

Article

Using Biased-Randomized Algorithms for the Multi-Period Product Display Problem with Dynamic Attractiveness

Mage Marmol ¹, Leandro do C. Martins ², Sara Hatami ² and Angel A. Juan ^{1,2,*}
and Vicenc Fernandez ³

¹ Marketing Department, Euncet Business School, 08225 Terrassa, Spain; mage.marmol@euncet.es

² IN3—Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; leandroc@uoc.edu (L.d.C.M.); shatami@uoc.edu (S.H.)

³ TechTalent-Lab, Universitat Politecnica de Catalunya—BarcelonaTech, 08222 Terrassa, Spain; vicenc.fernandez@upc.edu

* Correspondence: ajuanp@uoc.edu; Tel.: +34-932-53-2300

Received: 15 October 2019; Accepted: 29 January 2020; Published: 1 February 2020



Abstract: From brick-and-mortar stores to omnichannel retail, the efficient selection of products to be displayed on store tables, advertising brochures, or online front pages has become a critical issue. One possible goal is to maximize the overall ‘attractiveness’ level of the displayed items, i.e., to enhance the shopping experience of our potential customers as a way to increase sales and revenue. With the goal of maximizing the total attractiveness value for the visiting customers over a multi-period time horizon, this paper studies how to configure an assortment of products to be included in limited display spaces, either physical or online. In order to define a realistic scenario, several constraints are considered for each period and display table: (i) the inclusion of both expensive and non-expensive products on the display tables; (ii) the diversification of product collections; and (iii) the achievement of a minimum profit margin. Moreover, the attractiveness level of each product is assumed to be dynamic, i.e., it is reduced if the product has been displayed in a previous period (loss of novelty) and vice versa. This generates dependencies across periods. Likewise, correlations across items are also considered to account for complementary or substitute products. In the case of brick-and-mortar stores, for instance, solving this rich multi-period product display problem enables them to provide an exciting experience to their customers. As a consequence, an increase in sales revenue should be expected. In order to deal with the underlying optimization problem, which contains a quadratic objective function in its simplest version and a non-smooth one in its complete version, two biased-randomized metaheuristic algorithms are proposed. A set of new instances has been generated to test our approach and compare its performance with that of non-linear solvers.

Keywords: omnichannel retail stores; product display problem; multi-period decisions; dynamic attractiveness; biased-randomized heuristics

1. Introduction

As discussed in Verhoef et al. [1], customers today are changing the way they decide where, how, and even when to buy. With the rise of Internet-based technologies and mobile devices, different shopping channels have appeared and attracted customers’ attention. Hence, e-commerce not only offers customers the possibility of browsing through different stores in an online environment, but also the ability to get information, opinions, and a vast availability of stock. Omnichannel commerce is a fully-integrated approach to e-commerce that provides customers with a unified experience

across different shopping platforms, e.g., a personal computer, a physical retail center, or a mobile device. In an omnichannel environment, retailers at brick-and-mortar stores have to compete with other channels, and especially with the ‘showrooming’ behavior of customers. Showrooming occurs when customers go to a brick-and-mortar store to ‘touch and feel’ the product, but then complete the purchase online. Even when customers are in the position of choosing where and when to buy, most brands still generate a noticeable part of their sales revenue at brick-and-mortar stores, so they play a relevant role in capturing customers’ attention.

One of the strategies used by brick-and-mortar retailers to engage more customers is to offer them a variety of attractive products over a multi-period time horizon, which typically covers several weeks. As pointed out by Galino and Moreno [2], in order to achieve this goal the retailer needs to decide which combination of products has to be shown at the store, so that the combined attractiveness value is maximized. Some authors define attractiveness as the capacity to cause interest and attract the attention of another party [3]. According to Ellegaard and Ritter [4], value creation, interaction process, and emotions define the perceived attractiveness of one actor to another actor. The value creation refers to the potential value, while the interaction process deals with trust and commitment. Finally, emotions are the irrational part of decision making, which is not accessible by rational arguments. In this way, the attractiveness can be seen as an inter-linked concept which combines value, trust, commitment, and satisfaction [5]. According to Caro and Martínez-de Albéniz [6], “for many products, consumers tend to make different purchasing decisions over time. For example, most people would usually avoid eating the exact same meal in the exact same restaurant every day. This observed pattern has been called variety-seeking behavior”. The same authors state that “in apparel retailing, part of the success of fast fashion firms such as Zara and H&M relative to the incumbents such as the Gap can be attributed to more frequent assortment rotation, which generates the feeling of novelty among consumers”. Similarly, Caro et al. [7] studied the concept of product attractiveness in retail stores. In their own words: “carrying a static assortment—one that remains the same over time—becomes ineffective and possibly unprofitable because consumers are quickly bored with the choices within assortment and they divert their purchases to other consumption options. In other words, the customers’ preference for a particular product in the assortment decays over time, as it ages on the shelf”. These authors offer several examples of assortment renewal strategies involving clothing retailers such as H&M and Chico’s. They also discuss similar patterns in industries such as book stores and restaurants, which “frequently change the items on their menu to avoid customer satiation”. A similar concept can be found in Bernstein and Martínez-de Albéniz [8], who claim that “retailers in industries with short life cycles, such as apparel, have started updating their product offering with significant frequency. In particular, fast-fashion retailers such as Zara or H&M update their assortments periodically to induce frequent visits to their stores”. As exposed in Ferreira and Goh [9], “assortment rotation... has recently been used by both brick-and-mortar and online retailers as a strategy for gaining competitive advantage. A notable category of retailers who have employed this strategy successfully are fast-fashion retailers such as Zara and H&M, who have differentiated themselves from other retailers by rotating their assortment multiple times throughout the fashion industry standard 6-month selling season”. All in all, the attractiveness value of some products might decay as they are repeatedly displayed. Hence, the retailer must release new products. Similarly, if customers see the same products exposed in a store during several consecutive time periods (days or weeks), their willingness to visit that store will decrease. Accordingly, some popular retailers introduce new products into their stores almost on a daily basis. These dependencies across periods are considered in our work, which constitutes one of the main original contributions of our approach. Schnurr et al. [10] addresses novelty, placement, and consumers’ opinions as essential factors on the definition of product attractiveness: (i) unfamiliar—or new—products are perceived as more attractive and, consequently, of higher quality when placed in an attractive context; and (ii) the higher the attractiveness score of a product, the higher are the consumers’ intentions to purchase it. In this work, given a product included in a display table, we define its individual attractiveness value as the probability that the

product captures the attention (e.g., is selected and analyzed) of a standard customer visiting the shop. Being a probability, it will always be a value between 0 and 1 (or, equivalently, a score between 0 and 100). Attractiveness can also be measured by visual properties, and this is directly related to the existence of correlation between pairs of products (e.g., products that are complementary or substitute). Thus, retailers have to take into account customers’ purchases that occur in channels other than brick-and-mortar stores. Apart from experts’ opinion, a large amount of data can be obtained from customers’ behavior and preferences in an omnichannel environment. These data can provide retailers with vital information, such as which products generate a higher attraction level among customers of a certain retail store. Hence, identifying the best assortment of products to display has to be made considering customers’ preferences [11]. Selling strategies for retail stores should have the ability to offer customers a set of different surprising experiences. Using display tables to suggest a set of correlated articles in retail stores is one way to achieve the aforementioned goal. In the apparel sector, for example, a yellow sweater may be positively correlated with a white pair of jeans but negatively correlated with orange trousers (since yellow and orange are not colors that match according to certain fashion trends). Likewise, a skirt might be positively correlated with a top and negatively correlated with a pair of jeans, since both cloth pieces are bottom parts. Again, these dependencies across items should be considered. In practice, data gathered in an omnichannel database could be one of the most efficient ways to determine the correlations between pairs of products.

The problem of selecting, over a multi-period horizon, the most attractive configuration of products to be displayed in a limited space (e.g., a physical table at the store, an advertising brochure, or a website front page), is related to well-known problems such as the product recommendation problem [12], the shelf space allocation problem [13], and the assortment problem [14]. Figure 1 shows a simple example to illustrate the product display problem and its possible solutions. In this case, the solution consists of a 2-period time horizon. In each period, there are 3 display tables with a capacity of 5 items each. Each display table and item is represented by its identifier (ID). Hence, for instance, in period h_1 , display table 2 will be composed of items 11, 29, 45, 18, and 5.

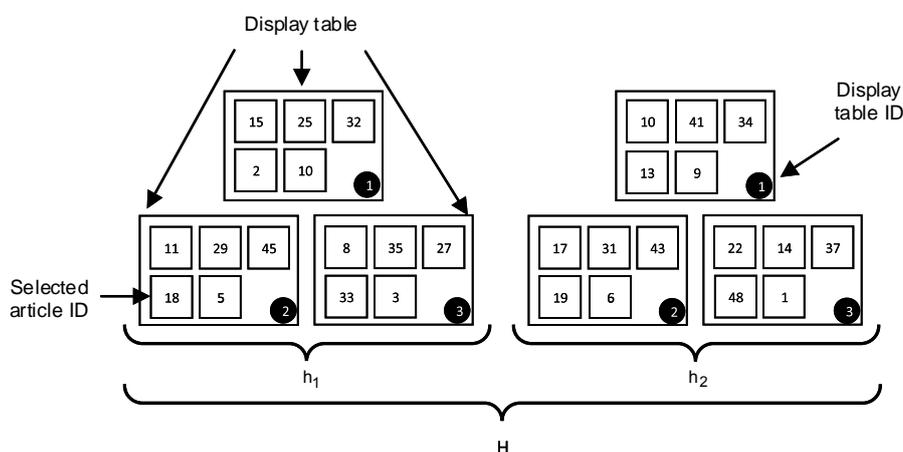


Figure 1. A solution representation for a simple multi-period product display problem.

When selecting a set of products to be displayed, one should always consider customer preferences and willingness to buy [15]. Current online display systems provide a list of products that are either based on the user’s past behavior or on decisions made by similar users. This strategy has been widely applied in e-commerce, where it has generated raised sales as well as customer satisfaction [16]. Companies such as Amazon use a method called collaborative filtering. Here, ratings and purchases made by similar users are considered to suggest products to online customers [17]. This product selection problem is also relevant for brick-and-mortar stores, since they might benefit from an optimal selection of products to be displayed over a multi-period time horizon. The main contributions of this paper are described next: (i) a novel mathematical formulation for the multi-period

product display problem with dynamic attractiveness levels is proposed in order to clearly define the problem under consideration—while analyzing a case study, the assumptions of this model were discussed with professionals of the retail sector, who were also students in an MBA offered at our business school—(ii) in order to solve this optimization problem in the context of a retail store with several display tables, biased-randomized (BR) versions of the greedy randomized adaptive search procedure (GRASP) and the iterated local search (ILS) are introduced; (iii) a set of novel benchmark instances, considering realistic constraints and different product characteristics, is proposed to test the quality of our approach when compared with non-linear solvers; and (iv) based on the outcomes of our experiments, a series of practical recommendations are provided. A complete introduction to biased-randomization techniques can be found in Grasas et al. [18]. A review of GRASP algorithms can be found in Festa and Resende [19], while a description of the ILS metaheuristic framework can be found in Lourenço et al. [20]. Finally, a recent study on the combination of biased-randomization techniques with GRASP is available in Ferone et al. [21]. Regarding the constraints considered in this work, they include diversity of fashion collections, selling-price categories, and marginal-profit categories. Likewise, dependencies across both periods and items are considered in our study. The rest of the paper is arranged as follows: Section 2 presents a literature review of related research; Section 3 describes the problem in more detail and provides a mathematical formulation for it; Section 4 introduces the proposed biased-randomized algorithms; Section 5 includes an explanation of the computational experiments carried out to test the quality of our approach, while Section 6 contains an analysis of the results; finally, Section 7 highlights the main conclusions of this work and proposes some lines for future research.

2. Related Work

This section has been divided into two subsections, each dealing with different problems strongly related to the product display problem considered in this paper.

2.1. The Product Assortment and Product Recommendation Problems

The product assortment problem involves finding the optimal combination of products to include in a limited portfolio of items to be manufactured or sold. One of the first works on the product assortment problem is due to Sadowski [22]. Since then, many researchers have explored this field, as illustrated in the survey made by Pentico [14]. This author classified the literature according to the following characteristics: demand, demand pattern, dimensions, number of stock sizes, substitution cost structure, and stocking pattern stability. In this sense, different authors studied the way customers make decisions and how that affects the stock offered or shown. Mahajan and van Ryzin [23] considered that customers choose only among products that are still in stock. Caro et al. [7] and Ulu et al. [24] developed models in which the assortment needs to be adapted over time. As identified by Mantrala et al. [25], product assortment not only has the constraints of physical space and retailers' budget, but an attractiveness factor—as perceived by customers—should also be taken into account. Caro et al. [7] considered a problem in which the attractiveness of products decays over time once they are introduced to the selected assortment. A related study on space and store operations is provided by Mou et al. [26], who considered how product attractiveness decreases with time. They also discuss the need for retailers to gather information from different channels in order to better plan their stock assortment.

Related to this, Honhon et al. [11] studied product substitution after a stock-out of a first-choice item occurs. Here, customers choose the products that are available at the time of their visit to a physical store. Similarly, other authors analyzed the management of multi-item retail inventory systems with demand substitution [27] and the dynamic assortment planning with demand learning [28]. According to the former authors, profitability depends on incorporating substitution effects in inventory management. Substitution increases the demand for other items and affects optimal stock levels. According to the latter authors, it is vital for retailers to select which products to

offer due to the limited display capacity in the physical stores. Hence, these authors described different stock assortment policies and introduce a model for dynamic assortment planning. Similarly, Honhon et al. [11] proposed a dynamic programming algorithm to determine the optimal assortment in a single-period problem with stock out-based substitution. Rajamma et al. [29] described a method for determining inventory depth and variety breadth, as well as the mix between basic and seasonal clothing in fashion retail. Strategic decisions on the right variety and depth of in-store stock have been developed by Mantrala et al. [25]. They provided reviews on how to customize retail assortment at the store level, rather than simply using a centrally planned assortment for all stores. A complete review of stock assortment is provided by Kök et al. [30].

The product recommendation problem has received increasing attention recently. According to Liu and Shih [31], “recommender systems rely on customer purchase history to determine customer preferences and to identify products that customers may purchase”. Li et al. [32] proposed a framework for a localized product recommendation system associated with automatic vending machines. Their system offers suitable recommendations of localized products to customers in different locations. They developed a hybrid technique using a metaheuristic approach, a clustering technique, as well as classification and statistical methods. The importance of product recommendation in today’s omnichannel retailing world is also mentioned by Balakrishnan et al. [33]. They adopted an intuitive co-clustering algorithm for locating useful patterns in a 0-1 matrix, which studies the buying behavior of customers using historical data on past purchases. In order to handle the product recommendation problem for e-commerce applications, Baykal et al. [34] proposed a co-operation framework for multiple role-based reasoning agents. Choi and Cho [35] presented a similar product-finding algorithm for the collaborative business companies that share a product taxonomy table and have exchangeable product information. Choi et al. [36] proposed an online product recommendation system, which combines implicit rating-based collaborative filtering (CF) and sequential pattern analysis (SPA). The system derives implicit ratings by applying CF to online transaction data—even when no explicit rating information is available—and integrates CF and SPA for improving recommendation quality.

Zhao et al. [37] developed a novel product recommender system called METIS. Their system identifies, almost in real-time, users’ purchase trials from their microblogs. Then, it makes product recommendations based on matching the users’ demographic information—extracted from their public profiles—with product demographics learned from these microblogs and additional online reviews. Zhao et al. [38] proposed a novel solution for ‘cross-site’ and ‘cold-start’ product recommendation, which recommends products from e-commerce websites to users at social networking sites in cold-start situations. The term cold-start refers to users who do not have historical records on the items they have purchased. The authors proposed learning both users’ and product feature representations via data collected from e-commerce websites, using recurrent neural networks and then applying a modified gradient boosting trees method to transform users’ social networking features into user purchase preferences. More recently, Kaminskas et al. [16] addressed a particular product recommendation problem regarding small-scale retail websites, where the small number of returning customers makes traditional user-centric personalization techniques inapplicable. Hence, these authors applied an item-centric product recommendation strategy that combines two well-known methods—association rules and text-based similarity—for generating recommendations based on a single ‘seed’ product. Furthermore, their approach was also used to recommend products based on a set of seed products in a user’s shopping basket. The effectiveness of their recommendation approach is demonstrated, in the product-seeded and basket-seeded scenarios, through a series of experiments employing real customer data.

Product recommendation systems are related to the product assortment problem: a set of correlated products must be selected to be exposed (or recommended) in an exposition area with limited capacity. This selection of products should help to improve the experience of customers when visiting a store. In effect, by exposing an appropriate set of items at the display tables it is possible to increase the level of attraction of customers to the store, which directly influences the

customers’ experience and, hence, the sales revenue. Although this Section shows that research has been undertaken on both product recommendation systems and the product assortment problem, there is still a lack of work on how to combine different products in retail display tables at brick-and-mortar stores, especially when considering a multi-period time horizon and dynamic attractiveness values.

2.2. The Shelf Space Allocation Problem

A related issue is the shelf space allocation problem [39,40], which is also linked to the optimal selection of products to be displayed on shelves with limited space. In this problem, however, several items of the same product can be selected. According to Hübner and Kuhn [13], there are two different questions associated with the planning: assortment planning and shelf space planning. The first relates to listing decisions based on consumer choice, whereas the second deals with the limited shelf space. Our work considers both aspects, that is, which products should be shown in a limited space. Retail stores, especially brick-and-mortar ones, need to make decisions on what stock should be displayed in order to increase the customer’s attention. Despite the proliferation of numerous software applications in shelf space management, which make use of historical data, new algorithms—as the ones introduced in this paper—are needed to deal with multi-period and dynamic versions of the product selection problem. Most of the literature about the shelf space allocation problem refers to supermarket products [41,42]. Other areas, such as fashion stores, have not received much attention. Parsons [43] analyzed how the atmosphere in fashion stores influences sales. Gao et al. [44] investigated how pre-packs are used in retail distribution and how this reduces handling costs. Notice that fashion and apparel customers do not approach the store with a clear purchase objective, as it would be the case for food in supermarkets or electronic equipment. Instead, the customer enters the store just to look and see if something attracts his / her attention. It is the store’s function to appeal to them to make them buy from the store and not from another shopping channel, e.g., a mobile device, tablet, or personal computer.

Table 1 summarizes the literature on product recommendation, product assortment, and shelf space allocation problems according to the solving methodology employed. Among them, we highlight the use of dynamic programming methods, heuristics/metaheuristics, and data mining/machine learning strategies such as collaborative filtering, association rules, and sequential pattern analysis. From the literature review, one can notice that approximate methodologies have gained more popularity over the years. This fact can be explained by the continuous growth in the size of realistic instances.

Table 1. Classification of main articles by problem and solving methodology.

Study	Problem			Methodology		
	Product Recommendation	Assortment and Stock Optimization	Shelf Space Allocation	Exact	Heuristic or Metaheuristic	Other
Sadowskit [22]		•		•		
Mahajan and van Ryzin [23]		•				•
Choi and Cho [35]	•				•	
Baykal et al. [34]	•				•	
Liu and Shih [31]	•				•	•
Honhon et al. [11]		•		•		
Choi et al. [36]	•					•
Sauré and Assaf [28]		•		•		
Caro et al. [7]		•		•	•	
Gao et al. [44]		•				•
Li et al. [32]	•				•	•
Zhao et al. [37]	•					•
Bianchi-Aguiar et al. [42]			•		•	
Flamand et al. [41]			•		•	
Kaminskas et al. [16]	•					•
Balakrishnan et al. [33]	•					•

3. A Formal Description of the Multi-Period Product Display Problem

Consider a warehouse holding a set of products or items. It has to supply a retail store for different time periods, which define the planning horizon. Each item belongs to a certain collection (e.g., shirts or jeans in the case of clothes), has a selling price—which might vary with time—and a marginal profit—which is typically given as a percentage of the selling price. Depending on its selling price, an item is classified as ‘expensive’ or not. In any given period the retail store contains a set of tables, each of them displaying a subset of non-repeated items. Each item has an initial attractiveness value, estimated from experts’ opinions and/or historical observations in an omnichannel environment—such as the number of times it has been selected and analyzed in the past, the feedback provided by the customers, etc. The attractiveness value can also depend upon other items currently being displayed in the table, since relations (or dependencies) between pairs of products may need to be considered.

Among all the available products in the warehouse, a subset of different items should be selected to be displayed on the retail display tables. The dependency between each pair of items is registered in a dependency matrix. An inter-period dependency is also considered. The attractiveness value of each item is reduced by a known quantity (typically expressed as a percentage) every time the product is repeatedly shown in two (or more) consecutive periods. In other words, if an item is repeatedly exposed during several consecutive periods of time, its novelty disappears and, as a consequence, its attractiveness value is reduced. On the other hand, whenever an item has not been shown in the previous period, its attractiveness value is increased due to the novelty effect. The goal is then to solve a multi-period product display problem with dynamic attractiveness levels. In this problem, a subset of items has to be selected to be displayed at each table-period combination in order to maximize the aggregated attractiveness level over all periods. In order to make the problem more realistic, a number of additional constraints are also considered in this paper:

1. Collection constraint: the subset of items assigned to each table should cover at least a minimum number of goods from each collection, $l_c \geq 0$.
2. Price constraint: a minimum number of products at each table, $l_p \geq 0$, should belong to the expensive category.
3. Profit constraint: the profit margin of each table should be greater than a threshold defined by the manager, $l_m \geq 0$.

A set of consecutive time periods H is considered, together with a finite set of items, I , which are hosted in a warehouse. Each item $i \in I$ is associated with a base price $p_i > 0$, a marginal benefit $m_{ih} \geq 0$ (which might be different at each period $h \in H$), and an initial attractiveness value $v_{i0} > 0$. The final selling price of each item $i \in I$ at period $h \in H$ is given by $p'_{ih} \geq p_i$, i.e., $m_{ih} = p'_{ih} - p_i$. These items belong to a set of collections $C = \{c_1, c_2, \dots, c_{|C|}\}$, where $I = \bigcup_{k=1}^{|C|} c_k$. The subset of expensive items is given by $I_p = \{i \in I / p_i \geq p_0\}$, where p_0 is a threshold price value defined by the manager.

At each period $h \in H$, a subset S_h items must be exposed using a set of homogeneous tables T , with each table containing a total of $n > 0$ items. The decision variable x_{ith} is equal to 1 if item i is selected for table t in period h , and to 0, otherwise. Thus, the set of non-repeated selected items for each table $t \in T$ in period $h \in H$ is given by $S_{th} = \{i \in I / x_{ith} = 1\}$, being $S_h = \bigcup_{t \in T} S_{th}$. A matrix $D = [d_{ij}]_{i,j \in I}$ provides the existing interaction value, $d_{ij} \in \mathbb{R}$, between any pair of items $i, j \in I$. These intra-period dependencies account for the fact that some items might be positively or negatively correlated with others (i.e., showing them together might generate synergies or, on the contrary, might reduce their aggregated attractiveness). Apart from this intra-period dependencies between pairs of items, inter-period dependencies are also considered to account for the product’s novelty (or the lack of it). Accordingly, the attractiveness value of every item is a dynamic input, i.e., it is reduced or increased by a certain percentage factor depending on whether the item was displayed or not in the previous period. Thus, $\forall h \in \{1, 2, \dots, |H|\}$, the attractiveness value of item i in period h , v_{ih} , is recursively defined as:

$$v_{ih} = f(v_{i(h-1)}, x_{it(h-1)}, a, b, u, w) = \begin{cases} \text{Max}\{a, (1 - u)v_{i(h-1)}\} & \sum_{t \in T} x_{it(h-1)} = 1 \\ \text{Min}\{b, (1 + w)v_{i(h-1)}\} & \sum_{t \in T} x_{it(h-1)} = 0 \end{cases} ,$$

where $0 \leq a < b$ are bounds for the attractiveness values and $u, w \geq 0$ are decreasing or increasing percentage factors, respectively.

With this notation, the addressed problem can be formulated as follows:

$$\text{Max} \sum_{h \in H} \sum_{t \in T} \sum_{i \in I} v_{ih} x_{ith} + \sum_{h \in H} \sum_{t \in T} \sum_{i \in I} \sum_{j \in I / i < j} d_{ij} x_{ith} x_{jth}. \tag{1}$$

Subject to:

$$\sum_{i \in I} x_{ith} = n \quad \forall t \in T, \forall h \in H \tag{2}$$

$$\sum_{t \in T} x_{ith} \leq 1 \quad \forall i \in I, \forall h \in H \tag{3}$$

$$\sum_{i \in c_k} x_{ith} \geq l_c \quad \forall c_k \in C, \forall t \in T, \forall h \in H \tag{4}$$

$$\sum_{i \in I_p} x_{ith} \geq l_p \quad \forall t \in T, \forall h \in H \tag{5}$$

$$\sum_{i \in I} m_{ih} x_{ith} \geq l_m \quad \forall t \in T, \forall h \in H \tag{6}$$

$$v_{ih} = f(v_{i(h-1)}, x_{it(h-1)}, a, b, u, w) \quad \forall i \in I, \forall h \in H \setminus \{0\} \tag{7}$$

$$x_{ith} \in \{0, 1\} \quad \forall i \in I, t \in T, h \in H \tag{8}$$

$$v_{ih} \in [a, b] \quad \forall i \in I, h \in H \tag{9}$$

The objective function (1) maximizes the total attractiveness of the planning horizon by considering the individual attractiveness of the items and the intra-period dependencies between each pair of selected items in each displaying table and period. Equation (2) ensures that the number of items on each table $t \in T$ does not exceed a pre-defined value n . Equation (3) guarantees that each item i cannot be selected more than once in a given period $h \in H$. Equation (4) confirms that, inside each period, each table covers at least l_c items from each collection c_k . Equation (5) guarantees that, inside each period, each table contains at least l_p expensive items. Equation (6) ensures, for each period, that the profit margin of each table should be greater than l_m . Equation (7) introduces the inter-periods dynamic component in the attractiveness value of the items. Notice that this equation transforms the objective function into a non-smooth one due to the definition of the v_{ih} values. Equation (8) states that all decision variables are binary. Finally, Equation (9) bounds the values that variable v_{ih} can take. A ‘relaxed’ version of this problem can be obtained when no bounds are imposed on the attractiveness values of each item. In that case, the objective function becomes quadratic since Equation (10) can be rewritten as:

$$v_{ih} = v_{i(h-1)}(1 - u) \sum_{t \in T} x_{it(h-1)} + v_{i-1}(1 + w) \sum_{t \in T} (1 - x_{it(h-1)}) \quad \forall i \in I, \forall h \in H \setminus \{0\} \tag{10}$$

4. Our BR-GRASP and BR-ILS Solving Approaches

In this paper, biased-randomized versions of the well-known GRASP [45] and the ILS [20] metaheuristics are proposed to solve the multi-period product display problem with dynamic attractiveness. Both algorithms consist of some common stages: (i) a construction stage, in which a feasible initial solution is built taking into account the constraints; and (ii) an improvement stage,

in which a local search is applied to the initial solution in order to enhance its quality. Apart from these two common stages, the ILS incorporates a perturbation phase and an acceptance criterion phase. As discussed in Resende and Ribeiro [46] and Graspas et al. [47], both GRASP and ILS are relatively easy-to-implement and flexible metaheuristic frameworks that have shown their efficiency in solving different optimization problems, including both deterministic and stochastic ones. They typically do not require many parameters or time-consuming fine-tuning processes. The previous properties make them especially suitable for industrial applications. Moreover, they have been successfully combined with biased-randomization techniques in multiple occasions [21,48–50].

In the BR-GRASP approach, a solution is built iteratively, element by element, and then improved via a local search procedure. This two-step process is repeated until a number of iterations (or a maximum running time) is reached. Then, the best-found solution is returned. On the other hand, the BR-ILS starts from a base solution, which is repeatedly perturbed (modified using a destruction-reconstruction process), enhanced via a local search procedure, and finally evaluated by an acceptance criterion until a stop condition is met. In both the BR-GRASP and the BR-ILS, a partial solution is constructed for each new period (taking into account the current configuration of the display tables). Then, this partial solution is improved through a local search procedure. The low-level details of these algorithms are provided next.

4.1. Introducing BR into GRASP and ILS

The use of Monte Carlo sampling techniques to enhance the performance of constructive heuristics was proposed by Faulin and Juan [51]. In our approach, more advanced biased-randomized techniques—based on the use of a geometric probability distribution—are used every time a new solution is constructed or partially reconstructed after a perturbation phase. BR techniques differ from standard selection strategies, which are usually based on a greedy criterion or on the use of a uniform probability distribution to select the next candidate from a list. Thus, for example, in a classical GRASP framework, a restricted list of candidates (RLC) is considered, and a uniform probability distribution is used to choose a candidate from this RLC. However, in a BR-GRASP, an unrestricted candidate list (UCL) is employed. This UCL is sorted according to some logical criterion, and a geometric distribution is used to select the next element from this sorted UCL [21]. The geometric distribution uses a single parameter, β , which is proportional to the probability of selecting the first element in the UCL. All elements in the UCL receive a probability of being selected, which is higher the closer the element is to the top of the sorted UCL. The same concept is also employed during the solution-construction processes inside our BR-ILS algorithm.

Pseudocode 1 illustrates this solution-construction process. All items (set I) are included in the UCL, which is sorted in descending order according to the ‘adjusted’ attractiveness value of each item, i.e., the original attractiveness value in the corresponding period is corrected to consider dependencies with respect to other items already in the display table. Then, the geometric distribution is used to randomly build a solution by selecting one ‘promising’ item at a time. Once selected, the element is removed from the UCL, the adjusted attractiveness values and profit margin of the remaining elements are updated, and the UCL is sorted again.

Pseudocode 1: Construction stage with biased randomization.

Data: Distribution parameter β for the geometric probability distribution

- 1 $s \leftarrow \emptyset$
- 2 Initialize candidate list set: $UCL \leftarrow I$
- 3 Order UCL according to sorting criterion $c(\cdot)$
- 4 **while** solution s is not completed **do**
- 5 Randomly select $pos \in \{1, \dots, |UCL|\}$ according to distribution $Geom(\beta)$
- 6 $s \leftarrow s \cup \{UCL[pos]\}$
- 7 $UCL \leftarrow UCL \setminus \{UCL[pos]\}$
- 8 Re-order UCL
- 9 **end**
- 10 return s

The efficiency of similar BR strategies has been extensively discussed in different studies, such as Dominguez et al. [52,53], Juan et al. [54], and Martin et al. [55].

4.2. Constructing an Initial Solution for the Current Period

For each period in the planning horizon, an initial solution is built by adding products to the display tables. We assume that these tables are empty at the beginning of the period. Each product is assumed to have an initial attractiveness value at the beginning of the first period. This attractiveness value is based on historical observations and, possibly, expert judgment.

Using the BR strategy previously described, a subset of items is assigned to a given display table. First, a subset S_{th} is built by selecting n products for table t in period h . At this point, the collection constraint is incorporated into the construction procedure by selecting a minimum number of items from each collection. Next, the price-related constraint and the profit-margin constraint are checked for the S_{th} subset. If both constraints are satisfied, the next table is considered. Otherwise, the solution is repaired. During the repair process, an item is randomly selected from S_{th} and replaced by another item not in S_{th} . In the case of the price constraint, for instance, this swapping process is based on the replacement of items from one price category (e.g., non-expensive) by products belonging to another one (e.g., expensive). In the case of the profit-margin constraint, the swapping process is based on the replacement of items with a low-profit margin by products with a high-profit margin. This process is repeated until a feasible configuration of tables is eventually achieved for the current period. After obtaining a feasible configuration for the current period, an improvement stage is applied as described next.

4.3. Improving the Solution of the Current Period

Pseudocode 2 illustrates the improvement procedure applied to each table in the current period. It starts from the first table t in the given period h . An item i is randomly selected from t and removed. Then it is replaced by another randomly-selected item $j \in I$ among those that can be inserted without violating any constraint. As a result, a new table t' is generated. The adjusted attractiveness value of t' is updated taking into account the dependencies between pairs of items. If the adjusted attractiveness value of t' is greater than that of t , the latter table is updated and the counter is reset. Otherwise, another item is randomly chosen until a maximum number of iterations is achieved without any improvement. The same process is applied to the remaining tables in the current period h . At each iteration of the constructive procedure, the articles' attractiveness and profit margins are conveniently updated for the next periods of the planning horizon.

Pseudocode 2: Improving the configuration of tables in period h .

Data: period h , maximum number of iterations without improvement $iter_{max}$

```

1 articleToRemove  $\leftarrow$  null
2 articleToInsert  $\leftarrow$  null
3 for each table  $t \in T$  of  $h$  do
4   counter  $\leftarrow$  0
5   while counter <  $iter_{max}$  do
6      $t' \leftarrow t$ 
7     itemToRemove  $\leftarrow$  random item  $i \in S_{t'h}$ 
8     itemToInsert  $\leftarrow$  random non-selected article  $j \in I$  that keeps the feasibility of  $t'$ 
9     Remove itemToRemove from  $t'$ 
10    Insert itemToInsert into  $t'$ 
11    Update total attractiveness and correlation of  $t'$ 
12    if  $tableAttractiveness(t') > tableAttractiveness(t)$  then
13       $t \leftarrow t'$ 
14      counter  $\leftarrow$  0
15    else
16      counter  $\leftarrow$  counter + 1
17    end
18  end
19 end

```

4.4. Perturbation Stage in the BR-ILS

Both the construction of an initial solution and the improvement process are employed in the BR-GRASP and the BR-ILS. However, the BR-ILS also makes use of a base solution which is modified via a perturbation stage. In our case, the perturbation is characterized by a destruction-reconstruction process as described in Pseudocode 3. After selecting a starting period, h_{start} , all posterior periods are destroyed and then re-built, using the previously described constructive and improvement stages.

The starting period is chosen according to a varying percentage of destruction, $k \in (0, 1]$. Hence, for example, if $k = 0.1$ then the last 10% of the periods are destroyed and reconstructed (due to the dependencies across periods, if a period is rebuilt all posterior periods need to be recomputed).

Pseudocode 3: Perturbation stage

Data: percentage of destruction k

```

1  $h_{start} \leftarrow |H| - k * |H|$ 
2 newSol  $\leftarrow$  baseSol
3 for each period  $h \in H$  of newSol starting from  $h_{start}$  do
4   newSol  $\leftarrow$  newSol  $\setminus$  { $h$ }
5    $h^* \leftarrow reconstruction(h)$ 
6    $h^* \leftarrow improvement(h^*)$ 
7   newSol  $\leftarrow$  newSol  $\cup$  { $h^*$ }
8 end
9 if newSol is better than baseSol then
10  baseSol  $\leftarrow$  newSol
11 end

```

4.5. Acceptance Criterion Stage in the BR-ILS

Finally, the ILS metaheuristic also incorporates an acceptance criterion to reduce the probability of getting trapped in a local minimum (Pseudocode 4). In our case, we use the demon-based acceptance

criterion described in Juan et al. [56]. In this criterion, newly generated solutions are compared with the base solution, and the former is updated in two cases: (i) when the new solution is better than the base solution; or (ii) when the new solution is worse than the base solution, but the difference in value is lower than the improvement (credit) obtained in the last update of the base solution.

Pseudocode 4: Credit-based acceptance criterion.

```

1  $\delta \leftarrow \text{cost}(\text{newSol}) - \text{cost}(\text{baseSol});$ 
2 if  $\delta \leq 0$  then
3    $\text{credit} \leftarrow -\delta$ 
4    $\text{baseSol} \leftarrow \text{newSol}$ 
5   if  $\text{cost}(\text{baseSol}) < \text{cost}(\text{bestSol})$  then
6      $\text{bestSol} \leftarrow \text{baseSol}$ 
7   end
8 end
9 if  $0 < \delta \leq \text{credit}$  then
10   $\text{credit} \leftarrow 0$ 
11   $\text{baseSol} \leftarrow \text{newSol}$ 
12 end

```

5. Computational Experiments

This section describes the experimental setup designed to evaluate the performance of our BR-GRASP and BR-ILS algorithms. To the best of our knowledge, this is the first work solving a rich and realistic version of the multi-period product display problem with dynamic attractiveness. Hence, we had to generate a complete set of benchmarks with different characteristics to comprehensively evaluate and test the proposed algorithms. These characteristics are: number of articles ($|I|$), number of display tables ($|T|$), number of collections ($|C|$), and number of items per display table (n).

For small-sized instances (aimed at being solved using non-linear exact methods), we set $|I| \in \{25, 50, 75, 100\}$, $|T| \in \{2, 3, 4\}$, $|C| = 5$, $|H| \in \{2, 3, 4\}$, and $n = 6$. Each of these instances was named according to these specifications. Thus, for example, instance 75i-5c-4p-4t-6it consists of 75 items, 5 collections, 4 periods, 4 tables per period, and 6 items per table. Similarly, for the large-sized instances we set $|I| \in \{500, 1000, 1500, 2000\}$, $|T| \in \{5, 10\}$, $|C| = 4$, $|H| = 12$, and $n = 10$. Each of these instances was named according to these specifications. Thus, for example, instance a500m5i1 consists of 500 products and 5 tables. The last index represents different instances using the same combination of items and tables. For the purpose of numerical experimentation, most of the specific inputs in these instances (e.g., initial attractiveness values, dependencies between pairs of products, item prices, and associated collection) have been randomly generated. In order to facilitate reproducibility of the experiments, all these instances and inputs are publicly available at https://www.researchgate.net/publication/330675091_instancesMPPDPDA.

Although all input data is available in the previous link, an overview of these inputs is provided next for the case of the large-sized instances. The final selling price of each product is generated according to a uniform distribution in the range $[10, 150]$ (monetary units). The profit-margin percentage for each product follows a uniform distribution in the range $[10\%, 35\%]$. The price and profit margin are considered to generate the absolute profit, which is used to check whether the respective constraints are satisfied. The initial attractiveness value for each item and the between-items dependencies are generated according to a uniform distribution in the range $[10, 100]$ and $[-35, 35]$, respectively. Regarding the considered constraints, the subset of selected products at each table should cover at least 20% of each collection. Regarding prices, the items are categorized into two different categories: those with a cost inferior to 60 monetary units are considered as non-expensive items. The rest are considered expensive. In our experiments, we required that the selected subset at each table should include at least 50% of expensive products. Finally, for each table, the profit margin

per table is set at 100 monetary units or more. To account for the ‘novelty factor’, when a product is displayed on a table during a given period, its attractiveness value is decreased by 10% for the next period (always considering that the minimum attractiveness value that a product can reach is bounded by the modeling parameter $a \geq 0$). Conversely, if a product is not displayed at a given period, its attractiveness value increases by 3% for the next period (up to a maximum value given by the modeling parameter $b > a$).

After some initial tests, a geometric probability distribution with a parameter β randomly chosen in the interval (0.80,0.99) was used for the biased-randomization process during the solution-construction stage. The stopping criterion for the BR-GRASP and BR-ILS is defined by a maximum computing time, t_{max} , defined as: $t_{max} = 0.5 \cdot |I| \cdot |T| \cdot |H|$. In practice, this represents approximately 15 seconds of execution for instances composed of 500 articles, 5 tables, and a planning horizon of 12 days, for example. Regarding the improvement stage, the stopping criterion is set to $it_{max} = 1000$ iterations without observing any improvement. Our algorithms were coded in Java and run on a standard PC with an Intel Core i5 CPU at 2.7 GHz and 8 GB RAM.

Tables 2 and 3 summarize the experimental setup and parameters for the computational experiments. Note that a new parameter, r , is introduced. This parameter refers to the percentage of profit margin (and price) reduction in specific sales periods.

Table 2. Problem parameters.

$ H $	h_s	h_e	r	u	w	l_c	l_p	l_m	b	a
12	11	12	10%	10%	3%	20%	50%	100	150	0

Table 3. Methodologies Parameters

α	β	t_{max}	it_{max}	k
0.1	[0.8,1)	$0.5 \cdot I \cdot T \cdot H $	1000	{0.1,0.3,0.4,0.55,0.7,0.8,1}

6. Analysis of Results

As discussed in Section 3, the relaxed version of the problem is non-linear. It might be solved using non-linear exact methods, at least up to a certain size. Hence, in order to compare the solutions generated by our BR-GRASP algorithm with the ones provided by the non-linear solver, we first run a set of small-sized instances. In a second experiment, the proposed large-sized instances are employed to test our methodologies under more realistic (large-scale and non-smooth) scenarios.

6.1. Small-Sized Instances and Limitations of Non-Linear Solvers

A set of small-sized instances was generated in order to test the performance of our BR-GRASP algorithm when compared to state-of-the-art non-linear solvers. As explained in the previous section, these instances differ in the number of articles, number of tables, and the length of the horizon. Table 4 presents the results of a numerical analysis discussing the limitations of these solvers even when dealing with the relaxed version of the problem. The overall best-known solution (BKS) for each instance is provided. According to our experiments, only instances with up to 600 binary variables and 200 constraints (e.g., 50 items, four periods, and three tables per period) can be solved in reasonable computing times using modern non-linear solver engines such as COIN-OR Bonmin and NEOS Bonmin [57]. Notice that our BR-GRASP algorithm (BR-GR) is very competitive when compared with the non-linear solver engines, obtaining better or similar solutions in much shorter computing times.

Table 4. Comparison of results between the non-linear solvers and our BR-GRASP approach (relaxed version of the problem).

Instance	Instance Details					BKS	GAP (%) w.r.t. BKS			Time (s)		
	I	H	T	bin var	constraints		COIN-OR	NEOS	BR-GR	COIN-OR	NEOS	BR-GR
25i-5c-2p-2t-6it	25	2	2	100	> 50	311.4	4.82	3.21	0.00	19	23	1
25i-5c-3p-2t-6it	25	3	2	150	> 75	440.4	2.04	2.68	0.00	10	23	1
25i-5c-4p-2t-6it	25	4	2	200	> 100	562.9	0.00	1.03	1.49	37	69	4
50i-5c-4p-2t-6it	50	4	2	400	> 200	768.7	0.91	0.00	0.85	196	46	5
50i-5c-4p-4t-6it	50	4	4	800	> 200	1250.3	0.00	2.62	2.58	> 700	> 300	3
75i-5c-2p-2t-6it	75	2	2	300	> 150	392.4	3.52	0.00	2.47	50	79	5
75i-5c-2p-3t-6it	75	2	3	450	> 150	557.7	3.39	1.27	0.00	148	165	5
75i-5c-3p-3t-6it	75	3	3	675	> 225	1033.9	N/A	N/A	0.00	> 3600	> 1200	64
75i-5c-3p-4t-6it	75	3	4	900	> 225	1352.1	N/A	N/A	0.00	> 3600	> 1200	83
100i-5c-2p-3t-6it	100	2	3	600	> 200	583.1	N/A	1.73	0.00	> 3600	> 120	34
100i-5c-3p-2t-6it	100	3	2	600	> 300	631.6	0.24	0.00	1.35	> 700	> 300	4
100i-5c-3p-4t-6it	100	3	4	1200	> 300	1122.7	N/A	N/A	0.00	> 3600	> 1200	64
100i-5c-4p-4t-6it	100	4	4	1600	> 400	1496.9	N/A	N/A	0.00	> 3600	> 1200	61

6.2. Large-Sized Instances with Fixed Profit Margins

Due to the satisfactory performance of our biased-randomized approach in small-sized instances, the BR-GRASP and the BR-ILS were also tested in solving more realistic (large-scale) instances of the complete (non-smooth) version of the problem. In this case, no variations in the selling price of the products were considered. A lower bound on the attractiveness value any item can achieve is imposed. In our experiments, we set $a = 0$, while b takes a sufficiently large value (i.e., in practice the attractiveness value of an item grows after each period in which the item has not been displayed). For each of the 40 generated instances, 10 runs were performed (each run using a different seed for the pseudo-random number generator). To evaluate the performance of each methodology, we consider the percentage gap between the best-found solution using that methodology (i.e., the one with the highest attractiveness value) and the BKS obtained with any solution methodology. Thus, the lower the gap, the better the performance of the methodology.

The solutions generated by our BR-GRASP and BR-ILS algorithms are compared with those generated by a greedy strategy, a ‘standard’ GRASP, and a ‘standard’ ILS (i.e., without considering BR techniques). The greedy methodology approximates human behavior (e.g., an expert manager) when selecting articles to be displayed. In the standard GRASP algorithm, and after some preliminary tests, the parameter α which controls the size of the RCL was set to 0.1. The standard ILS algorithm uses a greedy selection technique to build solutions. Regarding the perturbation stage, the destruction rate was defined by the set $k \in \{0.1, 0.3, 0.4, 0.55, 0.7, 0.8, 1\}$.

Table 5 provides the summarized results of the solutions generated by the greedy (G), BR-GRASP (BR-GR), GRASP (GR), ILS and BR-ILS methodologies. For each instance and methodology, the following data is provided: the attractiveness value of the best-found solution, the average (AVG) attractiveness value obtained after 10 runs, as well as its corresponding standard error (SE), the gap with respect to the BKS, and the average CPU time employed to find the corresponding solutions. The best-found solution for each instance is presented in bold.

Table 5. Comparison of the results obtained by the proposed methodologies.

Instance	Best Solution (attractiveness)					AVG Attractiveness (SE)					GAP (%) w.r.t. BKS					AVG Time (sec.)				
	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS
a500m5i1	100,303	104,887	104,726	103,882	104,447	100,303 (0)	104,540 (71)	104,418 (75)	103,676 (46)	104,131 (63)	4.37	0.00	0.15	0.96	0.42	0	8	7	10	12
a500m5i2	101,106	105,600	105,700	104,344	105,077	101,106 (0)	105,295 (59)	105,258 (83)	104,143 (53)	104,903 (48)	4.35	0.09	0.00	1.28	0.59	0	7	7	11	9
a500m5i3	100,313	104,955	104,721	103,930	104,688	100,313 (0)	104,596 (76)	104,506 (50)	103,614 (51)	104,295 (59)	4.42	0.00	0.22	0.98	0.25	0	8	7	12	12
a500m5i4	100,606	105,026	104,834	103,887	104,597	100,606 (0)	104,735 (59)	104,589 (48)	103,531 (61)	104,290 (59)	4.21	0.00	0.18	1.08	0.41	0	8	7	11	14
a500m5i5	100,633	105,745	105,700	104,647	105,018	100,633 (0)	105,293 (88)	105,165 (84)	104,127 (102)	104,701 (54)	4.83	0.00	0.04	1.04	0.69	0	7	8	10	10
a500m10i1	188,711	196,334	195,685	194,764	195,811	188,711 (0)	195,901 (106)	195,382 (77)	194,194 (113)	195,440 (98)	3.88	0.00	0.33	0.80	0.27	0	15	14	20	23
a500m10i2	189,668	197,098	196,662	195,567	196,288	189,668 (0)	196,734 (99)	196,347 (69)	195,086 (95)	195,957 (57)	3.77	0.00	0.22	0.78	0.41	0	14	15	21	20
a500m10i3	187,187	194,977	194,415	193,682	194,283	187,187 (0)	194,538 (108)	193,989 (66)	193,063 (111)	193,863 (65)	4.00	0.00	0.29	0.66	0.36	0	15	16	18	22
a500m10i4	187,302	196,927	195,653	195,125	195,885	187,302 (0)	196,421 (104)	195,232 (107)	194,526 (106)	195,611 (52)	4.89	0.00	0.65	0.91	0.53	0	15	15	18	20
a500m10i5	186,074	193,498	192,934	192,147	192,835	186,074 (0)	193,147 (83)	192,646 (77)	191,537 (116)	192,415 (93)	3.84	0.00	0.29	0.70	0.34	0	15	15	17	21
a1000m5i1	110,916	113,781	112,947	113,166	113,851	110,916 (0)	113,602 (45)	112,708 (58)	112,958 (58)	113,369 (73)	2.58	0.06	0.79	0.60	0.00	0	15	14	15	24
a1000m5i2	111,172	114,672	113,486	113,737	114,233	111,149 (0)	114,306 (53)	113,263 (37)	113,221 (107)	114,034 (32)	3.05	0.00	1.03	0.82	0.38	0	16	14	20	18
a1000m5i3	110,216	114,088	112,895	112,966	113,453	110,216 (0)	113,675 (73)	112,515 (66)	112,717 (50)	113,266 (58)	3.39	0.00	1.05	0.98	0.56	0	16	16	23	21
a1000m5i4	110,958	114,182	113,100	113,060	113,871	110,958 (0)	113,907 (71)	112,770 (61)	112,841 (51)	113,437 (65)	2.82	0.00	0.95	0.98	0.27	0	16	16	25	24
a1000m5i5	110,969	114,362	113,164	113,054	113,894	110,969 (0)	113,998 (68)	112,859 (52)	112,851 (44)	113,561 (51)	2.97	0.00	1.05	1.14	0.41	0	15	15	15	24
a1000m10i1	212,437	218,018	215,554	216,393	217,147	212,437 (0)	217,608 (86)	214,770 (117)	215,818 (110)	216,752 (80)	2.56	0.00	1.13	0.75	0.40	0	32	30	36	48
a1000m10i2	209,689	215,681	212,682	214,212	215,081	209,689 (0)	215,214 (87)	212,175 (83)	213,865 (67)	214,456 (114)	2.78	0.00	1.39	0.68	0.28	0	30	33	54	36
a1000m10i3	211,045	216,982	214,111	215,294	216,723	211,045 (0)	216,443 (89)	213,544 (94)	214,809 (107)	215,976 (111)	2.74	0.00	1.32	0.78	0.12	0	34	30	38	47
a1000m10i4	210,869	216,560	213,698	215,304	216,233	210,869 (0)	216,190 (109)	213,133 (123)	214,747 (112)	215,721 (73)	2.63	0.00	1.32	0.58	0.15	0	28	30	31	39
a1000m10i5	212,916	217,943	214,600	216,719	217,337	212,916 (0)	217,506 (84)	214,376 (58)	216,105 (144)	216,980 (64)	2.31	0.00	1.53	0.56	0.28	0	29	31	51	41
a1500m5i1	114,511	117,763	115,996	116,406	117,318	114,511 (0)	117,377 (74)	115,742 (50)	116,203 (69)	117,021 (67)	2.76	0.00	1.50	1.15	0.38	0	23	22	30	40
a1500m5i2	116,116	119,709	117,457	118,930	119,328	116,116 (0)	119,422 (42)	117,272 (32)	118,341 (114)	119,007 (63)	3.00	0.00	1.88	0.65	0.32	0	22	21	34	29
a1500m5i3	115,237	118,749	116,645	117,247	118,234	115,237 (0)	118,416 (71)	116,501 (32)	117,069 (43)	117,972 (56)	2.96	0.00	1.77	1.26	0.43	0	23	23	35	38
a1500m5i4	115,523	118,600	116,448	117,569	118,433	115,523 (0)	118,250 (72)	116,241 (56)	117,150 (84)	117,886 (69)	2.59	0.00	1.81	0.87	0.14	0	22	22	33	32
a1500m5i5	115,004	118,189	116,218	116,955	117,785	115,004 (0)	117,846 (76)	115,926 (57)	116,755 (69)	117,530 (54)	2.69	0.00	1.67	1.04	0.34	0	23	20	34	33
a1500m10i1	223,333	228,339	223,706	227,220	228,394	223,333 (0)	227,788 (119)	223,078 (93)	226,430 (125)	227,530 (101)	2.22	0.02	2.05	0.51	0.00	0	46	47	63	64
a1500m10i2	222,677	227,425	222,844	226,245	226,910	222,677 (0)	226,926 (96)	222,310 (78)	225,629 (113)	226,470 (98)	2.09	0.00	2.01	0.52	0.23	0	46	42	51	60
a1500m10i3	222,409	227,296	222,936	225,490	226,682	222,409 (0)	226,912 (119)	222,376 (109)	224,955 (96)	226,308 (91)	2.15	0.00	1.92	0.79	0.27	0	42	45	62	71
a1500m10i4	223,322	228,323	223,507	226,937	228,157	223,322 (0)	228,088 (55)	223,156 (79)	226,553 (96)	227,556 (92)	2.19	0.00	2.11	0.61	0.07	0	45	47	62	60
a1500m10i5	220,092	226,651	222,442	225,253	225,993	220,092 (0)	226,298 (61)	222,058 (68)	224,645 (133)	225,547 (80)	2.89	0.00	1.86	0.62	0.29	0	42	45	64	62
a2000m5i1	117,467	121,200	118,831	119,871	120,632	117,467 (0)	120,624 (74)	118,228 (74)	119,624 (54)	120,326 (60)	3.08	0.00	1.95	1.10	0.47	0	30	31	50	52
a2000m5i2	118,295	120,996	118,308	120,190	120,820	118,295 (0)	120,696 (68)	118,014 (52)	119,833 (81)	120,330 (72)	2.23	0.00	2.22	0.67	0.15	0	32	28	49	37
a2000m5i3	118,151	121,349	118,769	120,134	121,062	118,151 (0)	121,057 (53)	118,535 (57)	119,733 (70)	120,696 (70)	2.64	0.00	2.13	1.00	0.24	0	29	34	39	44
a2000m5i4	118,308	121,848	119,468	120,835	121,471	118,308 (0)	121,489 (76)	119,064 (68)	120,500 (58)	121,168 (57)	2.90	0.00	1.95	0.83	0.31	0	31	31	40	44
a2000m5i5	118,841	121,966	119,270	120,975	121,487	118,841 (0)	121,675 (42)	119,002 (46)	120,589 (74)	121,295 (65)	2.56	0.00	2.21	0.81	0.39	0	30	29	37	46
a2000m10i1	231,988	235,713	229,199	234,075	235,099	231,988 (0)	235,256 (84)	228,886 (62)	233,659 (106)	234,768 (72)	1.58	0.00	2.76	0.69	0.26	0	60	59	85	79
a2000m10i2	230,797	235,691	228,935	233,606	234,478	230,797 (0)	234,750 (135)	228,532 (76)	233,133 (83)	234,044 (62)	2.08	0.00	2.87	0.88	0.51	0	59	62	75	71
a2000m10i3	230,359	235,430	229,583	234,145	234,834	230,359 (0)	235,323 (33)	229,172 (80)	233,493 (184)	234,577 (48)	2.15	0.00	2.48	0.55	0.25	0	62	65	85	91
a2000m10i4	229,622	234,369	228,526	232,734	234,181	229,622 (0)	234,055 (64)	228,248 (55)	232,485 (61)	233,604 (87)	2.03	0.00	2.49	0.70	0.08	1	58	59	69	80
a2000m10i5	229,995	234,959	228,768	232,876	234,346	229,995 (0)	234,618 (98)	228,429 (76)	232,538 (85)	233,976 (69)	2.11	0.00	2.64	0.89	0.26	0	63	63	102	85
AVG	-	-	-	-	-	-	-	-	-	-	2.98	0.00	1.41	0.83	0.31	0	28	28	39	40

From Table 5, we can notice that the BR-GRASP and the BR-ILS (in that order) outperform the other methodologies. In particular, the BR-GRASP provides the BKS in 37 out of 40 tested instances. Although the BR-GRASP was not able to find the BKS in some cases, it was able to find solutions with a minor deviation (less than 0.10%) from it. According to our experiments, the greedy methodology runs extremely fast, since this methodology does not incorporate a local search mechanism and it is not embedded inside a multi-start process. Comparing the remaining methodologies, all of them require similar computing times, although the BR-GRASP reaches high-quality solutions faster than other approaches.

Figure 2 shows, for each methodology, a boxplot of the percentage gaps with respect to the BKS. From this Figure, one can conclude that the greedy methodology—which could be assimilated to a human behavior—represents by far the worst approach. In effect, it shows an average gap of about 3% with respect to the BKS, a gap that can grow up to 5% in some instances. It can be seen that biased-randomized techniques enhance the ILS and GRASP approaches. The main reason why our BR-GRASP performs slightly better than our BR-ILS might be the existence of inter-period dependencies. These inter-period dependencies might sometimes penalize partial-scope destruction-reconstruction processes (ILS) versus full-scope destruction-reconstruction ones (GRASP), since the former might get trapped more easily in a local minimum. ANOVA and Fisher tests were run in order to analyze if the performance differences were statistically significant. As expected, the ANOVA test resulted in the existence of significant differences among the algorithms (p -value = 0.000). Actually, according to the Fisher pairwise test, except for the comparison between algorithms BR-GRASP and BR-ILS, all other differences in performance are statistically significant.

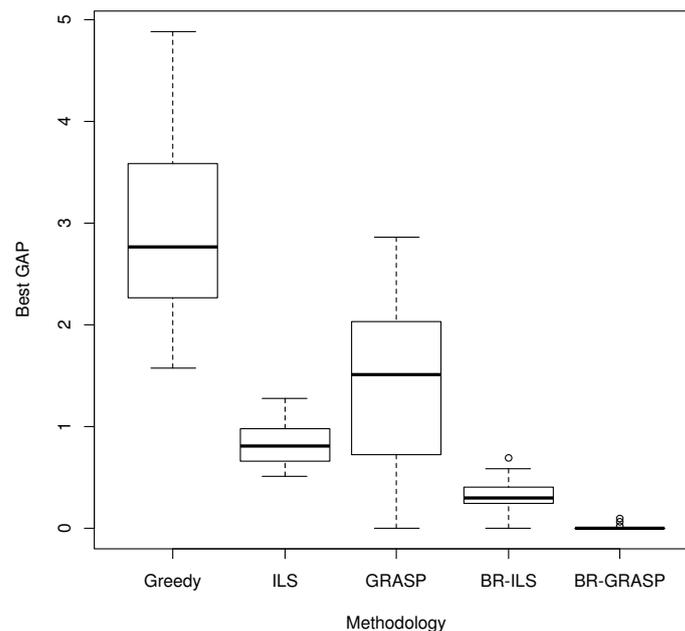


Figure 2. Comparison of percentage gaps w.r.t. the best-known solution (BKS) achieved by each methodology.

Figure 3 illustrates the convergence behavior with time of the BR-GRASP algorithm for an instance composed of 500 articles, 5 tables, and 5 periods. The convergence is analyzed by period P_i , $i \in \{1, 2, \dots, 5\}$, which runs for 1.25 seconds ($0.5 \times 500 \times 5$ milliseconds per period) for this problem. Notice that all periods present similar convergence behavior with time. Additionally, the range value of attractiveness is similar across the periods. This is explained by the small increment in attractiveness (3%) associated with new items, as well as by the large reduction in attractiveness (10%) associated with items that have lost their novelty effect.

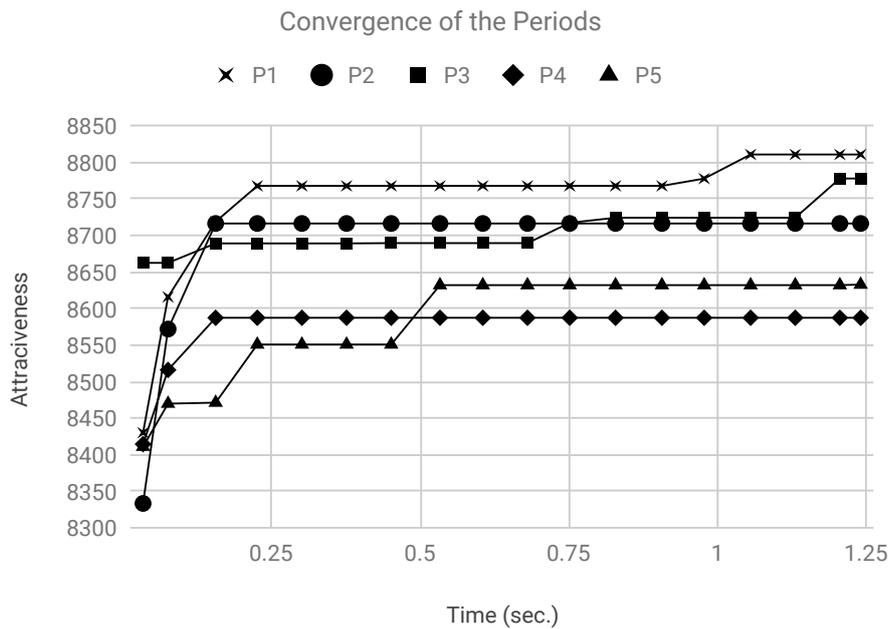


Figure 3. Convergence chart for the BR-GRASP in a particular instance.

6.3. Large-sized Instances with Varying Profit Margins (Selling Prices)

During the considered horizon, one could consider a reduction in product prices (and margins) due to the sales period. For running an experiment under these conditions, the parameters h_s and h_e , which represent the beginning and end of the especial sales period, were set to $h_s = 11$ and $h_e = 12$. The remaining parameters were not changed. Table 6 provides the summarized results of the solutions generated by our methodologies. For each problem magnitude, the first instance was considered, and a comparison among the solving methods was provided in terms of percentage gap.

From Table 6, one can notice that the BR versions of GRASP and ILS outperform the other solving methodologies. In this scenario with varying profit margin, the BR-GRASP is able to provide the BKS for the eight tested instances. For the remaining methodologies, their performance—in terms of gaps, SE, and CPU times—is similar to the one presented in Table 5.

Table 6. Comparison of the results obtained by the proposed methodologies.

Instance	Best Solution (attractiveness)					AVG Attractiveness (SE)					GAP (%) w.r.t. BKS					AVG Time (s)				
	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS	G	BR-GR	GR	ILS	BR-ILS
a500m5i1	100,167	104,906	104,748	103,981	104,422	100,167 (0)	104,497 (73)	104,440 (51)	103,482 (82)	104,104 (66)	4.52	0.00	0.15	0.88	0.46	0	8	8	8	10
a500m10i1	188,711	196,372	195,804	194,621	195,815	188,711 (0)	196,021 (78)	195,541 (78)	193,967 (108)	195,370 (78)	3.90	0.00	0.29	0.89	0.28	0	15	17	15	20
a1000m5i1	110,916	113,875	113,097	113,223	113,515	110,916 (0)	113,630 (53)	112,708 (48)	112,744 (79)	113,292 (54)	2.60	0.00	0.68	0.57	0.32	0	15	14	17	21
a1000m10i1	212,437	217,987	215,415	216,246	217,287	212,437 (0)	217,540 (111)	214,911 (100)	215,857 (95)	217,058 (60)	2.55	0.00	1.18	0.80	0.32	0	30	30	36	40
a1500m5i1	114,397	117,609	115,983	116,460	117,437	114,397 (0)	117,289 (61)	115,750 (47)	116,112 (72)	117,057 (58)	2.73	0.00	1.38	0.98	0.15	0	23	23	41	36
a1500m10i1	223,202	228,676	223,475	226,916	227,794	223,202 (0)	227,827 (159)	223,053 (75)	226,517 (81)	227,418 (68)	2.39	0.00	2.27	0.77	0.39	0	48	44	69	66
a2000m5i1	117,377	120,696	118,292	119,967	120,628	117,377 (0)	120,545 (41)	118,147 (43)	119,599 (59)	120,442 (42)	2.75	0.00	1.99	0.60	0.06	0	33	32	40	30
a2000m10i1	231,988	236,010	229,241	234,086	235,096	231,988 (0)	235,431 (111)	229,029 (60)	233,674 (89)	234,831 (64)	1.70	0.00	2.87	0.82	0.39	0	60	57	65	63
AVG	-	-	-	-	-	-	-	-	-	-	2.89	0.00	1.35	0.79	0.29	0	29	28	37	36

7. Conclusions

Increasing levels of competitiveness among brands as well as among channels of the same brand make it difficult for retailers in brick-and-mortar stores to engage customers while in the shop. One of the ways to attract them to the stores is to offer a different experience and a factor of surprise. Displaying a set of correlated and attractive products on retail display tables that vary often is a promising way to engage customers with a pleasant experience. From a managerial perspective, being able to know the selection of products that maximizes the attractiveness level enables a rationalization of the stock available in the store. Moreover, reducing the time required to make these decisions might significantly increase the productivity of the managers in charge of them.

In this paper, we propose a rich and realistic multi-period product display problem, as well as biased-randomized algorithms that allow us to solve it in an efficient way. In the considered problem, a set of correlated products has to be selected over multiple periods of time in order to maximize the total attractiveness level of the display tables in a retail store. A number of realistic characteristics and constraints have been incorporated in the problem to increase the potential applications of our work. Some of these are: (i) the inclusion of both expensive and non-expensive products on each display table and horizon; (ii) the achievement of a minimum profit margin per table and horizon; (iii) the consideration of dynamic (novelty-based) and correlated (combination-based) attractiveness levels; and (iv) the consideration of dynamic selling prices.

As solution approaches, a biased-randomized GRASP and a biased-randomized ILS have been proposed. To test these methodologies, a complete set of instances was generated by considering realistic assumptions and different design factors. In our approach, it is assumed that the attractiveness value of each product can be estimated using historical data obtained from an omnichannel environment. The experimental results show that both biased-randomized methodologies are able to provide, in short computing times, solutions that clearly outperform the human-behavior and other more standard methodologies. Additionally, a numerical study has shown that our biased-randomized algorithms are very competitive when compared with non-linear solver engines, obtaining better or similar solutions in much shorter computing times.

By increasing the attractiveness level of retail display tables in a short time horizon, managers can reduce customer attrition and, as a consequence, increase sales revenue in their stores. Using biased-randomized algorithms to maximize the attractiveness of products assigned to display tables in the considered scenario represents a clear enhancement over current practice, which might typically require many hours of a dedicated expert to generate even a feasible solution.

Author Contributions: conceptualization, M.M. and A.A.J.; methodology and software, L.d.C.M. and S.H.; writing, M.M., L.d.C.M.; validation, V.F. All authors have read and agree to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work has been partially supported by the Spanish Ministry of Science, Innovation, and Universities (RED2018-102642-T) and the Erasmus+ programme (2019-1-ES01-KA103-062602).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Verhoef, P.C.; Stephen, A.T.; Kannan, P.K.; Luo, X.; Abhishek, V.; Andrews, M.; Bart, Y.; Datta, H.; Fong, N.; Hoffman, D.L.; et al. Consumer Connectivity in a Complex, Technology-enabled, and Mobile-oriented World with Smart Products. *J. Interact. Market.* **2017**, *40*, 1–8. [[CrossRef](#)]
2. Galino, S.; Moreno, A. Integration of online and offline channels in retail: The impact of sharing reliable inventory availability information. *Manag. Sci.* **2014**, *60*, 1434–1451. [[CrossRef](#)]
3. Blau, P. *Exchange and Power in Social Life*; Routledge: New York, NY, USA, 2017.

4. Ellegaard, C.; Ritter, T. Attractiveness in business markets: Conceptualization and propositions. *White Paper* **2007**, 1–10. Available online: <https://www.impgroup.org/uploads/papers/5847.pdf> (accessed on 31 January 2020).
5. Halinen, A. *Relationship Marketing in Professional Services: A Study of Agency-Client Dynamics in the Advertising Sector*; Routledge: New York, NY, USA, 2012.
6. Caro, F.; Martínez-de Albéniz, V. Product and Price Competition with Satiation Effects. *Manag. Sci.* **2012**, *58*, 1357–1373. [[CrossRef](#)]
7. Caro, F.; Martínez-de Albéniz, V.; Rusmevichientong, P. The Assortment Packing Problem: Multiperiod Assortment Planning for Short-Lived Products. *Manag. Sci.* **2014**, *60*, 2701–2721. [[CrossRef](#)]
8. Bernstein, F.; Martínez-de Albéniz, V. Dynamic product rotation in the presence of strategic customers. *Manag. Sci.* **2017**, *63*, 2092–2107. [[CrossRef](#)]
9. Ferreira, K.; Goh, J. *Assortment Rotation and the Value of Concealment*; Harvard Business School Technology & Operations Mgt. Unit Working Paper; Harvard Business School Press: Boston, MA, USA, 2019; Volume 041. Available online: <http://nrs.harvard.edu/urn-3:HUL.InstRepos:30861196> (accessed on 31 January 2020).
10. Schnurr, B.; Brunner-Sperdin, A.; Stokburger-Sauer, N.E. The effect of context attractiveness on product attractiveness and product quality: The moderating role of product familiarity. *Market. Lett.* **2017**, *28*, 241–253. [[CrossRef](#)]
11. Honhon, D.; Gaur, V.; Seshadri, S. Assortment Planning and Inventory Decisions Under Stockout-Based Substitution. *Oper. Res.* **2010**, *58*, 1364–1379. [[CrossRef](#)]
12. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [[CrossRef](#)]
13. Hübner, A.H.; Kuhn, H. Retail category management: State-of-the-art review of quantitative research and software applications in assortment and shelf space management. *Omega* **2012**, *40*, 199–209. [[CrossRef](#)]
14. Pentico, D.W. The assortment problem: A survey. *Eur. J. Oper. Res.* **2008**, *190*, 295–309. [[CrossRef](#)]
15. Choi, S.H.; Kang, S.; Jeon, Y.J. Personalized recommendation system based on product specification values. *Expert Syst. Appl.* **2006**, *31*, 607–616. [[CrossRef](#)]
16. Kaminskis, M.; Bridge, D.; Foping, F.; Roche, D. Product-Seeded and Basket-Seeded Recommendations for Small-Scale Retailers. *J. Data Semant.* **2017**, *6*, 3–14. [[CrossRef](#)]
17. Ahn, H.J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Inform. Sci.* **2008**, *178*, 37–51. [[CrossRef](#)]
18. Grasas, A.; Juan, A.A.; Faulin, J.; de Armas, J.; Ramalhinho, H. Biased randomization of heuristics using skewed probability distributions: A survey and some applications. *Comput. Ind. Eng.* **2017**, *110*, 216–228. [[CrossRef](#)]
19. Festa, P.; Resende, M.G. An annotated bibliography of GRASP—Part I: Algorithms. *Int. Trans. Oper. Res.* **2009**, *16*, 1–24. [[CrossRef](#)]
20. Lourenço, H.R.; Martin, O.C.; Stützle, T. Iterated local search. In *Handbook of Metaheuristics*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 320–353.
21. Ferone, D.; Gruler, A.; Festa, P.; Juan, A.A. Enhancing and extending the classical GRASP framework with biased randomisation and simulation. *J. Oper. Res. Soc.* **2018**, 1–14. [[CrossRef](#)]
22. Sadowski, W. A Few Remarks on the Assortment Problem. *Manag. Sci.* **1959**, *6*, 13–24. [[CrossRef](#)]
23. Mahajan, S.; van Ryzin, G. Stocking Retail Assortments Under Dynamic Consumer Substitution. *Oper. Res.* **2001**, *49*, 334–351. [[CrossRef](#)]
24. Ulu, C.; Honhon, D.; Alptekinoglu, A. Learning consumer tastes through dynamic assortments. *Oper. Res.* **2012**, *60*, 833–849. [[CrossRef](#)]
25. Mantrala, M.K.; Levy, M.; Kahn, B.E.; Fox, E.J.; Gaidarev, P.; Dankworth, B.; Shah, D. Why is Assortment Planning so Difficult for Retailers? A Framework and Research Agenda. *J. Retail.* **2009**, *85*, 71–83. [[CrossRef](#)]
26. Mou, S.; Robb, D.J.; DeHoratius, N. Retail store operations: Literature review and research directions. *Eur. J. Oper. Res.* **2017**. [[CrossRef](#)]
27. Smith, S.A.; Agrawal, N. Management of Multi-Item Retail Inventory Systems with Demand Substitution. *Oper. Res.* **2000**, *48*, 50–64. [[CrossRef](#)]
28. Sauré, D.; Assaf, Z. Optimal dynamic assortment planning with demand learning. *Manuf. Serv. Oper. Manag.* **2013**, *15*, 387–404. [[CrossRef](#)]

29. Rajamma, R.K.; Paswan, A.K.; Ganesh, G. Services purchased at brick and mortar versus online stores, and shopping motivation. *J. Serv. Market.* **2007**, *21*, 200–212. [[CrossRef](#)]
30. Kök, A.G.; Fisher, M.L.; Vaidyanathan, R. Assortment planning: Review of literature and industry practice. In *Retail Supply Chain Management*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 99–153.
31. Liu, D.R.; Shih, Y.Y. Hybrid approaches to product recommendation based on customer lifetime value and purchase preferences. *J. Syst. Softw.* **2005**, *77*, 181–191. [[CrossRef](#)]
32. Li, Z.E.; Lu, Q.; Talebian, M. Online versus bricks-and-mortar retailing: A comparison of price, assortment and delivery time. *Int. J. Prod. Res.* **2014**, *53*, 1–14. [[CrossRef](#)]
33. Balakrishnan, J.; Cheng, C.H.; Wong, K.F.; Woo, K.H. Product recommendation algorithms in the age of omnichannel retailing—An intuitive clustering approach. *Comput. Ind. Eng.* **2018**, *115*, 459–470. [[CrossRef](#)]
34. Baykal, M.O.; Alhajj, R.; Polat, F. Co-operation framework of case-based reasoning agents for automated product recommendation. *J. Exp. Theor. Artif. Intell.* **2005**, *17*, 201–220. [[CrossRef](#)]
35. Choi, S.H.; Cho, Y.H. An utility range-based similar product recommendation algorithm for collaborative companies. *Expert Syst. Appl.* **2004**, *27*, 549–557. [[CrossRef](#)]
36. Choi, K.; Yoo, D.; Kim, G.; Suh, Y. A hybrid online-product recommendation system: Combining implicit rating-based collaborative filtering and sequential pattern analysis. *Electron. Commerce Res. Appl.* **2012**, *11*, 309–317. [[CrossRef](#)]
37. Zhao, X.W.; Guo, Y.; He, Y.; Jiang, H.; Wu, Y.; Li, X. We Know What YouWant to Buy: A Demographic-based System for Product Recommendation On Microblogs. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14), New York, NY, USA, 24–27 August 2014; Association for Computing Machinery: New York, NY, USA; pp. 1935–1944. [[CrossRef](#)]
38. Zhao, W.X.; Li, S.; He, Y.; Chang, E.Y.; Wen, J.R.; Li, X. Connecting Social Media to E-Commerce: Cold-Start Product Recommendation Using Microblogging Information. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1147–1159. [[CrossRef](#)]
39. Yang, M.H.; Chen, W.C. A study on shelf space allocation and management. *Int. J. Prod. Econ.* **1999**, *60*, 309–317. [[CrossRef](#)]
40. Yang, M.H. An efficient algorithm to allocate shelf space. *Eur. J. Oper. Res.* **2001**, *131*, 107–118. [[CrossRef](#)]
41. Flamand, T.; Ghoniem, A.; Maddah, B. Promoting impulse buying by allocating retail shelf space to grouped product categories. *J. Oper. Res. Soc.* **2016**, *67*, 953–969. [[CrossRef](#)]
42. Bianchi-Aguiar, T.; Silva, E.; Guimarães, L.; Carravilla, M.A.; Oliveira, J.F.; Amaral, J.G.; Liz, J.; Lapela, S. Using Analytics to Enhance a Food Retailer's Shelf-Space Management. *Interfaces* **2016**, *46*, 424–444. [[CrossRef](#)]
43. Parsons, A.G. Atmosphere in fashion stores: Do you need to change? *J. Fash. Mark. Manag.* **2011**, *15*, 428–445. [[CrossRef](#)]
44. Gao, L.; Thomas, D.J.; Freimer, M.B. Optimal inventory control with retail pre-packs. *Prod. Oper. Manag.* **2014**, *23*, 1761–1778. [[CrossRef](#)]
45. Feo, T.A.; Resende, M.G. Greedy randomized adaptive search procedures. *J. Glob. Optim.* **1995**, *6*, 109–133. [[CrossRef](#)]
46. Resende, M.G.; Ribeiro, C.C. *Optimization by GRASP*; Springer: Berlin/Heidelberg, Germany, 2016.
47. Grasas, A.; Juan, A.A.; Lourenço, H.R. SimILS: A simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization. *J. Simulat.* **2016**, *10*, 69–77. [[CrossRef](#)]
48. De Armas, J.; Juan, A.A.; Marquès, J.M.; Pedroso, J.P. Solving the deterministic and stochastic uncapacitated facility location problem: From a heuristic to a simheuristic. *J. Oper. Res. Soc.* **2017**, *68*, 1161–1176. [[CrossRef](#)]
49. Ferrer, A.; Guimarans, D.; Ramalhinho, H.; Juan, A.A. A BRILS metaheuristic for non-smooth flow-shop problems with failure-risk costs. *Expert Syst. Appl.* **2016**, *44*, 177–186. [[CrossRef](#)]
50. Gonzalez-Neira, E.M.; Ferone, D.; Hatami, S.; Juan, A.A. A biased-randomized simheuristic for the distributed assembly permutation flowshop problem with stochastic processing times. *Simulat. Model Pract. Theor.* **2017**, *79*, 23–36. [[CrossRef](#)]
51. Faulin, J.; Juan, A.A. The ALGACEA-1 method for the capacitated vehicle routing problem. *Int. Trans. Oper. Res.* **2008**, *15*, 599–621. [[CrossRef](#)]
52. Dominguez, O.; Juan, A.A.; Faulin, J. A biased-randomized algorithm for the two-dimensional vehicle routing problem with and without item rotations. *Int. Trans. Oper. Res.* **2014**, *21*, 375–398. [[CrossRef](#)]

53. Dominguez, O.; Guimarans, D.; Juan, A.A.; de la Nuez, I. A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls. *Eur. J. Oper. Res.* **2016**, *255*, 442–462. [[CrossRef](#)]
54. Juan, A.A.; Pascual, I.; Guimarans, D.; Barrios, B. Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem. *Int. Trans. Oper. Res.* **2015**, *22*, 647–667. [[CrossRef](#)]
55. Martin, S.; Ouelhadj, D.; Beullens, P.; Ozcan, E.; Juan, A.A.; Burke, E.K. A multi-agent based cooperative approach to scheduling and routing. *Eur. J. Oper. Res.* **2016**, *254*, 169–178. [[CrossRef](#)]
56. Juan, A.A.; Lourenço, H.R.; Mateo, M.; Luo, R.; Castella, Q. Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues. *Int. Trans. Oper. Res.* **2014**, *21*, 103–126. [[CrossRef](#)]
57. Bonami, P.; Lee, J. BONMIN user’s manual. *Numer. Math.* **2007**, *4*, 1–32.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).