WORKING PAPER SERIES

Integrated Order Picking and Vehicle Routing with Due Dates

Daniel Schubert/André Scholz/Gerhard Wäscher

Working Paper No. 7/2017

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

FACULTY OF ECONOMICS AND MANAGEMENT

Impressum (§ 5 TMG)

Herausgeber:
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Der Dekan

Verantwortlich für diese Ausgabe:
Daniel Schubert, André Scholz, Gerhard Wäscher
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Postfach 4120
39016 Magdeburg
Germany

http://www.fww.ovgu.de/femm

Bezug über den Herausgeber
ISSN 1615-4274

Integrated Order Picking and Vehicle Routing with Due Dates

D. Schubert, A. Scholz, G. Wäscher

Abstract

Supermarkets typically order their goods from a centrally located distribution center (warehouse). Each order that the warehouse receives is characterized by the requested items, the location of the respective supermarket and a due date by which the items have to be delivered. For processing an order, a human operator (order picker) retrieves the requested items from their storage locations in the warehouse first. The items are then available for shipment and loaded on the vehicle which performs the tour including the respective location of the supermarket. Whether and to which extent a due date is violated (tardiness) depends on the composition of the tours, the corresponding routes and the start dates of the tours (vehicle routing subproblem). The start date of a tour, however, is also affected by the assignment of orders to pickers and the sequence according to which the orders are processed by the pickers (order picking subproblem). Although both subproblems are closely interconnected, they have not been considered simultaneously in the literature so far. In this paper, an iterated local search algorithm is designed for the simultaneous solution of the subproblems. By means of extensive numerical experiments, it is shown that the proposed approach is able to generate high-quality solutions even for large instances. Furthermore, the economic benefits of an integrated solution are investigated. Problem classes are identified, where the sequential solution of the subproblems leads to acceptable results, and it is pointed out in which cases an integrated solution is inevitable.

Keywords: Vehicle Routing, Order Picking, Parallel Machine Scheduling, Iterated Local Search

Corresponding author:
André Scholz
Otto-von-Guericke University Magdeburg, Faculty of Economics and Management
Postbox 4120, 39016 Magdeburg, Germany
Phone: +49 391 67 51841
Fax: +49 391 67 48223
Email: andre.scholz@st.ovgu.de

1 Introduction

Supermarkets are typically supplied once per workday (DVZ, 2013) from a centrally located distribution center (warehouse) with perishable goods. When the warehouse has received a certain number of orders from the supermarkets, the orders are assigned to human operators (order pickers) who retrieve the respective items from their storage locations. Each order picker processes the orders one by one in a particular sequence. All items belonging to an order are grouped together on transport devices like pallets or boxes and then, together with the items from other orders, loaded on vehicles (trucks) which deliver the goods to the respective supermarkets. This gives rise to a vehicle routing problem, namely how the items of the various orders are to be assigned to vehicles and in which sequence the supermarkets are to be visited on each tour. The solution to the vehicle routing problem determines the actual delivery dates, i.e. the points in time when each supermarket is being served. However, a vehicle can only leave from the warehouse and the respective tour can only be started when the items of all orders allocated to the tour have been retrieved completely. Thus, the actual delivery dates are also affected by the assignment of orders to order pickers and the sequence in which the orders are processed.

In practice, distribution centers and supermarkets have agreed on deadlines by which the ordered items have to be delivered. For supermarkets, complying with such deadlines is of uttermost importance as only very limited safety stocks exist and empty shelves will result in clients satisfying their demands at competitor outlets. For the distribution centers, a violation of the deadlines will, therefore, result in – often heavy – fines or even in the loss of customers if the deadlines are violated more permanently. However, due to short response times which have also been agreed between distribution centers and supermarkets, deadlines are often difficult to meet. Thus, deadlines will not only have to be considered for the determination of vehicle tours but also for the assignment of orders to order pickers and the scheduling of the orders. Consequently, dispatchers of the picking and shipping processes in warehouses are confronted with a complex decision problem. Given a set of orders from supermarkets and corresponding deadlines, it has to be decided (1) how these orders have to be assigned to order pickers, (2) how the orders assigned to each order picker have to be sequenced, (3) how the orders have to be allocated to vehicles, and (4) in which sequence the supermarkets should be visited by each vehicle such that the violation of the deadlines is minimized. We will refer to this problem as the order assignment and sequencing, and vehicle routing problem (OASVRP).

So far, both in literature and in practice of warehouse management and control, the OASVRP has not been dealt with holistically. Instead, the order assignment and the order sequencing problem on the one hand and the vehicle routing problem, on the other hand, are treated and solved separately (Schmid et al., 2013). With respect to the previously sketched interdependencies between these problems, it can be expected, though, that an integrative solution approach to the OASVRP can provide a significant source of the reduction of costs and improved customer service by allowing for an improved compliance with given delivery deadlines. Our goal, therefore, is twofold: First, we intend to present a solution approach to the OASVRP which provides high-quality solutions within an acceptable amount of computing time. Since the above-mentioned subproblems are NP-hard already, we concentrate on a heuristic solution approach. In particular, an iterated local search algorithm is proposed. This type of metaheuristic is chosen since it has already been proven to provide excellent results for other challenging optimization problems in warehouse management. Second, by means of this algorithm, we will analyze whether, under which conditions, and to what extent benefits arise from dealing with the OASVRP holistically.

Special attention will be given to the fact that in practice large problem instances have to be solved. For instance, the EDEKA group Minden-Hannover, a large cooperative of independent supermarkets in Germany, serves 1513 supermarkets from nine warehouses (EDEKA, 2017), i.e. each warehouse has to provide goods for more than 150 customers on average.

The remainder of the paper is organized as follows: In Section 2, the OASVRP will be stated in greater detail. The related literature will be reviewed in Section 3. Section 4 comprises the presentation of the proposed iterated local search approach, where the generation of an initial solution, the structure of the neighborhoods used within the improvement phase and the design of the perturbation phase will be described in particular. Section 5 is devoted to the numerical experiments which have been carried out in order to evaluate the performance of the proposed algorithm but also in order to identify the benefits resulting from a holistic approach to the OASVRP. The paper concludes with a summary and an outlook on further research (Section 6).

2 Problem description

The order assignment and sequencing, and vehicle routing problem (OASVRP), which will be described in this section, deals with picking requested items from a warehouse and delivering them to the respective customers. Let a set of orders be given, each of which specifying certain items and the corresponding demands from a particular customer. Furthermore, each order has been assigned a deadline (due date) according to which the items have to be received by the customer. The items of each order have to be shipped as a unit, split deliveries are not permitted. In order to make a customer order available for shipping, the requested items have to be collected from the warehouse. Each customer order is processed separately, i.e. it may not be merged (batched) with other customer orders.

Human operators (order pickers) walk or ride through the warehouse, retrieving the items from known storage locations. Picking is performed on (picker) tours through the warehouse, i.e. each order picker starts from the depot, visits the locations of the items to be collected and, afterwards, returns to the depot where he/she deposits the collected items. The distance which has to be covered for collecting all items of an order and, correspondingly, the time, which is required to do so, is dependent on the sequence according to which the order picker visits the locations. The determination of the sequence is part of the picker routing problem. Thus, a solution to the picker routing problem, e.g. obtained by application of so-called routing strategies (Roodbergen, 2001), gives the processing time of an order, i.e. the time which passes from the moment the order picker leaves the depot until the moment he/she returns to the depot. As customer orders are processed separately, the processing times can be computed in advance and assumed to be given.

The number of order pickers is limited. Thus, each order picker will have to process several orders in sequence. When all items of an order have been collected and forwarded to the depot by a picker, the order is considered as finalized and available for shipping. The point in time when an order is finalized will be denoted as the release date of an order. It is determined by its processing time and the sum of the processing times of the orders which have been processed before by the respective order picker. In other words, the release date of an order is dependent on how the orders are assigned to order pickers and how they are scheduled.

A fleet of homogeneous vehicles is based at the warehouse. The vehicles perform tours from the warehouse to the customer locations and back on which the requested items are delivered to the customers. Thus, for each vehicle tour, it has to be decided which customers should be served and in which sequence they should be visited. Each tour is started by loading an available vehicle with the items requested by the customers assigned to the respective tour. Only items from orders finalized for shipping may be loaded, and all items of an order must be loaded completely on the same vehicle. The customer locations are visited one after another and the respective requested items are unloaded. Service times have to be taken into account for the loading operations at the warehouse as well as for the unloading operations at the customer locations. The length of a tour (i.e. its duration) can then be defined as the sum of all service times required at the warehouse and at the customer locations visited, plus the travel times needed by the vehicle for moving from the warehouse to the first customer location, between the customer locations, and from the last customer location back to the warehouse. It is limited by a driving time constraint (Prescott-Gagnon et al., 2010). The point in time when a vehicle returns to the warehouse after having visited all customers of a tour will be denoted as the completion date of this tour.

Loading of a vehicle for a particular tour could be started as soon as picking of all orders which have been assigned to the tour has been finalized. However, the number of vehicles is limited, and vehicles may have to perform multiple tours. Loading of the orders for a particular tour, thus, may have to wait until the vehicle has returned from a previous tour. The start date of a tour is correspondingly defined as the maximum of the release dates of all orders assigned to the tour and the completion date of the tour previously performed by the vehicle.

Each customer order is characterized by a certain due date, which has been agreed on by the warehouse and the customer. The point in time when the requested items of a customer order are actually unloaded at the customer location will be named the delivery date of the order. An order which has not been received by the customer by the due date results in customer dissatisfaction, fines or even in the loss of the customer if such delays happen to occur over a longer period of time. Complying with agreed due dates, therefore, is of uttermost importance to the economic success of distribution warehouses and will provide the core criterion for the evaluation of how the warehouse manages to process customer orders. Since the consequences of delayed deliveries are often dependent on the length of the delays, we will refer to the total tardiness of all customer orders here (also see, e.g., Ullrich (2013) who have used the total tardiness as an evaluation criterion in similar problem settings). In case that the due date of an order is not met, its tardiness equals the difference between the delivery date and the due date. If the order is delivered in time, the tardiness of the order amounts to zero. Then, the total tardiness of a set of customer orders equals the sum of the tardiness of all orders in the set.

The OASVRP can now be stated. Let the following be given:

- a set of customers and their locations, a limited number of order pickers, and a homogeneous fleet of vehicles,
- a set of customer orders with agreed due dates and (picking) processing times,
- travel times between the customer locations and between the warehouse and the customer locations,
- a limit on the tour length,
- a service time required for loading the vehicles at the warehouse and a service time required for unloading the vehicles at the customer locations.

The following six questions have then to be answered (simultaneously) such that the total tardiness of all customer orders is minimized and the given limit on the tour length is not exceeded:

1) For each customer order, to which order picker should it be assigned?
2) For each order picker, in which sequence should the assigned customer orders be processed?
3) For each customer location, to which tour should it be assigned?
4) For each tour, in which sequence should the assigned customer locations be visited?
5) For each tour, to which vehicle should the tour be assigned?
6) For each vehicle, in which sequence should the tours assigned to the vehicle be processed?

Fig. 1 illustrates a solution of an instance of the OASVRP with nine customer orders, two pickers and two vehicles. Fig. 1a depicts the temporal aspects by means of a Gantt chart; it further demonstrates the assignment of customer orders to order pickers and the sequence according to which orders are processed, as well as the assignment of tours to vehicles and the sequence according to which the tours are performed. Picker #1 processes customer order #1 first, then continues with order #2, order #3 and order #5. Correspondingly, picker #2 starts with processing customer order #6, followed by orders #7, #4, #8, and #9. The (picking) processing time of each order is represented by the length of the corresponding rectangle. The right end of each rectangle provides the release date of the corresponding order.

Four tours have been built for delivering the requested items to the customers. Fig. 1b provides a graph of the corresponding routes. From Fig. 1a it can be taken that each vehicle executes two tours. E.g., vehicle #1 visits customer #1 first and then proceeds to customer #2 on a first tour; on a second tour customers #9 and #3 are visited. The length of each rectangle represents the time which is needed by the vehicle for traveling from the warehouse or a previous customer to the respective customer location plus the service time for unloading the vehicle at the customer location. The rectangle at the beginning of each tour indicates the service time for loading the vehicle at the warehouse, while a rectangle labeled with 0 represents a trip of the vehicle from the last customer of the tour back to the warehouse.

Fig. 1a also demonstrates that a tour cannot be started before all corresponding orders have been finalized at the warehouse. E.g., loading of vehicle #1 for the first tour (1, 2) starts as soon as picking of customer orders #1 and #2 has been completed. After having visited customer #2, vehicle #1 returns to the warehouse where it remains idle until picking of the last order of its second tour has been finalized. Loading of vehicle #1 for the second tour (9, 3) commences when order #9 has been provided. The first tour (7, 8, 6) of vehicle #2 cannot be started before picking of the last order (order #8) has been finalized. While this tour is being carried out, picking of all orders of the second tour (4, 5) is completed. Thus, loading of vehicle #2 can immediately be started upon its return to the warehouse from the first tour.

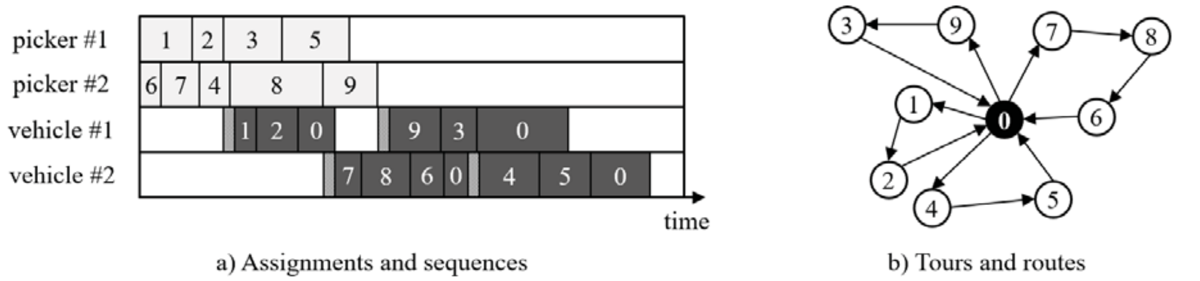

Fig. 1: Example solution

We note that the OASVRP is actually composed of two subproblems, namely an order assignment and sequencing problem (OASP), where customer orders have to be assigned to order pickers and sequences have to be determined in which the orders should be processed, and a vehicle routing problem (VRP), where the vehicles may perform multiple tours and release dates and due dates have to be considered. The interface between these two problems is provided by the release dates of the orders on the one hand and the start dates of the tours on the other hand. Before we introduce a solution approach to the OASVRP, we will review the literature related to the two subproblems and the OASVRP.

3 Literature review

3.1 Order assignment and sequencing problem

If we assume that customers are served individually and not on a tour together with other customers, then the time that it takes to transport the requested items from the depot to a customer is fixed and the latest release date of each customer order could be computed from the given due dates. This gives rise to the order assignment and sequencing problem which can be stated as follows: Let a set of customer orders with (latest) release dates and processing times, and a limited number of order pickers be given. How should the customer orders be assigned to the order pickers and in which sequence should the assigned customer orders be processed by each order picker such that the total tardiness of all orders is minimized?

To the best of our knowledge, the OASP has not been addressed in the context of order picking so far. However, the OASP is equivalent to the Identical Parallel Machine Problem (IPMP) from scheduling (Pinedo, 2016), where the customer orders are processed by machines instead of order pickers.

The IPMP is known to be NP-hard (Pinedo, 2016) and only a few exact solution approaches are available.

The first exact approach, a dynamic programming-based algorithm, was introduced by Gupta & Maykut (1973). Branch and bound algorithms were proposed by Azizoglu & Kirca (1998), Yalaoui & Chu (2002) and Shim & Kim (2007) which managed to solve problems with up to 15 customer orders and three machines, 20 customer orders and two machines, and 30 customer orders and five machines, respectively.

Due to the fact that exact algorithms can only deal with small-sized problems, heuristic approaches have been developed for solving problems of practical size. In particular, priority rule-based algorithms (Ho & Chang, 1991) have been introduced, where customer orders are sorted first according to a priority rule and then assigned one by one to the next available machine. The priority of a customer order can be dependent on its due date, on its processing time and/or on the ratio between the average processing time and the average due date. Alidaee & Rosa (1997) extended the modified-due-date (MDD) rule for the single machine problem proposed by Baker & Bertrand (1982). The MDD rule combines elements of the earliest-due-date (EDD) rule, where priority values of orders increase with a decreasing due date, and the shortest-processing-time rule, according to which higher priorities are assigned to orders with shorter processing times. Other approaches than priority-rule-based algorithms have been proposed by Koulamas (1997), who designed a decomposition approach as well as a hybrid simulated annealing algorithm. According to Ullrich (2013), the current state-of-the-art heuristic has been developed by Biskup et al. (2008). In this approach, customer orders are first sorted according to the EDD rule. Several incomplete initial solutions are then generated, where each initial solution is iteratively completed. In each iteration, exactly one non-assigned customer order is inserted into the partial solution. For each machine, one customer order is selected according to the MDD rule, resulting in a set of potentially assignable customer orders. One customer order from this set is then selected and optimally inserted into the partial solution.

3.2 Vehicle routing problems with multiple use of vehicles, release dates and due dates

The second subproblem of the OASVRP deals with the determination of routes for the vehicles which are used for delivering the requested items to the customers after finalized orders have been provided at the warehouse. It represents a variant of the classic vehicle routing problem where each customer order is characterized by a release date, i.e. a point in time when it becomes available for shipment at the depot, and a due date, i.e. the point in time by which it should have been received at the customer location. A set of homogeneous vehicles is available for transporting the requested items to the customers. Several customer locations may be visited on each tour and the vehicles may be used for multiple tours; however, the length of each tour is limited.

A VRP with multiple use of vehicles and a tour length constraint has been introduced by Fleischmann (1990). He extended the classic capacitated vehicle routing problem and designed a savings based heuristic. Taillard et al. (1996) and Brandão & Mercer (1997) proposed tabu search algorithms in order to solve the VRP with multiple use of vehicles. A constructive heuristic has been introduced by Petch & Salhi (2004), and Olivera & Viera (2007) suggested an extended tabu search approach to this problem.

For the VRP with hard time windows and multiple use of one vehicle, Azi et al. (2007) proposed an exact algorithm. Azi et al. (2010) extended this work to the case of multiple vehicles. They suggested a branch-and-price algorithm that is able to solve instances with up to 50 customers. More recently, Azi et al. (2014) designed an adaptive large neighborhood search algorithm for the heuristic solution of large-sized instances.

A VRP with release dates has been considered by Cattaruzza et al. (2016). They designed a genetic algorithm for the vehicle routing problem with hard time windows and release dates.

Due dates can be considered as a special case of time windows in which the lower bound is sufficiently small and the upper bound of the time window is a soft constraint. Taillard et al. (1997) suggested a tabu search algorithm for a VRP with a hard constraint regarding the lower bound and a soft constraint with respect to the upper bound of the time window. For the VRP with soft time windows, Chiang & Russell (2004) and Fu et al. (2008) designed tabu search algorithms. Liberatore et al. (2011) also considered the VRP with soft time windows and developed a branch-and-price algorithm.

3.3 Integrated scheduling and vehicle routing problems

As mentioned before, the OASP is equivalent to the IPMP. This scheduling problem has also been considered in conjunction with distribution problems which involve routing decisions. Table 1 provides an overview of publications related to such integrated scheduling and vehicle routing problems (ISVRP). (We refer to Chen (2010) for a very detailed review.) The second column of the table depicts the number of machines considered in the scheduling subproblem. The third, fourth and fifth column refer to the routing subproblem. The third column provides the number of available vehicles, while the entry "infinite" indicates that a sufficiently large number of vehicles has been assumed. The fourth column indicates whether each vehicle may only perform a single tour or whether it can be used for multiple tours, while the fifth column informs whether a limit on the length of each tour has been considered.

Table 1: Integrated scheduling and vehicle routing problems

reference	# machines	# vehicles	use of vehicles	tour length
Hurter & Van Buer (1996)	single	infinite	single	unlimited
Van Buer et al. (1999)	single	intinite	multiple	limited
Chen & Vairaktarakis (2005)	multiple	infinite	single	unlimited
Li et al. (2005)	single	single	multiple	unlimited
Low et al. (2013)	single	multiple	single	unlimited
Low et al. (2014)	single	infinite	single	unlimited
Li et al. (2016)	single	infinite	single	unlimited
Ullrich (2013)	multiple	multiple	multiple	unlimited
this paper (OASVRP)	multiple	multiple	multiple	limited

Hurter & Van Buer (1996) were the first who considered an ISVRP (Gao et al., 2015). They investigated a newspaper production and distribution problem. Different types of newspapers have to be delivered from a distribution center to drop-off points. All drop-off points have to be served by an identical deadline. The delivery of the newspapers is performed on tours by a fleet of homogeneous vehicles, but only one type of newspapers can be included in a single tour. Van Buer et al. (1999) extended this problem by allowing multiple tours per vehicle and multiple types of newspapers per tour. Chen & Vairaktarakis (2005) studied several variants of the ISVRP with an unrestricted number of homogeneous vehicles. Each vehicle is allowed to perform at most one tour and can visit a restricted number of locations per tour. In the production subproblem, both a single machine and multiple machines are assumed. Li et al. (2005) considered an ISVRP, where all customer orders are processed by a single machine. A single vehicle is available which may perform multiple tours. Low et al. (2013) investigated an ISVRP which integrates a scheduling problem with one machine and a VRP with hard time windows. This problem was extended by Low et al. (2014) who took a fleet of heterogeneous vehicles into account. Moreover, instead of hard time windows, soft time windows were assumed. The work of Li et al. (2016) combines a scheduling problem including a single machine with routing decisions including an unlimited number of homogeneous vehicles. In Ullrich (2013), multiple machines are used in the production process and a limited number of heterogeneous vehicles is used for the delivery of customer orders. For machines and vehicles, ready dates are given, i.e. the corresponding machine or vehicle must not necessarily be available at the beginning of the planning horizon.

As can be seen from Table 1, the problem considered by Ullrich (2013) resembles the OASVRP dealt with in this paper, except for the limitation of the tour lengths. The author proposed a genetic algorithm and investigated the benefits from an integrated solution of the scheduling and routing problems. However, the performance of the algorithm deteriorates drastically with an increasing number of customer orders. E.g., for instances with 70 orders, the quality of solutions provided by the genetic algorithm is hardly superior to the quality of solutions generated by a simple construction procedure (Ullrich, 2013, p. 163).

Apart from incorporating the tour length limitation, we will, therefore, pay particular attention to the development of an algorithm for the OASVRP which is capable of providing high-quality solutions to practical-sized problem instances in reasonable computing times. The algorithm, based on an iterated local search approach, will be presented in the next section.

4 Iterated local search approach

4.1 General principle

Iterated local search (ILS) has successfully been adapted to many kinds of optimization problems. It can be considered as the state-of-the-art algorithm for operations research problems, among others for various types of vehicle routing problems (Vidal et al., 2013), for single machine (Grosso et al., 2004; Congram et al., 2002) and identical parallel machine scheduling problems (Brucker et al., 1996, 1997) as well as for the order batching and sequencing problem (Henn & Schmid, 2013).

The general principle of ILS can be described as follows (Lourenço et al., 2010): Starting with an initial solution σ_{ini}, an improvement phase is executed in order to determine a local optimum, resulting in the first incumbent solution σ_{inc} and, at the same time, the best solution σ^* found so far. Perturbation and improvement phases are then alternately performed until a termination condition is met. In the perturbation phase, the incumbent solution σ_{inc} is randomly modified in order to avoid the ILS getting stuck in a local optimum. Based on the modified solution, a (new) local optimum is determined by means of the improvement phase. If the resulting solution represents a new best solution, then the best solution σ^* as well as the incumbent solution σ_{inc} are updated. Otherwise, σ^* remains unchanged and σ_{inc} is only altered if an acceptance condition is met. Depending on the acceptance condition, it may be possible to accept a solution with a worse objective function value than the incumbent solution. A pseudocode of the ILS approach is depicted below.

Algorithm 1 General principle of iterated local search

Input: problem data

Output: solution σ^* to the OASVRP and corresponding total tardiness $f(\sigma^*)$

```
generation of an initial solution σ_ini
σ_inc := improvement(σ_ini)
σ_best := σ_inc
while termination condition is not met do
    σ̃ := perturbation(σ_inc)
    σ* := improvement(σ̃)
    if f(σ*) < f(σ_best) then
        σ_best := σ*
    end if
    σ_inc := acceptance condition(σ*, σ_inc)
end while
```

4.2 Initial solution

For the generation of an initial solution, the OASVRP is divided into its two subproblems, which are then solved sequentially. First, a solution to the VRP is constructed. Tours and corresponding routes are generated by adapting the EDD rule originally designed for the IPMP (Baker & Bertrand, 1981). According to this rule, all customer orders are sorted in a non-descending order of the due dates. Then, in this sequence, the orders are assigned to the vehicle which currently possesses the shortest total travel time. More precisely, a customer order is assigned to the last position of the currently last route of the vehicle chosen. A new tour is opened each time the maximum tour length would be exceeded. In order to provide a feasible solution to the VRP, order release dates have to be taken into account. Regarding the OASVRP, the release date of an order is defined by the point in time when the order is finalized for shipment at the warehouse, which is not known at this stage of the algorithm. Therefore, release dates are estimated by assuming that the number of order pickers is identical to the number of vehicles and each order picker processes all orders assigned to a certain vehicle in the sequence provided by the above-described modification of the EDD rule. Based on the estimated release dates, the start date of each tour is determined. The estimated release dates are taken as (planned) start dates of the tours.

The solution of the VRP is then taken as input for the solution of the OASP which determines the release dates of the orders. Regarding the OASVRP, picking of an order has to be finalized before the start date of the tour in which it is included. In the context of the resulting OASP, the tardiness of an order is then defined as the non-negative difference between the (planned) start date of the corresponding tour and the release date of the order. As mentioned in Section 3.1, the OASP is equivalent to the IPMP. Thus, in order to solve this problem, the approach of Biskup et al. (2008) is applied.

If the approach of Biskup et al. (2008) leads to a solution with a total tardiness equal to 0, combining this solution with the solution to the VRP results in a feasible solution to the OASVRP as well. Otherwise, a tour including orders for which picking is completed after the (planned) start date of the tour has to be postponed and the start dates of the respective tour and the subsequent tours are corrected correspondingly. After having obtained a feasible solution, start dates may be updated in order to ensure that each tour starts as early as possible.

4.3 Improvement phase

As has been explained in Section 2, solving the OASVRP involves six different types of decisions which have to be taken simultaneously. Since a simple local search procedure will not be able to deal with all aspects of the problem, a more complex improvement procedure will be used. In fact, a variable neighborhood descent (VND) algorithm has been designed in order to tackle all decision types.

VND was first introduced in Hansen & Mladenovic (2001). In this approach, the solution space is explored using a sequence of neighborhood structures $\mathcal{N}_1, \ldots, \mathcal{N}_L$. Starting with a solution σ, a local optimum regarding the first neighborhood structure $\mathcal{N}_1$ is determined. If the resulting solution provides a better objective function value than the best solution found so far, this solution becomes the new best solution and $\mathcal{N}_1$ is explored again. Otherwise, the algorithm continues with exploring the next neighborhood structure. Each time a local optimum represents a new best solution, the algorithm continues with $\mathcal{N}_1$. The VND approach terminates when no improvement has been found in the last neighborhood structure $\mathcal{N}_L$. In this case, the best solution σ^* is a local optimum with respect to all neighborhood structures.

The improvement phase of the ILS approach presented in this paper consists of two VND algorithms, dealing with decisions related to the VRP (VND_VRP) and the OASP (VND_OASP), respectively. A pseudocode of the improvement phase is given in Algorithm 2.

Algorithm 2 Improvement phase

```
Input: problem data, solution σ with objective function value f(σ)
Output: local optimum σ*, f(σ*)
  do
      σ* := σ
      σ := VND_VRP (σ)
      σ := VND_OASP (σ)
  while f(σ) < f(σ*)
```

Each VND algorithm deals with one subproblem only. This approach is chosen because of the structure of the OASVRP. Decisions regarding the VRP only affect this subproblem. Moves related to these decisions can be performed quite easily. In contrast to that, any changes regarding the OASP have an impact on the release dates of the orders and, therefore, such moves will also affect the start dates of the tours. Since moves regarding the OASP affect both subproblems, their execution is much more time-consuming. As a consequence, a VND approach is first applied to the VRP and modifications to solutions of the OASP are not carried out before a local optimum regarding the VRP has been found.

In the VND algorithm for the VRP, four neighborhood structures $\mathcal{N}_1^{\mathrm{VRP}}, \ldots, \mathcal{N}_4^{\mathrm{VRP}}$ are contained:

$\mathcal{N}_1^{\mathrm{VRP}}$: a consecutive sequence of orders is moved to another position of the same tour;
$\mathcal{N}_2^{\mathrm{VRP}}$: two tours assigned to different vehicles are exchanged;
$\mathcal{N}_3^{\mathrm{VRP}}$: a consecutive sequence of orders is moved to a tour assigned to another vehicle;
$\mathcal{N}_4^{\mathrm{VRP}}$: a consecutive sequence of up to two orders is removed from a tour and it is assigned to the same vehicle building a new tour.

The impact of moves performed in the VND_VRP procedure is exemplified for the neighborhood structure $\mathcal{N}_4^{\mathrm{VRP}}$ (see Fig. 2). Regarding $\mathcal{N}_4^{\mathrm{VRP}}$, a neighbor solution is constructed by first choosing a tour assigned to a certain vehicle. A consecutive sequence of orders is then removed from the tour, i.e. the tour is divided into two subsets. The first subset contains all orders which are still included in the tour. No further changes will be performed to this tour. The second subset consisting of the removed orders will form a new tour and will be assigned to another position of the same vehicle. The position is chosen in such a way that the total tardiness is minimized.

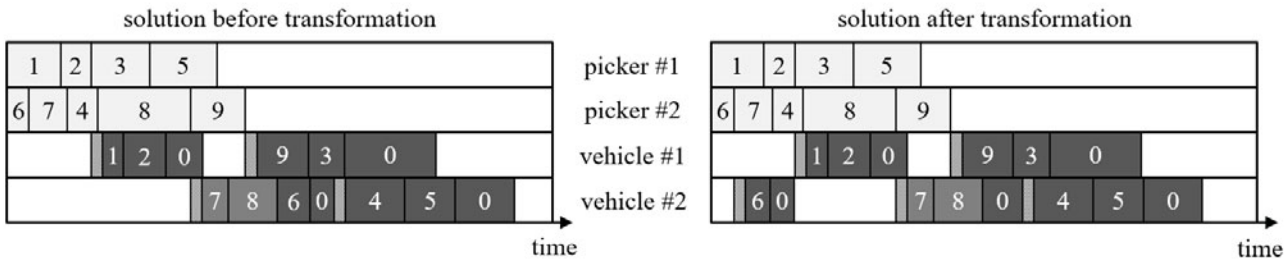

Fig. 2: Example of a move regarding neighborhood structure $\mathcal{N}_4^{\mathrm{VRP}}$

In the example depicted in Fig. 2, a possible move regarding $\mathcal{N}_4^{\mathrm{VRP}}$ is performed. The orders #7 and #8 are removed from the tour $(7, 8, 6)$ originally assigned to vehicle #2. The remaining tour only contains order #6 and can now start much earlier as order #6 is processed by an order picker at the very beginning of the planning horizon. The new tour $(7, 8)$ is inserted as the second one for vehicle #2 and can be started after order #8 has been provided. As can be seen from Fig. 2, the start date of this tour is identical to the start date of the tour $(7, 8, 6)$ in the solution before the transformation. Obviously, the length of the tour $(7, 8)$ is shorter than the length of the tour $(7, 8, 6)$. Therefore, the subsequent tour $(4, 5)$ can be started earlier now, resulting in earlier delivery dates for both orders.

The moves included in the VND_VRP procedure affect the tours assigned to one ($\mathcal{N}_1^{\mathrm{VRP}}$ and $\mathcal{N}_4^{\mathrm{VRP}}$) or two ($\mathcal{N}_2^{\mathrm{VRP}}$ and $\mathcal{N}_3^{\mathrm{VRP}}$) vehicles. Tours assigned to other vehicles will remain unchanged, but also decisions related to the OASP will not be affected. As mentioned before, moves changing the assignment of customer orders to order pickers or the sequence according to which customer orders are processed by an order picker are much more complex. Therefore, the VND_OASP procedure includes the following two neighborhood structures only:

$\mathcal{N}_1^{OASP}$: an order is moved to another position of the same picker;

$\mathcal{N}_2^{OASP}$: two orders assigned to different pickers are exchanged.

In Fig. 3, an example of a move regarding $\mathcal{N}_2^{OASP}$ is depicted, where the assignment to the order pickers is exchanged for orders #1 and #8. The release date of order #1 increases, resulting in a later start date of the corresponding tour. The start date of the following tour $(9, 3)$ is now determined by the completion date of the tour $(1, 2)$ instead of the release date of order #9. Consequently, the tour $(9, 3)$ is also postponed. Regarding vehicle #2, it can be seen that the start dates of the tours significantly decrease due to the exchange of orders #1 and #8 because the release date of order #8 decreases.

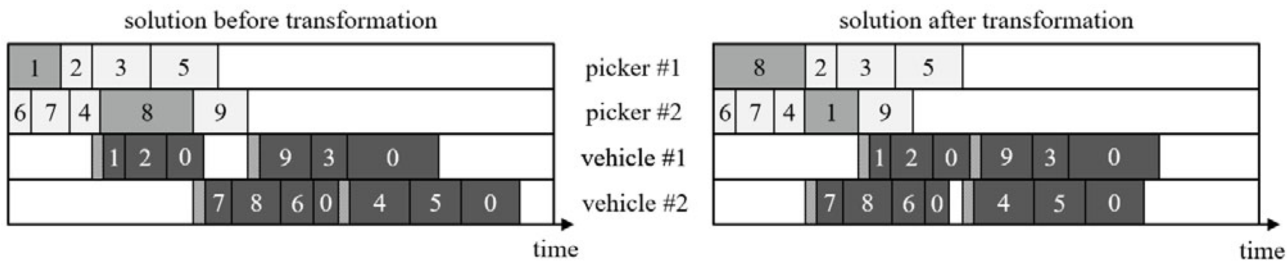

Fig. 3: Example of a move regarding neighborhood structure $\mathcal{N}_2^{\mathrm{OASP}}$

4.4 Perturbation phase

After a local optimum has been identified in the improvement phase, the solution is randomly modified in the perturbation phase. The design of the perturbation phase is pivotal for the performance of an ILS approach. If the modifications are too small, a further application of the improvement phase will result in the same local optimum. If too many changes are applied to the local optimum, the promising part of the solution space is left and the ILS algorithm turns into an improvement procedure with multiple random starts (Lourenço et al., 2010).

For the perturbation phase, we decided to use moves related to the OASP, as their impact on the solution is expected to be larger than that of VRP moves (see the previous subsection). A move in the perturbation phase is defined by the exchange of two sequences of consecutive customer orders which are assigned to different order pickers (see Fig. 4). The lengths of the two sequences are chosen randomly and may be different from each other. The maximum length of a sequence determines the degree of modification performed in the perturbation phase.

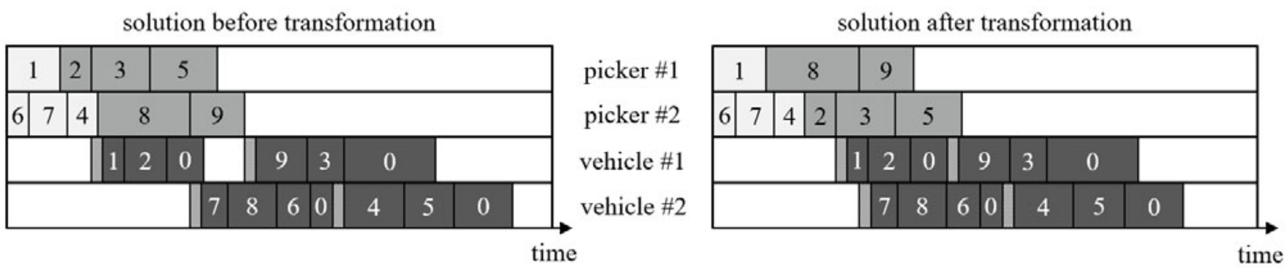

Fig. 4: Example of a move in the perturbation phase

In Fig. 4, three consecutive orders assigned to picker #1 are exchanged with two orders processed by picker #2. Consequently, the release dates change for all of the five orders. This affects the start date of all tours. Regarding vehicle #1, the first tour is postponed because of the increased release date of order #2. Nevertheless, the subsequent tour $(9, 3)$ is started earlier now. For vehicle #2, the transformation leads to earlier start dates for both tours. The start date of tour $(7, 8, 6)$ decreases due to the smaller release date of order #8, and the subsequent tour can then be started earlier as well since its start date is determined by the completion date of tour $(7, 8, 6)$.

4.5 Acceptance and termination condition

In addition to the perturbation phase, an adequate acceptance condition helps to overcome local optima. A solution of inferior quality may be accepted if no improvements have been found for a certain number $\tilde{n}$ of consecutive iterations and if the solution provides an objective function value not too far from the objective function of the current best solution. The acceptance condition of our ILS was proposed by Dueck & Scheuer (1990). According to this condition, a solution σ is accepted if its objective function value $f(\sigma)$ is not larger than $(1 + \alpha) \cdot f(\sigma^*)$, where $f(\sigma^*)$ denotes the objective function value of the current best solution σ^* and α is a parameter indicating the relative amount of deterioration allowed. At the beginning of the algorithm, α is initialized by 0, i.e. a solution is only accepted if it represents an improvement compared to the current best solution. Each time no solution is accepted for $\tilde{n}$ consecutive iterations, α is increased. Whenever the incumbent solution is updated, α is set to 0 again. Thus, the longer the ILS algorithm gets stuck in a local optimum the higher the relative amount of deterioration allowed gets. This type of acceptance condition has also been applied by Polacek et al. (2004) to a VRP with time windows and multiple depots, by Tarantilis et al. (2004) to a VRP with a heterogeneous fleet and by Henke et al. (2015) to a VRP with multiple compartments.

A time limit has been chosen as the termination condition. After each iteration, it is checked whether the time limit has been exceeded. If this is the case, the ILS approach is terminated. Otherwise, at least one additional iteration is performed.

5 Numerical experiments

5.1 Test problem instances and parameter settings

In order to evaluate the performance of the ILS algorithm as well as to determine the benefits from solving the OASVRP as a holistic problem instead of dealing with the OASP and the VRP sequentially, extensive numerical experiments have been conducted. Since the problem instances of Ullrich (2013) were not available, new test problem instances have been generated. The generation of the instances followed the procedures of Ullrich (2013) for the VRP and of Scholz et al. (2016) for the OASP.

For the numerical experiments, problem instances with 100 and 200 customer orders have been generated. Instances of this size have also been used by Henn (2015) and Scholz et al. (2016) for different types of order picking problems and they are also considered as realistic problem sizes for VRPs (Desaulniers et al., 2014). The customer orders are processed by 2, 3 or 5 order pickers in the warehouse. For the determination of the processing times of the customer orders, a block layout with 10 picking aisles is assumed for the picking area. A class-based procedure has been assumed for assigning the articles to the storage locations (Henn, 2015; Scholz et al., 2016). The routes of the order pickers are constructed by means of the S-shape strategy which represents the routing strategy most frequently used in practice (Roodbergen, 2001). Processing times of orders will increase with an increasing number of blocks. Instances with 1 block (short processing times) and 3 blocks (long processing times) are considered.

Identical to Ullrich (2013), the number of vehicles available for the delivery of the customer orders is set either to 4, 6, 8 or 10. The respective customer locations are chosen randomly, while the corresponding coordinates for the horizontal and vertical dimensions are selected from the interval $[1, 100]$ for instances including 100 customer orders and from $[1, 150]$ for problems with 200 orders. The location of the warehouse is fixed to the coordinates $(50, 50)$ and $(75, 75)$, respectively. The travel times are then defined by the euclidean distances between the locations (Ullrich, 2013). The time for loading the vehicle (service time at the depot) is set to 20 minutes, while 5 minutes are required for unloading the required items (service time at a customer location). The maximum tour length is set to 8 hours.

Finally, a due date is assigned to each customer order. The due dates are determined based on the procedure of Ullrich (2013). According to this procedure, the due dates are dependent on the number of customer orders N, the number of vehicles K, the number of order pickers M as well as on the processing times p_n $(n = 1, \ldots, N)$ of the orders and the travel and service times. Additionally, a parameter θ is introduced describing how difficult it is to meet the due dates. The due date of customer order n is then a realization of the random variable D_n which is defined as follows (Ullrich, 2013):

$$D_n = p_n + t_{0n} + s_0 + s_n + \Gamma + \Delta \tag{1}$$

On the one hand, D_n includes order-specific data such as the travel time t_{0n} between the depot and the location of customer n and the service time (s_n) at the customer location. Due to the integration of the service time at the depot (s_0) as well as the random variables Γ and Δ, general problem data is included in the calculation on the other hand. Γ and Δ are uniformly distributed over the discrete sets $\{0, \ldots, \lfloor\theta\,(\max_{n=1,\ldots,N} p_n)\,(N/\,(K + M))\rfloor\}$ and $\{0, \ldots, \lfloor\theta\,(\max_{n=1,\ldots,N} p_n)\rfloor\}$, respectively. In the numerical experiments, θ is set to 0.5 (tight due dates) and to 1.0 (loose due dates).

The combination of all parameter values gives rise to 96 different problem classes. For each class, 48 test problem instances have been generated, resulting in 4608 instances in total. The ILS algorithm has been implemented using Visual Studio C++ 2015. The numerical experiments have been performed by means of a Haswell system with up to 3.2 GHz and 16 GB RAM per core.

Regarding the ILS approach, the following settings have been chosen. The maximum length of a sequence of consecutive orders exchanged is set to 5 for the perturbation phase. The parameter α included in the acceptance criterion is increased by 0.1 after 50 consecutive iterations each without finding a new best solution. The time limit for the ILS has been fixed to 30 minutes for instances with 100 customer orders and to 60 minutes for problems with 200 orders.

5.2 Generation of upper bounds

As has been shown by Ullrich (2013), only very small problem instances can be solved to optimality within a reasonable amount of computing time. Therefore, upper bounds are generated in order to evaluate the performance of the ILS approach. In fact, three procedures for the determination of upper bounds are applied.

As for the first procedure (Ullrich, 2013), each customer order is assumed to be served on a separate tour. This assumption reduces the OASVRP to a hybrid flow shop problem with M parallel machines at the first stage and K parallel machines at the second stage. Processing times at the first stage are given by the processing times p_n ($n = 1, \ldots, n$) of the orders, while the times for delivering the customer orders (given by $s_0 + t_{0n} + t_{n0} + s_n$) represent the processing times at the second stage (Ullrich, 2013). This problem is then solved by applying the MDD rule which has been proven to perform quite well for multiple stage hybrid flow shop problems with due dates (Brah, 1996). Ullrich (2013) compared the solutions generated by a genetic algorithm to this upper bound and pointed out that the genetic algorithm was not able to find solutions of superior quality for problem instances with 70 or more customer orders. This observation indicates that the genetic algorithm does not perform well for those problems. Therefore, in our experiments, this upper bound (UB_1) is used in order to identify whether the ILS approach is suitable for dealing even with very large instances.

The general principle for the generation of the second upper bound (UB_2) was also proposed by Ullrich (2013). He suggested to divide the problem into its two subproblems and then solve them one after another to optimality. The author started with the VRP, continued with the OASP and got back again to the VRP. The procedure then terminates since performing further iterations have proven not to lead to significant improvements regarding the solution quality. Ullrich (2013) computed upper bounds of this type for very small problem instances including 7 customer orders only. Therefore, in order to be able to calculate the bounds for larger instances, we use the same principle but the subproblems are solved heuristically. At first, the procedure for the determination of an initial solution (see Section 4.2) is used, i.e. the VRP is solved and then the algorithm of Biskup et al. (2008) is applied to the OASP. As suggested by Ullrich (2013), the VRP is then solved again. Here, the VND_VRP procedure (see Section 4.3) is applied. This procedure for the generation of an upper bound is much more complex than the previous one, as a state-of-the-art algorithm is used for solving the OASP and a VND approach for solving the VRP. Upper bounds of this second type are generated in order to determine the benefits from dealing with the OASVRP as a holistic problem instead of solving the subproblems in sequence.

The determination of the third upper bound (UB_3) also originates from the ILS approach. The initial solution is constructed and one improvement phase of the ILS algorithm is performed. The quality of this bound is at least as good as the quality of the second bound. The third bound is used for the investigation of the impact of the perturbation phase on the quality of solutions provided by the ILS approach.

5.3 Evaluation of the solution quality of the ILS algorithm

In Tables 2 and 3, the average total tardiness (tard_i) in minutes is depicted for the upper bounds UB_i ($i \in \{1, 2, 3\}$) as well as for the solutions provided by the ILS approach (tard_{ILS}) for problem classes with 100 and 200 customer orders, respectively. Furthermore, the average improvements (imp_i) [in %] are presented in comparison to upper bound UB_i. In the tables, K denotes the number of vehicles, B represents the number of blocks and θ is the parameter used for the generation of the due dates (see Section 5.1).

Performance of the ILS algorithm for large-sized instances

Comparing the objective function values of solutions obtained by the ILS approach to the upper bound UB_1, significant improvements regarding the total tardiness can be observed. On average, the reduction ranges from 4.6% (100 orders, 2 pickers, $\theta = 0.5$, $B = 1$, $K = 10$) to 94.0% (200 orders, 5 pickers, $\theta = 1$, $B = 3$, $K = 4$). The magnitude of the improvement varies very strongly between different problem classes. This can be explained by the performance of the approach for generating UB_1. Application of the MDD rule leads to rather good solutions to the OASP. The VRP is solved on the basis of the assumption that each customer is served on a separate tour. This assumption is not critical as long as processing the orders in the warehouse consumes more time than the separate delivery of each order, i.e. when many more vehicles than order pickers are available or when the processing times of the orders are large in comparison to their travel times. Furthermore, the upper bound may have a good quality in case of loose due dates.

Due to the increasing ratio between vehicles and order pickers, the amount of improvement decreases with an increasing number of vehicles (75.4% for $K = 4$ and 32.6% for $K = 10$) and increases with an increasing number of order pickers (30.6% for 2 pickers and 75.6% for 5 pickers). As has been anticipated, these two parameters have the largest impact on the amount of improvement. If few vehicles are available for the delivery, the solutions to the VRP can significantly be improved by serving several customer orders on the same tour when the orders can be processed by many order pickers. Besides the number of pickers and the number of vehicles, the number of blocks and the parameter θ affect the amount of improvement. An increasing number of blocks results in a reduction of the average improvement. While the total tardiness can be reduced by 56.7% in a single-block layout, the improvement amounts to 47.5% when the picking area is composed of three blocks. The reason can be found in the processing times which increase when the picking area includes a larger number of blocks. Picking the orders gets more time-consuming and the advantage of serving several customers on a single vehicle tour diminishes. The amount of improvement also decreases with an increasing value of θ, as the bound can be improved by 55.5% for $\theta = 0.5$ and by 47.7% for $\theta = 1$. Problem classes characterized by a large θ contain instances with loose due dates. In this case, delivering more than one order per tour is not that important. Thus, the quality of the bound increases and the amount of improvement obtained by application of the ILS approach decreases. Furthermore, it can be observed that larger reductions of the total tardiness are achieved for instances with a larger number of orders (43.8% for 100 orders and 60.4% for 200 orders).

Table 1: Evaluation of the ILS approach for problem classes with 100 customer orders

			2 pickers							3 pickers							5 pickers						
θ	B	K	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$
1	1	4	1111	61.9	593	28.7	549	23.0	423	1139	79.8	528	56.5	449	48.8	230	1196	91.2	576	81.8	495	78.8	105
1	1	6	707	30.6	605	19.0	572	14.3	490	691	60.9	381	29.2	343	21.2	270	724	84.5	317	64.5	259	56.5	112
1	1	8	565	6.9	643	18.1	605	13.0	526	488	39.2	390	23.9	359	17.4	297	500	74.1	262	50.5	216	40.0	130
1	1	10	576	5.4	643	15.2	609	10.5	545	397	21.7	395	21.4	362	14.2	311	373	63.4	231	40.8	195	29.9	136
1	3	4	901	47.0	647	26.1	616	22.4	478	938	75.5	482	52.3	416	44.7	230	1029	93.3	564	87.8	477	85.5	69
1	3	6	657	10.9	729	19.7	694	15.6	586	551	44.6	430	29.1	400	23.6	305	597	81.5	294	62.4	239	53.7	111
1	3	8	714	10.7	773	17.5	736	13.4	638	407	15.2	458	24.5	428	19.3	345	391	65.7	271	50.5	227	40.9	134
1	3	10	774	11.7	796	14.1	767	11.0	683	423	9.9	492	22.5	459	16.9	381	286	48.9	259	43.6	222	34.3	146
0.5	1	4	1276	54.3	772	24.5	722	19.3	583	1261	71.4	682	47.1	602	40.0	361	1238	84.1	675	70.8	598	67.0	197
0.5	1	6	823	25.2	745	17.3	709	13.2	616	793	51.9	503	24.2	464	17.9	381	784	75.0	405	51.5	345	43.0	196
0.5	1	8	665	6.6	742	16.3	704	11.7	622	584	34.1	485	20.6	451	14.7	385	555	63.6	340	40.7	291	30.5	202
0.5	1	10	660	4.6	726	13.2	694	9.3	630	471	16.8	483	18.9	451	13.0	392	415	51.8	298	33.1	258	22.6	200
0.5	3	4	1193	38.5	942	22.0	911	19.4	734	1177	62.8	737	40.6	665	34.2	438	1193	81.2	752	70.2	667	66.3	224
0.5	3	6	879	10.5	961	18.2	923	14.8	786	741	36.0	623	23.8	590	19.6	474	735	67.7	448	46.9	386	38.4	238
0.5	3	8	901	9.7	957	15.0	921	11.7	813	572	12.6	628	20.3	593	15.7	500	521	51.5	405	37.6	360	29.8	253
0.5	3	10	920	10.7	943	12.9	908	9.6	821	556	9.1	620	18.4	586	13.8	506	400	35.7	379	32.2	339	24.1	257
average			833	21.6	763	18.6	728	14.5	623	699	40.1	520	29.6	476	23.4	363	684	69.6	405	51.4	348	46.3	169

Table 2: Evaluation of the ILSA for problem classes with 200 customer orders

			2 pickers							3 pickers							5 pickers						
θ	B	K	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$	$tard_1$	imp_1	$tard_2$	imp_2	$tard_3$	imp_3	$tard_{ILSA}$
1	1	4	7742	76.5	4388	58.5	4222	56.9	1820	7853	86.8	4497	77.0	4351	76.2	1034	8035	92.6	4670	87.3	4511	86.9	591
1	1	6	4938	58.5	2715	24.5	2543	19.4	2050	5013	76.6	2643	55.6	2386	50.8	1173	5169	88.6	2663	77.9	2383	75.3	589
1	1	8	3701	39.6	2493	10.4	2422	7.7	2234	3667	64.8	1980	34.7	1819	29.0	1293	3637	82.9	1941	67.9	1770	64.8	623
1	1	10	2929	21.0	2578	10.2	2482	6.7	2315	2846	51.4	1668	17.1	1577	12.4	1382	2858	76.7	1498	55.6	1333	51.1	665
1	3	4	6772	69.8	3391	39.9	3026	32.6	2039	7070	85.5	3529	70.9	3049	66.3	1027	7394	94.0	3909	88.7	3406	87.1	440
1	3	6	4336	42.3	2920	14.3	2847	12.1	2503	4400	70.2	2349	44.2	2114	38.0	1311	4590	88.4	2528	78.9	2286	76.7	533
1	3	8	3243	16.7	3037	11.0	2950	8.4	2702	3150	50.6	1867	16.7	1797	13.5	1555	3227	80.3	1689	62.4	1489	57.4	635
1	3	10	3273	12.0	3166	9.0	3072	6.2	2880	2458	31.6	1935	13.2	1855	9.4	1680	2478	71.0	1290	44.3	1135	36.7	718
0.5	1	4	8242	70.6	4950	51.0	4772	49.2	2425	8294	81.4	4948	68.8	4785	67.8	1541	8284	88.6	4967	80.9	4798	80.3	947
0.5	1	6	5381	52.6	3247	21.4	3070	16.9	2552	5341	70.0	3056	47.6	2803	42.9	1600	5342	83.0	2991	69.7	2697	66.2	907
0.5	1	8	3990	34.5	2889	9.5	2816	7.1	2615	3910	57.8	2318	28.8	2151	23.3	1651	3827	76.3	2203	58.9	2014	55.0	906
0.5	1	10	3236	18.1	2924	9.3	2830	6.3	2651	3072	44.9	1965	13.9	1876	9.9	1691	2979	69.3	1735	47.2	1565	41.5	915
0.5	3	4	7940	61.4	4557	32.7	4173	26.5	3066	8068	76.3	4554	58.0	4100	53.4	1912	8200	86.6	4773	77.0	4272	74.3	1098
0.5	3	6	5168	35.3	3828	12.7	3750	10.9	3343	5140	60.6	3167	36.1	2935	31.0	2024	5239	79.5	3171	66.2	2923	63.2	1072
0.5	3	8	3950	13.8	3778	9.9	3687	7.7	3404	3774	43.6	2512	15.2	2433	12.5	2130	3756	70.3	2239	50.2	2017	44.8	1114
0.5	3	10	3911	10.9	3819	8.7	3718	6.2	3485	2984	26.6	2503	12.5	2418	9.4	2191	2897	60.0	1756	34.0	1600	27.6	1158
average			4922	39.6	3417	20.8	3274	17.6	2630	4815	61.2	2843	38.2	2653	34.1	1575	4869	80.5	2751	65.5	2513	61.8	807

This can be explained by the fact that no advantage of the larger solution space is taken by the procedure for the construction of the upper bound, whereas the number of moves performed in the improvement phase of the ILS algorithm significantly increases with an increasing number of customer orders.

On average, across all problem classes, the first upper bound can be improved by 52.1%, which demonstrates that significant reductions of the total tardiness are obtained. In contrast to the genetic algorithm of Ullrich (2013), which is not able to significantly improve the upper bound for instances including 70 or more customer orders, the proposed ILS algorithm results in serious improvements even for very large instances. Furthermore, the impact of the parameters on the amount of improvement matches the expectations based on the quality of the upper bound, which leads us to the conclusion that the ILS algorithm is well designed.

Benefits of a holistic solution of the OASVRP

While the generation of the first upper bound makes use of two simple construction procedures, the second bound is provided by application of a state-of-the-art algorithm to the OASP and a VND approach to the VRP. Nevertheless, compared to the second upper bound, the ILS algorithm results in remarkable improvements, which vary between 8.7% (200 orders, 2 pickers, $\theta = 0.5$, $B = 3$, $K = 10$) and 88.7% (200 orders, 5 pickers, $\theta = 1$, $B = 3$, $K = 4$). Over all problem classes, the total tardiness can be reduced by 37.8% on average, which clearly demonstrates that solving the OASVRP as a holistic problem is pivotal for obtaining high-quality solutions.

The results from the experiments indicate that a simultaneous solution of the OASP and the VRP is more advantageous if the number of vehicles is not too large in comparison to the number of order pickers. If few vehicles are available for the delivery of the orders, more orders will be contained in a single tour. The start dates of the tours are then dependent on the release dates of several orders, i.e. the solution of the VRP is strongly affected by the solution of the OASP and the other way round. If the number of vehicles is very large, the tours include few or even a single order only. In this case, the VRP gets less important and the OASP can be solved without taking the vehicle tours into account. A similar argumentation holds for the impact of the number of blocks on the size of the improvement. Increasing processing times caused by a larger number of blocks produces the same effect as a decreasing number of order pickers does since fewer orders can be processed within the same amount of time. Thus, fewer customers will be visited on a tour, decreasing the benefits from solving the subproblems simultaneously. Regarding the number of customer orders, the results show that – what concerns the joint approach – a larger number of orders provides more space for improvement, increasing the reduction of the total tardiness by 8.3 percentage points (33.2% reduction for 100 orders and 41.5% reduction for 200 orders).

Impact of the perturbation phase on the solution quality

The third upper bound is obtained by application of the improvement phase to the solution provided by the procedure for the generation of the second bound. The improvements obtained by the ILS approach range between 6.2% (200 orders, 2 pickers, $\theta \in \{0.5, 1\}$, $B = 3$, $K = 10$) and 87.1% (200 orders, 5 pickers,

$\theta = 1$, $B = 3$, $K = 4$). On average, the improvement amounts to 32.5%, clearly demonstrating that more than a simple improvement procedure is required for providing high-quality solutions to such a complex problem. The application of the perturbation phase is pivotal in order to overcome local optima and to guide the search to the promising part of the solution space.

The amount of improvement is mainly dependent on the number of order pickers and the number of vehicles. Regarding the number of order pickers, it can be observed that the reduction of the total tardiness, given by UB_3, gets much larger with an increasing number of pickers. In fact, the average amount of reduction equals to 16.1% in the case of 2 pickers, while the total tardiness can be reduced by 53.1% for problems including 5 pickers. This can be explained by the fact that the perturbation phase exchanges randomly-chosen orders between two pickers. If 2 pickers are available only, the selection of the order pickers is fixed. Thus, the probability that the perturbation phase leads to solutions already investigated earlier is significantly increased in this case. Concerning the number of vehicles, the same behavior can be observed as for the comparison with UB_2: If the number of vehicles gets very large in comparison to the number of order pickers, many orders can be delivered on a separate tour. Thus, the benefit from solving the OASP and the VRP simultaneously diminishes, which also reduces the range in which improvements can be obtained. The impact of the processing times and the parameter θ is much less significant than the impact of the number of pickers and the number of vehicles. The amount of improvement obtained by application of the ILS approach increases with decreasing processing times, i.e. with a decreasing number of blocks, and an increasing value of θ. Furthermore, larger improvements are obtained for instances with a larger number of customer orders. While the total tardiness can be reduced by 28.1% when 100 customers are considered, an average reduction of 37.8% is obtained for instances with 200 orders. However, these results have to be taken carefully since much more computing time is spent on solving instances including 200 orders by application of the ILS approach.

5.4 Considerations regarding computing times

The generation of the upper bounds requires a few seconds of computing time, whereas the computing time of the ILS approach has been fixed to one hour for problem instances with 200 customer orders. The improvements obtained in comparison to the bounds provides information on the benefits of applying the ILS algorithm instead of using simple construction procedures or sequential solution approaches. However, no reliable conclusions on the performance of the proposed ILS algorithm can be drawn from the results. In particular, it is not known whether the time limit has appropriately been chosen. In Fig. 5, information on the development of the average solution quality over time is given for instances from three problem classes. Problem classes with 200 customer orders are considered, implying that the ILS approach is terminated after one hour of computing time. For each point in time, Fig. 5 depicts the relative deviation [in %] of the total tardiness provided by the best solution found after one hour from the tardiness of the current best solution.

The first problem class (2 pickers, $\theta = 0.5$, $B = 3$, $K = 10$), which has been considered, is characterized by very low improvements (6.2%) with respect to UB_3. After 10% of the total computing time, the tardiness provided by the current best solution can only be improved by 2% on average within the remaining 90% of the computing time. The reason can be found in the design of the perturbation phase. As mentioned before, fewer decisions have to be taken in the perturbation phase in the case of 2 pickers, which significantly reduces the number of possible moves. For the second problem class (3 pickers, $\theta = 1$, $B = 3$, $K = 6$), UB_3 can be improved by 38.0%, which can be interpreted as a fairly average amount of improvement. In this case, the time limit of one hour seems to be chosen appropriately. The total tardiness is reduced by 10% in the last 90% of the computing time, representing a significant amount of improvement. Thus, the algorithm should not be stopped at an early stage. Furthermore, the algorithm seems to converge as the reduction found amounts to 1.1% for the last 50% and 0.1% for the last 10% of the total computing time. The largest improvements are obtained for the problem class (5 pickers, $\theta = 1$, $B = 3$, $K = 4$), where UB_3 is reduced by 87.1% on average. The development of the solution quality over time clearly indicates that a larger amount of computing time is required for generating high-quality solutions. Even the tardiness obtained after 50% of the total computing time can be reduced by 11.1%. In the last 10% of the time, the objective function value can still be improved by almost 1%.

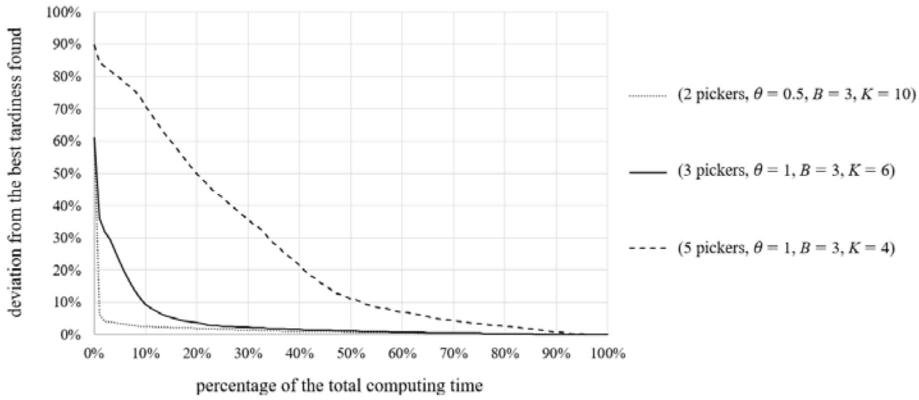

Fig. 5: Development of the solution quality over one hour of computing time

As a preliminary conclusion, it can be stated that the time limit of 1 hour is more than sufficient for solving instances from the class with 2 order pickers. For the classes considered above, which include 3 or 5 pickers, it is further investigated whether the computing time has been chosen sufficiently or not. Therefore, for these two problem classes, the development of the solution quality over four hours of computing time is depicted in Fig. 6.

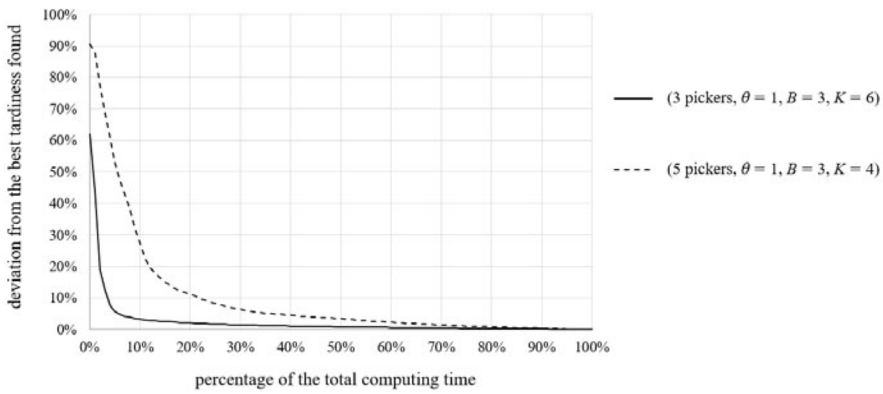

Fig. 6: Development of the solution quality over four hours of computing time

If three additional hours are spent on solving an instance from the class with 3 pickers, the total tardiness can be reduced by 1.6% on average. Thus, we conclude that the time limit of one hour is appropriate for those instances as the solution quality does not improve significantly. This is not true for problem instances from the class containing 5 order pickers. On average, the objective function value of the best solution found after one hour of computing time can be improved by 8.2%, which shows that the total tardiness can significantly be reduced by spending a larger amount of computing time. However, in the last hour, the reduction obtained amounts to less than 1.0%, indicating that four hours of computing time are sufficient in order to tackle problems with 5 order pickers. Summing up, it can be pointed out that the amount of computing time required for obtaining solutions of good quality increases with an increasing number of order pickers, which could be expected as the problem gets more complex when more pickers are available.

6 Conclusions and outlook

In this paper, we investigated the order assignment and sequencing, and vehicle routing problem (OASVRP), which is particularly pivotal for an efficient organization of the distribution processes in the retail industry. In the considered scenario, the orders are first processed in the warehouse by retrieving the respective requested items from their storage locations. After having completed the picking operations, vehicles will perform tours in order to deliver the requested items to the customers.

Order picking and vehicle routing operations are closely interconnected since a vehicle tour cannot start before all requested items of the orders included in the tour have been provided by the warehouse. Nevertheless, the integrated solution of these two subproblems has not been addressed in the literature so far. For solving the OASVRP, an algorithm of Ullrich (2013) could be adapted to this problem. However, the computational performance of this approach is limited. In order to introduce a more competitive approach, in particular for large problem instances, an iterated local search (ILS) algorithm for the OASRP has been proposed in this paper. Due to the complexity and the characteristics of the OASVRP, the improvement procedure includes two alternating variable neighborhood descent algorithms which tackle one subproblem each. By means of the ILS approach, the benefits from dealing with the OASVRP as a holistic problem could be investigated even for problem instances of a size encountered in practice.

Extensive numerical experiments have been conducted. In the first part of the experiments, it is demonstrated that the proposed ILS approach is suitable for solving large-sized instances. The second part of the experiments is devoted to the investigation of the benefit from integrating order picking and vehicle routing operations. It has been shown that the solution of the OASVRP as a holistic problem reduces the total tardiness by up to 88.7%, while the average reduction over all problem classes amounts to 37.8% compared to a sequential solution of the respective subproblems. Furthermore, problem characteristics have been pointed out under which a separate solution of the subproblems leads to acceptable results, and problem classes have been identified, where the consideration of the OASVRP as a holistic problem is inevitable in order to provide high-quality solutions.

Further research could concentrate on an extension of the problem regarding the picking operations. In this paper, the processing times of the orders can considered as given due to the assumption that customer orders have to be processed separately. However, the picking device may enable the order pickers to temporarily store a larger number of items, which allows for processing several customer orders on the same tour. In this case, it has to be decided which customer orders are included in a picker tour. Furthermore, the routes could not be determined in advance anymore, which makes the resulting problem even more complex. Nevertheless, integrating this aspect would clearly represent a worthwhile endeavor because the processing times in the warehouse would significantly decrease if customer orders do not have to be processed separately.

Regarding the vehicle routing subproblem, a straightforward extension represents the introduction of time windows for the delivery of the customers. In particular, early deliveries may cause problems when no room is available for temporarily storing the items. Such problems would be avoided by the integration of an earliest possible delivery date. A further interesting extension can be found in the consideration of vehicles with multiple compartments. Supermarkets receive several kinds of food, which have to be transported under different cooling conditions. These products can be transported on the same tour if the vehicle can be divided into different compartments, where each compartment represents a certain temperature zone (Hübner & Ostermeier, 2016).

References

Alidaee, B. & Rosa, D. (1997): Scheduling parallel machines to minimize total weighted and unweighted tardiness. *Computers & Operations Research* 24, 775–788.

Azi, N; Gendreau, M. & Potvin, J.-Y. (2007): An exact algorithm for a single-vehicle routing problem with time windows and multiple routes. *European Journal of Operational Research* 78, 755–766.

Azi, N; Gendreau, M. & Potvin, J.-Y. (2010): An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *European Journal of Operational Research* 202, 756–763.

Azi, N; Gendreau, M. & Potvin, J.-Y. (2014): An adaptive large neighborhood search for a vehicle routing problem with multiple routes. *Computers & Operations Research* 41, 167–173.

Azizoglu, M. & Kirca, O. (1998): Tardiness minimization on parallel machines. *International Journal of Production Economics* 55, 163–168.

Baker, K.R. & Bertrand, J.W.M. (1981): An investigation of due-date assignment rules with constrained tightness. *Journal of Operations Management* 1, 109–120.

Baker, K.R. & Bertrand, J.W.M. (1982): A dynamic priority rule for scheduling against due-dates. *Journal of Operations Management* 3, 37–42.

Biskup, D.; Herrmann, J. & Gupta, J.N.D. (2008): Scheduling identical parallel machines to minimize total tardiness. *International Journal of Production Economics* 115, 134–142.

Brah, S.A. (1996): A comparative analysis of due date based job sequencing rules in a flow shop with multiple processors. *Production Planning & Controls* 7, 362–373.

Brandão, J. & Mercer, A. (1997): A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. *European Journal of Operational Research* 100, 180–191.

Brucker, P.; Hurink, J. & Werner, F. (1996): Improving local search heuristics for some scheduling problems. Part I. *Discrete Applied Mathematics* 65, 97–122.

Brucker, P.; Hurink, J. & Werner, F. (1997): Improving local search heuristics for some scheduling problems. Part II. *Discrete Applied Mathematics* 72, 47–69.

Cattaruzza, D.; Absi, N. & Feillet, D. (2016): The multi-trip vehicle routing problem with time windows and release dates. *Transportation Science* 50, 676–693.

Chen, Z.-L. & Vairaktarakis, G.L. (2005): Integrated scheduling of production and distribution operations. *Management Science* 51, 614–628.

Chen, Z.-L. (2010): Integrated production and outbound distribution scheduling: Review and extension. *Operations Research* 58, 130–148.

Chiang, W.-C. & Russell, R.A. (2004): A Metaheuristic for the Vehicle-Routeing Problem with Soft Time Windows. *The Journal of the Operational Research Society* 55, 1298–1310.

Congram, R.K.; Potts, C.N. & van de Velde, S.L. (2002): An iterated dynasearch algorithm for the single-machine total weighted tardiness scheduling problem. *INFORMS Journal on Computing* 14, 52–67.

Desaulniers, G.; Madsen, O.B.G. & Ropke, S. (2014):: The vehicle routing problem with time windows. *Vehicle routing: Problems, methods and applications* 2nd edition, Toth, P. & Vigo, D. (eds). Philadelphia: Society for Industrial and Applied Mathematics and the Mathematical Optimization Society, 119–160.

Dueck, G. & Scheuer, T. (1990): Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics* 1, 161–175.

DVZ – Deutsche Verkehrszeitung (2013): Wir liefern künftig im 24-Stunden-Rhythmus. URL: http://www.dvz.de/rubriken/logistik-verlader/single-view/nachricht/24-stunden-belieferung-ueber-alle-sortimente.html.

EDEKA Minden-Hannover: Zahlen, Daten & Fakten. URL: http://www.edeka-verbund.de/Unternehmen/de/edeka_minden_hannover/unternehmen_minden_hannover/zahlen_daten_fakten/zahlen_daten_fakten_minden_hannover.jsp

Fleischmann, B. (1990): The vehicle routing problem with multiple use of vehicles. Working paper, Fachbereich Wirtschaftswissenschaften, Universität Hamburg.

Fu, Z.; Eglese, R. & Li, L.Y.O. (2008): A unified tabu search algorithm for vehicle routing problems with soft time windows. *Journal of the Operational Research Society* 59, 663–673.

Gao, S.; Qi, L. & Lei, L. (2015): Integrated batch production and distribution scheduling with limited vehicle capacity. *International Journal of Production Economics* 160, 13–25.

Grosso, A.; Della Croce, F. & Tadei, R. (2004): An enhanced dynasearch neighborhood for the single-machine total weighted tardiness scheduling problem. *Operations Research Letters* 32, 68–72.

Gupta, J.N.D. & Maykut, A. R. (1973): Concepts, theory, and techniques –Scheduling jobs on parallel processors with dynamic programming. *Decision Sciences* 4, 447–457.

Hansen, P. & Mladenovic, N. (2001): Variable neighborhood search: principles and applications. *European Journal of Operational Research* 130, 449–467.

Henke, T; Speranza, M.G. & Wäscher, G. (2015): The multi-compartment vehicle routing problem with flexible compartment sizes. *European Journal of Operational Research* 246, 730–743.

Henn, S. & Schmid, V. (2013): Metaheuristics for order batching and sequencing in manual order picking systems. *Computers & Industrial Engineering* 66, 338–351.

Henn, S. (2015): Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses. *Flexible Services and Manufacturing* 27, 86–114.

Ho, J.C. & Chang, Y.-L. (1991): Heuristic for minimizing mean tardiness of m parallel machines. *Naval Research Logistics* 38, 367–381.

Hurter, A.P. & Van Buer, M.G. (1996): The newspaper production/distribution problem. *Journal of Business Logistics* 17, 85–107.

Hübner, A. & Ostermeier, M. (2016): A multi-compartment vehicle routing problem with loading and unloading costs. Working Paper, Catholic University Eichstätt-Ingolstadt.

Koulamas, C. (1997): Decomposition and hybrid simulated annealing heuristics for the parallel-machine total tardiness problem. *Naval Research Logistics* 44, 109–125.

Li, C.-L.; Vairaktarakis, G.L. & Lee, C.-Y. (2005): Machine scheduling with deliveries to multiple customer locations. *European Journal of Operational Research* 164, 39–51.

Li, K.; Zhou, C.; Leung, J.Y.-T. & Ma, Y. (2016): Integrated production and delivery with single machine and multiple vehicles. *Expert Systems With Applications* 57, 12–20.

Liberatore, F.; Righini, G. & Salani, M. (2011): A column generation algorithm for the vehicle routing problem with soft time windows. *4OR* 9, 49–82.

Lourenço, H.R.; Martin, O.C. & Stützle, T. (2010): Iterated local search: Framework and applications. *Handbook of Metaheuristics*, 2nd edition., Gendreau, M. & Potvin, J.-Y. (eds.). International Series in Operations Research & Management Science 146. New York et al.: Springer, 363–397.

Low, C.; Li, R.-K. & Chang, C.-M. (2013): Integrated scheduling of production and delivery with time windows. *International Journal of Production Research* 51, 897–909.

Low, C.; Chang, C.-M.; Li, R.-K. & Huang, C.-L. (2014): Coordination of production scheduling and delivery problems with heterogeneous fleet. *International Journal of Production Research* 153, 139–148.

Olivera, A. & Viera, O. (2007): Adaptive memory programming for the vehicle routing problem with multiple trips. *Computers & Operations Research* 34, 28–47.

Petch, R & Salhi, S. (2004): A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Applied Mathematics* 133, 69–92.

Pinedo, M.L. (2016): Scheduling: Theory, Algorithms, and Systems. 5th edition, Springer, Cham et al.

Polacek, M; Hartl, R.F. & Doerner, K. (2004): A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of Heuristics* 10, 613–627.

Prescott-Gagnon, E.; Desaulniers, G.; Drexl, M. & Rousseau, L.-M. (2010): European driver rules in vehicle routing with time windows. *Transportation Science* 44, 455–473.

Roodbergen, K. J. (2001): Layout and Routing Methods for Warehouses. Trial, Rotterdam.

Schmid, V.; Doerner, K.F. & Laporte, G. (2013): Rich routing problems arising in supply chain management. *European Journal of Operational Research* 224, 435–448.

Scholz, A.; Schubert, D. & Wäscher, G. (2016): Order picking with multiple pickers and due dates – Simultaneous solution of order batching, batch assignment and sequencing, and picker routing problems. Working Paper No. 5/2016, Faculty of Economics and Management, Otto-von-Guericke-Universität Magdeburg.

Shim, S.-O. & Kim, Y.-D. (2007): Scheduling on parallel identical machines to minimize total tardiness. *European Journal of Operational Research* 177, 135–146.

Taillard, É.D.; Laporte, G. & Gendreau, M. (1996): Vehicle routeing with multiple use of vehicles. *The Journal of the Operational Research Society* 47, 1065–1070.

Taillard, É.D.; Badeau, P.; Gendreau, M.; Guertin, F. & Potvin, J.-Y. (1997): A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportations Science* 31, 170–186.

Tarantilis, C.D.; Kiranoudis, C.T. & Vassiliadis, V.S. (2004): A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. *European Journal of Operational Research* 152, 148–158.

Ullrich, C.A. (2013): Integrated machine scheduling and vehicle routing with time windows. *European Journal of Operational Research* 227, 152–165.

Van Buer, M.G.; Woodruff, D.L. & Olson, R.T. (1999): Solving the medium newspaper production/distribution problem. *European Journal of Operational Research* 115, 237–253.

Vidal, T.; Crainic, T.G.; Gendreau, M. & Prins, C. (2013): Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research* 231, 1–21.

Yalaoui, F. & Chu, C. (2002): Parallel machine scheduling to minimize total tardiness. *International Journal of Production Economics* 76, 265–279.

Otto von Guericke University Magdeburg
Faculty of Economics and Management
P.O. Box 4120 | 39016 Magdeburg | Germany

Tel.: +49 (0) 3 91/67-1 85 84
Fax: +49 (0) 3 91/67-1 21 20

www.fww.ovgu.de/femm

ISSN 1615-4274