The value of integrating order picking and vehicle routing decisions in a B2C e-commerce environment

Peer-reviewed author version

MOONS, Stef; BRAEKERS, Kris; RAMAEKERS, Katrien; CARIS, An & Arda, Yasemin (2019) The value of integrating order picking and vehicle routing decisions in a B2C e-commerce environment. In: INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH, 57 (20), p. 6405-6423.

DOI: 10.1080/00207543.2019.1566668
Handle: http://hdl.handle.net/1942/27620

AUTHOR VERSION - ACCEPTED

The value of integrating order picking and vehicle routing decisions in a B2C e-commerce environment

S. Moons[a,*], K. Braekers[a], K. Ramaekers[a], A. Caris[a] and Y. Arda[b]

[a]UHasselt - Hasselt University, Research Group Logistics, Agoralaan - Building D, 3590 Diepenbeek, Belgium;
[b]QuantOM, HEC Liège - Management School of the University of Liège, Rue Louvrex 14, 4000 Liège, Belgium.

ARTICLE HISTORY
Compiled January 17, 2019

ABSTRACT
In B2C e-commerce sales, customers expect a fast and low-cost delivery. To be able to fulfil these customer expectations, both warehouse and distribution operations have to be performed in an efficient and effective way. Ideally, these two supply chain functions should be considered simultaneously in an integrated problem since they are interrelated. In this paper, a record-to-record travel algorithm is proposed to solve the integrated order picking-vehicle routing problem (I-OP-VRP). Experiments with both small-size and large-size instances are conducted. Furthermore, the integrated approach is compared with an approach in which both problems are solved sequentially. Results show that integration leads to increased service levels, i.e., it allows to shorten the time between placing an order and receiving the goods. On top, the integrated approach leads to costs savings of on average 1.8%. Thus, integration is indispensable for a fast and cost-efficient delivery of goods.

KEYWORDS
integration; order picking; vehicle routing problem; meta-heuristic algorithm; e-commerce logistics

1. Introduction

Business-to-consumer (B2C) e-commerce sales are increasing yearly in Europe. In 2016, they increased with about 15.5% (Ecommerce Foundation 2017), resulting in an annual delivery of 4.2 billion parcels (Ecommerce Europe 2016). Handling this large number of orders in a short time period puts the logistics activities of the supply chain under pressure. At the same time, customers have high expectations on the delivery of their online purchases. To meet the expectations at low cost, e-commerce companies have to thoroughly reconsider their operations. Instead of individually optimising every single process of the supply chain, related problems need to be handled simultaneously. Integration results in larger savings than the improvement of individual supply chain functions. Therefore, companies ideally have to adopt an integrated approach to recognize the relationship between supply chains functions, e.g., warehouse and delivery operations (Chen 2004).

*Corresponding author: S. Moons (stef.moons@uhasselt.be)

When a customer purchases goods online, these products first need to be picked in a distribution centre (DC). After completing the picking process, the goods have to be delivered to the preferred delivery location of the customer. Thus, since orders can only be delivered after they are picked in the DC, picking and delivery decisions are interrelated. To obtain more efficient schedules, the order picking and delivery decisions need to be taken simultaneously. Such an integrated approach will lead to better solutions than when solving both problems separately and sequentially (Moons et al. 2018).

Although integrating supply chain functions into a single optimisation problem will most probably improve the company's performance, historically these problems are handled in an uncoordinated way. In literature, distribution operations are mostly integrated with production tasks. The first studies on the integration of production scheduling and vehicle routing decisions at the operational level were published in the 1990s. Especially since 2010, there is an increasing interest in this research domain. A detailed review on integrated production scheduling-vehicle routing problems can be found in Moons et al. (2017).

Production and warehousing activities have many concepts in common as stated in Moons et al. (2018). Nevertheless, differences occur between these two problems. A warehouse environment is often more complex than a production environment. In a production context, the main decision to be taken is the choice and design of the machine environment including the number of machines. In a warehouse context, however, multiple decisions have to be taken concerning, e.g., the number of order pickers to hire, the routing policy, storage location policy, batching policy, and zoning policy.

Speranza (2018) identifies a more systemic, or integrated, approach as one of the major research directions based on the current trends in transportation and logistics. Recently, an increasing number of studies have been conducted on integrated, or rich, vehicle routing problems (VRPs). Schmid, Doerner, and Laporte (2013) provide an overview of interesting extensions to the classical VRP. Examples from literature are location-routing problems, inventory-routing problems, production-routing problems, and routing problems with loading constraints.

Integrating order picking and vehicle routing decisions is a new problem variant in the research on integrated problems. In the integrated order picking-vehicle routing problem (I-OP-VRP), picking lists and vehicle routes are determined simultaneously. Requirements and constraints of both the order picking problem (OPP) and the VRP are considered at the same time. For example, delivery time windows are taken into account when picking lists are established. In uncoordinated approaches, a picking deadline is usually adopted, i.e., the picking process must finish before the beginning of the delivery operations. In the integrated approach, however, no such deadline exists. Consequently, there is higher flexibility for determining picking lists and delivery routes. In the I-OP-VRP, the following decisions need to be made: (1) the assignment and scheduling of orders to order pickers, (2) the assignment of orders to vehicles, and (3) the construction of vehicle routes. The outcome is a detailed schedule so that all orders are picked and delivered within the time window chosen by the customer.

The integration of order picking and vehicle routing decisions is a recent research area. Therefore, a standard problem is considered such that the focus is on the value of integration and not the complexity of the problem or the solution methodology. In the order picking subproblem, a discrete order picking policy, i.e., picking each customer order in an individual picking route, is applied. A discrete order picking policy is easy to understand for order pickers, and no sorting procedure of the picked items need to be conducted at the end of the picking tour resulting in a lower possibility of errors.

The vehicle routing subproblem considers a homogeneous fleet of vehicles and hard time windows.

The contributions of this paper are: (1) the proposition of a meta-heuristic algorithm to efficiently solve the I-OP-VRP; (2) the introduction of a set of realistic benchmark instances; and (3) the quantification of the value of integrating order picking and vehicle routing decisions. The remainder of this paper is structured as follows. An overview of related literature is given in Section 2. Section 3 describes the considered problem in detail. The notation used in this paper is introduced in Section 4. A mathematical formulation for the integrated order picking-vehicle routing problem addressed in this paper is presented in Section 5. A heuristic based on a record-to-record travel (RRT) algorithm to solve the integrated order picking-vehicle routing problem is presented in Section 6. Experiments using the developed heuristic algorithm are conducted in Section 7. In Section 8, the value of integration is examined. In general, similar findings can be noticed for large-size instances compared to these in Moons et al. (2018) for small-size instances. Service level can be improved and total operational cost can be decreased by integrating both problems. Finally, in Section 9, conclusions and future research opportunities are highlighted.

2. Related literature

Integrating order picking and vehicle routing decisions is a relatively new research domain. A first step towards the integration of order picking and delivery operations into a single optimisation problem is made by Low, Li, and Chang (2013), Low et al. (2014), Low, Chang, and Gao (2017), Zhang, Wang, and Huang (2016), and Zhang, Wang, and Huang (2017). The studies by Low, Li, and Chang (2013), Low et al. (2014), and Low, Chang, and Gao (2017) consider the integration of a scheduling problem in a DC and a VRP. Nevertheless, the authors describe the problem using production related terminology instead of order picking terminology. For instance, to calculate the processing time of an order in the DC a unit processing time of a retailer is multiplied by the demand of that retailer. However, order picking processing times are not proportional to the demand requested, but mainly depend on the travel times within the warehouse. In Low, Li, and Chang (2013), an INLP model is used to minimise the time to deliver all customer orders. In subsequent studies of Low et al. (2014) and Low, Chang, and Gao (2017), the objective is cost minimisation considering fixed vehicle costs, transportation costs, and penalty costs incurred for the violation of a time window. Solution procedures based on Genetic algorithms (GA) are developed to solve the problem. In Low, Li, and Chang (2013) and Low et al. (2014), two versions of a GA are proposed in each study, of which the second GA is an adaptive GA with dynamically modified parameter values. In Low, Chang, and Gao (2017), a backward adaptive genetic algorithm and a forward adaptive genetic algorithm are developed.

Zhang, Wang, and Huang (2016) and Zhang, Wang, and Huang (2017) integrate an order picking system with distribution operations. To the best of the authors' knowledge, Zhang, Wang, and Huang (2016) and Zhang, Wang, and Huang (2017) are the first to investigate the integration of order picking and distribution problems. The main focus of the studies is on the order picking subproblem where orders arrive dynamically over time. In Zhang, Wang, and Huang (2016), the delivery operations are outsourced to a third-party logistics (3PL) service provider. Zhang, Wang, and Huang (2017) explicitly take into account the distribution operations. However, the customer locations are divided in different zones, and each zone is delivered by a direct shipment from the DC. Consequently, in both studies no vehicle routing decisions have to be made. In a real-world e-commerce context, however, multiple customers are delivered in a single route, and as such, vehicle routing decisions should be considered.

Recently, the first studies on an I-OP-VRP have been published by Moons et al. (2018) and Schubert, Scholz, and Wäscher (2018). Moons et al. (2018) examine an I-OP-VRP in a B2C e-commerce context. A MIP formulation for an I-OP-VRP with time windows to minimise the sum of the order picking and delivery costs is presented. Comparing an uncoordinated and integrated approach for small-size instances with up to 20 customer orders indicates that integration can result in higher service levels and cost savings of 14% on average. The problem considered in this paper is the same as the one of Moons et al. (2018), except for the objective function. The objective is still to minimise total costs, but a more realistic assessment of these costs is made: routing costs are composed of both distance- and time-based components (instead of only time), and waiting times of vehicles at the start of a route are no longer penalised. Additionally, while Moons et al. (2018) only solve small-size instances with CPLEX using the mathematical formulation proposed, the current paper proposes a RRT heuristic which allows to solve large-size problem instances and to analyse the value of integration on these larger instances.

Schubert, Scholz, and Wäscher (2018) study an I-OP-VRP for the supply of perishable goods to supermarkets. The authors develop an iterated local search (ILS) algorithm. A variable neighbourhood descent method, with four neighbourhoods impacting the VRP and two adapting the OPP, is implemented in the ILS to solve the problem. Experiments with instances with 100 and 200 customers are executed. The objective is to minimise total tardiness with respect to delivery due dates. The overall average reduction in tardiness is 37.8% compared to an uncoordinated sequential approach.

Both Moons et al. (2018) and Schubert, Scholz, and Wäscher (2018) apply a discrete order picking policy in which each order is picked in an individual tour and use homogeneous vehicles for the delivery operations. The picking routes within a warehouse to retrieve the requested goods from their storage location are solved in advance as a separate problem in both studies and are used as input for the I-OP-VRP. The current study (and the one of Moons et al. (2018)) differs from the study of Schubert, Scholz, and Wäscher (2018) in the following ways. First, Schubert, Scholz, and Wäscher (2018) consider only delivery due dates and focus on maximising the service level, at any cost, by minimising late deliveries (i.e., tardiness). In contrast, here the objective is to minimise the total costs incurred to ensure a desired service level. Hard deliveries time windows (instead of soft due dates) are imposed, taking into account the fact that customers perceive a higher service level when they can select the time window in which the goods need to be delivered, such that they do not have to stay at home an entire day waiting for the delivery of their parcel to avoid a failed delivery. Second, next to a tour length restriction, vehicle capacity constraints are considered in paper. Third, while Schubert, Scholz, and Wäscher (2018) allow vehicles to make multiple tours, in this study each vehicle can conduct at most a single tour.

In this study, the focus is on the total operational cost of picking and delivery operations as well as on the service level offered. Furthermore, the algorithm is used to quantify the benefit of integrating both supply chain functions. Additionally, for the delivery aspect of the problem, the heuristic algorithm in this paper makes use of well-known local search operators which have proven their effectiveness on the VRP with time windows in the past, in contrast to Schubert, Scholz, and Wäscher (2018) who use less common operators to adapt the routes. The same picking operators are used in both papers. The operators in this paper are designed in such a way that they adapt the solution of a single subproblem, i.e., only changing the picking lists or the vehicle routes. When changing the solution of one subproblem results in an infeasibility for the second subproblem, appropriate changes to the latter subproblem are made to maintain solution feasibility. For example, when by changing the picking schedules the picking process of some orders is not finished before the departure of the delivery vehicles, then the vehicle routes themselves are adapted to solve the infeasibility.

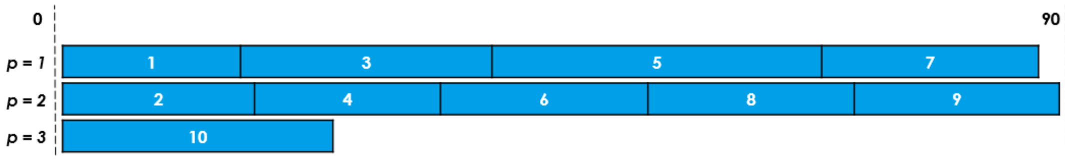
3. Problem description

In the I-OP-VRP considered in this paper, a number of orders, each consisting of one or more articles (order lines) have to be picked in a DC and subsequently delivered to their final customer destination. Both order picking schedules and vehicle routes need to be determined. All customer orders have to be handled in a DC, in which multiple order pickers are working in parallel in a single zone in the DC. In case of high customer demand, additional order pickers can be temporarily hired from a fixed pool of workers. To avoid congestion in the aisles of the warehouse, the number of order pickers that can work during a specified time period is limited. The labour cost, which is incurred for each minute working, is different for both types of order pickers. Temporarily hired order pickers have a higher labour cost compared to regular order pickers. Each order picker is allowed to work a maximum amount of time during a single shift. A picking schedule and vehicle routes have to be determined for a single shift of order pickers and drivers.

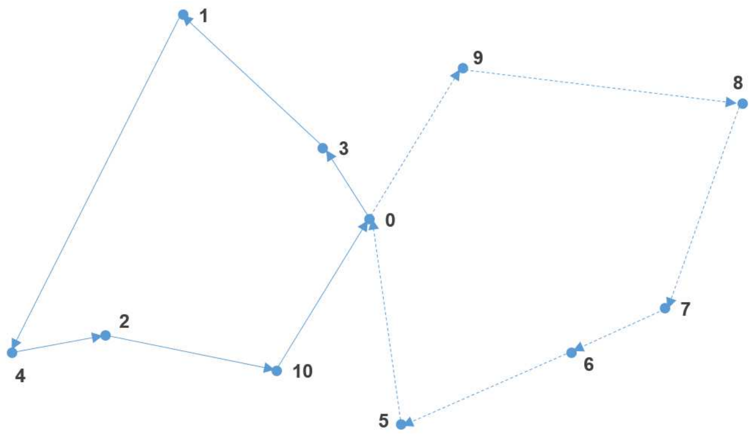
A discrete order picking policy in which each order is picked in an individual route through the warehouse is applied using a picker-to-product system. Splitting an order into suborders or combining orders in batches is not allowed. The storage locations of the goods in the DC are known in advance. Therefore, the picking routes can be determined in a separate problem and are used as input for the I-OP-VRP. The picking time of an order is independent of whether it is picked by a regular or a temporary order picker as these travel at the same speed.

The deliveries are executed by a homogeneous fleet of vehicles (vans), which is originally located at the DC. Each vehicle has to return to the DC at the end of a route and is allowed to conduct at most a single route. A customer order cannot be split over multiple routes. The delivery cost consists of two components: a cost for each kilometre travelled and a labour cost of the driver, which is incurred from the moment the loading of the vehicle is started. This is a common cost structure in transport economics (Blauwens, De Baere, and Van De Voorde 2016). The working time during a driver's shift is limited. During the purchasing process, customers can select a single time window from a list of options. The goods purchased need to be delivered within the time period selected by the customer. The time window bounds cannot be violated. At each delivery location, a service time is considered. At the start of a route, a fixed loading time at the DC is taken into account. The objective of the I-OP-VRP is to minimise total costs of the order picking and delivery operations.

In Figure 1, an example of a solution for the picking schedules and vehicle routes of a small instance is shown. As can be seen, two vehicle routes need to be conducted. In one route, the orders which are picked in the first part of the picking schedules are delivered. The second route delivers the orders which are picked later.



(a) Picking schedules



(b) Vehicle routes

Figure 1. Example of picking schedules and vehicle routes for a random instance

4. Notation

In the mathematical model, the following sets, indices, parameters, and decision variables are used:

Sets and indices

$I = \{0, ..., n\}$	set of customer orders, indices i and j, where $i = j = 0$ indicates the DC
$P = \{1, ..., \bar{p}, ..., \hat{p}\}$	set of order pickers, index p, where $\{1, ..., \bar{p}\}$ indicates regular order pickers and $\{\bar{p} + 1, ..., \hat{p}\}$ temporary order pickers
$V = \{1, ..., \bar{v}\}$	set of vehicles, index v

Parameters

C_v	capacity of vehicle v, in number of items
w_i	capacity utilisation (or size) of customer order i, in number of items
pt_i	time needed to pick customer order i, in minutes
ot_i	order time of customer order i, in minutes
pd	picking due date, in minutes
rd_i	earliest completion time for loading customer order i in a vehicle ($i \geq 1$), in minutes
s_i	service time at delivery destination of customer order i, in minutes; index $i = 0$ indicates the loading time at the DC

t_{ij}	time to travel from the delivery destination of customer order i to the delivery destination of customer order j, in minutes. When two orders belong to the same customer, then $t_{ij} = 0$.
d_{ij}	distance to travel from the delivery destination of customer order i to the delivery destination of customer order j, in kilometres. When two orders belong to the same customer, then $d_{ij} = 0$.
$[a_i, b_i]$	lower bound a_i and upper bound b_i of delivery time window of customer i ($i \geq 1$); index $i = 0$ indicates the time window in which vehicles can leave and return to the DC, in minutes
$creg/ctemp$	labour cost minute of a regular/temporary order picker
$wt_{reg}^{max}/wt_{temp}^{max}$	maximum working time of a regular/temporary order picker, in minutes
ctt_v	kilometre cost coefficient of vehicle v
ctl_v	hourly cost coefficient of vehicle v
TL_{max}	maximum tour length, in minutes
Decision variables	
STO_i	picking start time of customer order i ($i \geq 1$), in minutes
CTO_i	picking completion time of customer order i ($i \geq 1$), in minutes
STT_v	loading start time of vehicle v, in minutes
TL_v	tour length of vehicle v, in minutes
DT_i	delivery time of customer order i ($i \geq 1$), i.e., start of unloading, in minutes
X_{ip}	binary variable which is equal to 1 ($X_{ip} = 1$) if customer order i is picked by order picker p
U_{ijp}	binary variable which is equal to 1 ($U_{ijp} = 1$) if customer order j is picked immediately after customer order i ($i \neq j$) by order picker p
Y_{iv}	binary variable which is equal to 1 ($Y_{iv} = 1$) if customer order i is delivered by vehicle v
Z_{ijv}	binary variable which is equal to 1 ($Z_{ijv} = 1$) if customer order j is delivered immediately after customer order i ($i \neq j$) by vehicle v

5. Mathematical formulation

In this section, a mathematical formulation for the I-OP-VRP is presented, which is based on the one formulated in Moons et al. (2018). The objective function is adapted by replacing the delivery cost components by an hourly coefficient cost ctl_v and a kilometre cost ctt_v. The hourly cost consists mainly of the labour cost of the driver. Additionally, the cost of insurance, depreciation, and road tax are included in this hourly cost coefficient. The kilometre cost coefficient includes the cost of fuel, tires, maintenance, and fines (Blauwens, De Baere, and Van De Voorde 2016).

$$\min creg \cdot \sum_{i=1}^{n} pt_i \cdot \sum_{p=1}^{\bar{p}} X_{ip} + ctemp \cdot \sum_{i=1}^{n} pt_i \cdot \sum_{p=\bar{p}+1}^{\hat{p}} X_{ip} + \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{v=1}^{\bar{v}} ctt_v \cdot d_{ij} \cdot Z_{ijv} + \sum_{v=1}^{\bar{v}} ctl_v \cdot TL_v \quad (1)$$

subject to

$$\sum_{p=1}^{\hat{p}} X_{ip} = 1, \qquad \forall i \in I \setminus \{0\} \quad (2)$$

$$X_{ip} = \sum_{j=0}^{n} U_{ijp} = \sum_{j=0}^{n} U_{jip}, \qquad \forall i \in I, p \in P, i \neq j \quad (3)$$

$$\sum_{j=1}^{n} U_{0jp} \leq 1, \qquad \forall p \in P \quad (4)$$

$$STO_i \geq ot_i, \qquad \forall i \in I \quad (5)$$

$$STO_j \geq CTO_i - M^1 \cdot \left(1 - \sum_{p=1}^{\hat{p}} U_{ijp}\right), \qquad \forall i, j \in I, i \neq j, M^1 = pd \quad (6)$$

$$CTO_i = STO_i + pt_i, \qquad \forall i \in I \quad (7)$$

$$CTO_i \leq pd, \qquad \forall i \in I \quad (8)$$

$$\sum_{i=1}^{n} pt_i \cdot X_{ip} \leq wt_{reg}^{max}, \qquad \forall p = 1, ..., \bar{p} \quad (9)$$

$$\sum_{i=1}^{n} pt_i \cdot X_{ip} \leq wt_{temp}^{max}, \qquad \forall p = \bar{p}+1, ..., \hat{p} \quad (10)$$

$$Z_{ijv} = 0, \qquad \forall i, j \in I \setminus \{0\}, i \neq j, \forall v \in V, a_i + s_i + t_{ij} \geq b_j \quad (11)$$

$$\sum_{v=1}^{\bar{v}} Y_{iv} = 1, \qquad \forall i \in I \setminus \{0\} \quad (12)$$

$$Y_{0v} \geq Y_{iv}, \qquad \forall i \in I \setminus \{0\}, v \in V \quad (13)$$

$$Y_{jv} = \sum_{i=0}^{n} Z_{ijv} = \sum_{i=0}^{n} Z_{jiv}, \qquad \forall j \in I, v \in V, i \neq j \quad (14)$$

$$\sum_{i=1}^{n} w_i Y_{iv} \leq C_v, \qquad \forall v \in V \quad (15)$$

$$rd_i \leq STT_v + M_i^2 \cdot (1 - Y_{iv}), \qquad \forall i \in I \setminus \{0\}, v \in V, M_i^2 = rd_i \quad (16)$$

$$a_0 \leq STT_v, \qquad \forall v \in V \quad (17)$$

$$STT_v + s_0 + t_{0j} \leq DT_j + M_j^3 \cdot (1 - Z_{0jv}), \qquad \forall j \in I \setminus \{0\}, v \in V, M_j^3 = b_0 + s_0 + t_{0j} - a_j \quad (18)$$

$$DT_i + s_i + t_{ij} \leq DT_j + M^4_{ij} \cdot \left(1 - \sum_{v=1}^{\bar{v}} Z_{ijv}\right), \qquad \forall i, j \in I \setminus \{0\}, i \neq j,$$

$$M^4_{ij} = b_i + s_i + t_{ij} - a_j \qquad (19)$$

$$a_i \leq DT_i \leq b_i, \qquad \forall i \in I \setminus \{0\} \qquad (20)$$

$$DT_i + s_i + t_{i0} \leq b_0 + M^5_i \cdot \left(1 - \sum_{v=1}^{\bar{v}} Z_{i0v}\right), \qquad \forall i \in I \setminus \{0\},$$

$$M^5_i = b_i + s_i + t_{i0} - b_0 \qquad (21)$$

$$DT_i + s_i + t_{i0} - STT_v \leq TL_v + M^6_i \cdot (1 - Z_{i0v}), \qquad \forall i \in I \setminus \{0\}, v \in V,$$

$$M^6_i = b_i + s_i + t_{i0} \qquad (22)$$

$$TL_v \leq TL_{max}, \qquad \forall v \in V \qquad (23)$$

$$CTO_i, STO_i \geq 0, DT_i \geq 0, \qquad \forall i \in I \qquad (24)$$

$$STT_v, TL_v \geq 0, \qquad \forall v \in V \qquad (25)$$

$$Y_{iv}, Z_{ijv} \in \{0, 1\}, \qquad \forall i, j \in I, i \neq j, v \in V \qquad (26)$$

$$X_{ip}, U_{ijp} \in \{0, 1\}, \qquad \forall i, j \in I, i \neq j, p \in P \qquad (27)$$

The objective function (1) minimises the sum of the labour cost of the order pickers, the hour coefficient cost, and the kilometre travel cost. Each customer order must be assigned to exactly one order picker as stated by constraints (2). Constraints (3) indicate that each customer order needs to have the same number of predecessors and successors. At most one order can be picked as first in a picking sequence as indicated by inequalities (4). Constraints (5) specify the earliest possible start time of the picking process of a customer order. The start time and completion time of picking a customer order is determined by constraints (6) and (7), respectively. Inequalities (8) impede that the picking due date is violated. The working time of regular and temporary order pickers is limited by constraints (9) and (10), respectively. A customer cannot be visited before another customer if the time window of the former one starts after the end of this of the latter one as stated by constraints (11). Constraints (12) guarantee that each customer order must be delivered by exactly one vehicle. The DC needs to be visited in each route as ensured by inequalities (13). Constraints (14) specify that a vehicle has to arrive at and leave each customer order location once. Constraints (15) limit the capacity of a vehicle. The earliest possible start time of a vehicle tour is specified by inequalities (16) and (17). Constraints (18) and (19) determine the delivery time of each order. The delivery time needs to be within the time window of a customer as ensured by inequalities (20). Constraints (21) indicate that each vehicle has to be back at the DC on time. The maximum tour length is restricted by inequalities (22) and (23). Constraints (24)-(26) indicate the domain of the decision variables.

6. Solution procedure

The proposed solution approach is based on a record-to-record travel (RRT) algorithm, which is a deterministic algorithm similar to simulated annealing (SA) and is first introduced by Dueck (1993) as a variant on threshold accepting (TA, or deterministic annealing). In SA, a new better solution is always accepted, while a new worse solution is accepted with a gradually lowered probability. In RRT, each new solution not worse than the best solution, i.e., *record*, is always accepted. Furthermore, a solution which only deviates a certain fixed percentage from the record is also accepted. In RRT, a new solution is always compared with the best solution found so far, while in TA a new solution is compared with the last accepted solution. Furthermore, in SA a larger number of parameters need to be tuned in comparison with a RRT algorithm. RRT algorithms have efficiently been used to solve VRP variants (e.g., Li, Golden, and Wasil 2007; Groër, Golden, and Wasil 2009). The first step of the proposed algorithm is to generate an initial solution. Next, to improve the quality of this solution, five local search operators are used iteratively in a RRT framework for a number of iterations.

6.1. *Initial solution: constructive heuristic*

An initial solution is created by using a constructive heuristic consisting of two parts, one for each subproblem. For the assignment of orders to pickers the same procedure is used as in Belo-Filho, Amorim, and Almada-Lobo (2015) for the assignment of orders to production lines in an integrated production scheduling-vehicle routing problem. The initial vehicle routes are created by applying the cheapest insertion principle. This procedure is also applied by Du, Li, and Chou (2005) for a dynamic VRP, and by Liu, Li, and Liu (2017) for a VRP with release dates, both in a B2C e-commerce context.

Before assigning orders to pickers, the minimum number of order pickers needed $NPick_{min}$ is calculated as follows: $\lceil$total order picking time/maximum working time of a picker$\rceil$. The orders are assigned to the pickers iteratively, following a non-decreasing order of the upper bound of the delivery time window chosen by the customer. The assignment procedure depends on $NPick_{min}$. If $NPick_{min}$ is less than or equal to the number of regular pickers, then an order is assigned to the first position of the picking schedule of each picker in the set of $NPick_{min}$ pickers. Afterwards, orders are assigned to the next schedule positions of each required picker until all orders are assigned. Before an order is assigned to a picker, feasibility is checked concerning the maximum allowed picker working time and delivery deadline. If $NPick_{min}$ is greater than the number of regular pickers, then temporary order pickers are required. In this case, orders are first assigned as much as possible to regular order pickers, and thereafter the remaining unassigned orders are assigned to temporary order pickers using the same procedure. If, after this procedure, still some orders are not assigned to a picker, then an additional picker is added to the set of $NPick_{min}$ pickers until all orders are assigned to a picker.

In Figure 1(a), an example of the assignment procedure for the picking schedule is shown. Ten orders need to be picked with a total picking time of 185 minutes. Two regular order pickers are available with each a maximum working time of 90 minutes. Consequently, a third temporary order picker needs to be hired. First, orders are as much as possible assigned to the two regular pickers in an alternating way. Next, the remaining order is assigned to the third temporarily hired order picker.

The cheapest insertion principle (Rosenkrantz, Stearns, and Lewis 1977) is used for the assignment of orders to vehicle routes. The first order is assigned to an empty route. For the next orders, each possible insertion position is checked and the cheapest feasible option is selected. Feasibility needs to be checked concerning the vehicle capacity, maximum allowed route length, and delivery time windows. The time window feasibility check is executed in constant time using the earliest departure time and latest arrival time as described in Vidal et al. (2015). Additionally, the order has to be picked before the start of the route. If this relationship is violated, the picking schedule is adapted such that the order can be inserted at the cheapest position without violating the picking-delivery relationship of other orders (in a similar way as described in the last paragraph of Section 6.3). If this is not possible, the order is assigned to the next best feasible insertion position in a route.

For example, if the best insertion position is in a vehicle with a departure time of 180 and the picking operations of that order are completed at 190, the order needs to be rescheduled such that the picking process is completed earlier. However, when this would lead to new violations of the picking-delivery relationship for other orders, the order cannot be rescheduled. The order needs to be assigned to another vehicle with a later departure time, e.g., 200.

6.2. *Local search operators*

In order to improve the initial solution, five local search operators are used. To adapt the constructed routes, the following three operators are applied: $\mathit{2\text{-}Opt}_{VRP}$ (intra-route), $\mathit{exchange}_{VRP}$, and $\mathit{relocate}_{VRP}$ (intra- and inter-route). These operators are often used to solve a VRP with time windows (Toth and Vigo 2014). Furthermore, also Du, Li, and Chou (2005) and Liu, Li, and Liu (2017) use the relocate and exchange operator to improve their initial solution for a VRP in an e-commerce environment. The contribution of each operator is examined in Section 7.4. The $\mathit{2\text{-}Opt}_{VRP}$ operator reverses the direction of a subpath. Two edges are removed and replaced by two new edges within the same route (Croes 1958). The $\mathit{exchange}_{VRP}$ operator swaps two customer orders within the same route or between two routes. The operator can only swap an order within the same route if there are at least three edges between the orders in order to avoid overlap with the $\mathit{2\text{-}Opt}_{VRP}$ operator. The $\mathit{relocate}_{VRP}$ operator removes a customer order from a route and reinserts it at another position in the same route or in another route (Savelsbergh 1992). The moves are evaluated for feasibility concerning vehicle capacity, time windows, route length and the OPP-VRP relationship. When the OPP-VRP relationship is violated, a move is not discarded directly but it is checked whether feasibility can be maintained by adapting picking schedules using the procedure explained in the last paragraph of Section 6.3.

The order picking schedules are changed with similar operators: $\mathit{exchange}_{OPP}$ and $\mathit{relocate}_{OPP}$. The OPP-operators are only applied to moves between temporary and regular order pickers since only that type of moves could possibly result in a lower total picking cost. Swapping orders between regular order pickers or relocating an order from one regular to another regular order picker will never lead to lower labour costs because regular order pickers all have the same labour cost. However, when other picking sequences in the picking lists of regular order pickers are needed to obtain feasible or cheaper vehicle routes, these changes will be found by a VRP operator as will be explained later. The $\mathit{exchange}_{OPP}$ operator swaps a customer order currently being picked by a temporary order picker with a customer order being picked by a regular order picker. The $\mathit{relocate}_{OPP}$ operator removes a customer order from the picking schedule of a temporary order picker and reinserts it in the schedule of a regular order picker. Feasibility checks are executed with respect to maximum working time and the OPP-VRP relationship.

Algorithm 1 Detailed outline of solution procedure with record-to-record travel

Parameters: α, I_{max}, $I_{max}^{non-impr}$, $M_{max}^{non-impr}$
    α: deviation rate
    I_{max}: maximum number of iterations
    $I_{max}^{non-impr}$: maximum number of consecutive iterations without improving the record
    $M_{max}^{non-impr}$: maximum number of consecutive moves without improving the record
Variables: $numb\_it$, I^{non_impr}, M^{non_impr}
    $numb\_it$: iteration number
    I^{non_impr}: number of consecutive iterations without improving the record
    M^{non_impr}: number of consecutive moves without improving the record
Solutions: S_B = best solution, S_A = last accepted solution
    S_0 = initial solution, S = current solution
Operators = $\{$*2-Opt*$_{VRP}$, *exchange*$_{VRP}$, *relocate*$_{VRP}$, *exchange*$_{OPP}$, *relocate*$_{OPP}\}$
Determine S_0
$S_B := S_0$; $S_A := S_B$; $record := Z[S_B]$; $deviation := \alpha \cdot Z[S_B]$
$numb\_it := 0$, $I^{non\_impr} := 0$, $M^{non\_impr} := 0$
repeat
    Shuffle operators randomly leading to an operator sequence numbered from 1 to 5
    for $operator\_sequence\_number = 1$ to 5 **do**
        Select random vehicle or order picker from S_A
        $M^{non\_impr} := 0$
        repeat
            Select random order in vehicle route or order picker schedule
            Conduct best move on S_A resulting in solution S
            if $Z[S] < record + deviation$ **then**
                $S_A := S$
                if $Z[S] < record$ **then**
                    $S_B := S$
                    update *record* and *deviation*
                    $M^{non\_impr} := 0$
                else
                    $M^{non\_impr} := M^{non\_impr} + 1$
                end if
            end if
        until $M^{non\_impr} = M_{max}^{non-impr}$
    end for
    if *record* is updated **then**
        $I^{non\_impr} := 0$
    else
        $I^{non\_impr} := I^{non\_impr} + 1$
    end if
    if $I^{non\_impr} = I_{max}^{non-impr}$ **then**
        $S_A := S_B$
        $I^{non\_impr} := 0$
    end if
    $numb\_it := numb\_it + 1$
until $numb\_it = I_{max}$

6.3. *Algorithmic framework*

A detailed outline of the solution procedure is given in Algorithm 1. In each iteration of the local search, the five operators are executed in a random order (line 8). Each operator is executed for a single randomly selected vehicle or order picker and starts from the last accepted solution S_A (line 10). Within a vehicle route or picking schedule a random customer order is selected for which the operator is executed (line 13). Within the loop (line 9-26), for each operator the best feasible move for the selected order is conducted. A new solution S is accepted if its objective value $Z[S]$ is less than the best objective value found so far $Z[S_B]$ (*record*) plus a deviation (line 15-16). The deviation value is a fraction α of the record value. Additionally, if the new objective value is less than the record, it becomes the new best solution. In this case, the record and deviation value are updated (line 17-20). Otherwise, the number of non-improving moves $M^{non-impr}$ is increased (line 22). The selected operator is executed for the same vehicle or order picker as long as the number of consecutive moves without improving the record $M^{non-impr}$ is less than a predefined maximum number of consecutive non-improving moves $M^{non-impr}_{max}$ (line 25). Each time the operator is executed, a random order is selected within the same vehicle route or order picking schedule. The next operator continues with the last accepted solution. If after executing the five operators the record is not updated, the number of non-improving iterations $I^{non-impr}$ is increased (line 30). When a maximum number of consecutive iterations without improvement of the record $I^{non-impr}_{max}$ is reached, then the last accepted solution is replaced by the best solution S_B (line 32-35). The RRT heuristic is executed for a maximum number of iterations I_{max} (line 36).

In the VRP-operators, the best move is selected based on the impact of the operator on the total distribution costs, i.e., drivers' labour cost and kilometre cost. In the OPP-operators, the impact of the best move is calculated based on the total labour cost of both types of order pickers. Before actually executing a move, it needs to be evaluated whether the relationship between OPP and VRP is not violated by the move. This is where the integration of the OPP and the VRP is mainly implemented in the RRT heuristic. It is checked whether after a move the order picking process of each order is still finished before the departure time of the vehicle that delivers the order. If a violation occurs, the RRT heuristic tries to solve the violation such that the best move can be conducted. The procedure to solve a violation depends on the type of operator that causes the violation. An outline of the procedure is given in Algorithm 2. When the OPP-VRP relationship is violated by a VRP-operator, the violated orders are removed from their original position in the picking schedules (line 2) and reinserted at other positions without creating a violation for other orders. A reinsertion position is searched in the picking schedule of each order picker, starting with the first order picker at the end of the schedule (line 3-12). A removed order is inserted at the first position found which solves the violation (line 6-7). If it is not possible to reinsert all the violated orders, the VRP-move is considered not feasible. If an OPP-operator leads to violations, the violated orders are removed from the vehicle routes (line 14) and reinserted in routes with a later departure time. Each possible vehicle route is considered, starting with the first vehicle at the first position in the route (line 15-24). The removed order is inserted in the first possible route which solves the violation (line 18-19). Again, when it is not possible to reinsert the orders, then the OPP-move is considered not feasible. Additionally, a move by a VRP- or OPP-operator is considered non-acceptable if, after repairing violations, it leads to a total cost increase which is larger than the deviation value.

Algorithm 2 Outline procedure to fix violations of OPP-VRP relationship

Violations occurring due to execution of VRP operator
1: Let u_p be a vector representing the sequence of orders picked by an operator p
2: Remove violated orders from picking schedule and add them to a list $L1$
3: **for** $i = 1$ to $|L1|$ **do**
4:     **for** $p = 1$ to $\hat{p}$ **do**
5:         **for** $k = |u_p|$ to 0 **do**
6:             **if** insertion at this position is feasible **then**
7:                 Insert order i in position k of vector u_p
8:                 Go to Line 11
9:             **end if**
10:         **end for**
11:     **end for**
12: **end for**

Violations occurring due to execution of OPP operator
13: Let l_v be a vector representing the sequence of customers (and DC) visited by a vehicle v
14: Remove violated customers from vehicle route and add them to a list $L2$
15: **for** $i = 1$ to $|L2|$ **do**
16:     **for** $v = 1$ to $\bar{v}$ **do**
17:         **for** $k = 1$ to $|l_v| - 1$ **do**
18:             **if** insertion at this position is feasible **then**
19:                 Insert client i in position k of vector l_v
20:                 Go to Line 23
21:             **end if**
22:         **end for**
23:     **end for**
24: **end for**

7. Validation of heuristic algorithm

In this section, computational experiments are described to evaluate the performance of the proposed RRT algorithm. The experiments are executed on a 12-core Xeon E5-2680v3 CPUs with 128 GB RAM. The algorithm is implemented in C++. Optimal solutions for the small-size instances are obtained by ILOG CPLEX 12.7.1, using the mathematical formulation in Section 5.

7.1. *Data generation*

Since this study is one of the first in which an I-OP-VRP is solved in an e-commerce context, no benchmark instances exist. Thus, to conduct experiments, artificial instances are generated based on real-life data or related studies in the field of B2C e-commerce. Instances with three different problem sizes are generated, i.e., 10, 15, and 100 customer orders. For each instance size, 50 instances are generated resulting in 150 instances in total. The randomly generated instances are available online at `http://alpha.uhasselt.be/kris.braekers` (after acceptance). For the instances with 10 and 15 customer orders, two regular order pickers, one temporary order picker, and three vehicles are available. For the larger instances with 100 orders, nine regular order pickers, three temporary order pickers, and seven vehicles are used. The total number of order pickers available is calculated as follows: $\lceil$(maximum picking time of an order $\cdot$ number of orders)/maximum working time of a picker$\rceil$.

The order sizes are randomly generated from $TRIA(1, 2, 6)$, where $TRIA(a, c, b)$ defines a triangular distribution with a the minimum value, c the mode, and b the maximum value. The average order size is 3 items. Random numbers generated from a triangular distribution are rounded to the closest integer. The same mean order size is used in a mail order or B2C e-commerce problem setting by Ruben and Jacobs (1999), Petersen (2000), and Zhang, Wang, and Huang (2016). Since the storage locations of the goods in the DC are not known, no routing policy can be applied. Therefore, the picking times, in minutes, are randomly generated from $U(10, 27)$, where $U(x_1, x_2)$ defines a uniform distribution between x_1 and x_2. The average order processing time is 18.5 minutes as in Gong and de Koster (2008), who consider online retailers. All orders are available for picking at the start of the planning horizon. Pickers are allowed to work 240 minutes during a shift. After consulting a large international logistics service provider, the hourly labour cost of a regular and a temporary order picker is set equal to 25 and 30 euro, respectively.

The deliveries are executed by a fleet of homogeneous vans with a capacity of 100 items, similarly as in Cárdenas, Beckers, and Vanelslander (2017) who based this value on data of a Belgian logistics carrier working in a B2C e-commerce context. The delivery locations are located in a 50x50-square with the DC located in the centre, as in Liu, Li, and Liu (2017). The hourly labour cost of the driver is 25 euro (VIL 2016). Additionally, a cost of 0.22 euro per kilometre travelled is incurred (Blauwens, De Baere, and Van De Voorde 2016). Each driver is allowed to work 480 minutes. Loading the vehicles at the DC takes 20 minutes, similar as in the e-grocery problem of Punakivi and Saranen (2001) and in the I-OP-VRP of Schubert, Scholz, and Wäscher (2018). The average service time at the customer location, i.e., unloading of a parcel, is equal to four minutes (VIL 2016). The service time of an order is generated from a triangular distribution $TRIA(2, 4, 6)$.

Customers of e-commerce companies can often select a time window within which they want the goods to be delivered from a limited number of options. A survey in the United Kingdom has indicated that in case customers are allowed to choose the length of the delivery time slot approximately 52% would prefer a two-hour time window (Interactive Media in Retail Group 2014). Additionally, real-world B2C e-commerce companies offering this service mostly propose time slots with a two-hour width (e.g., Albert Heijn n.d.; Coolblue n.d.). Nine partly-overlapping time window slots are used in our experiments: seven time windows have a two-hour width, and two have a four-hour width. When the last customer in a route is visited, the vehicle has to return to the DC before the end of its time window. The time window upper bound of the DC b_0 is equal to the earliest time window lower bound a, i.e., 176, plus the maximum driver work time TL_{max}, i.e., 480 minutes, resulting in a value of 656. In Table 1, the time window options used in the computational experiments are shown.

Table 1. Time window options

Time window	Width
[176, 296]	2h
[236, 356]	2h
[296, 416]	2h
[356, 476]	2h
[416, 536]	2h
[176, 416]	4h
[416, 656]	4h
[476, 596]	2h
[536, 656]	2h

Since the problem starts in an empty state, the assumption is made that if a customer purchases goods online, then at least a two-hour time period is provided for order picking. In total, each order picker is allowed to work four hours during a single shift. Vehicles can leave the DC when needed to deliver goods on time, while still other orders are being picked by order pickers. The lower bound of the earliest time window is equal to 120 minutes plus the loading time at the DC plus the largest travel time between the DC and the farthest customer location. This customer location is located at the corner of the square, and the largest travel time is calculated as follows: $\sqrt{(x_{max}/2)^2 + (y_{max}/2)^2}$, where x_{max} and y_{max} refer to the coordinates of a corner point in a two-dimensional x-y-square. In case of a 50x50-square, both x_{max} and y_{max} are equal to 50. The travel times are calculated taking the rounded Euclidean distance between the locations of the customer orders.

7.2. *Parameter tuning*

In a record-to-record travel algorithm, the main parameter is the deviation rate α. Furthermore, the number of iterations I_{max}, the maximum number of consecutive non-improving iterations $I_{max}^{non-impr}$, and the maximum number of consecutive non-improving moves by an operator $M_{max}^{non-impr}$ need to be determined. The maximum number of iterations is the stopping criterion of the heuristic algorithm, and, therefore, this value is determined upfront by manual parameter testing using various values for the other parameters. Once the number of iterations is fixed, the remaining parameters are tuned using the irace package of López-Ibáñez et al. (2016). The irace package is a software package implementing iterated racing procedures which are used for automatically configuring parameters of algorithms. Table 2 shows the ranges within which the parameters are tuned, and the tuned value for each instance size. The irace package is executed for each problem size separately because the problem size can have an impact on the tuned values. When the irace package is used for all instances simultaneously, the result will be a parameter combination which would lead to good solutions on average. However, on each instance individually the results could be worse. The ranges are based on manual parameter testing experiments. As can be seen in Table 2, the tuned values differ between the problem sizes. The deviation rate decreases with the instance size, while the value of other parameters increases with the instance size.

Table 2. Parameter list and tuning

	10 orders		15 orders		100 orders	
Parameter	Range	Tuned value	Range	Tuned value	Range	Tuned value
I_{max}	-	700	-	6,000	-	250,000
α	(0, 0.20)	0.15	(0, 0.20)	0.15	(0.00, 0.05)	0.01
$I_{max}^{non-impr}$	(0, 10)	5	(0, 25)	10	(5, 30)	26
$M_{max}^{non-impr}$	(0, 15)	8	(5, 25)	21	(5, 35)	32

7.3. *Results*

For each instance, 20 runs are conducted. For the small-size instances with 10 and 15 customer orders, the optimal solution is obtained by CPLEX, except for three instances with 15 orders for which the optimal solution could not be found within 500 hours. For these three instances, the RRT heuristic obtains the same solution in all 20 runs. The gap between the heuristic solution found in a run and the optimal solution is calculated. For the three instances for which the optimal solution is not known, the gap between the best heuristic solution found and the solution of a run is computed. Table 3 shows a summary of the results for the experiments with the small-size instances. The average gap is 0.00% for all instances with 10 and with 15 customer orders. Thus, the RRT algorithm finds the optimal solution in all 20 runs for each of these instances for which the optimal solution is known. In Table 3, the computation time for CPLEX and a single run of the heuristic algorithm is indicated. Solving the problem to optimality by CPLEX takes approximately 2.5 minutes for the 10 orders instances and approximately 47 hours for the 15 orders instances. Increasing the problem size with five customers already has a large impact on the computation time. The RRT heuristic is capable of finding the same solution in less than one second. This indicates that the algorithm is an effective and efficient tool for solving small-size instances of the I-OP-VRP.

Table 3. Summary results of experiments with small-size instances

Instance size	avg. gap (%)	avg. time CPLEX (s)	avg. time RRT 1 run (s)	avg. # pick.	avg. # veh.
10 orders	0.00	145.12	0.0073	1.09	1.52
15 orders	0.00	170,598.30	0.2226	1.98	1.52

For the large-size instances, more detailed results are presented in Table 4. The instance number is indicated in column 1. Column 2 shows the best solution obtained by the RRT algorithm, Z_B^{RRT}. The average objective value over the 20 runs, Z_{avg}^{RRT}, is indicated in column 3. The average and maximum gap between the heuristic solution and the best heuristic solution is presented in column 4 and 5, respectively. The average run time is indicated in column 6. The objective value of the initial solution $Z[S_0]$ is given in column 7, and the percentage gap ($\Delta Z[S_0]$ in %) between the initial solution and the best heuristic solution, computed as $(Z[S_B] - Z[S_0])/Z[S_0]$, is indicated in column 8.

For the large-size instances with 100 customer orders, no optimal solution can be obtained in a short amount of computation time. To indicate the impact of the RRT heuristic on the solution, the percentage difference between the initial solution and the best heuristic solution found is provided. The best objective value found is on average 25.73% better than the initial solution after running the algorithm less than two minutes on average. Thus, the heuristic developed in this paper is clearly capable of drastically improving the initial solution. On average, approximately nine pickers are needed and four vehicles are used for the picking and delivery operations.

7.4. *Analysis of RRT heuristic*

This section analyses some design choices of the RRT heuristic described in Section 6. First, the contribution of each local search operator is investigated. Second, the comparison is made between a random and fixed sequence of executing the local search operators in the RRT heuristic.

In the RRT heuristic, five operators are implemented. Two operators work on the picking schedules, three on the vehicle routes. In the experiments described in the previous section, all operators are included in the solution algorithm. Seven variants of the RRT algorithm are analysed to determine the contribution of the operators. In five variants, a single operator is excluded from the algorithm. In variant 6, the two

Table 4. Results of instances with 100 customer orders

	Z_B^{RRT}	Z_{avg}^{RRT}	avg. gap (%)	max. gap (%)	avg. time (s) RRT	$Z[S_0]$	$\Delta Z[S_0]$ (%)
1	1,498.93	1,517.47	1.24	2.94	132.80	2,054.36	-27.04
2	1,406.49	1,415.51	0.64	1.54	92.99	1,698.16	-17.18
3	1,464.97	1,492.77	1.90	3.42	99.71	1,965.31	-25.46
4	1,438.71	1,456.02	1.20	3.48	102.31	1,920.00	-25.07
5	1,478.91	1,492.42	0.91	1.61	126.24	1,897.72	-22.07
6	1,427.16	1,446.65	1.37	3.70	93.25	2,044.89	-30.21
7	1,398.66	1,418.48	1.42	2.77	112.80	1,928.66	-27.48
8	1,393.79	1,412.63	1.35	3.43	93.40	1,984.53	-29.77
9	1,462.48	1,474.13	0.80	2.27	121.62	1,922.74	-23.94
10	1,469.39	1,488.50	1.30	2.91	89.09	2,170.43	-32.30
11	1,452.67	1,471.38	1.29	3.37	91.31	1,963.27	-26.01
12	1,469.20	1,481.63	0.85	2.71	123.19	2,041.06	-28.02
13	1,474.82	1,503.99	1.98	3.23	118.28	1,932.30	-23.68
14	1,455.98	1,484.18	1.94	3.55	105.74	2,001.20	-27.24
15	1,463.92	1,490.37	1.81	3.93	105.42	1,986.90	-26.32
16	1,483.62	1,498.98	1.04	2.50	119.98	1,954.27	-24.08
17	1,480.86	1,505.84	1.69	2.76	115.61	1,942.99	-23.78
18	1,508.28	1,538.20	1.98	3.23	140.67	2,077.76	-27.41
19	1,414.32	1,432.52	1.29	2.44	92.21	1,805.31	-21.66
20	1,447.74	1,464.93	1.19	2.50	91.39	1,872.61	-22.69
21	1,464.29	1,490.64	1.80	4.53	94.35	1,934.62	-24.31
22	1,383.80	1,397.89	1.02	3.02	111.38	1,794.22	-22.87
23	1,421.17	1,441.03	1.40	2.94	87.92	1,818.99	-21.87
24	1,458.45	1,474.44	1.10	2.61	85.68	1,824.21	-20.05
25	1,390.22	1,405.91	1.13	2.48	96.72	1,915.04	-27.41
26	1,419.64	1,436.48	1.19	3.18	124.99	1,993.55	-28.79
27	1,479.82	1,497.25	1.18	2.34	144.93	1,982.34	-25.35
28	1,427.82	1,455.81	1.96	3.72	92.59	1,988.16	-28.18
29	1,510.16	1,524.69	0.96	2.86	98.92	2,068.59	-27.00
30	1,434.40	1,452.83	1.28	2.52	113.89	1,877.88	-23.62
31	1,468.97	1,496.35	1.86	3.68	107.08	2,025.53	-27.48
32	1,466.69	1,477.57	0.74	2.19	94.67	2,000.73	-26.69
33	1,429.21	1,451.49	1.56	3.13	106.90	1,968.89	-27.41
34	1,478.17	1,497.55	1.31	2.81	106.88	2,118.02	-30.21
35	1,405.78	1,432.62	1.91	3.79	109.90	1,924.98	-26.97
36	1,468.77	1,494.02	1.72	3.07	104.59	2,001.63	-26.62
37	1,507.19	1,534.74	1.83	3.77	104.38	2,105.66	-28.42
38	1,455.90	1,474.49	1.28	3.63	95.27	2,029.42	-28.26
39	1,463.12	1,479.91	1.15	3.12	120.70	1,774.67	-17.56
40	1,413.29	1,433.10	1.40	2.91	105.68	1,953.46	-27.65
41	1,485.62	1,502.22	1.12	2.46	125.23	2,046.59	-27.41
42	1,477.12	1,495.20	1.22	3.20	108.43	2,066.81	-28.53
43	1,413.17	1,428.19	1.06	2.22	100.76	1,889.83	-25.22
44	1,486.38	1,503.41	1.15	2.23	94.23	1,907.33	-22.07
45	1,497.29	1,513.78	1.10	2.62	104.86	1,944.22	-22.99
46	1,450.01	1,470.89	1.44	3.60	104.35	1,826.86	-20.63
47	1,448.89	1,468.56	1.36	2.72	103.49	2,061.17	-29.71
48	1,445.91	1,462.24	1.13	2.39	98.73	1,971.60	-26.66
49	1,394.60	1,408.34	0.99	2.41	97.56	1,914.95	-27.17
50	1,428.61	1,443.71	1.06	3.62	96.30	1,977.97	-27.77
		avg. gap	1.33	avg. time	106.12	avg. Δ	-25.73

picking operators are excluded, while in variant 7 all VRP operators are left out. Each variant is executed for all 50 instances of each problem size and 20 runs are conducted for each instance.

Table 5 shows the contribution of the operators for each problem size. The gap between the best solution obtained by the RRT heuristic with all operators included and the solution obtained by the specific scenario for each run is calculated. The average gap over all runs and instances is calculated. The higher the gap, the more

Table 5. Contribution of local search operators

	Avg. gap (%)		
Scenario	10 orders	15 orders	100 orders
All operators included	0.0000	0.0000	1.3310
$exchange_{VRP}$ excluded	0.0007	0.0001	1.4539
$relocate_{VRP}$ excluded	1.6826	3.0866	7.5912
$2\text{-}Opt_{VRP}$ excluded	0.0015	0.0003	1.3046
$exchange_{OPP}$ excluded	0.0000	0.0001	1.3212
$relocate_{OPP}$ excluded	0.0000	0.0481	2.7074
All OPP operators excluded	0.0007	0.0661	3.0222
All VRP operators excluded	4.8138	9.3977	34.5683

impact the operator has on the solution quality. The *relocate* operators of both the OPP and the VRP have the largest individual impact. The *exchange* and $2\text{-}Opt_{VRP}$ operators have a smaller impact on the solution quality. When all VRP operators are removed, the solution quality decreases with approximately 34% for the largest instances. The solutions obtained by this variant differ only slightly from the initial solutions, with only an average improvement of 0.22%.

In the original RRT heuristic, the five local search operators are executed in a random sequence in each iteration. Additional experiments are conducted to investigate whether the sequence of the operators influences the performance of the solution algorithm. Six different fixed sequences are tested. In sequence 1a-1c, the VRP operators are executed first, followed by the two OPP operators. In sequence 2a and 2b, first, the OPP operators are used and thereafter the VRP operators. In sequence 3, the VRP and OPP operators are executed in an alternating way. The following sequences are tested:

- 0 Random sequence of operators
- 1a $exchange_{VRP}$ - $relocate_{VRP}$ - $2\text{-}Opt_{VRP}$ - $exchange_{OPP}$ - $relocate_{OPP}$
- 1b $relocate_{VRP}$ - $exchange_{VRP}$ - $2\text{-}Opt_{VRP}$ - $exchange_{OPP}$ - $relocate_{OPP}$
- 1c $2\text{-}Opt_{VRP}$ - $exchange_{VRP}$ - $relocate_{VRP}$ - $exchange_{OPP}$ - $relocate_{OPP}$
- 2a $exchange_{OPP}$ - $relocate_{OPP}$ - $exchange_{VRP}$ - $relocate_{VRP}$ - $2\text{-}Opt_{VRP}$
- 2b $relocate_{OPP}$ - $exchange_{OPP}$ - $exchange_{VRP}$ - $relocate_{VRP}$ - $2\text{-}Opt_{VRP}$
- 3 $exchange_{VRP}$ - $exchange_{OPP}$ - $relocate_{VRP}$ - $relocate_{OPP}$ - $2\text{-}Opt_{VRP}$

The gap between the best solution found in the basis scenario 0 with a random sequence and the solution found by the variant is calculated. In the experiments with large-size instances the best solution is not found in each run. Therefore, an average gap for the original variant is computed as well by calculating the difference between the best solution found and the solution of each run. The average gap over 20 runs and 50 instances for each variant are presented in Table 6. For the small-size instances, the random sequence obtains the best solutions for all instances in all runs. Most of the fixed sequences do not find the best solution in all runs. For the large-size instances, all scenarios, except 1c, have a lower average gap than scenario 0. However, the average gaps between the scenarios only slightly differ. Thus, based on these results, no scenario outperforms the other for all problem sizes.

Table 6. Impact of sequence of local search operators

	Avg. gap (%)		
Scenario	10 orders	15 orders	100 orders
0	0.0000	0.0000	1.3310
1a	0.0007	0.0000	1.2611
1b	0.0000	0.0001	1.2703
1c	0.0003	0.0001	1.3599
2a	0.0007	0.0000	1.1944
2b	0.0007	0.0001	1.2205
3	0.0007	0.0001	1.1974

8. Value of integration

In Moons et al. (2018), the value of integration is examined for small-size instances using CPLEX. In the current study, the RRT heuristic is used to quantify the value of integrating order picking and vehicle routing decisions for large-size instances. Therefore, an uncoordinated approach is compared with an integrated approach. An uncoordinated version of the RRT heuristic is proposed. A picking due date is introduced in the order picking subproblem to separate both subproblems. The picking process of all orders has to be completed before the due date. No order can leave the DC before the due date. The release date of each order in the VRP is equal to the picking due date. Furthermore, a cut-off time is determined, which indicates the latest time in which orders should be requested if they need to be picked up by a vehicle at the due date. All orders placed before the cut-off time are picked before the due date.

The situation described is fully uncoordinated which often occurs in real life when e-commerce companies outsource their delivery operations to a 3PL service provider. The 3PL service provider picks up the customer orders at the DC once a day. All customer orders need to be picked before this moment such that the goods can be loaded onto the vehicles. The exact pickup time is negotiated between both parties, but is fixed for a longer time horizon. A first step towards a more integrated approach can be a flexible pickup time, which is set on a daily basis based on the customer orders of that specific day. An efficient information system is needed to exchange the information between both parties. In the experiments in this section, the fully uncoordinated approach is considered to indicate the impact of integrating both subproblems.

An overview of the uncoordinated version of the RRT heuristic is given in Algorithm 3. The uncoordinated solution method is mainly the same as the integrated version except that the method is divided in two parts. In the first part of the uncoordinated version of the RRT heuristic (line 2-7), an initial solution for the OPP is constructed. Here, the orders are sorted based on their picking time in descending order. Then, the two OPP operators are used to improve the initial solution. The second part of the uncoordinated heuristic algorithm focuses on the vehicle routing subproblem (line 8-13). The picking due date is used as release date for the orders in the VRP. An initial VRP solution is constructed which is afterwards improved using the three VRP operators.

In the experiments with the integrated approach in Section 7, a minimum picking time of two hours (120 minutes) is provided for the order picking operations. Vehicles cannot leave the DC in this time period. Based on this, the delivery time window bounds are determined. The earliest time window bound is calculated using the min-

Algorithm 3 Outline of uncoordinated solution approach

Parameters: $numb\_it$, I_{max}
  $numb\_it$: iteration number
  I_{max}: maximum number of iterations
Solving the OPP problem
Generate initial OPP solution S_0
$numb\_it := 0$
repeat
  Local search within a record-to-record travel framework using two OPP operators
  $numb\_it := numb\_it + 1$
until $numb\_it > I_{max}$
Solving the VRP problem
Generate initial VRP solution S_0
$numb\_it := 0$
repeat
  Local search within a record-to-record travel framework using three VRP operators
  $numb\_it := numb\_it + 1$
until $numb\_it > I_{max}$

imal two-hour picking time, taking into account the loading time and the travel time from the DC to the farthest customer. To compare an uncoordinated approach and an integrated approach, two scenarios are possible for the uncoordinated approach:

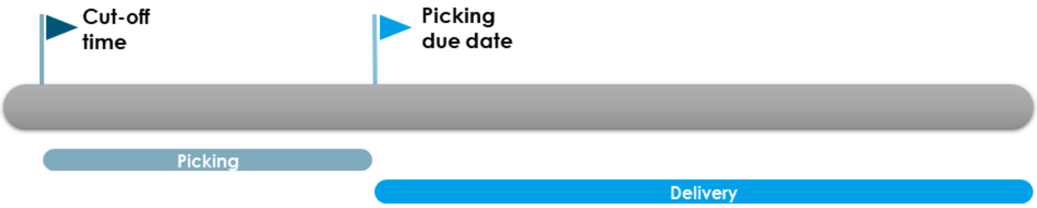
(1) Cut-off time two hours before picking due date: In this scenario, the pickers have two hours to complete all picking operations. Although order pickers are allowed to work four hours within a single shift, there is only a time period of two hours between the cut-off time and the picking due date. Thus, the order pickers cannot work four hours as in the integrated approach. The same number of orders needs to be picked in a shorter amount of time. However, using the same number of order pickers would result in infeasible solutions. With 12 pickers working two hours, a total picking time of 24 hours is available. Nevertheless, all instances generated have a total picking time which is greater than 24 hours. Therefore, in this scenario, the number of pickers available is increased to obtain feasible solutions: 17 regular pickers instead of 9 and 6 temporary pickers instead of 3. Customers can still be delivered within the same time windows as before.

(2) Cut-off time four hours before picking due date: Customers have to request the goods two hours earlier than in the integrated approach in order to receive them within the same time windows. The order pickers can work four hours as in the integrated approach. Alternatively, the delivery time windows can be postponed with two hours compared to the previous experiments. In both approaches, the service level offered to the customer is lower in the uncoordinated approach. The time period between the request of a product and the delivery of the good is extended.

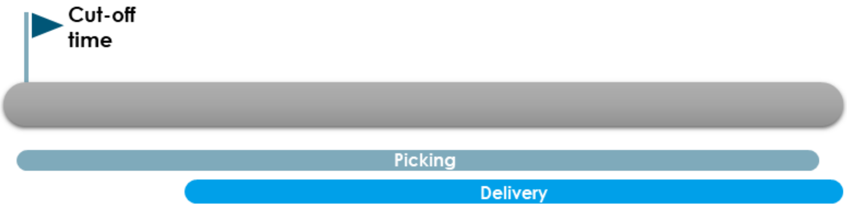
Thus, in scenario 1 additional order pickers are needed to avoid infeasibility, and in scenario 2 the service level offered is decreased.

Figure 2 shows the time line for each approach. The delivery time windows are spread over eight hours in all scenarios. Figure 2(a) indicates the time line for scenario 1 with a two-hour picking time available before the picking due date. In Figure 2(b), scenario 2 is presented with a four-hour time period available before the due date. Finally, the integrated approach is shown in Figure 2(c). The picking operations start at the same moment as in scenario 1, but do not have a due date. The picking of each

(a) Uncoordinated approach - scenario 1



(b) Uncoordinated approach - scenario 2



(c) Integrated approach

Figure 2. Timeline for uncoordinated and integrated approaches

order has to be finished such that the goods can be delivered within it time window selected by the customer.

Similarly as in the previous experiments, 20 replications are conducted for each instance. Each solution obtained by the uncoordinated approach is compared with the best solution found for each instance by the integrated approach. Column 2 of Table 7, shows the difference in total cost, which indicates the value of integration or cost reduction rate (CRR). Columns 3 to 6 present the difference per cost component. A positive percentage indicates that the integrated approach outperforms the uncoordinated approach.

Table 7. Comparison of uncoordinated and integrated approaches

Scenario	CRR	ΔTC_{creg}	ΔTC_{ctemp}	ΔTC_{cd_v}	ΔTC_{ct_v}
(1)	1.84%	0.11%	-12.10%	3.75%	4.29%
(2)	1.83%	0.15%	-14.00%	3.75%	4.29%

CRR = cost reduction rate; TC_{creg} = total regular picking cost; TC_{ctemp} = total temporary picking cost; TC_{cd_v} = total labour cost drivers; TC_{ct_v} = total travel cost

Integrating both problems lead to savings in variable costs of approximately 1.80% on average in both scenarios, with savings up to 5.3%. In the integrated approach, the regular order picking cost is slightly lower, while the labour cost of the temporary order pickers is on average higher in comparison with an uncoordinated approach. Hiring temporary order pickers, which have higher labour costs, can be beneficial if this leads to lower distribution costs. The cost increase in the order picking problem is compensated by cost savings in the vehicle routing problem. Thus, by integrating both problems an overall optimum can be found instead of optimising both problems individually. The impact on the vehicle routing costs is the same in both uncoordinated scenarios since the time windows and customer locations are the same. The average number of vehicles needed, approximately five, does not change when integrating both subproblems.

Apart from these savings in operational costs, the integrated approach offers either a large reduction in fixed costs or a drastic increase in the service level. In scenario (1), the main difference between the uncoordinated and integrated approach is the number of pickers needed. In the uncoordinated approach with a picking due date at 120, the order pickers have two hours each to pick all goods. Thus, the same number of orders have to be picked in a shorter time period. Consequently, a higher number of pickers are needed to pick the same number of orders, i.e., 16 instead of 9. As mentioned before, a higher number of regular pickers is available. Thus, although the changes in the total labour cost of regular pickers are small, hiring new pickers does have an additional cost in real life.

In scenario (2), the total costs does not change significantly, but there is an impact on the service level. Customers have to order their goods two hours earlier to have these delivered in the same time window as in the integrated approach. Consequently, the service level offered to the customers decreases. Thus, by integration, companies can offer their customers the opportunity to purchase goods closer in time to their preferred delivery time using the same number of pickers and vehicles.

Table 8. Computation times of uncoordinated and integrated approaches

Scenario	avg. time (s)
Integrated	106.12
Uncoordinated (1)	84.71
Uncoordinated (2)	87.71

In Table 8, the average computation times of a single run of the algorithm of the integrated and both uncoordinated approaches are indicated. Executing a single run of the RRT algorithm for the integrated approach takes on average 106 seconds, while the algorithms for the uncoordinated approach run for about 85 seconds, independent of the scenario. The slightly higher computation times for the integrated approach may be explained by the need for the procedure to fix violations of the OPP-VRP relationship, described in Algorithm 2. In the uncoordinated approaches, this procedure is never executed as the two subproblems are solved separately. In addition, as the integrated approach has a larger search space and leads to better results, it is conceivable that on average it may take more iterations to reach the maximum number of non-improving iterations for a local search operator. Nevertheless, the small increase in computation time is negligible compared to the benefits which can be obtained by integrating both subproblems.

To conclude, the integration of order picking and vehicle routing operations results on average in a lower total cost. Furthermore, e-commerce companies can allow their customers to purchase goods online later and still have their goods delivered within the same time window without the need of a higher number of pickers or vehicles.

Similar findings were noticed in Moons et al. (2018) for small-size instances using a different objective function. In Moons et al. (2018), costs are incurred for waiting times before the actual start of a vehicle route. In this study, such waiting costs are no longer considered. Due to excluding the waiting costs, average total cost savings are lower in this paper, i.e., 1.80%, than in the study of Moons et al. (2018), i.e., 14%. Thus, even when waiting times are not taken into account, integration is valuable since the service level can be improved and total cost can be decreased.

9. Conclusion and future research

To meet the higher expectations of customers of B2C e-commerce companies, the logistics activities need to be reconsidered. To increase efficiency, the order picking and delivery decisions ideally need to be integrated into a single optimisation problem. This paper proposed a record-to-record travel algorithm to solve an integrated order picking-vehicle routing problem (I-OP-VRP). Five well-known local search operators, such as exchange and relocate, were implemented in a record-to-record travel solution method. The parameters of the heuristic were tuned using an automatic configuration algorithm. Computational experiments were conducted with instances with different problem sizes. For small-size instances for which the optimal solution is known, the record-to-record travel algorithm obtained the optimal solution in all runs conducted in approximately one second. Instances with 100 orders could be solved within 2 minutes.

Furthermore, the value of integration was examined by comparing an uncoordinated and integrated approach. Two different uncoordinated scenarios were compared with the integrated approach. In the uncoordinated approach, a picking due date strictly separated the order picking and delivery operations. Integration has two benefits for e-commerce companies. First, cost savings of on average 1.8% and even up to 5.3% could be obtained by integrating both problems. Total labour costs decreased because a lower number of order pickers were required. Second, e-commerce companies which integrate their operations could offer a higher service level. Customers could request their goods later in time while still be delivered within the same time windows. The time period between the request of an order and the delivery of goods was shortened by integration. Thus, integration can lead to a faster and more cost-efficient picking and delivery which is a competitive advantage in B2C e-commerce business.

The I-OP-VRP considered in this paper is a standard problem which can be extended with more real-world characteristics in further research. First, since in B2C e-commerce DCs a large number of orders need to be handled, often a batch picking policy is applied instead of a discrete order picking policy. Therefore, in future research, the proposed record-to-record travel algorithm may be extended with batch picking in the order picking subproblem. Other research directions could be the integration of other warehouse decisions such as the determination of picking routes and zoning decisions. Second, customers can order goods online 24/7 and the demand is generally not known in advance. The developed solution method should be adapted to be applicable in a dynamic environment. However, the proposed method is efficient and fast, thus can be useful in such a context.

Acknowledgements

The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government - department EWI.

References

Albert Heijn, "Hoe wil je je boodschappen ontvangen? (How do you want your groceries to be delivered?)" n.d. `https://www.ah.nl/kies-moment/bezorgen/1099AM`.

Belo-Filho, M.A.F., Amorim, P., and Almada-Lobo, B. 2015. "An adaptive large neighbourhood search for the operational integrated production and distribution problem of perishable products." *International Journal of Production Research* 53 (20): 6040–6058.

Blauwens, G., De Baere, P., and Van De Voorde, E. 2016 "Cost Calculation in a Transport Firm". In *Transport Economics*, Chap. 5, edited by Blauwens, G., De Baere, P., and Van De Voorde, E. 6th ed. Berchem: De Boeck.

Cárdenas, I., Beckers, J., and Vanelslander, T. 2017. " E-commerce last-mile in Belgium: Developing an external cost delivery index." *Research in Transportation Business & Management* 24: 123–129.

Chen, Z.-L. 2004. "Integrated Production and Distribution Operations: Taxonomy, Models, and Review." In *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-Business Era*, edited by Simchi-Levi, D., Wu, S.D., and Shen, Z.-J. 711–746. Boston: Kluwer Academic Publishers.

Coolblue, "Op afspraak bezorgd (Delivered by appointment)." n.d. `https://www.coolblue.be/nl/klantenservice/2824/informatie-over/op-afspraak-bezorgd.html`.

Croes G.A. 1958. "A Method for Solving Traveling-Salesman Problems." *Operations Research* 6 (6): 791–812.

Du, T.C., Li, E.Y., and Chou, D. 2005. "Dynamic vehicle routing for online B2C delivery." *Omega* 33 (1): 33–45.

Dueck, G. 1993. "New Optimization Heuristics." *Journal of Computational Physics* 104 (1): 86–92.

Ecommerce Europe. 2016. *European B2C E-commerce Report 2016*. Technical Report. Brussels: Ecommerce Europe.

Ecommerce Foundation. 2017. *European Ecommerce Report 2017*. Technical Report. Amsterdam: Ecommerce Foundation.

Gong, Y., and de Koster, R. 2008. "A polling-based dynamic order picking system for online retailers." *IIE Transactions* 40 (11): 1070–1082.

Groër, C., Golden, B., and Wasil, E. 2009. "The Consistent Vehicle Routing Problem." *Manufacturing & Service Operations Management* 11 (4): 630–643.

Interactive Media in Retail Group. 2014. *IMRG UK Consumer Home Delivery Review 2014*.

Li, F., Golden, B., and Wasil, E. 2007. "The open vehicle routing problem: Algorithms, large-scale test problems, and computational results." *Computers & Operations Research* 34 (10): 2918–2930.

Liu, L., Li, K., and Liu, Z. 2017. "A capacitated vehicle routing problem with order available time in e-commerce industry." *Engineering Optimization* 49 (3): 449–465.

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Birattari, M., and Stützle, T. 2016. "The irace package: Iterated racing for automatic algorithm configuration." *Operations Research Perspectives* 3: 43–58.

Low, C., Chang, C.M., and Gao, B.Y. 2017. "Integration of production scheduling and delivery in two echelon supply chain." *International Journal of Systems Science: Operations & Logistics* 4 (2): 122–134.

Low, C., Chang, C.M., Li, R.K., and Huang, C.L. 2014. "Coordination of production scheduling and delivery problems with heterogeneous fleet." *International Journal of Production Economics* 153: 139–148.

Low, C., Li, R.K., and Chang, C.M. 2013. "Integrated scheduling of production and delivery with time windows." *International Journal of Production Research* 51 (3): 897–909.

Moons, S., Ramaekers, K., Caris, A., and Arda, Y. 2017. "Integrating production scheduling and vehicle routing decisions at the operational decision level: A review and discussion." *Computers & Industrial Engineering* 104: 224–245.

Moons, S., Ramaekers, K., Caris, A., and Arda, Y. 2018. "Integration of order picking and vehicle routing in a B2C e-commerce context." *Flexible Services and Manufacturing Journal* 30 (4): 813–843.

Petersen, C.G. 2000. "An Evaluation of Order Picking Policies for Mail Order Companies". *Production and Operations Management* 9 (4): 319–335.

Punakivi, M., and Saranen, J. 2001. "Identifying the success factors in e-grocery home delivery." *International Journal of Retail & Distribution Management* 29 (4): 156–163.

Rosenkrantz, D.J., Stearns, R.E., and Lewis II, P.M. 1977. "An Analysis of Several Heuristics for the Traveling Salesman Problem". *SIAM Journal on Computing* 6: 563–581.

Ruben, R.A., and Jacobs, F.R. 1999. "Batch Construction Heuristics and Storage Assignment Strategies for Walk/Ride and Pick Systems." *Management Science* 45 (4): 575–596.

Savelsbergh, M.W.P. 1992. "The Vehicle Routing Problem with Time Windows: Minimizing Route Duration." *ORSA Journal on Computing* 4 (2): 146–154.

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

Schubert, D., Scholz, A., and Wäscher, G. (2018). "Integrated order picking and vehicle routing with due dates." *OR Spectrum*. Forthcoming.

Solomon, M. M. 1987. "Algorithms for the vehicle routing and scheduling problems with time window constraints." *Operations research* 35: 254–265.

Speranza, M.G. 2018. "Trends in transportation and logistics." *European Journal of Operational Research* 264: 830–836.

Toth, P., Vigo, D., eds. 2014. *Vehicle routing: problems, methods, and applications.* 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics.

Vidal, T., Crainic, T.G., Gendreau, M., and Prins, C. 2015. "Timing problems and algorithms: Time decisions for sequences of activities." *Networks* 65 (2): 102–128.

VIL. 2016. "E-green." *Vlaams Instituut voor de Logistiek.*

Zhang, J., Wang, X., and Huang, K. 2016. "Integrated on-line scheduling of order batching and delivery under B2C e-commerce." *Computers & Industrial Engineering* 94: 280–289.

Zhang, J., Wang, X., and Huang, K. 2017. "On-line scheduling of order picking and delivery with multiple zones and limited vehicle capacity." *Omega* 79: 104–115.