European Journal of Operational Research 293 (2021) 495–510

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Production, Manufacturing, Transportation and Logistics

Optimizing routing and delivery patterns with multi-compartment vehicles

Markus Frank[a], Manuel Ostermeier[b], Andreas Holzapfel[c], Alexander Hübner[b], Heinrich Kuhn[a,*]

[a] Supply Chain Management & Operations, Catholic University of Eichstätt-Ingolstadt, Germany
[b] Supply & Value Chain Management, Technical University of Munich, Germany
[c] Logistics Management, Geisenheim Univeristy, Germany

ARTICLE INFO

Article history:
Received 14 April 2020
Accepted 18 December 2020
Available online 25 December 2020

Keywords:
Supply chain management
Multi-temperature logistics
Transportation
Retailing
Adaptive large neighborhood search

ABSTRACT

Retailers usually apply repetitive weekly delivery patterns when scheduling the workforce for shelf replenishment, defining cyclic transportation routes and managing warehouse capacities. In doing so, all logistics subsystems are jointly scheduled. Grocery products require different temperature zones. As long as transport was in separated vehicles due to temperature requirements, it was not possible to coordinate deliveries across different temperature zones. The recent introduction of multi-compartment trucks has changed this and allows joint deliveries. This simultaneous delivery of multiple product segments impacts repetitive weekly delivery patterns as, for example, low volume segments can be delivered more frequently if they are transported together with high volume segments.

We address the problem of defining delivery patterns for delivery with multi-compartment vehicles. After deriving decision-relevant costs, we propose a novel model that defines the Periodic Multi-Compartment Vehicle Routing Problem. The model is solved by an integrated framework that determines delivery patterns within an Adaptive Large Neighborhood Search in combination with a Large Neighborhood Search for solving the routing problem. We analyze the impact of selecting delivery patterns across product segments and show the efficiency of our integrated planning approach using numerical studies. Joint planning generates cost savings of up to 15%. Furthermore, we show that the algorithm provided can also improve single-segment problems by 3% compared to a state-of-the art benchmark. Beyond that we demonstrate the applicability and advantage of our approach in a case study with a large German grocery retailer.

1. Introduction and motivation

Retailers constantly strive for excellence in logistics due to tight margins, heavy competition and high customer expectations. This may be achieved with new technologies and advanced planning approaches that enable better coordination within and between subsystems in a retail supply chain. Grocery stores exhibit a repetitive sales pattern. Consequently, grocers define store-specific delivery patterns (DP) and cyclically supply stores with their requirements of products and goods. The DPs constitute a defined combination of weekdays on which a store is supplied. They are usually defined as part of the tactical planning for standard weeks without external influences (e.g., public holidays) (Kuhn & Sternbeck, 2013). Defining the DPs also means determining the delivery frequency. The delivery frequency, however, impacts the volume per store delivery, which in turn affects the associated logistics costs in the distribution center (DC), transportation and store. For example, while a larger delivery volume is beneficial for picking and transportation processes due to economies of scale, it is unfavorable for store processes as storage space within stores is usually very limited (Taube & Minner, 2018). Deliveries that do not fit onto the shelves require extra handling and intermediate storage in the backroom (Kotzab & Teller, 2005; Reiner, Teller, & Kotzab, 2013). The resulting trade-off for distribution, warehouse and store costs has to be considered within the planning process (Sternbeck & Kuhn, 2014).

Furthermore, the distribution process in grocery retail involves multi-temperature logistics due to the different temperature re-

* Corresponding author.
E-mail address: heinrich.kuhn@ku.de (H. Kuhn).

https://doi.org/10.1016/j.ejor.2020.12.033

quirements across products. These requirements range right from sub-zero temperatures for the transport of frozen products, lightly cooled products, through to ambient products without any temperature regulation. The exact temperatures are strictly regulated by law and the cooling chain must not be violated during the processing of orders. Retailers usually categorize product segments to different product groupings with similar temperature requirements (e.g., deep-frozen, fresh, dairy, ambient) and organize warehouses by temperature zones. These segments were generally distributed individually in the past. However, multi-compartment vehicles (MCVs) – recently introduced in a flexible version for food transportation – allow multi-temperature transportation. These trucks enable the joint transportation of multiple segments, i.e., products with differing temperature requirements on the same vehicle. The loading area of an MCV can be split flexibly into different compartments for each tour, and the temperature of each compartment can be adjusted individually (Ostermeier, Henke, Hübner, & Wäscher, 2020; Ostermeier & Hübner, 2018). This allows for high flexibility in assigning orders to tours and sequencing the individual routes. MCVs also open up new possibilities for the definition of DPs. The DPs of different segments of a store can be aligned to achieve transportation synergies, which may result in higher delivery frequencies for the store. For instance, frozen products are often delivered once or twice per week due to small order volumes. If combined with other segments (e.g., fresh or ambient products), the frequency can be adjusted to enable more frequent deliveries. The simultaneous supply of different segments reduces the number of stops per route (Hübner & Ostermeier, 2019) and increases the probability that the products delivered could be entirely stacked on the shelf, since greater delivery frequency will decrease the product volume per delivery (Donselaar, Gaur, Woensel, Broekmeulen, & Fransoo, 2010; Reiner et al., 2013; van Zelst, van Donselaar, van Woensel, Broekmeulen, & Fransoo, 2009).

In current literature the definition of DPs differs for each product segment and generally assumes store deliveries with single-compartment vehicles (SCV) (e.g., Gaur and Fisher (2004); Holzapfel, Hübner, Kuhn, and Sternbeck (2016); Sternbeck and Kuhn (2014); Taube and Minner (2018)). The joint delivery of multiple segments is not considered. This raises the question of how the combination of the supply across multiple product segments influences the definition of store-specific DPs for individual segments, and how the altered DPs affect total logistics costs. To address this question, we formulate the Periodic Multi-Compartment Vehicle Routing Problem (PMCVRP), including the definition of DPs and decisions on the corresponding delivery schedules. To further detail the problem, we provide the related problem characteristics and literature in Sections 2 and 3. Section 4 presents the PMCVRP that considers several product segments demanding different temperature zones. The PMCVRP simultaneously decides on (i) the optimal delivery frequency and days for each segment and each store, and (ii) the optimal delivery of the associated store orders with multi-compartment vehicles. The decision model formulated explicitly takes into account the interdependency between delivery frequency and routing decisions. The resulting problem is NP-hard since it is a generalization of the capacitated VRP (Toth & Vigo, 2014), and thus a heuristic solution approach is presented for practice-relevant problem sizes in Section 5. We introduce an approach that iteratively addresses the multi-period problem of defining DPs with an Adaptive Large Neighborhood Search (ALNS) and the corresponding routing problem with MCVs with a Large Neighborhood Search (LNS). To the best of our knowledge, this is the first comprehensive model and solution approach for this problem. Section 6 provides numerical studies, and Section 7 summarizes our findings and refers to future research opportunities.

2. Grocery supply chain, associated processes and costs

2.1. Distribution processes in grocery retailing

Grocery retailers channel about 70% to 90% of shipment volumes to their stores via DCs. Most retailers operate their own vertically integrated logistics network with several central and regional DCs, a vehicle fleet and a large number of local stores to manage (Kuhn & Sternbeck, 2013). Usually between 50 and 400 outlets are served from a single DC (Glatzel, Großpietsch, & Hübner, 2012). In this context, the internal grocery retail supply chain can be divided into three logistics subsystems: DC, transportation and store. The store delivery process can be characterized as follows. The products of a store order are picked onto pallets or role cage in the DC. Next, trucks transport the goods to the stores. Store employees then bring the load carriers to the show room and direct shelf filling takes place (Reiner et al., 2013). If products do not fit onto the shelves, the remaining units are carried to the backroom of the store. Refilling takes place later, when space becomes available due to consumer purchases (Holzapfel et al., 2016; Kotzab & Teller, 2005; Kuhn & Sternbeck, 2013).

Groceries are stored in and transported from DCs to stores in different temperature zones. The specific temperature requirements during storage and transportation are subject to legal regulations. In the European Union, temperatures of -20°C to -18°C for deep-frozen products, $+2^{\circ}$C to $+7^{\circ}$C for cooled products (like meat and dairy products), and $+4^{\circ}$C to $+7^{\circ}$C for fruits and vegetables are mandatory. For some fresh products, retailers apply further product-specific temperature zones to obtain a longer shelf life (e.g., a maximum temperature of $+2^{\circ}$C for fresh fish and seafood). Only ambient products like dry goods and beverages do not need to adhere to specific transportation temperature requirements. Considering the mandated temperature zones plus ambient products, there are at least four different zones in grocery distribution. On the grounds of temperature requirements, retailers store, pick, and prepare the deliveries in temperature-specific DC areas. The traditional approach is to distribute goods separately for each product segment with their specific temperature requirements. Recent truck models are equipped with temperature-specific compartments that allow the transport of different product segments in the corresponding chambers (compartments) of one truck (Ostermeier & Hübner, 2018). For example, when considering deep-frozen and ambient products ordered by the same outlet, the use of such MCVs makes it possible to deliver both product segments on the same truck at the same time. Whereas the transport of the different product segments needs to be planned separately when SCVs with one temperature zone are applied, it becomes necessary to jointly plan flows across segments when MCVs are available. The loading area of an MCV is customized for each tour. Each compartment can be adjusted to a given temperature according to the requirements of loaded product segments. The delivery process with MCVs starts with the collection of orders for all segments assigned to the corresponding tour. Collection involves the approach of multiple shipping gates as each segment is stored in a separate area at the DC (see Fig. 1). After all segments are loaded, the MCV jointly supplies the corresponding stores with the different product segments. Fig. 1 illustrates the overall process of an MCV tour with four segments.

2.2. Selection of delivery patterns

Theoretically, each store could be supplied individually whenever an order is triggered. However, retailers limit the delivery frequency to a certain degree for practical reasons and use weekly delivery cycles. Applying such repetitive and store- and segment-

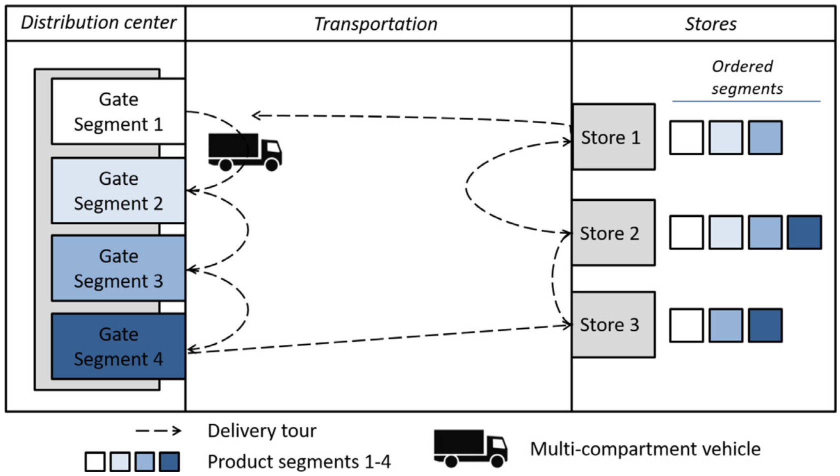

Fig. 1. Distribution process with MCVs.

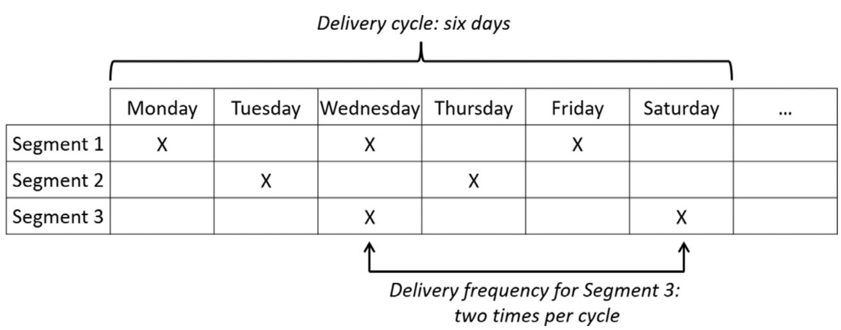

Fig. 2. Example of delivery patterns for a store with three segments.

specific DPs has several reasons. Retailers usually apply periodic inventory review policies (see e.g., Broekmeulen, van Donselaar, Fransoo, and Woensel (2006); Curşeu, van Woensel, Fransoo, van Donselaar, and Broekmeulen (2009); Minner and Transchel (2010)). A replenishment order is issued whenever store inventory falls to or below a reorder level. Such a cyclic ordering policy allows order volumes to be adapted and eases subsequent logistics planning. The orders arrive at a store on identical weekdays each week. Scheduling the workforce for the shelf replenishment process is therefore much easier. Likewise, in terms of transportation, such cyclic ordering and defined delivery days offer the opportunity to design cyclic master routes for each week. At the DC, shift planning can be adjusted with regard to expected picking volumes that are dependent on the delivery frequency determined across all stores (Holzapfel et al., 2016). Finally, retail practice considers the selection of DPs as an important lever to balance DC, transportation and instore requirements (Hübner, Kuhn, & Sternbeck, 2013; Sternbeck & Kuhn, 2014).

Assuming a one-week delivery cycle with six delivery days allows for one to six deliveries per week and store, resulting in $2^6 - 1 = 63$ possible DPs. This yields $63^{|N|}$ possible combinations for $|N|$ stores. The majority of retailers apply store- and segment-specific DPs. This means that individual patterns for each product segment are defined for each store. This is motivated by the fact that both stores (regarding sales volumes and shelf capacity) and segments (regarding freshness requirements) are heterogeneous. As such, the combinatorial challenge increases to $63^{|N|\times|S|}$ possibilities, where $|S|$ indicates the number of product segments. An example for store- and segment-specific DPs is shown in Fig. 2. It illustrates three different DPs for the corresponding segments of a single store.

2.3. Identification of decision-relevant costs and constraints for defining DPs and using MCVs

A DP defines the delivery frequency (e.g., three times per week) and the corresponding delivery days (e.g., each Monday, Wednesday and Friday). It therefore also determines the delivery quantities for each day selected. The daily demand of each store for each segment can be estimated and builds the foundation of the planning. When using daily demand, the weekly seasonality is also incorporated (e.g., higher demand on Saturdays). The demand between order intervals is aggregated to the preceding delivery, e.g., if a delivery happens on Monday and Wednesday, the demand for Tuesday will be fulfilled on Monday. This means that the delivery size of each segment and each store in each period results from selecting a DP. A DP with a higher weekly delivery frequency leads to more but smaller deliveries while the total delivery quantity remains constant. Hence, as order intervals are a result of the DPs applied, volume effects along the supply chain occur that strongly influence operations and costs along the whole grocery supply chain. When applying MCVs, delivery days can be optimized across segments. This also impacts the frequency and size of deliveries of the segments.

The following details the processes, constraints and costs involved in the corresponding subsystems. The analysis is based on our work with a case company and related literature, in particular Sternbeck and Kuhn (2014), Holzapfel et al. (2016) and Hübner and Ostermeier (2019).

Distribution center. The store orders are processed in segment-specific areas of the DC. These areas particularly fulfill the specific temperature requirements of the individual products of the seg-

Subsystem	Distribution Center		Transportation			Store		
Relevant Costs	Order processing costs	Picking & packing costs	Loading costs	Travel costs	Unloading costs	Direct shelf filling costs	Shelf refilling costs	Order placement costs
Impacting Variables	Delivery pattern		Number of compart-ments	Routing		Delivery pattern		
Aggregated Costs	Pattern-dependent costs		Loading costs	Transportation costs		Pattern-dependent costs		
Relevant Constraints	Minimum & maximum picking capacity		Vehicle capacity			Store receiving capacity		

Fig. 3. Decision-relevant operations, distribution costs and constraints along the internal retail supply chain.

ments. The picking volume in each of these areas is generally limited on each working day of the week. In addition, retailers define minimum workload levels for each area to balance the workloads between consecutive working days. This avoids workload peaks and eases shift scheduling. Furthermore, each order causes *order processing costs* at the associated area of the DC. These are fixed costs for each order. A higher delivery frequency therefore leads to an increasing number of orders and higher overall order processing costs at the DC. After the order processing, the respective products are picked and then placed on a load carrier. The entire order is finally packaged and placed at the DC area's gate for loading and transportation. These are variable costs that depend on the delivery size. A higher delivery frequency leads to smaller pick sizes and thus to higher overall *picking and packing costs*.

Transportation. Subsequent to the picking and packing in the DC, the product segments are loaded onto trucks with limited capacity. The associated costs are denoted as *loading costs*. As described above, the distribution process with MCVs requires the collection of segments from different DC areas. These costs depend on the number of segments assigned to a tour and thus on the number of compartments required on the vehicle. For the actual delivery tour, transportation costs arise that involve costs for traveling between the locations and costs for unloading the goods at the stores. The *travel costs* depend on the distance covered by a truck between the locations, i.e., DCs and stores. The transportation costs obviously increase with higher frequency and more tours, but may also decrease if segments are transported jointly across segments. *Unloading costs* occur when a truck stops at a store and unloads the delivery. These costs are induced by setup times for unloading goods from the vehicles and goods receiving processes in the store. The latter include tasks of store employees for checking the items received and complete administrative steps for the goods reception. The resulting costs are fixed costs for each receiving process. The entire unloading costs can therefore be reduced if DPs are synchronized across multiple segments, since this will reduce the number of stops required at the stores.

Store. At each arrival of a new delivery at a store, the orders are further processed. The entire store receiving capacity is generally limited because of space and workforce limitations. This limits the entire volume that can be delivered on a single day. A delivery is either immediately used for direct shelf filling purposes or stored in a backroom until refilling is required. The associated refilling costs are independent of the usage of MCVs. The following costs only depend on the delivery frequency and size.

First of all, *direct shelf filling costs* represent the transport of goods received from the store inbound area to the shelves and putting the units onto the shelves. They are store- and segment-specific and depend on the different settings concerning store layout and shelf types. If the quantity of a product delivered exceeds the shelf capacity, the remaining units have to be brought to the backroom and are stored there until the required capacity is available due to customer purchases and a refill can take place. The related costs are denoted as *shelf refilling costs* from the backroom. This additional refill process is considerably more costly than direct shelf filling. With a higher frequency of deliveries and smaller delivery sizes, this process becomes less frequent and less capacity may be required to store additional units in the backroom that did not fit on the shelves. Finally, new orders are submitted as soon as the reorder level is reached. The reorder level depends on the order size and therefore on the delivery frequency. Each order causes fixed *order placement costs* at the stores. These costs increase with higher delivery frequency.

Summary. Some of the costs above depend on the same decisions and can therefore be summarized in a single cost parameter to streamline the cost model. First of all, a pattern-dependent cost parameter is introduced that comprises costs for order processing and picking and packing at the DC as well as direct shelf filling, shelf refilling, and order placement costs at the store. The DP defines – via the chosen delivery days – the total number of deliveries per period, e.g., per week, and the corresponding delivery sizes of each segment per store delivery, which affect all the cost factors mentioned. Secondly, travel and unloading costs can be summarized, as they both depend on the routing. Fig. 3 summarizes the decision-relevant costs and constraints per subsystem considered in the present paper. In addition it lists the relevant constraints that have to be taken into account.

3. Related literature

The problem considered generally belongs to the class of periodic vehicle routing problems (PVRP). There is a wide range of publications available concerning the PVRP (e.g., Campbell & Wilson (2014)). Classical PVRP literature however neglects several essential characteristics that are relevant when planning DPs and MCVs in grocery retailing. We therefore focus our literature review on publications related to DP and MCV planning.

Literature on DP planning. Publications on DP planning consider – contrary to pure PVRP publications – pattern-dependent costs, and analyze their influences on overall planning. They especially take into account that the delivery sizes per day depend on the DPs chosen. An approach to determine a weekly delivery schedule is provided by Gaur and Fisher (2004) based on a periodic inventory routing problem. Ronen and Goodhart (2008) consider a related problem and include DC costs and additional extensions, such as limited picking capacity, a heterogeneous fleet, and daily minimum utilization rates for DC and transportation subsystems. Similar stores are clustered and patterns are predefined for these clusters using an MIP. Furthermore, a PVRP is applied for the routing. Clustering, pattern-definition and routing is not done sequentially without any feedback loops. This causes the problem that once patterns are assigned to the stores they cannot be changed any more although the routing step may reveal that adapting the delivery patterns could capture additional savings (Holzapfel et al., 2016). In addition, they neglect instore operational costs. Sternbeck and Kuhn (2014) are the first to examine the logistics processes comprehensively in DCs, transportation and stores and their dependencies on DPs. They develop a binary integer program that minimizes the sum of all relevant costs identified and apply it to a real-life case. Transportation costs are approximated with a cost matrix dependent on distance and order size. Actual tours are not considered. Holzapfel et al. (2016) also take into account DC, transportation and instore logistics and propose an advanced solution approach that clusters stores and approximates transportation costs using the logic of Fisher and Jaikumar (1981). Taube and Minner (2018) focus on handling costs at the DC and stores. They consider a classical joint replenishment problem with stochastic demand and present decomposition approaches and a genetic algorithm to solve it. After experimenting with random data instances, they use the most promising model for a case study with a European retailer.

To sum up, this literature stream is related to our problem setting, but falls short in approximating transportation costs without directly solving the related VRP or neglecting instore costs. Furthermore, MCVs and product flows across segments have not been investigated so far.

Literature on the MCVRP. The largest body of MCVRP literature deals with applications in fuel distribution and fixed compartment sizes (e.g., Avella, Boccia, and Sforza (2004); Coelho and Laporte (2015)). Yet in our problem context, the flexibility of compartments is a central characteristic and we therefore focus on related publications. The first comprehensive formulation of a vehicle routing problem with both fixed and flexible compartments is presented by Derigs et al. (2011). The authors use and evaluate a whole string of heuristic solution methods (construction-, search- and metaheuristics) for the problem, while focusing on their application in food and petrol distribution. Henke, Speranza, and Wäscher (2015) discuss an MCVRP with flexible compartments for application in German glass waste collection. A VNS is used to improve an initial solution generated by a randomized construction procedure. Koch, Henke, and Wäscher (2016) and Henke, Speranza, and Wäscher (2019) consider a similar problem formulation but propose different solution approaches. Koch et al. (2016) present a genetic algorithm that may also be modified for a multi-period context and Henke et al. (2019) develop a branch-and-cut algorithm to address the problem. Hübner and Ostermeier (2019) consider an MCVRP in the context of grocery retailing, taking into account MCV-specific costs for the first time. An LNS is applied to solve the corresponding problem. Ostermeier and Hübner (2018) also extend this research and present a vehicle selection model for the MCVRP. Furthermore, Ostermeier, Martins, Amorim, and Hübner (2018) consider the use of flexible compartments and corresponding loading issues. The authors present a mathematical formulation for the extended MCVRP and solve the problem with a branch-and-cut approach as well as an adapted LNS. Besides flexible compartments, Hsiao, Chen, and Chin (2017) also consider the flexible adjustment of compartment temperatures and present a biogeography-based optimization approach. Martins, Ostermeier, Amorim, Hübner, and Almada-Lobo (2019) present an MCVRP for multiple periods that considers consistent deliveries across segments but uses the given DPs as input parameter. They solve the resulting multi-period MCVRP with product-oriented time windows using an ALNS. For a review on MCVRP literature, we further refer to Ostermeier et al. (2020).

To sum up this literature stream, we can state that current MCVRP literature for flexible compartments has been developed only recently. Multi-period problems are rare and – if available – do not consider the assignment of DPs and, particularly, the choice of delivery days.

Summary. Despite the numerous publications on DPs and MCVs, none of the above integrates the diverse problem characteristics mentioned. We address this gap in literature and present a problem formulation that considers the joint selection of DPs across stores and segments in the circumstances of MCV deliveries. Moreover, the interrelation of the joint DP selection of product segments on warehousing, MCV routing, and store operations requires an integrative planning approach. Only the simultaneous consideration of all decision-relevant costs and constraints ensures feasible and cost-optimal decisions for the entire distribution process. As such, our work extends the literature on both DP and MCV planning in grocery distribution. Our research specifically makes a contribution to:

- Identifying decision-relevant costs in warehousing, transportation and instore operations when selecting DPs across product segments and using MCVs for their joint store deliveries;
- Formulating a novel model, i.e., the PMCVRP that simultaneously defines cost-minimal DPs and MCV delivery tours for a diverse set of product segments in a multiple period environment;
- Developing a sophisticated heuristic solution algorithm that finds good solutions in acceptable computation times for the defined PMCVRP model, and
- Generating numerical examples with simulated and actual retail data to obtain insights into the value of integration when DPs of products with different temperature requirements are jointly delivered from the DCs to the stores.

4. Decision model

In the present section we formulate the mathematical model of the decision problem described. The model is based on the PVRP where transportation is executed by MCVs, and DC-related, transportation and store-related costs are considered that depend on the DP chosen. We denote this problem as PMCVRP. The model is formulated as follows using the notation given in Table 1.

Let $G = (N_0, E)$ be an undirected, weighted graph consisting of a vertex set $N_0 = \{0, 1, \ldots, |N|\}$, representing the location of the depot (0) and the locations of stores ($N = \{1, \ldots, |N|\}$), and a set of edges $E = \{(i, j) : i, j \in N_0\}$, representing the connection between different locations. Each edge is associated with non-negative travel costs c_{ij}^{travel}. The product segments are denoted by the set $S = \{1, \ldots, |S|\}$. The stores are supplied using a heterogeneous fleet of MCVs denoted by the set of vehicles $K = \{1, \ldots, |K|\}$. The vehicle fleet is assumed to be sufficiently large to satisfy the total demand. The compartment setting for each MCV is adjustable, i.e., the number and size of compartments is not predetermined but part of the decision problem. Further, the total vehicle capacity Q^{veh} is not affected by the specific compartment setting due to the given flexibility. Each MCV is used for one tour per period at most. The planning horizon comprises one delivery cycle with a given set of delivery periods, $T = \{1, \ldots, |T|\}$. Further, a set of possible delivery patterns $P = \{1, \ldots, |P|\}$ is introduced that covers all possible store- and segment-specific delivery schedules. Generally, this set can include all feasible weekday combinations of all frequencies, but a prior limitation, e.g., dependent on certain segment or store features, is reasonable.

Every store has a positive demand for each segment across the planning horizon. The delivery quantity o_{psit} indicates the demand of store i for product segment s at period t when DP p is selected.

Table 1
Notation used to model PMCVRP.

Sets	
K	Set of vehicles $K = \{1, ..., \|K\|\}$
N	Set of stores $N = \{1, ..., \|N\|\}$, $N_0 = \{0, 1, ..., \|N\|\}$ with 0 as depot
P	Set of delivery patterns $P = \{1, ..., \|P\|\}$
S	Set of product segments $S = \{1, ..., \|S\|\}$
T	Set of periods $T = \{1, ..., \|T\|\}$
Parameters	
a_{pt}	$a_{pt} = 1$, if period t, $t \in T$, is included in pattern p, $p \in P$, 0 otherwise
c_s^{load}	Loading costs for segment s, $s \in S$
c_{ij}^{tran}	Transportation costs for approaching location j after i for $i, j \in N_0$, where $c_{ij}^{\text{tran}} = c_{ij}^{\text{travel}} + c_j^{\text{unload}}$
c_{psi}^{pat}	Pattern-dependent costs of segment s, $s \in S$, and store i, $i \in N$, when pattern p, $p \in P$, is selected
o_{psit}	Delivery quantity of segment s, $s \in S$, for store i, $i \in N$, in period t, $t \in T$, when pattern p, $p \in P$, is selected
Q^{veh}	Vehicle capacity
Q_s^{pickmin}	Minimum picking capacity for segment s, $s \in S$, at the DC
Q_s^{pickmax}	Maximum picking capacity for segment s, $s \in S$, at the DC
Q_i^{recmax}	Maximum receiving capacity for store i, $i \in N$
Decision and auxiliary variables	
u_{skt}	Binary; indicating whether segment s, $s \in S$, is delivered by vehicle k, $k \in K$, in period t, $t \in T$
x_{ijkt}	Binary; indicating whether vehicle k, $k \in K$, travels from location i to j for $i, j \in N_0$, in period t, $t \in T$
y_{sikt}	Binary; indicating whether store i, $i \in N$, receives segment s, $s \in S$, in period t, $t \in T$, by vehicle k, $k \in K$
z_{psi}	Binary; indicating whether pattern p, $p \in P$, is selected for segment s, $s \in S$, and store i, $i \in N$

It depends on the pattern assigned, and is part of the decision problem. The total demand of a segment is split across actual days of delivery according to the chosen DP. The quantity of a delivery must include the demand of the period during which the delivery takes place and the demand of all following periods until the next scheduled delivery. In line with this, the parameter a_{pt} indicates whether period t is included in pattern p or not. Moreover, single orders for one segment in one period may not be split up across different vehicles.

The cost parameters are defined as follows. The loading costs c_s^{load} represent the costs for stopping at a segment-specific gate at the depot and for loading the order onto the truck. Travel costs c_{ij}^{travel} include the costs for the travel from location i to location j. Unloading costs c_j^{unload} cover the costs for each stop at a store. For the sake of simplicity, we summarize travel and unloading costs in a generalized cost term for transportation, and define $c_{ij}^{\text{tran}} := c_{ij}^{\text{travel}} + c_j^{\text{unload}}$. Finally, the pattern-dependent costs c_{psi}^{pat} indicate the costs that occur when store i is supplied with segment s according to pattern p. The pattern-dependent costs comprise both depot- and store-specific handling costs as described in Section 2. The following binary decision variables are applied:

- x_{ijkt} indicates whether vehicle k travels from location i to j within period t, $k \in K, i, j \in N_0, t \in T$.
- y_{sikt} indicates whether store i receives segment s by vehicle k within period t, $s \in S, i \in N, k \in K, t \in T$.
- z_{psi} indicates whether pattern p is selected for segment s and store i, $p \in P, s \in S, i \in N$.

Additionally, we introduce the auxiliary binary variables u_{skt} indicating if vehicle k contains at least one order of segment s on day t. The mathematical model for the PMCVRP can then be formulated as follows.

$$min \quad TC = \sum_{s\in S}\sum_{k\in K}\sum_{t\in T} c_s^{\text{load}} \cdot u_{skt} + \sum_{i\in N_0}\sum_{\substack{j\in N_0\\ i\neq j}}\sum_{k\in K}\sum_{t\in T} c_{ij}^{\text{tran}} \cdot x_{ijkt} + \sum_{p\in P}\sum_{s\in S}\sum_{i\in N} c_{psi}^{\text{pat}} \cdot z_{psi} \tag{1}$$

subject to

$$\sum_{p\in P} z_{psi} = 1 \qquad \forall s \in S, \forall i \in N \tag{2}$$

$$\sum_{\substack{i\in N_0\\ i\neq j}} x_{ijkt} = \sum_{\substack{i\in N_0\\ i\neq j}} x_{jikt} \qquad \forall k \in K, \forall t \in T, \forall j \in N_0 \tag{3}$$

$$\sum_{k\in K}\sum_{j\in N_0} x_{0jkt} \leq |K| \qquad \forall t \in T \tag{4}$$

$$\sum_{s\in S} y_{sjkt} \leq |S| \cdot \sum_{i\in N_0} x_{ijkt} \qquad \forall j \in N, \forall t \in T, \forall k \in K \tag{5}$$

$$\sum_{i\in L}\sum_{j\in L} x_{ijkt} \leq |L| - 1 \qquad \forall t \in T, \forall k \in K, \forall L \subseteq N, |L| \geq 2 \tag{6}$$

$$\sum_{j\in N_0} x_{0jkt} \leq 1 \qquad \forall t \in T, \forall k \in K \tag{7}$$

$$\sum_{p\in P}\sum_{s\in S}\sum_{i\in N} o_{psit} \cdot z_{psi} \cdot y_{sikt} \leq Q^{\text{veh}} \qquad \forall t \in T, \forall k \in K \tag{8}$$

$$Q_s^{\text{pickmin}} \leq \sum_{p\in P}\sum_{i\in N} o_{psit} \cdot z_{psi} \leq Q_s^{\text{pickmax}} \qquad \forall s \in S, \forall t \in T \tag{9}$$

$$\sum_{p\in P}\sum_{s\in S} o_{psit} \cdot z_{psi} \leq Q_i^{\text{recmax}} \qquad \forall i \in N, \forall t \in T \tag{10}$$

$$\sum_{k\in K} y_{sikt} = \sum_{p\in P} z_{psi} \cdot a_{pt} \qquad \forall s \in S, \forall i \in N, \forall t \in T \tag{11}$$

$$\sum_{i\in N} y_{sikt} \leq u_{skt} \cdot |N| \qquad \forall s \in S, \forall k \in K, \forall t \in T \tag{12}$$

$$u_{skt} \in \{0, 1\} \qquad \forall s \in S, \forall k \in K, \forall t \in T \tag{13}$$

$$x_{ijkt} \in \{0, 1\} \qquad \forall i, j \in N_0, \forall k \in K, \forall t \in T \tag{14}$$

$$y_{sikt} \in \{0, 1\} \qquad \forall s \in S, \forall i \in N, \forall k \in K, \forall t \in T \tag{15}$$

$$z_{psi} \in \{0, 1\} \qquad \forall p \in P, \forall s \in S, \forall i \in N \qquad (16)$$

The objective function (1) minimizes the total costs (TC), consisting of loading, transportation (including unloading), and pattern-dependent costs that arise for every pattern that is assigned to a segment-store combination (s, i), $s \in S, i \in N$. Constraints (2) ensure that exactly one delivery pattern per product segment is assigned for each store. Constraints (3) represent the flow conservation, guaranteeing that every store visited is also left again. Additionally, each vehicle has to start from the depot as defined by Constraints (4). Constraints (5) guarantee that a store is visited if a corresponding order is loaded. The subtour elimination constraints are denoted by Constraints (6). According to Constraints (7), every vehicle may be used only once per day. Constraints (8) ensure that the vehicle capacity is not exceeded. Constraints (9) ensure that the picking effort at each segment-specific DC area neither falls below the minimum nor exceeds the maximum picking capacity on each day t. Constraints (10) consider the maximum receiving capacity of each store i on each delivery day t. Constraints (11) ensure that if a store receives a product segment on day t according to the selected DP, the corresponding segment has to be assigned to a vehicle on this delivery day. Further, if at least one order of segment s is assigned to vehicle k, the corresponding compartment is required and thus, u_{skt} is activated (Constraints (12)). Lastly, the decision and auxiliary variables are defined as binary by Constraints (13)–(16).

The PMCVRP extends both PVRP and MCVRP. As such, it generalizes the well-known CVRP that is known to be an NP-hard optimization problem (see e.g., Laporte (2009); Toth and Vigo (2014)). Exact solution approaches are only able to solve small problem instances. In our application we consider industry cases with hundreds of stores that are served from temperature-specific DCs with a diverse set of product segments. In these cases, heuristics are required to provide solutions for the PMCVRP.

5. Solution approach

We propose a heuristic algorithm to solve the PMCVRP. The algorithm iteratively optimizes the assignment of DPs for each segment-store combination and solves the corresponding MCVRP in each period of the planning horizon. Fig. 4 illustrates the general framework of the algorithm proposed. It contains three major parts that are described in more detail within the upcoming section. After generating an initial solution (see Section 5.1), the algorithm performs two sequential stages within its second part (see Section 5.2). In Stage 1, an ALNS framework is used to determine individual DPs for each segment-store combination (see Section 5.2.1). This results in new partial solutions that define the delivery quantities for each period and each segment-store combination of the entire planning horizon. Stage 2 then solves the resulting MCVRPs applying an LNS approach in each period of the planning horizon (see Section 5.2.2). The ALNS optimizes delivery patterns across all periods, whereas the LNS only optimizes the routing within a period. Finally a Simulated Annealing approach is used to decide on the next candidate schedule to work on during the subsequent iteration. This part of the algorithm also adapts the parameters of the search process (Section 5.3).

Please note that we use the following terminology within the detailed description of the algorithm. Deliveries are set by segment and store and are therefore uniquely defined for each segment-store combination subject to the assigned patterns. We will therefore use the term “segment-store combination” (s, i), $s \in S, i \in N$ to uniquely define the object and planning entity. We use this term whenever we consider the characteristics attributed to a store and the corresponding segment (e.g., weekly demand for segment s by store i). Furthermore, a DP p is assigned to each segment-store combination, indicated by the triple (p, s, i), $p \in P, s \in S, i \in N$.

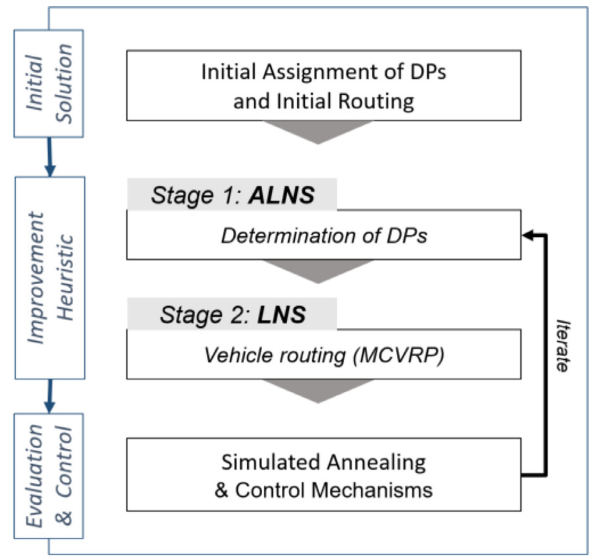

Fig. 4. Algorithmic structure.

5.1. Initial solution

The solution approach starts with a random initial assignment of patterns. This assignment specifies the delivery days and the associated delivery quantities for each segment-store combination. It also determines the entire delivery requirements in each period. Afterwards, the procedure applies the Savings Algorithm by Clarke and Wright (1964) to construct feasible delivery tours in each period. The Savings Algorithm was used in many VRP and MCVRP formulations and provides fast and reasonable initial solutions (Toth & Vigo, 2014). We use the parallel version of the algorithm as it provides better solutions than the sequential approach (Laporte, 2009). The procedure starts constructing single tours for every segment-store delivery scheduled in each period of the planning horizon. Afterwards we calculate the associated savings values for all pairs of segment-store combinations $[(s, i), (\tilde{s}, j)]$, $s, \tilde{s} \in S, i, j \in N$, if they were jointly delivered: $\Delta c_{si,\tilde{s}j} = c^{\text{travel}}_{i0} + c^{\text{travel}}_{0j} - c^{\text{travel}}_{ij}$. Here, 0 represents the depot and c^{travel}_{ij} denotes the travel costs between the respective locations i and j $(i, j \in N)$. Iterating across the sorted decreasing list of savings-values $\Delta c_{si,\tilde{s}j}$, the corresponding tours of the pairs (s, i) and $(\tilde{s}, j)$ are merged if feasible.

5.2. Improvement heuristic

In each iteration of our solution approach we change large parts of the current solution by assigning new delivery patterns to segment-store combinations (*Stage 1*) and solving the corresponding routing problems (*Stage 2*). Please note that we are keeping track of all partial solutions created (*Stage 1*), whereas the routing for each period (*Stage 2*) is only executed if a new solution (i.e., assignment of patterns) has been reached. In addition, we verify the feasibility of each pattern assignment in respect of the lower and upper picking limits at the DC and the store’s maximum receiving capacity. In the event that it is not feasible, the partial solution is rejected and *Stage 1* is repeated. In the following both stages are described in detail.

Table 2
Operators used within the ALNS algorithm.

Operator h	Operator type		Operator name
1	(i)	Score-related	Proximity Operator
2			Segment Bundle Operator
3			Sales Volume Operator
4	(ii)	Cost-related	Pattern-dependent Cost Operator
5	(iii)	Move-related	Move-One Operator
6			Move-Two Operator
7	(iv)	Random	Random Operator

5.2.1. Stage 1: ALNS for optimizing DPs

The ALNS approach introduced by Shaw (1997) was effectively applied when solving multiple variants of the VRP (e.g., Shaw (1997), Ropke and Pisinger (2006)), and it was particularly effective when solving PVRPs (e.g., Zajac (2017)) and MCVRPs (e.g., Martins et al. (2019)). On the basis of an initial solution, it typically uses several remove and insertion operators to destroy and repair large parts of the solution in each iteration. In order to adapt the ALNS approach to the present problem and to the state of the search process, a weight is assigned to each operator that determines how often it is selected during the search process. The weights are adjusted dynamically depending on the past performance of the respective operator with respect to the overall solution. We describe this adjustment procedure in detail in Section 5.3. We use the ALNS as it enables us to embed a whole set of operators derived from and built with problem-specific knowledge. Further, the adaptive mechanism decides which operators to use for which type of problem instance.

Most ALNS approaches include remove and reinsertion operators for customers or orders to recreate large parts of the solution. In this aspect, our ALNS differs from other formulations. Rather than removing and reinserting orders or deliveries, the operators used in our approach select new patterns for segment-store combinations, i.e., they decide on how often and on which days a store receives the respective segments. Traditional remove and insert operators are not applicable for our problem as these usually assume that the order sizes per customer and period are independent of the solution. In our case, however, the order size per store delivery depends on the chosen DP. For example, modifying a current DP will at least omit, add or move one delivery day. Consequently, this will change the delivery size and the related costs of the associated deliveries since we assume a pre-defined weekly demand pattern for all segment-store combinations. Modifying an individual pattern of a segment-store combination may therefore result in a completely new delivery schedule. Table 2 summarizes the ALNS operators applied. It comprises the following operator types: (i) score-related, (ii) cost-related, (iii) move-related and (iv) random. Each operator will change the patterns in an iteration for a given number of segment-store combinations.

(i) Score-related operators. The structure of our score-related operators is based on the well-known Removal Operator by Shaw (Shaw, 1997). For each score-related operator we define a relatedness measure $R_{si,\tilde{s}j}$ for two segment-store combinations (s,i) and $(\tilde{s},j)$, $i,j \in N$, $s,\tilde{s} \in S$. The algorithmic structure of the score-related operators is identical despite the different relatedness measures. We therefore present this general five-step structure first and then detail the individual operators. The general structure of the score-related operators is further given in Algorithm 1.

After a random segment-store combination (s,i) has been selected in *Step 1*, the relatedness $R_{si,\tilde{s}j}$ between delivery schedules for segment-store combination (s,i) and all other combinations $(\tilde{s},j)$, $\tilde{s} \in S, j \in N$, is calculated in *Step 2* and ranked in ascending order according to the relatedness measure calculated, $R_{si,\tilde{s}j}$. The more related the attributes of two segment-store combinations, the more likely it is to obtain synergies in a joint consideration and the higher the expected additional cost savings by aligning the respective patterns. Subsequent to the relatedness calculated, an additional parameter α is used that determines the degree of randomization of the search. More precisely, after sorting all segment-store combinations $(\tilde{s},j)$ according to their score, a random number ζ, $\zeta \in [0,1]$ is drawn in *Step 3*, selecting the combination that lays ζ^{α} down the ranking. If the combination selected $(\tilde{s},j)$ has a different pattern p', $p' \in P$ compared to the combination (s,i), a new pattern is assigned to combination $(\tilde{s},j)$ in *Step 4*. Here, the new pattern is chosen randomly among all patterns $\tilde{p}$, $\tilde{p} \in P$ that have a higher pattern similarity ($\omega_{psi,\tilde{p}\tilde{s}j}$) than the previously assigned pattern. The pattern similarity is calculated using Eq. (17). This metric is determined by the ratio of matching periods $\eta^{t}_{psi,\tilde{p}\tilde{s}j}$ for pattern p and $\tilde{p}$, to the total number of periods $|T|$. Matching periods are days where both patterns intend to carry out a store delivery.

Algorithm 1 Score-related operators.

Input: Solution $\mathbb{S}$, set of patterns P, number of segment-store combinations to be changed c, degree of randomization α
List $L = \emptyset$
Randomly select a segment-store combination (s,i), $s \in S, i \in N$ and add it to list L
while $|L| < c$ **do**
  Step 1: randomly select a segment-store combination from list L, $(s,i) \in L, s \in S, i \in N$
  Step 2: compute score $R_{si,\tilde{s}j}$ for all segment-store combinations $(\tilde{s},j) \notin L, \tilde{s} \in S, j \in N$, and sort by their score in ascending order
  Step 3: draw a random number $\zeta \in [0,1]$ and select the segment-store combination $(\tilde{s},j)$, $\tilde{s} \in S, j \in N$, that lays ζ^{α} down the ranking
  if pattern p assigned to (s,i) differs from pattern p' assigned to $(\tilde{s},j)$ (i.e., $p \neq p'$, $p,p' \in P$), **then**
    Step 4: assign new pattern to segment-store combination $(\tilde{s},j)$ that is selected randomly among all patterns $\tilde{p}$, $\tilde{p} \in P$, with a higher *pattern similarity* $\omega_{psi,\tilde{p}\tilde{s}j}$ to the current pattern p of (s,i)
  else
    continue;
  end if
  Step 5: add $(\tilde{s},j)$ to L
end while
return new partial solution $\mathbb{S}^*$ with the updated pattern assignments for all segment-store combinations considered (s,i), $s \in S, i \in N$.

$$\omega_{psi,\tilde{p}\tilde{s}j} = \frac{\sum_{t=1}^{|T|} \eta^{t}_{psi,\tilde{p}\tilde{s}j}}{|T|} \tag{17}$$

This process is repeated until the patterns for c segment-store combinations are adjusted. The resulting new solution $\mathbb{S}^*$ is then the input for solving the MCVRPs in *Stage 2*.

Proximity Operator. The first score-related operator is based on the idea that it is usually cost-efficient to serve stores in geographical proximity using the same vehicle. In order to enable a conjoint delivery in each period of the planning horizon, the patterns of these segment-store combinations should be as similar as possible. If two neighboring stores and their segments or two segments of the same store share the same delivery days (but only if this is the case), they should be placed on the same tours by the subsequent routing decision and transportation synergies can be realized. To do this, the Proximity Operator tries to assimilate the patterns of segment-store combinations by selecting combinations (s, i) and $(\tilde{s}, j)$ with a lower value $R^1_{si,\tilde{s}j}$ subject to the patterns p and $\tilde{p}$ currently selected.

$$R^1_{si,\tilde{s}j} = \beta \cdot \frac{c^{\text{travel}}_{ij}}{\bar{c}^{\text{travel}}} + (1-\beta) \cdot \omega_{psi,\tilde{p}\tilde{s}j} \tag{18}$$

$R^1_{si,\tilde{s}j}$ comprises two components that are weighted by β: the geographical proximity of the respective stores i and j and the current similarity of their patterns p and $\tilde{p}$. The first metric is expressed by the travel costs c^{travel}_{ij} between the store locations of i and j and the maximum travel costs between any two store combinations $\bar{c}^{\text{travel}}$. Second, the pattern similarity $\omega_{psi,\tilde{p}\tilde{s}j}$ (see Eq. (17)) of the current patterns of the segment-store combinations considered is included as we aim at changing combinations with dissimilar patterns.

Segment Bundle Operator. A further operator bundles deliveries across segments. Scheduling deliveries from different stores but the same segment in the same period can lead to savings on loading costs by decreasing the number of loading gates a single MCV has to approach at the depot. However, savings on loading costs may be exceeded by additional travel costs that arise if deliveries from the same segment but not from the same delivery area are placed on a single tour. To avoid this, the Segment Bundle Operator tries to assimilate patterns of segment-store combinations that are located in the same neighborhood and concern identical segments. It combines the metric for the travel costs between segment-store combinations with the segment similarity. The segment similarity, defined as $\sigma_{si,\tilde{s}j}$, indicates whether the segments considered are equal or not:

$$\sigma_{si,\tilde{s}j} = \begin{cases} 0, & \text{if } s = \tilde{s} \\ 1, & otherwise \end{cases} \tag{19}$$

The relatedness measure $R^2_{si,\tilde{s}j}$ for this operator is denoted by Eq. (20). Again, we use a weight δ for the two components of $R^2_{si,\tilde{s}j}$. To achieve the desired effect, the weights here are to be chosen differently to the Proximity Operator, with a stronger focus on the similarity of segments than on the relative travel costs of the corresponding stores.

$$R^2_{si,\tilde{s}j} = \delta \cdot \frac{c^{\text{travel}}_{ij}}{\bar{c}^{\text{travel}}} + (1-\delta) \cdot \sigma_{si,\tilde{s}j} \tag{20}$$

Sales Volume Operator. The third score-related operator is based on the overall segment-specific demand of a store for the entire planning period. This total demand is denoted by Ψ_{si}, $s \in S, i \in N$. It may be favorable to align deliveries for stores with comparable demand since we consider heterogeneous stores with different store sizes and sales volumes for each segment. The operator offers the option to copy cost-efficient patterns already found for a segment-store combination to another store with a similar demand structure. Unlike the first two operators, it solely aims to reduce pattern-dependent costs and hence supports the diversification of our search algorithm. Again, the more similar the total demands, the lower the calculated $R^3_{si,\tilde{s}j}$ (see Eq. (21)). This represents the absolute difference of the total demand of the combinations (s, i) and $(\tilde{s}, j)$.

$$R^3_{si,\tilde{s}j} = |\Psi_{si} - \Psi_{\tilde{s}j}| \tag{21}$$

(ii) Cost-related Operator. Cost-intensive patterns may be assigned in the course of the ALNS as they may be favorable in terms of transportation costs. Yet depending on the problem instance's cost structure, the pattern-dependent costs may exceed the savings achieved. Moreover, specific delivery period combinations or delivery frequencies can be extraordinarily costly. We therefore introduce an operator directly focusing on pattern-dependent costs that counteracts this effect. The operator therefore considers the cost of a chosen pattern p, $p \in P$, for a segment-store combination (s, i) and aims to find a new pattern p', $p' \in P : p' \neq p$, with lower pattern-dependent costs. Algorithm 2 presents the algorithmic structure of this operator. Please note that Algorithm 2 is similar to Algorithm 1 but not identical. The pattern-dependent cost operator directly considers the cost delta between the current, c^{pat}_{psi}, and the minimum pattern-dependent costs, $c^{\text{pat}}_{\underline{p}si}$ of a pattern-segment-store combination (p, s, i), $\Delta c^{\text{pat}}_{psi} = c^{\text{pat}}_{psi} - c^{\text{pat}}_{\underline{p}si}$. DP $\underline{p}$, leading to the minimum pattern-dependent costs, is determined as follows: $\underline{p} = \arg\min_{p \in P} [\, c^{\text{pat}}_{psi} \,]$. The Pattern-dependent Cost Operator evaluates whether a pattern that results in lower costs is available. In *Step 2* of Algorithm 2 we do not select the segment-store combination that promises the highest savings, but the combination that lays ζ^{α} down the descending cost ranking (cf. score-related operators). If the minimum pattern-dependent costs during an iteration are not yet reached, a new pattern p' is selected randomly among the set of all patterns P, implying lower pattern-dependent costs. Due to its nature, the Pattern-dependent Cost Operator can be seen as a regulatory operator. For the overall problem it is not effective to select the patterns with minimum costs for each segment-store combination as this would usually lead to high transportation costs. Yet the pattern-dependent costs are one main driver of a successful search, and low-priced patterns have to be utilized in different combinations.

Algorithm 2 Pattern-dependent Cost Operator.

```
Input: Solution 𝕊, set of patterns P, number of segment-store combinations to be changed c, pattern-dependent costs of all combinations c^pat_psi, degree of randomization α, list O of all segment-store combinations (s, i), s ∈ S, i ∈ N
List L = ∅
while |L| < c do
    Step 1: sort all segment-store combinations (s, i) ∈ O by c^pat_psi in descending order
    Step 2: draw a random number ζ ∈ [0, 1] and select the segment-store combination (s, i), that lays ζ^α down the ranking
    if pattern-dependent costs of current pattern p of segment-store combination (s, i) do not equal c^pat_(p_)si, then
    Step 3: assign a new pattern p', p' ∈ P, to segment-store combination (s, i), which is selected randomly among all patterns P with lower c^pat_p'si than the current pattern of segment-store combination (s, i)
    else
        continue;
    end if
    Step 4: remove (s, i) from O, add (s, i) to L
end while
forall (s, i) ∈ O do
    keep the assigned pattern p, p ∈ P
return new partial solution 𝕊* with the updated pattern assignments for all segment-store combinations considered (s, i), s ∈ S, i ∈ N.
```

(iii) Move-related operators. Two move-related operators are introduced to increase the diversification of the search algorithm as they do not use a specific search criterion but randomly choose orders to be considered for moves. A move, in this context, is the change of delivery periods while maintaining the identical number of deliveries per week. For example, if a store receives a product segment on day 1 and 2, the Move-One Operator moves one and only one of these delivery days to another period where currently a delivery is not intended. So the DP may change to deliveries on day 2 and day 5. The DP featuring a delivery frequency of $|T|$ times within the planning horizon stays unconsidered since none of the delivery days scheduled can be moved. The pseudo-code of the Move-One Operator is denoted in Algorithm 3.

Algorithm 3 Move-One Operator.

```
Input: Solution 𝕊, set of patterns P, number of segment-store combinations to be changed c, list O of all segment-store combinations (s, i), s ∈ S, i ∈ N
List L = ∅
while |L| < c do
    Step 1: randomly select a segment-store combination (s, i) ∈ O
    if the frequency of the current pattern p of (s, i) is unequal |T|, then
    Step 2: randomly select one delivery period t_1, t_1 ∈ {T | a_{pt_1} = 1} of the current pattern p of segment-store combination (s, i)
    Step 3: move the delivery from period t_1 to period t_2, t_2 ∈ {T | a_{pt_2} = 0} in the current pattern p of (s, i) and assign the resulting pattern p' to (s, i)
    end if
    Step 4: remove (s, i) from O, add (s, i) to L
end while
forall (s, i) ∈ O do
    keep the assigned pattern p, p ∈ P
return new partial solution 𝕊* with the updated pattern assignments for all segment-store combinations considered (s, i), s ∈ S, i ∈ N.
```

The Move-Two Operator moves two delivery periods planned to two delivery periods not yet scheduled. The DPs that intend deliveries on $|T|$, $|T|-1$ or 1 periods of the planning horizon are not considered since no feasible moves exist for these patterns. We use the Move-One Operator in those cases, except when the delivery frequency equals $|T|$.

(iv) Random Operator. The Random Operator is introduced as an additional diversification operator that also changes the patterns. It randomly selects segment-store combinations and randomly assigns a new pattern from all possible patterns.

5.2.2. Stage 2: LNS for solving the routing in each period

Stage 2 within the improvement phase of the entire algorithm addresses the routing problem. This stage solves the MCVRP for each period t, $t \in T$, assuming the DPs selected in Stage 1. We apply the LNS framework suggested by Hübner and Ostermeier (2019) for solving the MCVRP since they assume an equivalent cost structure to ours. In addition, the LNS approach enables high-quality solutions in short computation times (see also Derigs et al. (2011)) that are particularly relevant in our case since we need to solve the MCVRP in each iteration for each period of the planning horizon. The LNS approach of Hübner and Ostermeier (2019) uses the Savings Algorithm by Clarke and Wright (1964) to generate an initial solution for the routing in each period. This is conducted equivalently as described in Section 5.1. Based on the initial solution, Shaw Removal and Regret-k Insertion are used within the LNS. The Shaw Removal is based on Shaw (1997), but is modified to consider the joint delivery of multiple segments. In the present implementation of the LNS we further adapt the Shaw Removal operator to account for the dynamic structure of the delivery sizes. The algorithmic structure presented in Algorithm 1 resembles the structure of Shaw Removal as it also serves as the basis for our score-related operators. The associated relatedness measure is given in Eq. (22).

$$R^S_{si,\tilde{s}j} = \mu \cdot \frac{c^{tran}_{ij}}{\bar{c}^{tran}} + \nu \cdot \sigma_{si,\tilde{s}j} + \xi \cdot \frac{|o_{psit} - o_{\tilde{p}\tilde{s}jt}|}{o^{max}} \tag{22}$$

$R^S_{si,\tilde{s}j}$ takes three metrics into account: transportation costs, product segment and delivery sizes, weighted by μ, ν, and ξ, respectively. The metrics for transportation costs and segment similarity are identical to those used in the Segment Bundle Operator. Additionally, the delivery size is included as swapping deliveries with similar sizes result in the faster generation of new feasible solutions. Delivery sizes are compared using the difference in size of deliveries o_{psit} and $o_{\tilde{p}\tilde{s}jt}$ (for the given delivery period t and the corresponding patterns p and $\tilde{p}$) in relation to the highest delivery quantity across all deliveries (o^{max}). After the defined number of deliveries has been removed (see Algorithm 1), removed deliveries are reinserted applying Regret-k Insertion (Ropke & Pisinger, 2006). It calculates the regret values, i.e., differences, between the best insertion possibility of a delivery and the k-best options. The delivery with the highest difference (regret) is inserted in each iteration. This allows a more foresighted insertion that takes future costs into account. The Regret Operator is indispensable for the search as it significantly improves the solution quality of MCVRPs (see Derigs et al. (2011)). Finally, Record-To-Record Travel as introduced by Dueck (1993) is used as an acceptance criterion for the LNS. Accordingly, a new solution is accepted as a new incumbent solution if it lays within a defined deviation (D) from the best solution found so far. The LNS terminates after a predefined number of iterations without a solution improvement.

5.3. Evaluation and control mechanism of entire algorithm

Simulated Annealing. While testing our approach we found that the search process tends to get trapped in local minima. We therefore use a Simulated Annealing framework to govern the search and enable broader diversification. Accordingly, a new solution $\mathbb{S}^*$ found within the improvement heuristic is accepted if it is better than the best-known solution so far, $\mathbb{S}_{best}$, or the incumbent solution, $\mathbb{S}_{inc}$. Further, for a higher degree of diversification, a worse solution is accepted as an incumbent solution with the probability $e^{-\frac{f(\mathbb{S}')-f(\mathbb{S}_{inc})}{E}}$. This probability is then decreased in the course of the search process. The temperature $E > 0$ is initialized using E_{start} and decreased in each successive iteration by the cooling rate factor $d \in]0;1[$. For the calculation of E_{start} we adapted the method of Ropke and Pisinger (2006) to fit the requirements of the PMCVRP. Consequently, E_{start} is set such that a solution subsequently obtained is accepted with a probability of 0.5, i.e., $E_{start} = -\frac{g \cdot f(\mathbb{S}_{start})}{\ln 0.5}$, if its objective function value is g percent worse than the starting solution.

Additional diversification. Apart from being used as operator within the ALNS, the Random Operator is deployed as an additional tool of diversification. This is why we introduce a reset border λ. If λ iterations are made without a new best solution being found, the Random Operator is used, changing a high number of segment-store combinations and thereby destroying a large part of the current solution.

Termination criterion. If the number of ALNS iterations without a new best solution found reaches a predefined limit, the search process is stopped. This limit is independent of the reset border λ and is never readjusted in the course of the solution approach.

Adaptive operator selection. The final step of each ALNS iteration is the adaptation of operator weights used within the ALNS approach. As stated above (Section 5.2.1), the operator selection within the ALNS is based on individual weights for each operator. As proposed in Ropke and Pisinger (2006), we use a roulette wheel selection principle where the probability Φ_h of operator

Table 3
Overview of numerical experiments.

Section	Experiments and purpose	Stores	Segments	Data set	Number of instances
6.1	Runtime analysis	25, 50, 100	3	Solomon (1987)	18
6.2	Single segment benchmark	30, 40	1	Holzapfel et al. (2016)	120
6.3	Value of MCV integration and joint deliveries for DP planning	50	2–4	Simulated data, informed by real-world data	20
6.4	Case study	376	3	Real-world data	1

h, $h \in \{1, 2, \ldots, 7\}$ (see Table 2) being selected for the current iteration is determined by its weight ρ_h and $\Phi_h = \frac{\rho_h}{\sum_h \rho_h}$. In the beginning of the search, the likelihood of selection is equal for all operators, i.e., the weights of all operators are set to 1. Later, the weights are adjusted depending on their performance in the previous search leg. Note, a single *search leg* is defined by a specific number of consecutive iterations. A score Θ_h is introduced to measure the performance of operator h during the last search leg. The performance of the operators is measured by evaluating the overall solution obtained, i.e., the total costs according to Eq. (1). Θ_h is increased by θ_1 if the operator results in a new best solution, by θ_2 if the operator results in a new incumbent solution, and by θ_3 if the operator results in a new solution, but is not accepted. At the end of each search leg, the weights ρ_h for every operator h are updated according the average scores achieved, $\frac{\Theta_h}{\Omega_h}$ using Eq. (23). Ω_h denotes the number of times operator h is selected in the last search leg. The magnitude of change for the weights is controlled by the *smoothing factor* $\tau \in]0;1[$. After all new weights have been calculated, all Θ_h are reset to zero for the next search leg.

$$\rho_h = (1-\tau)\cdot\rho_h + \tau\cdot\frac{\Theta_h}{\Omega_h} \qquad \forall h \in \{1, \ldots, 7\} \tag{23}$$

Post-optimization of routing. After the stop criteria of the ALNS are met, we apply a post-optimization step to improve the final routing solutions, i.e., the MCVRPs for each period of $|T|$. We therefore apply an extended LNS to the routing problem of each period, increasing the LNS search limit of unsuccessful iterations significantly. Please note that using a higher limit for the LNS is only feasible at the end of the ALNS search, as the LNS is frequently applied during the search (i.e., $|T|$ times per ALNS iteration) and runtimes would increase exponentially.

6. Numerical experiments

Numerical experiments are applied to evaluate the performance of our solution approach and the interdependence between the planning of DPs and MCVs. The runtime is analyzed in Section 6.1. Section 6.2 compares our approach to the results of Holzapfel et al. (2016) to provide a benchmark with regard to solution quality. The impact of using MCVs instead of SCVs for the determination of DPs and the overall solution structure is assessed in Section 6.3. Finally, we apply our approach to a real-world case at a major German retailer in Section 6.4. Table 3 gives an overview of the numerical experiments and the data sets used.

Data applied. Each test instance is defined by the number of stores (and their spatial distribution), number of segments (and the number of their products) as well as the planning horizon, which in turn determines the number of possible DPs. If not stated otherwise, we consider five days ($|T| = 5$) as planning horizon with all possible DPs (i.e., $2^5 - 1 = 31$ combinations). The number of possible DPs may, however, be reduced due to non-feasible combinations, i.e., we check if a DP violates vehicle or store capacities (Q^{ved} and Q_i^{recmax}) for all given segment-store combinations. We apply a daily demand for each segment-store combination and specify the ranges for each data set. The daily demand for the specific weekdays depends on the weekly seasonality obtained from data of a benchmark case (see Section 6.2) and a real case study (see Section 6.4). The shelf capacity for each product was set equally for all stores in all tests. It was determined using the ratio of average weekly product demand to product shelf capacity as given in Holzapfel et al. (2016). Vehicle capacity is set to 2,700 transportation units (TU). Also, we adopt the empirical cost parameter setting by Holzapfel et al. (2016) for store- and DC-related costs as well as MCV-related loading and unloading cost parameters from Hübner and Ostermeier (2019). The exact values of case study related data are subject to non-disclosure agreements.

Implementation details. The algorithm-specific parameter setting used for our experiments is summarized in Table 4. We adopted the corresponding values reported in literature (see Table 4) for the majority of parameters as these yield excellent results for our setting. The weights δ and ϵ in the bundle operator segment have been tuned within our tests. The cooling rate (d) was adjusted compared to values reported in Ropke and Pisinger (2006) due to a differing number of iterations and different objective value ratios. The number of segment-store combinations (c) was chosen depending on the corresponding problem size as depicted in Table 5. All experiments were executed with a limit of 3,000 iterations without a new best solution found. The adaptive weights were adjusted after every search segment of 50 iterations. The LNS for the evaluation of the MCVRPs terminates after 100 unsuc-

Table 4
Algorithm parameters used.

Parameter	Value	Function	Origin
α	6	Degree of randomization	Ropke and Pisinger (2006)
β, γ	0.8, 0.2	Weights of proximity operator at ALNS	Derigs et al. (2011)
δ, ϵ	0.3, 0.7	Weights of segment bundle operator at ALNS	Own experiments
μ, ν, ξ	0.6, 0.2, 0.2	Weights at LNS	Hübner and Ostermeier (2019)
k	2	Regret insertion parameter at LNS	Hübner and Ostermeier (2019)
D	0.003	Deviation allowed at LNS	Hübner and Ostermeier (2019)
d	0.99975	Simulated annealing cooling rate	Own experiments
g	0.03	Start temperature control parameter	Ropke and Pisinger (2006)
λ	200	Reset border	Derigs et al. (2011)
θ_1, θ_2, θ_3	33, 9, 11	Operator score increase for new solutions	Ropke and Pisinger (2006)
r	0.1	Reaction factor for operator weight adjustment	Ropke and Pisinger (2006)

Table 5
Setting of parameter *c*.

Instance size	No. of segment-store combinations	
# stores	Minimum	Maximum
25	5	15
50	5	20
100	5	30
>200	5	50

cessful iterations. The extended LNS, which is applied only in the post-optimization when the ALNS is terminated, stops after 2,000 unsuccessful iterations. Our algorithm is also built on a stochastic search procedure. To balance this out, we apply the same instance multiple times, depending on the data set used. The frequency is denoted in the respective tests. The algorithm described in Section 5 was implemented in Java 8 and used for all following experiments.

6.1. Runtime analysis

Runtime data. We use the VRPTW data sets provided by Solomon (1987) to analyze the computation times of our solution approach. The data set comprises instances with 25, 50 and 100 customers and is subdivided into three categories: (C) clustered stores, (R) uniformly random distributed stores and a (RC) mixture of both. Solomon (1987) provides two different spatial distributions of stores for each category, resulting in six instances for each of the classes with 25, 50 and 100 stores. We use the number and the spatial distribution of customers from these data sets. The actual distances are multiplied such that the delivery area resembles a realistic distribution area in retail practice, similar to our case study (see Section 6.4). This allows maintenance of the general cost parameter setting as otherwise the share of travel costs would be underestimated. We set up three product segments for a five-day week.

Runtime comparison. The runtime analysis is summarized by each instance class in Table 6. As the definition of DPs is a tactical planning problem, the runtimes are still within an acceptable range, considering DPs are not defined on a weekly but monthly or yearly basis. The runtime strongly increases with the increase of problem sizes. This is due to the increase in segment-store combinations and the resulting complexity for pattern and routing decisions. More than 90% of computational time is consumed by the Regret-*k*-Insertion heuristic within the LNS for the daily routing. However, the Regret-*k*-Insertion is substantial to obtain good routing solutions as it significantly improves the solution quality of the search (see also Derigs et al. (2011)). The regret value has to be recalculated for each insertion and each possible position on trucks and therefore consumes significant computation time. Several parameters impact the runtime of the Regret-*k*-Insertion heuristic. The number of orders to be removed, the degree of regret (k), and the overall number of orders on each day are the most important drivers. In our problem, the number of orders per day changes dynamically and also the corresponding delivery volume. In contrast to other PVRPs, both delivery frequency and days are permanently changed within our approach.

6.2. Comparison with single segment benchmark

Benchmark approach. The effectiveness of our approach is shown by a comparison with Holzapfel et al. (2016). This approach is a special variant of our problem (multi-period, single segment and SCVs), and the only available benchmark for our setting. The authors solve the allocation of DPs by an optimally solved general assignment problem (GAP) while approximating the resulting transportation costs using the approach of Fisher and Jaikumar (1981). Contrary to our approach, day-to-day vehicle routing is not part of their solution procedure. They assume stable base tours, i.e., assignments of store orders to vehicles for each day of the planning horizon. In order to make a fair comparison, we re-evaluated the DP assignment from Holzapfel et al. (2016) using our general approach to solve the VRPs on each day of the planning horizon. This means, we also apply the LNS approach proposed in the present paper for each instance derived from Holzapfel et al. (2016) to further improve the results presented there. In doing so, we apply the extended LNS for each delivery day five times and keep the best daily solution found. This procedure entirely corresponds to the assumptions and the implementation of our overall modeling and solution approach.

Benchmark data. Holzapfel et al. (2016) apply scenarios for a single segment across six delivery days with 10, 20, 30 and 40 stores and three different delivery area sizes. The stores are randomly located within a delivery area of 50 km × 50 km (“Metropolitan”), 200 km × 200 km (“District”) or 400 km × 400 km (“State”). All demand, store and cost parameters in Holzapfel et al. (2016) are set according to empirical data of a partner company as well as to data collected by Kuhn and Sternbeck (2013) and Sternbeck and Kuhn (2014). In the benchmark data, not all of the $2^6 - 1 = 63$ DPs are feasible for all stores due to vehicle (Q^{vec}) or receiving capacities at stores (Q_i^{recmax}). Consequently, around one-third of DPs can be excluded upfront with respect to individual segment-store combinations. Since smaller problem classes are not relevant for our application, we focus on instances with 30 and 40 stores, totaling 120 instances, for the benchmark calculations.

Benchmark comparison. The overall approach suggested in the present paper achieved costs savings in all instances but one (see Fig. 5) compared to the adapted Holzapfel et al. (2016) approach.

In detail, two effects can be observed. First, the larger the delivery area, the greater are the improvements achieved. With increasing delivery area size and thus higher travel costs, routing becomes more important. Since Holzapfel et al. (2016) approximate the travel costs, actually solving the VRP gains importance. Secondly, it can be observed that the average improvement for instances with 40 stores is slightly lower than with 30 stores. This can be attributed to the increasing impact of DP selection. As more customers are involved, it is more important to find the optimal DP assignment in order to exploit the bundling effects in transportation. As Holzapfel et al. (2016) solve the allocation of DPs optimally using approximations for the resulting travel costs, it be-

Table 6
Total computation times for different problem sizes, in hours.

No. of stores	25			50			100		
No. of segment-store combinations	75			150			300		
Type of spatial store distribution	C	R	RC	C	R	RC	C	R	RC
Runtime, *min*	0.13	0.21	0.16	0.35	0.57	0.34	1.83	1.30	1.09
Runtime, *average*	0.36	0.65	0.57	1.50	1.60	0.93	4.77	4.72	3.93
Runtime, *max*	0.74	1.07	0.84	3.33	2.80	1.94	15.48	10.17	9.28

Table 7
Cost savings in comparison to the adapted benchmark of Holzapfel et al. (2016) (best of 5 runs).

Delivery area	Metropolitan		District		State		Average
Number of stores	30	40	30	40	30	40	
Total cost savings	0.56%	0.44%	2.05%	1.55%	3.71%	2.81%	1.85%
Savings share of[a]							
Travel costs	0.50%	0.36%	1.93%	1.46%	3.58%	2.73%	1.76%
Unloading costs	0.03%	−0.06%	−0.06%	−0.05%	−0.03%	−0.06%	−0.04%
Pattern-dependent costs	0.03%	0.13%	0.18%	0.14%	0.16%	0.13%	0.13%

[a] Relative share of savings in relation to total costs of Holzapfel et al. (2016).

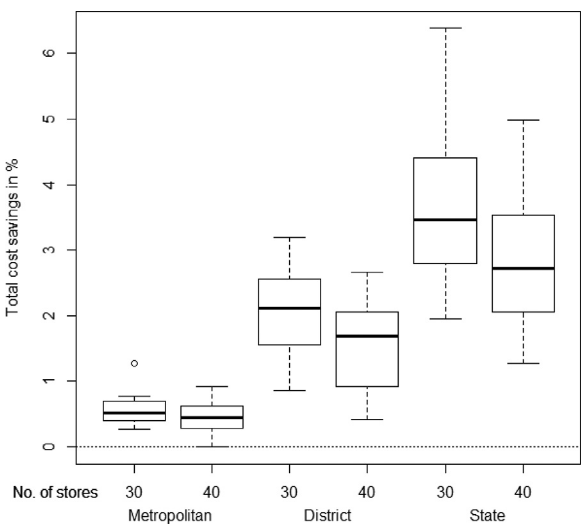

Fig. 5. Distribution of total cost savings depending on delivery area and number of stores in comparison to the adapted benchmark of Holzapfel et al. (2016) (best of 5 runs).

comes more difficult for our approach to generate additional improvements via delivery pattern assignment. Table 7 shows the cost savings in percent of total costs obtained by Holzapfel et al. (2016). Our results show that total cost savings originate almost entirely from travel costs, whereas the relative pattern-dependent cost savings are on average only around 0.13%.

Concerning computation times, Holzapfel et al. (2016) report an average of 2.7 minutes in addition to a pre-processing time of about 5 minutes for instances with 40 customers, while the computation times of our approach amount to an average of 5.23 minutes per single run for the same instances.

In conclusion, we observe that our solution approach is able to solve the related benchmark problem effectively. It improves the benchmark solutions by around 1.85% on average across all problem classes when applying the best solution of five runs. The average of all five runs still improves the results by around 1.25% across all problem classes. Moreover, we would like to note that our approach aims at problems with multiple segments and the corresponding cost savings when segments are jointly delivered. This effect is not taken into account within the benchmark comparison as only a single segment is considered.

6.3. Planning of delivery patterns across segments with multi-compartment vehicles

A core aspect of our work is the use of MCVs for distribution and the corresponding impact on determining DPs. The different product flows and consequently joint deliveries are only possible with MCVs. In line with this, we compare the results of our solution approach with MCVs to a solution using SCVs only where the product segments are distributed separately. In the SCV scenario

Table 8
(Mean) segment share of total order volume.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Segment 1	80%	50%	$33.\overline{3}\%$	$54.\overline{4}\%$
Segment 2	10%	30%	$33.\overline{3}\%$	$24.\overline{4}\%$
Segment 3	10%	20%	$33.\overline{3}\%$	$21.\overline{1}\%$

we apply our solution approach for each product segment separately, thus generating independent single segment solutions. The scenario with joint planning and delivery is denoted as “MCV” and the separate one as “SCV”.

Test data. For the following analysis we use simulated data sets. It compromises four scenarios with five instances for each, totaling 20 data sets. The generation of these data sets is informed by the real-world data from our case study. All instances comprise 50 stores, which are randomly located around a DC within a delivery area of 230 km × 180 km. Demands are simulated for three product segments. The total demand across the delivery week follows a normal distribution with $\mu = 1,400$ and $\sigma = 800$. Stores generally feature – depending on their size – different sales volumes across segments. The weekly demand is therefore randomly multiplied by a factor of 0.5 (low), 1.0 (medium) or 1.5 (high) to simulate different store sizes. The daily demand is subject to weekly seasonality with the distribution factors $\{0.149, 0.233, 0.205, 0.211, 0.202\}$ for a five-day week. The shelf capacity for each product is set equally for all stores, but randomly across products, according to the ratios of demand to shelf capacity reported by Holzapfel et al. (2016). Based on the medium store size, this results in normally distributed shelf capacity with $\mu = 1,000$ and $\sigma = 400$. As the available real-world data lacks information on picking and store receiving capacities, these are set as unconstrained. We apply our heuristic five times to each instance and compare the best results achieved.

The segment share of the total order volume may have a major impact. One could expect that the more segments are available, the higher the potential for cost savings through the joint delivery of segments. We therefore apply four different scenarios. The demand share of each segment for each scenario is given in Table 8. In Scenarios 1 to 3, each store follows the segment shares indicated. In Scenario 1 for example, Segments 1, 2 and 3 comprise 80%, 10% and 10% of the total order volume, respectively. Scenario 4 combines Scenarios 1 to 3 by randomly assigning one of the given scenarios to each store.

Results. We first analyze the overall cost structure for all scenarios. Table 9 illustrates the share of cost components of the total costs across the different scenarios.

The cost structure of the MCV and SCV scenarios identify pattern-dependent costs as main cost driver of the PMCVRP. Pattern-dependent costs account for almost two-thirds of total costs with an average of 63% (MCV) and 61% (SCV). The share of travel costs on the other hand is only half as high with 27% (MCV) and 31% (SCV). This underlines the importance of DP selection for a low cost solution. We further compare potential cost sav-

Table 9
Share of costs components of total costs.

Scenario	1		2		3		4	
	MCV	SCV	MCV	SCV	MCV	SCV	MCV	SCV
Loading costs	5.6%	1.7%	5.7%	1.7%	5.7%	1.6%	5.3%	1.6%
Travel costs	26.5%	28.9%	26.7%	30.9%	27.4%	34.3%	27.7%	30.2%
Unloading costs	4.3%	5.5%	4.2%	6.3%	4.1%	6.5%	4.6%	5.7%
Pattern-dependent costs	63.6%	63.8%	63.4%	61.0%	62.8%	57.6%	62.2%	62.4%

Table 10
Cost savings of MCV compared to SCV.

Scenario	1	2	3	4
Total costs saving	8.28%	13.12%	15.15%	10.60%
Savings share of[a]				
Loading costs	−3.43%	−3.30%	−3.20%	−3.12%
Travel costs	4.67%	7.73%	11.14%	5.44%
Unloading costs	1.60%	2.69%	2.98%	1.58%
Pattern-dependent costs	5.44%	6.00%	4.23%	6.69%

[a] Relative share of savings in relation to total SCV costs

ings and other performance indicators across all scenarios when all segments are jointly planned and MCVs instead of SCVs are used. Tables 10 and 11 summarize our findings. In all of our scenarios, the joint planning and usage of MCVs results in significant cost reductions, with up to 15% of total cost savings (Scenario 3).

As expected, loading costs increase in all scenarios when MCVs instead of SCVs are used. They occur for each segment loaded on a vehicle. The loading costs for MCV vs. SCV are approximately two times higher. This shows that on nearly all tours three segments are combined when MCVs are available. The main savings are achieved by lower pattern and travel costs. They originate in the joint delivery of segments and are therefore dependent on the demand scenario considered. Pattern-dependent savings are highest when the segment demand structure is most heterogeneous (Scenarios 2 and 4). DPs across the different segments are aligned and thus costs decrease. Travel cost savings are higher in scenarios where the segment volumes are equally spread across segments as shown in Scenario 3. With equally spread volumes, it is very likely that different segments are supplied together, whereas in Scenario 1 (80%/10%/10% volume shares) bundling effects across segments are limited. The same can be observed with respect to unloading costs: most stops at stores can be avoided in Scenario 3 (306 (SCV) vs. 164 (MCV) stops).

The corresponding solution structure for each scenario is further depicted in Table 11. The total number of deliveries differs as we determine the delivery frequency and therefore the split of total demand across weekdays. The most striking impact of MCVs on the solution structure is increasing delivery frequency and therefore more deliveries in total. For instance, the average number of deliveries rises in Scenario 1 from around 251 to 412 deliveries, an increase of over 64%. The average number of tours remains relatively stable in all scenarios, and consequently the number of orders delivered per tour increases. Interestingly, a greater increase in delivery frequency does not lead to a greater decrease in pattern-dependent costs. Considering Scenario 1, the frequency increases from 1.67 to 2.75 and pattern-dependent costs are reduced by 5.44%. In contrast, Scenario 4 reveals a reduction of 6.69%, while the increase in delivery frequency is lower when MCVs are used (1.83 to 2.60 deliveries). This can be attributed to the fact that the store-individual optimal delivery frequencies are not generally chosen when using SCVs or MCVs. Instead, a different DP and frequency is chosen as another option enables a higher total cost saving (e.g., due to reduced transportation costs). Also, the average capacity utilization per tour improves by about 1.27% on average when using MCVs, as there are more deliveries, and thus more possible loading combinations.

6.4. Case study

Case study data. To conclude our numerical analysis, we present a case study with a major German retailer. Our partner company uses MCVs for distribution but had not considered the impact of MCVs on the determination of DPs. We therefore apply our approach to the given problem data to compare the selection of DPs when the benefits of MCVs are taken into account during the decision process. The case study covers a representative five-day week with orders of 376 stores to be delivered from a single depot. Most stores have a relatively small demand such that most of them are only served once a week. The order structure is heterogeneous, with about 50% of stores ordering one segment, about 20% ordering two segments, and about 30% ordering all three segments. Moreover, the most frequently ordered segment accounts for around 80% of the total order volume. The other two segments account for roughly half of the remaining volume each. Consequently, this order structure resembles Scenario 1 above. The order volumes of the three segments follow a weekly seasonality. Regarding store sizes, the 10% of stores with the highest sales also account for more than 40% of the order volume. This means that the majority of stores have a small demand volume. As in the previous data sets, the shelf space data was supplemented based on an average-sized store demand assuming the same ratio between product demand and shelf capacity as in Holzapfel et al. (2016).

In the following we analyze the potential cost savings if the retailer coordinates DP and MCV planning. In doing so, we compare the retailer's approach denoted as “status quo approach” with our approach denoted as “integrated approach”. The status quo approach equals our approach, but assumes the DPs currently applied by the retailer. We therefore assume that the retailer already applies the same MCVRP solution approach as we do. This means we only evaluate the effect of planning DPs across segments and do not mix this with potential effects resulting from different approaches used to solve the MCVRP. We apply both approaches ten times and compare the respective best solution.

Results. Table 12 presents the resulting cost savings in % of the status quo total costs; and Table 13 displays some performance indicators and reveals insights into the respective solution structure achieved. The runtime of our integrated solution approach amounts to an average of 2.89 hours.

The integrated approach that jointly determines DPs across segments results in total cost savings of 7.68% compared to the status quo approach. The savings mainly result from improved pattern and travel costs, which are facilitated by improved DP assignment. Only the segment-store combinations with very high demand are supplied twice a week. Almost all other combinations are delivered once a week. The number of stops and tours required are therefore reduced, resulting in higher vehicle capacity utilization. The reduction of delivery tours is an essential driver of cost reduction as it leads to personnel cost savings. This means that at our case company the number of delivery tours should be further decreased as the savings outweigh the increase of instore logistics costs al-

Table 11
Solution structure of MCV compared to SCV, entire planning period.

Scenario	1		2		3		4	
	MCV	SCV	MCV	SCV	MCV	SCV	MCV	SCV
∅ number of orders delivered	412.8	251.2	463.2	313.0	411.8	306.0	389.4	274.2
∅ number of tours	29.0	29.0	30.2	30.6	27.8	28.0	28.2	29.0
∅ number of orders/tour	14.2	8.7	15.3	10.2	14.8	10.9	13.8	9.5
∅ number of stops	178.4	251.2	180.2	313.0	164.8	306.0	198.2	274.4
∅ capacity utilization	89.9%	88.5%	89.3%	88.3%	88.9%	88.7%	90.5%	87.9%
∅ delivery frequency	2.75	1.67	3.09	2.09	2.74	2.04	2.60	1.83

Table 12
Cost savings of the integrated approach compared to status quo.

Total cost savings	7.68%
Savings share of[a]	
Loading costs	0.36%
Travel costs	4.08%
Unloading costs	0.97%
Pattern-dependent costs	2.27%

[a] Relative share of savings in relation to total costs of status quo

Table 13
Solution structure resulting from the integrated approach compared to the status quo approach.

	Status quo approach	Integrated approach
Number of orders delivered per week	709	630
Number of tours per week	24	22
Number of stops per week	418	381
∅ capacity utilization	86.7%	94.6%

though the company's solution already reveals a low average delivery frequency. In conclusion, we can state that an integrated solution approach enables better evaluation of the complete planning problem. When delivery options with MCVs are taken into account, DPs can be adjusted to align deliveries across the complete planning horizon and to ultimately reduce total costs. This can result in both increasing (see Section 6.3) or decreasing frequency, as shown in the case study.

7. Conclusion

In this paper we introduced a new MCVRP variant, the PMCVRP, that addresses the selection of delivery patterns when MCVs are used for distribution. The PMCVRP is a multi-period MCVRP applied for grocery distribution. However, the practical relevance is not limited to this application as it can easily be adapted to other application areas in which periodicity of deliveries is relevant (e.g., fuel distribution or agricultural problems). The problem presented combines the research on DP planning with an MCVRP and consequently closes an existing gap in literature by identifying new options for delivery planning. More precisely, the objective of our work is to highlight the impact on DP planning when the deliveries of different product segments can be combined across the planning horizon when using MCVs. This paper identifies the decision-relevant processes and corresponding costs for both the choice of patterns and the use of MCVs, and presents a formal model description. The resulting problem is solved using an ALNS approach for assigning patterns and an LNS for solving the routing. It is tailored to the given problem specifics. The performance of the algorithm proposed is compared to an existing approach in literature to show its efficiency and effectiveness. This revealed that our approach is able to improve given solutions for DP planning. In subsequent numerical experiments we analyze the interdependencies between routing with MCVs and DP planning. We show that, depending on the given problem characteristics, the PMCVRP leads to a different solution structure (i.e., altered delivery frequencies) and reduces total costs compared to the prevailing planning with SCVs as it combines different product flows and adjusts the corresponding patterns of stores accordingly. Finally, we consider a case study with a major German retailer for the supply of stores with small order volumes. The case study shows the practical relevance of our approach and improves the planning solution of the retailer by around 8% if DP planning is solved using MCVs.

The research on MCVRPs has steadily grown over the past years and our work further contributes to this field by closing another existing gap in literature. However, there are still numerous possibilities for future research. First, we consider an MCVRP for master route planning and assume given demands for an average week. In this context, the consideration of stochastic demand could further improve the planning as a more realistic evaluation of costs would be possible considering realistic demand fluctuations. In general, the consideration of stochastic demands is still neglected in most MCVRP applications (see Ostermeier et al. (2020)). Second, our cooperation with industry shows that due to given economic developments and changed conditions, the existing delivery fleet usually consists of heterogeneous vehicles for different purposes, including both MCVs and SCVs. Consequently, the consideration of a heterogeneous fleet within the PMCVRP would be a valuable next step. Third, the PMCVRP aims at minimizing total costs, consisting of pattern- and routing-dependent cost factors. The consideration of further impacts on profits (e.g., service level agreements, tardy deliveries) as well as ecological aspects (energy consumption, joint vs. split delivery) would be a valuable avenue for future research directions. Lastly, the ALNS and LNS approaches perform well when solving the PMCVRP and MCVRP, respectively. Recently, other solution approaches are suggested that show promising results in related application areas, such as population-based search algorithms relating to waste collection (see Rabbani, Farrokhi-asl, & Rafiei (2016)).

Acknowledgements

We owe a debt of gratitude to the industry experts who have provided empirical data and practical insights as well as encouraged our research topic. We further want to thank the editor and reviewers for their valuable advice on improving this work.

References

Avella, P., Boccia, M., & Sforza, A. (2004). Solving a fuel delivery problem by heuristic and exact approaches. *European Journal of Operational Research, 152*(1), 170–179.
Broekmeulen, R., van Donselaar, K., Fransoo, J., & Woensel, T. v. (2006). The opportunity of excess shelf space in grocery retail store. *Operations Research, 49*, 710–719.
Campbell, A. M., & Wilson, J. (2014). Forty years of periodic vehicle routing. *Networks, 63*(1), 2–15.
Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*(4), 568–581.

Coelho, L. C., & Laporte, G. (2015). Classification, models and exact algorithms for multi-compartment delivery problems. *European Journal of Operational Research, 242*(3), 854–864.

Curşeu, A., van Woensel, T., Fransoo, J., van Donselaar, K., & Broekmeulen, R. (2009). Modelling handling operations in grocery retail stores: An empirical analysis. *Journal of the Operational Research Society, 60*(2), 200–214.

Derigs, U., Gottlieb, J., Kalkoff, J., Piesche, M., Rothlauf, F., & Vogel, U. (2011). Vehicle routing with compartments: Applications, modelling and heuristics. *OR Spectrum, 33*, 885–914.

Donselaar, K. H. v., Gaur, V., Woensel, T. v., Broekmeulen, R. A., & Fransoo, J. C. (2010). Ordering behavior in retail stores and implications for automated replenishment. *Management Science, 56*(5), 766–784.

Dueck, G. (1993). New optimization heuristics. *Journal of Computational Physics, 104*(1), 86–92.

Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks, 11*(2), 109–124.

Gaur, V., & Fisher, M. (2004). A periodic inventory routing problem at a supermarket chain. *Operations Research, 52*, 813–822.

Glatzel, C., Großpietsch, J., & Hübner, A. (2012). Higher margins through efficient supply chains. *Akzente*, (2), 16–21.

Henke, T., Speranza, M. G., & Wäscher, G. (2015). The multi-compartment vehicle routing problem with flexible compartment sizes. *European Journal of Operational Research, 246*(3), 730–743.

Henke, T., Speranza, M. G., & Wäscher, G. (2019). A branch-and-cut algorithm for the multi-compartment vehicle routing problem with flexible compartment sizes. *Annals of Operations Research, 275*(2), 321–338.

Holzapfel, A., Hübner, A., Kuhn, H., & Sternbeck, M. (2016). Delivery pattern and transportation planning in grocery retailing. *European Journal of Operational Research, 252*(1), 54–68.

Hsiao, Y.-H., Chen, M.-C., & Chin, C.-L. (2017). Distribution planning for perishable foods in cold chains with quality concerns: Formulation and solution procedure. *Trends in Food Science & Technology, 61*, 80–93.

Hübner, A., & Ostermeier, M. (2019). A multi-compartment vehicle routing problem with loading and unloading costs. *Transportation Science, 53*(1), 282–300.

Hübner, A. H., Kuhn, H., & Sternbeck, M. G. (2013). Demand and supply chain planning in grocery retail: An operations planning framework. *International Journal of Retail & Distribution Management, 41*(7), 512–530.

Koch, H., Henke, T., & Wäscher, G. (2016). A genetic algorithm for the multi-compartment vehicle routing problem with flexible compartment sizes. Working Paper No. 04/2016, Otto-von-Guericke-University Magdeburg,.

Kotzab, H., & Teller, C. (2005). Development and empirical test of a grocery retail instore logistics model. *British Food Journal, 107*, 594–605.

Kuhn, H., & Sternbeck, M. G. (2013). Integrative retail logistics: An exploratory study. *Operations Management Research, 6*(1–2), 2–18.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science, 43*(4), 408–416.

Martins, S., Ostermeier, M., Amorim, P., Hübner, A., & Almada-Lobo, B. (2019). Product-oriented time window assignment for a multi-compartment vehicle routing problem. *European Journal of Operational Research, 276*, 893–909.

Minner, S., & Transchel, S. (2010). Periodic review inventory-control for perishable products under service-level constraints. *OR Spectrum, 32*(4), 979–996.

Ostermeier, M., Henke, T., Hübner, A., & Wäscher, G. (2020). Multi-compartment vehicle routing problems: State-of-the-art, modeling framework and future directions. *European Journal of Operational Research*. https://doi.org/10.1016/j.ejor.2020.11.009.

Ostermeier, M., & Hübner, A. (2018). Vehicle selection for a multi-compartment vehicle routing problem. *European Journal of Operational Research, 269*, 682–694.

Ostermeier, M., Martins, S., Amorim, P., & Hübner, A. (2018). Loading constraints for a multi-compartment vehicle routing problem. *OR Spectrum, 40*, 997–1027.

Rabbani, M., Farrokhi-asl, H., & Rafiei, H. (2016). A hybrid genetic algorithm for waste collection problem by heterogeneous fleet of vehicles with multiple separated compartments. *Journal of Intelligent & Fuzzy Systems, 30*, 1817–1830.

Reiner, G., Teller, C., & Kotzab, H. (2013). Analyzing the efficient execution on in-store logistics processes in grocery retailing – the case of dairy products. *Production and Operations Management, 22*(4), 924–939.

Ronen, D., & Goodhart, C. A. (2008). Tactical store delivery planning. *Journal of the Operational Research Society, 59*(8), 1047–1054.

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science, 40*(4), 455–472.

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems. APES Group, Dept of Computer Science, University of Strathclyde, Glasgow, Scotland, UK,.

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research, 35*(2), 254–265.

Sternbeck, M. G., & Kuhn, H. (2014). An integrative approach to determine store delivery patterns in grocery retailing. *Transportation Research Part E, 70*, 205–224.

Taube, F., & Minner, S. (2018). Data-driven assignment of delivery patterns with handling effort considerations in retail. *Computers & Operations Research, 100*, 379–393.

Toth, P., & Vigo, D. (2014). Vehicle routing: Problems, methods, and applications, second edition. *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics.

van Zelst, S., van Donselaar, K., van Woensel, T., Broekmeulen, R., & Fransoo, J. (2009). Logistics drivers for shelf stacking in grocery retail stores: Potential for efficiency improvement. *International Journal of Production Economics, 121*(2), 620–632.

Zajac, S. (2017). An adaptive large neighborhood search for the periodic vehicle routing problem. In T. Bektaş, S. Coniglio, A. Martinez-Sykora, & S. Voß (Eds.), *Computational logistics* (pp. 34–48). Cham: Springer International Publishing.