m_2$ |
| asym-CON8-3 | 10/318.52 | 857.35 | 1.3 | — | $m_2$ |
| asym-CON8-4 | 9/329.11 | 816.81 | 1.5 | — | $m_1$ |
| asym-CON8-5 | 9/289.70 | 798.07 | 1.5 | — | $m_2$ |
| asym-CON8-6 | 9/275.29 | 718.36 | 2.2 | — | $m_1$ |
| asym-CON8-7 | 9/310.88 | 863.39 | 0.5 | — | $m_1$ |
| asym-CON8-8 | 9/308.97 | 808.17 | 3.9 | — | $m_2$ |
| asym-CON8-9 | 9/269.56 | 843.84 | 3.0 | — | $m_2$ |
| class 7 | | | | | |
| asym-CMT1x | 3/160 | 510.23 | 0.0 | 283.72 | $m_2$ |
| asym-CMT1y | 3/160 | 511.38 | 0.0 | 728.51 | $m_1$ |

column "$t_{cpu}$" displays a horizontal bar.

For all instances with 40 customers, the commodity-flow model produces the best results. The approach solves five out of six problem instances in class 4 to optimality. For instances in class 5 it is difficult to generate optimal solutions, because the lower bounds obtained from the linear programming relaxation of the formulations are relatively weak at the upper levels of the search tree. However, three instances are solved to optimality and for the others, the gaps are very small. The optimal solution of instance R101_40_02 and the upper bounds of instances 3C_40_50_1 and 3C_40_50_3 are lower than the optimal solutions published in Dell'Amico, Righini, and Salani (2006) (deviations are 0.8%, 1.2%, and 0.7%).

Class 6 consists of random test instances, where two different geographical scenarios are examined; a uniformly distributed and an urban configuration of customers. The instances of class 7 are derived from the Christofides, Mingozzi, and Toth (1979) benchmark for the CVRP. Both test sets consist of instances with 50 customers. Optimal or near optimal solutions for the original symmetric instances can be found in Subramanian, Uchoa, and Ochi (2010) or Subramanian, Uchoa, Pessoa, and Ochi (2011). In order to construct asymmetric instances for the VRPSDP, the instances are modified (subsection 5.1). Table 5 depicts the corresponding run times and solution gaps for the asymmetric benchmark sets under consideration. For all instances, the vehicle-flow model provides the best solution quality. All instances with three or four vehicles are solved to optimality. For instances with nine or ten vehicles the enumeration could not be completed within the time limit. This means that the number of vehicles is a significant parameter in order to identify the difficulty of the underlying problem.

The analytical study of both models shows that the

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

**Table 6: Average numbers of cuts; vehicle-flow and commodity-flow model**

| | | | model $m_1$ | | | | model $m_2$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| class | $\|C\|$ | *no* | clique | implied-bound | flow-cover | user | implied-bound | flow-cover | flow-path | user |
| 1 | 20 | 20 | 2.4 | 6.6 | 2.6 | 31.5 | 0.2 | 11.3 | 1.7 | 19.7 |
| 1 | 30 | 20 | 3.1 | 90.2 | 8.3 | 81.5 | 0.1 | 8.2 | 0.9 | 54.7 |
| 1 | 40 | 20 | 5.8 | 398.1 | 43.5 | 235.6 | 0.0 | 6.6 | 0.0 | 163.7 |
| 1 | 50 | 20 | 6.4 | 577.9 | 66.3 | 261.1 | 0.0 | 4.6 | 0.1 | 185.8 |
| 1 | 60 | 20 | 8.1 | 1066.0 | 118.1 | 353.3 | 0.0 | 3.7 | 0.0 | 270.4 |
| 2 | 19 | 8 | 17.9 | 11.0 | 0.0 | 15.0 | 54.4 | 33.6 | 0.0 | 6.0 |
| 3 | 15 | 10 | 7.0 | 95.6 | 4.3 | 25.1 | 90.6 | 71.0 | 5.8 | 23.4 |
| 3 | 17 | 10 | 7.4 | 114.4 | 7.4 | 32.6 | 152.9 | 102.3 | 9.4 | 26.9 |
| 3 | 20 | 5 | 8.8 | 58.5 | 3.8 | 33.5 | 148.0 | 104.0 | 12.3 | 24.3 |
| Min | 22 | 1 | 6.0 | 35.0 | 3.0 | 76.0 | 138.0 | 175.0 | 35.0 | 54.0 |
| 4 | 20 | 6 | 8.8 | 168.7 | 13.5 | 93.8 | 187.2 | 90.2 | 8.3 | 59.3 |
| 4 | 40 | 6 | 10.7 | 1,780.5 | 148.3 | 264.3 | 45.8 | 8.7 | 1.0 | 175.8 |
| 5 | 20 | 9 | 36.3 | 249.7 | 20.4 | 124.2 | 354.0 | 157.9 | 20.1 | 100.8 |
| 5 | 40 | 9 | 74.4 | 2,103.8 | 96.6 | 348.1 | 248.4 | 30.0 | 5.9 | 201.1 |
| 6 | 50 | 40 | 15.6 | 1,519.2 | 148.5 | 283,5 | 0.0 | 30.2 | 3.8 | 208.7 |
| 7 | 50 | 2 | 10.0 | 824.0 | 123.0 | 231.5 | 0.0 | 21.0 | 3.5 | 258.5 |
| average | | | 12.2 | 676.4 | 63.0 | 176.4 | 51.5 | 31.9 | 3.4 | 127.4 |

CPU time needed to obtain an optimal solution depends on the number of customers, the number of vehicles, and the structure of the considered instances. The approach based on the vehicle-flow formulation is effective in finding optimal solutions for small-scale symmetric instances, and in finding good upper bounds for asymmetric instances that contain either clustered customers or randomly generated customers (instances of type "C" or "R"; see classes 6 and 7). In contrast, the approach based on the commodity-flow model turns out to be more robust in generating upper bounds as well as optimal solutions for medium-scale symmetric problem instances. Moreover, model $m_2$ performs well for real-world asymmetric instances that contain a mix of clustered and randomly generated customer locations (instances of type "RC"; see class 1).

Table 6 shows the average numbers of cuts added during the optimization process. Thereby, the cuts of the individual models, i.e., $m_1$ and $m_2$, are separated into different columns. We constitute the rounded capacity cuts as "user cuts" just like CPLEX does.

Considering the average numbers of cuts, it is obvious that more cuts are generated for the vehicle-flow model than for the commodity-flow model. For the vehicle-flow model, user cuts and implied-bound cuts are particularly created in the sub-problems of the branch-and-bound tree. Rounded capacity cuts are known to be effective for routing problems (see, e.g., Fukasawa, Longo, Lysgaard, De Aragao, Reis, Uchoa et al. 2006) and implied-bound cuts are useful if the binary variables imply bounds on continuous variables (relevant for load constraints (6)–(8), and (14)–(16)). A clique can be introduced if at most one variable in the group of binary variables can be positive in any integer feasible solution (constraints (3) and (4)). For the commodity-flow model, mainly user cuts are generated. Furthermore, implied-bound cuts are only added for symmetric instances (classes 2 to 5), this means that inequalities (23), (26), and (27) are stronger for asymmetric instances. Flow-cover and flow-path cuts treat the continuous variables as nodes and apply binary variables to indicate the flow entering and leaving a node.

## 6 Conclusion

In this paper, we have presented the vehicle routing with simultaneous delivery and pick-up that typically occurs in closed-loop supply chains. In order to solve the VRPSDP, a vehicle-flow and a commodity-flow formulation are proposed. Considering the models, optimal solutions for small-scale and medium-scale problem instances can efficiently be generated by a standard solver (e.g., CPLEX). To facilitate the solution process, we add problem-specific preprocessing techniques and cutting planes. The computational tests have

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

been performed on symmetric and asymmetric benchmark instances with up to 60 customers. Most instances are solved to optimality and for the remaining instances good upper bounds could be generated.

Future work will include the explicit consideration of multiple objective functions in the mathematical models. In addition, combining the VRPSDP with the problem of optimizing transport flows to solve the joint problem of determining the optimal connections between pairs of nodes and finding the optimal set of vehicle routes may be an interesting issue.

## Acknowledgments

We are indebted to Jeng-Fung Chen, Jan Dethloff, Matteo Salani, and Anand Subramanian for kindly providing us with their VRPSDP benchmark instances. The benchmarks presented herein and the results obtained may be downloaded from the web page: http://www.wiwi.tu-clausthal.de/abteilungen/unternehmensforschung/forschung.

## References

Ai, The Jin and Voratas Kachitvichyanukul (2009): A Particle Swarm Optimization for the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Computers and Operations Research*, 36 (5): 1693-1702.

Akçalı, Elif, Sila Çetinkaya, and Halit Üster (2009): Network Design for Reverse and Closed-Loop Supply Chains: An Annotated Bibliography of Models and Solution Approaches, *Networks*, 53 (3): 231-248.

Angelelli, Enrico and Renata Mansini (2002): The Vehicle Routing Problem With Time Windows and Simultaneous Pick-Up and Delivery, in: Andreas Klose, Maria Gracia Speranza, and Luk N. Van Wassenhove (eds.): *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, Springer: Berlin, 249-267.

Bianchessi, Nicola and Giovanni Righini (2007): Heuristic Algorithms for the Vehicle Routing Problem With Simultaneous Pick-Up and Delivery, *Computers and Operations Research*, 34 (2): 578-594.

Bostel, Nathalie, Pierre Dejax, and Zhiqiang Lu (2005): The Design, Planning, and Optimization of Reverse Logistics Networks, in: André Langevin and Diane Riopel (eds.): *Logistics Systems: Design and Optimization*, Springer: New York, NY, 171-212.

Chen, Jeng-Fung and Tai-Hsi Wu (2006): Vehicle Routing Problem With Simultaneous Deliveries and Pickups, *Journal of the Operational Research Society*, 57 (5): 579-587.

Christofides, Nicos, Aristide Mingozzi, and Paolo Toth (1979): The Vehicle Routing Problem, in: Nicos Christofides, Aristide Mingozzi, Paolo Toth, and Claudio Sandi (eds.): *Combinatorial Optimization*, Wiley: Chichester, UK et al., 315-338.

De Koster, René B. M., Marisa P. De Brito, and Majsa A. Van De Vendel (2002): How to Organise Return Handling: An Exploratory Study With Nine Retailer Warehouses, *International Journal of Retail and Distribution Management*, 30 (8): 407-421.

Dell'Amico, Mauro, Giovanni Righini, and Matteo Salani (2006): A Branch-and-Price Approach to the Vehicle Routing Problem With Simultaneous Distribution and Collection, *Transportation Science*, 40 (2): 235-247.

Dethloff, Jan (2001): Vehicle Routing and Reverse Logistics: The Vehicle Routing Problem With Simultaneous Delivery and Pick-Up, *OR Spektrum*, 23 (1): 79-96.

Fukasawa, Ricardo, Humberto Longo, Jens Lysgaard, Marcus P. De Aragao, Marcelo Reis, Eduardo Uchoa, and Renato F. Werneck (2006): Robust Branch-and-Cut-and-Price for the Capacitated Vehicle Routing Problem, *Mathematical Programming*, 106 (3): 491-511.

Gajpal, Yuvraj and Prakash Abad (2009): An Ant Colony System (ACS) for Vehicle Routing Problem With Simultaneous Delivery and Pickup, *Computers and Operations Research*, 36 (12): 3215-3223.

Garey, Michael R. and David S. Johnson (1979): *Computers and Intractablility: A Guide to the Theory of NP-Completeness*, Freeman: San Francisco, CA et al.

Gillett, Billy E. and Leland R. Miller (1974): A Heuristic Algorithm for the Vehicle Dispatch Problem, *Operations Research*, 22 (2): 340-349.

Gomes Salema, Maria I., Ana P. Barbosa Póvoa, and Augusto Q. Novais (2009): A Strategic and Tactical Model for Closed-Loop Supply Chains, *OR Spectrum*, 31 (3): 573-599.

Gomes Salema, Maria I., Ana P. Barbosa Póvoa, and Augusto Q. Novais (2010): Simultaneous Design and Planning of Supply Chains With Reverse Flows: A Generic Modelling Framework, *European Journal of Operational Research*, 203 (2): 336-349.

Gu, Qiaolun and Hanhua Ji (2008): An Integrated Logistics Operational Model for Remanufacturing/Manufacturing System Based on the Consumer Market, *International Journal of Logistics Systems and Management*, 4 (1): 21-39.

Halse, Karsten (1990): Vehicle Routing Problem With Simultaneous Delivery and Pick-Up Points, Technical Report No. 4/90, Technical University of Denmark, Lingby.

BuR – Business Research
Official Open Access Journal of VHB
German Academic Association for Business Research (VHB)
Volume 6 | Issue 1 | May 2013 | 77–92

Homberger, Jörg and Hermann Gehring (2002): Parallelization of a Two-Phase Metaheuristic for Routing Problems With Time Windows, *Journal of Heuristics*, 8 (3): 251-276.

Irnich, Stefan (2008): Resource Extension Functions: Properties, Inversion, and Generalization to Segments, *OR Spectrum*, 30 (1): 113-148.

Lysgaard, Jens (2003): CVRPSEP: A package of Separation Routines for the Capacitated Vehicle Routing Problem, Working Paper, University of Aarhus.

Min, Hokey (1989): The Multiple Vehicle Routing Problem With Simultaneous Delivery and Pick-up Points, *Transportation Research Part A*, 23A (5): 377-386.

Mitra, Subrata (2005): An Algorithm for the Generalized Vehicle Routing Problem With Backhauling, *Asia-Pacific Journal of Operational Research*, 22 (2): 153-169.

Montané, Fermín Alfredo T. and Roberto D. Galvão (2006): A Tabu Search Algorithm for the Vehicle Routing Problem With Simultaneous Pick-Up and Delivery Service, *Computers and Operations Research*, 33 (3): 595-619.

Nagy, Gábor and Saïd Salhi (2005): Heuristic Algorithms for Single and Multiple Depot Vehicle Routing Problems With Pickups and Deliveries, *European Journal of Operational Research*, 162 (1): 126-141.

Parragh, Sophie N., Karl F. Doerner, and Richard F. Hartl (2008): A Survey on Pickup and Delivery Problems, Part I: Transportation Between Customers and Depot, *Journal für Betriebswirtschaft*, 58 (1): 21-51.

Privé, Julie, Jacques Renaud, Fayez Boctor, and Gilbert Laporte (2006): Solving a Vehicle-Routing Problem Arising in Soft-Drink Distribution, *Journal of the Operational Research Society*, 57 (9): 1045-1052.

Rieck, Julia and Jürgen Zimmermann (2009): A Hybrid Algorithm for Vehicle Routing of Less-Than-Truckload Carriers, in: Martin J. Geiger, Walter Habenicht, Marc Sevaux, and Kenneth Sörensen (eds.): *Metaheuristics in the Service Industry*, Springer: Berlin, 155-171.

Ryan, David M., Curt Hjorring, and Fred Glover (1993): Extensions of the Petal Method for Vehicle Routeing, *Journal of the Operational Research Society*, 44 (3): 289-296.

Salhi, Saïd and Gábor Nagy (1999): A Cluster Insertion Heuristic for Single and Multiple Depot Vehicle Routing Problems With Backhauling, *Journal of the Operational Research Society*, 50 (10): 1035-1042.

Sheu, Jiuh-Biing, Yi-Hwa Chou, and Chun-Chia Hu (2005): An Integrated Logistics Operational Model for Green-Supply Chain Management, *Transportation Research Part E*, 41 (4): 287-313.

Solomon, Marius M. (1987): Algorithms for the Vehicle Routing and Scheduling Problem With Time Window Constraints, *Operations Research*, 35 (2): 254-265.

Subramanian, Anand, Lúcia M. A. Drummond, Cristiana Bentes, Luiz S. Ochi, and Ricardo Farias (2010): A Parallel Heuristic for the Vehicle Routing Problem With Simultaneous Pickup and Delivery, *Computers and Operations Research*, 37 (11): 1899-1911.

Subramanian, Anand, Eduardo Uchoa, and Luiz S. Ochi (2010): New Lower Bounds for the Vehicle Routing Problem With Simultaneous Pickup and Delivery, in: Paola Festa (ed.): *Experimental Algorithms*, Springer: Berlin, 276-287.

Subramanian, Anand, Eduardo Uchoa, Artur A. Pessoa, and Luiz S. Ochi (2011): Branch-and-Cut With Lazy Separation for the Vehicle Routing Problem With Simultaneous Pickup and Delivery, *Operations Research Letters*, 39 (5): 338-341.

Toth, Paolo and Daniele Vigo (2002): An Overview of Vehicle Routing Problems, in: Paolo Toth and Daniele Vigo (eds.): *The Vehicle Routing Problem*, SIAM: Philadelphia, PA, 1-26.

Wassan, Niaz A., Abdul H. Wassan, and Gábor Nagy (2008): A Reactive Tabu Search Algorithm for the Vehicle Routing Problem With Simultaneous Pick-Ups and Deliveries, *Journal of Combinatorial Optimization*, 15 (4): 368-386.

Zachariadis, Emmanouil E., Christos D. Tarantilis, and Chris T. Kiranoudis (2009): A Hybrid Metaheuristic Algorithm for the Vehicle Routing Problem With Simultaneous Delivery and Pick-Up Service, *Expert Systems With Applications*, 36 (2): 1070-1081.

Zachariadis, Emmanouil E., Christos D. Tarantilis, and Chris T. Kiranoudis (2010): An Adaptive Memory Methodology for the Vehicle Routing Problem With Simultaneous Pick-Ups and Deliveries, *European Journal of Operational Research*, 202 (2): 401-411.

## Biographies

**Julia Rieck** studied Applied Mathematics at the University of Göttingen and the University of Hamburg. She obtained her PhD in 2008 and now holds the position of a researcher and teaching assistant at Clausthal University of Technology. Her research interests include vehicle routing and location problems, distribution network design, and project scheduling problems.

**Jürgen Zimmermann** is Professor of Operations Research at Clausthal University of Technology since 2002. He received his Diploma in Applied Mathematics and his PhD at the University of Karlsruhe, where he also completed his Habilitation. His research interests include project management and scheduling, machine scheduling, and routing problems.