The Rank Pricing Problem with Ties

Concepción Domínguez, Martine Labbé, Alfredo Marín

The Rank Pricing Problem with Ties

Concepción Domínguez[1,*] Martine Labbé[2] Alfredo Marín[3]

[1,2] Université Libre de Bruxelles, Belgium
[1,2] Inria Lille-Nord Europe, France
[1,3] Universidad de Murcia, Spain

May 28, 2020

Abstract

In the Rank Pricing Problem (RPP), a firm intends to maximize its profit through the pricing of a set of products to sell. Customers are interested in purchasing at most one product among a subset of products. To do so, they are endowed with a ranked list of preferences and a budget. Their choice rule consists in purchasing the highest-ranked product in their list and whose price is below their budget. In this paper, we consider an extension of RPP, the Rank Pricing Problem with Ties (RPPT), in which we allow for indifference between products in the list of preferences of the customers. Considering the bilevel structure of the problem, this generalization differs from the RPP in that it can lead to multiple optimal solutions for the second level problems associated to the customers. In such cases, we look for pessimistic optimal solutions of the bilevel problem : the customer selects the cheapest product.

We present a new three-indexed integer formulation for RPPT and introduce two resolution approaches. In the first one, we project out the customer decision variables, obtaining a reduced formulation that we then strengthen with valid inequalities from the former formulation. Alternatively, we follow a Benders decomposition approach leveraging the separability of the problem into a master problem and several subproblems. The separation problems to include the valid inequalities to the master problem dynamically are shown to reduce to min-cost flow problems. We finally carry out extensive computational experiments to assess the performance of the resolution approaches.

Keywords: Combinatorial Optimization, Pricing Problems, Integer Programming, Bilevel Programming, Benders Decomposition

*Corresponding author. E-mail address: concepcion.dominguez@um.es (C. Domínguez)

1 Introduction

A key decision for a company is its pricing strategy, i.e. the choice of the best possible price for their products considering customers' behavior. More generally, considering a set of customers with certain preferences over the products available, what should be the price of each product so as to maximize the company's profit?

Pricing problems are challenging due to their bilevel structure, since they take into account customers' purchasing decisions when setting the prices of the products. Customers' choice rule can be modeled in a variety of ways. In the Rank Pricing Problem (RPP), customers are unit-demand (i.e. interested in purchasing at most one unit of one product) and they possess their own ranking of the candidate products yielding to an incomplete list of preferences for each customer. Once the prices are set by the company, customers purchase their highest-ranked product among the ones they can afford (if any). Modeling customers' behavior through a ranked list of preferences is versatile and offers a general framework. It allows to model customers' choices based on both compensatory decision processes (like assigning a utility to the products and purchasing the highest-ranked one) and noncompensatory decision processes (such as ranking product attributes in terms of importance, and comparing them following a lexicographic rule).

In this work, we consider a generalization of RPP in which customers are not forced to define a strict preference between all the pairs of candidate products. Instead, we allow for indifference and consider ties in the list of preferences. We name this problem the *Rank Pricing Problem with Ties* (RPPT).

Considering ties in the preference lists of the customers leads to a different bilevel structure of the problem. As detailed in Calvete et al. [8], in the RPP (without ties), the second level problem associated to each customer has a unique optimal solution for a given vector of prices of the products. However, in this extension, the indifference results in second level problems that may have multiple optimal solutions. In RPPT, we consider the *pessimistic optimal solution.* In case of indifference, the customers' selection of products is the most *natural* for the customers since it is based on the price - they purchase (one of) the cheapest products.

To the best of our knowledge, the study of RPPT has not yet been addressed in the literature. In this paper, we tackle its resolution by means of exact optimization methods. Specifically, we begin with a formal introduction of RPPT and propose an integer linear model with three-indexed variables. Next, we derive two resolution methods for our three-indexed model.

The first method is based on a reduced model for RPPT that uses a much smaller set of variables. Since the linear relaxation of this reduced model provides a weaker bound, we project out the variables of the three-indexed model by means of Farkas' Lemma to obtain a set of valid inequalities strengthening the Reduced Model. Due to the particular structure of the rank pricing problem, the separation problem relative to these valid inequalities can be transformed into a min-cost flow problem. In this way, we avoid solving a linear problem with a commercial solver and instead apply a suitable resolution algorithm, making the separation procedure computationally efficient.

The second resolution approach is based on Benders decomposition and takes advantage of the structure of the problem. First we reformulate the three-indexed model, obtaining a master problem with a set of constraints whose separation can be done by solving linear subproblems. Then we are able to identify a small (polynomial) subset of constraints from the previous set to obtain a reduced master formulation that constitutes a valid formulation for RPPT. The rest of the constraints (now valid inequalities) are separated in our resolution method and included dynamically, in a branch-and-cut framework. Although the valid inequalities are different from the reduced model ones, the separation procedure is analogous to the first one. To speed up the cut separation in the linear relaxation phase, we include an in-out method, a technique used to stabilize and accelerate the convergence of the cut loop.

We also provide a preprocessing techniques section where we reduce the size of the instances by making use of the properties of the problem. We conclude our paper with the results of extensive numerical experiments, where we compare the two resolution methods proposed in terms of number of nodes of the branching tree, integrality gap and computational time, and we show the efficiency of the valid inequalities and the preprocessing techniques.

The article is organized as follows. In Section 2, we provide a literature review. Section 3 states the notation used throughout the paper and Section 4 is devoted to the presentation of the three-indexed model. Section 5 includes all the results regarding the reduced model. In Section 6, we provide the results concerning the Benders decomposition resolution approach. Section 7 includes the preprocessing techniques, and Section 8 contains the computational study. We provide some conclusions in Section 9.

2 Literature review

The Rank Pricing Problem as stated here (but under a different name) was introduced by Rusmevichientong et al. [27]. Motivated by the availability of data from a website offering car recommendations to customers, they proposed pricing problems with unlimited supply and unit-demand customers, and three different objectives, namely a min-buying, a max-buying and a rank-buying objective. They show that those problems are NP-complete in the strong sense and introduce a heuristic approximation algorithm together with performance bounds. Aggarwal et al. [1] and Briest and Krysta [7] take up the work in [27] and present complexity results and approximation algorithm schemes for RPP and variants of it. To the best of our knowledge, Calvete et al. [8] proposed the first formulations for the RPP, and thus [8] is a good starting point when tackling RPPT.

Pricing optimization problems in combination with ranking-based customers' preferences are scarce in the literature, since many of them consider the maximization of the customers' utility. However, the modelization of the customers' selection rule by means of a ranked list of preferences appears in many related fields. A closely related problem to our own is the Product Line Design (PLD) problem. This problem aims at selecting a subset of products to be produced (generally from a bigger given set) in order to maximize the company's revenue. The modelization of the customers in PLD is typically made in two ways. In the probabilistic choice behavior (studied by Green and Krieger [15, 16], McBride and Zufryden [23], Dobson and Kalish [12] and Belloni et al. [3], among others), each segment of customers probabilistically chooses from the available options. In the first-choice (also called ranking-based) behavior, customers deterministically select the product from the offered line that maximizes their utility. Some references are those by Chen and Hausman [10], Schön [28, 29] and Kraus and Yano [20]. A very recent work by Bertsimas and Mišić [5] studies the PLD problem, introducing a new mixed-integer formulation, theoretically analyzing it, and presenting a solution approach based on Benders decomposition that significantly outperforms the previous results. As we will address in the following sections, this paper has been the motivation for introducing Benders decomposition as a plausible technique to tackle the resolution of RPPT.

A similar modelization of the customers' selection rule can also be found in the field of Discrete Location. More specifically, the Simple Plant Location Problem with Order (SPLPO) consists in locating a set of facilities assuming that customers rank the potential facilities and they attend their most preferred among the open ones. SPLPO was introduced in 1987 by Hanjoul and Peeters [18], who developed a heuristic and was further studied by Hansen et al. [19], Vasilyev and Klimentova [31] and Cánovas et al. [9]. Other works deal with a particular ordering of the facilities through the concept of closest assignment. Espejo et al. [13] give a thorough review and comparison of the different closest assignment constraints encountered in the location literature, and study their generalization in the case of ties between distances.

Although the optimality criteria differ from our own, bipartite matching problems with preferences also model the customers' choice by means of a ranked list of preferences. In particular, very well-known problems like the Stable Marriage (SM) problem include preference lists as the agents' choice. The first integer formulations were introduced by Vande Vate [30] and by Gusfield and Irving [17]. An extension of Vande Vate's model to include incomplete lists of preferences was given by Rothblum [26]. More recently, extensions of these models have been introduced by Kwanashie and Manlove [21] and Delorme et al. [11] to tackle a one-to-many generalization of SM problem, namely the Hospital-Residents (HR) problem, as well as the Stable Marriage with Ties (SMT) and the Hospital-Residents with Ties (HRT) generalizations. An in-depth review on structural and algorithmic results on matching problems with preferences can be found in Manlove [22].

3 Notation and relationship with problem PLD

The aim of RPPT is to establish the prices of the products of the company so as to maximize its revenue, taking into account that we assume unit-demand customers who, once the prices are settled, will purchase their highest-ranked product among the ones they can afford (if any).

Besides, if a customer is indifferent between two products and he can afford both, he will purchase the cheapest one (or one of the cheapest randomly if there are more than one).

As for the notation, let $K = \{1, \dots, |K|\}$ be the set of customers and $I = \{1, \dots, |I|\}$ the set of products. Each customer $k \in K$ has a subset of acceptable products $I^k \subseteq I$ so that k would rather not make any purchase than purchasing a product $i \notin I^k$. Similarly, we say that a customer k is acceptable for a product i if it belongs to $K_i := \{k \in K : i \in I^k\}$. Without loss of generality, we assume $I^k \neq \emptyset \ \forall k \in K$, $K_i \neq \emptyset \ \forall i \in I$.

The acceptable products for k (i.e. the products in I^k) are ranked by k from the *best* to the *worst* in a preference list. However, some customers may not be able to define a clear strict preference over certain products, and they are allowed to express indifference in their preference lists. We denote $i \prec_k j$ when we say that a customer $k \in K$ *prefers* product i to j, and we use $i \sim_k j$ if k is *indifferent between* two products i and j. Therefore, there exists a weak order on the set I^k for each $k \in K$. Furthermore, $\sim_k$ is an equivalence relation (reflexive, symmetric, transitive) which defines a partition $\mathscr{S}^k = \{S_1^k, \dots, S_{n^k}^k\}$ of the set I^k such that $i, j \in S_n^k$ if $i \sim_k j$ and $i \in S_n^k$, $j \in S_{n'}^k$ with $n < n'$ if $i \prec_k j$. Notice that for a given customer k, $\prec_k$ defines a total order on the set of the equivalence classes $\mathscr{S}^k$.

Each customer k is endowed with a budget. In order to keep notation consistent in the formulation, and given that different customers may have the same budget, we define set $M = \{1, \dots, |M|\}$ as the set of indices that refer to the different budgets of the customers, and $\{b^m\}_{m \in M}$ as the set of different budgets, so that $b^{m_1} < b^{m_2}$ if $m_1 < m_2$. Further, we define a function $\sigma : K \to M$ such that $\sigma(k) = m$ if the budget of customer k is the m-th smallest budget b^m.

As explained in Rusmevichientong et al. [27], there is always an optimal solution of RPP in which the prices of the products are equal to a customer budget b^m, $m \in M$. Since this result is also valid for RPPT, we define $M_i := \{m \in M : \exists k \in K_i \text{ with } \sigma(k) = m\}$ as the set of indices of budget values that are candidates to be the optimal price of product i. Moreover, for $k \in K_i$, $M_i^k := \{m \in M_i : m \leq \sigma(k)\}$ represents the set of indices m of candidate prices b^m at which k can purchase i in a feasible solution. Finally, we define $M_{S_n^k} = \cup_{i \in S_n^k} M_i^k$ as the subset of indices $m \in M$ of candidate prices b^m at which k could purchase some product $i \in S_n^k$.

Table 1: Preference matrix, vector of budgets and an optimal solution to an instance of RPPT

	Prod. 1	Prod. 2	Prod. 3	Prod. 4	Prod. 5	Budgets
Customer 1	1	3	1*	-	2	120
Customer 2	2	1*	1	-	-	95
Customer 3	1	2	4	1*	3	82
Customer 4	-	3	1	3	2*	82
Customer 5	-	1	3	2*	-	79
Customer 6	2	-	1	2	1*	65
Customer 7	3	2	5	1	4*	64
Customer 8	1	4	2	-	3*	53
Optimal prices	-	95	120	79	53	585

Example 3.1. *Table 1 shows an instance of* RPPT *with* $|K| = 8$ *and* $|I| = 5$ *and an optimal solution. The entry* (k, i) *of the preference matrix denotes the index* n *of the equivalence class* S_n^k *to which* i *belongs for* k *(the symbol - indicates that the corresponding product* $i \notin I^k$*). Clearly, the smaller the entry of the preference matrix, the greater the preference of the customer over that product. Customer 1 is thus interested in all products except for product 4, that is,* $I^1 = \{1, 2, 3, 5\}$*, and from the preference matrix we deduce* $1 \sim_1 3 \prec_1 5 \prec_1 2$*, so we have* $|\mathscr{S}^1| = n_1 = 3$ *and* $S_1^1 = \{1, 3\}$*,* $S_2^1 = \{5\}$*,* $S_3^1 = \{2\}$*. Similarly, the acceptable set of customers for product 1 is* $K_1 = \{1, 2, 3, 6, 7, 8\}$*.*

There are 7 different customer budgets: $b^1 = 53, b^2 = 64, \ldots, b^5 = 82, b^6 = 95, b^7 = 120$*. Following the notation,* $\sigma(1) = 7$*, i.e., customer 1 has the* 7^{th} *smallest budget (i.e. the greatest one),* $\sigma(2) = 6$*,* $\sigma(3) = \sigma(4) = 5$*, et cetera. Furthermore, the last row of the table shows a vector of optimal prices along with the objective value (585). The purchasing decision of every customer in this optimal solution is represented by an asterisk next to the entry of the matrix associated to the product he purchases.*

The set of indices of budget values that are candidates to be the optimal price of product 4 are $M_1 = \{2, 3, 4, 5\}$*, and in the optimal solution, 4 has price* $b^4 = 79$*. Likewise, the set of indices of candidate prices at which customer 6 may purchase product 4 is* $M_4^6 = \{2, 3\}$*. And the set of indices of candidate prices at which customer 6 may purchase a product from* $S_2^6 = \{1, 4\}$ *is* $M_{S_2^6} = \{1, 2, 3\}$*.*

Notice that, even if there are less products than customers and six customers interested in product 1, this product remains unsold in the optimal solution. One could think that, since customer 7 purchases a product with price 53 but he has a budget of 64 and prefers product 1, setting the price 64 for product 1 would lead to a feasible solution with greater objective value. However, the fact that ties are allowed in RPPT *prevents this solution from being optimal. Indeed, in this case customers 1 and 3 would also purchase product 1 (given that they are indifferent between 1 and the product they are currently purchasing but 1 has a smaller price), and therefore the revenue would be 525 instead of 585.*

In order to relate the RPPT with the Product Line Design problem (PLD), let us first properly introduce the latter. In PLD, we are given a set of products S with fixed prices, and the aim is to select a subset of them $S' \subset S$ of size p to build a product line. We are also given a set of unit-demand customers K. Each customer $k \in K$ is interested in a subset of products $S^k \subset S$ and ranks the products in S^k creating a list of preferences (the preferences are strict). Thus, $i \prec_k j$ for $i, j \in S^k$ if k prefers i over j. Once the product line is established, each customer is assumed to purchase the highest-ranked product in $S' \cap S^k$, if any. The problem consists in finding the product line that maximizes the profit of the company.

Now, let us assume we have ties in the list of preferences of the customers in PLD. We can name this problem the *Product Line Design problem with Ties* (PLDT). In such case, since all the products have a fixed price, there are no ties between two products with different prices (because if a customer ranks two products equally and one is cheaper, he purchases the cheapest one when possible). Therefore, there can only be ties between products with the same price. Furthermore, for all product lines in which there are two products $i, j \in S' \cap S^k$ for some customer k, and $i \sim_k j$, k will purchase either i or j. In sum, PLD and PLDT have the same structure, and PLDT does not require additional constraints to translate the pessimistic assumption.

Clearly, assuming $p = |S|$ in PLD (or PLDT), RPPT is a generalization of PLD where the prices of the products are not fixed. Furthermore, we now show that RPPT can be seen as a particular case of PLDT with a larger number of products and in which the number of products to select in the product line $p = |S|$.

The prices of the products are given in PLDT. On the contrary, in RPPT they are not fixed, but the candidate prices belong to the sets $\{b^m\}_{m \in M}$ of budgets of the customers. Therefore, to transform an RPPT instance into a PLDT instance, we define the set of products $S := \{(i, b^m) : i \in I, m \in M_i\}$. Similarly, we define $S^k := \{(i, b^m) : i \in I^k, m \in M_i^k\} \subseteq S$. Regarding the customers' lists of preferences, we assume that $i \prec_k j$ for $i, j \in I^k$ implies $(i, b^m) \prec_k (j, b^{m'})$ $\forall\, m \in M_i$, $m' \in M_j$. As for $i, j \in I^k$ with $i \sim_k j$ in RPPT, it holds $(i, b^m) \prec_k (j, b^{m'})$ if $m < m'$ and $(i, b^m) \sim_k (j, b^{m'})$ if $m = m'$.

Let (i, b^m) and $(i, b^{m'})$, with $m < m'$, be two products of the PLDT version of an RPPT instance. If they both belong to S', then $\forall k \in K$ with $(i, b^m), (i, b^{m'}) \in S^k$ it holds $(i, b^m) \prec_k (i, b^{m'})$, so product $(i, b^{m'})$ is not sold. Hence, we do not need to add any additional constraint imposing that each product can only be sold at one candidate price. This also implies that at most $|I|$ products will be sold in any optimal product line S', even if we do not impose a limit on its size p.

4 Three-Indexed Model for RPPT

In this section, we propose a mixed-integer formulation using two sets of variables. Firstly, we define binary variable v_i^m, $\forall i \in I$, $\forall m \in M_i$, that takes value 1 if the price of product i is equal to the m-th smallest budget b^m. For each $k \in K$, and considering the partition $\mathscr{S}^k$, we define

y_n^{km}, $\forall k \in K$, $n \in N^k := \{1, \ldots, n^k\}$, $m \in M_{S_n^k}$, that takes value 1 in a solution provided that customer k purchases a product $i \in S_n^k$ at price b^m.

With these sets of variables, we present a first model called the Three-Indexed Model (3IM) for RPPT:

$$
\begin{aligned}
\text{(3IM)} \quad \max_{\mathbf{v},\mathbf{y}} \quad & \sum_{k\in K}\sum_{n\in N^k}\sum_{m\in M_{S_n^k}} b^m y_n^{km} && \text{(1a)}\\
\text{s.t.} \quad & \sum_{m\in M_i} v_i^m \le 1 \qquad \forall i \in I, && \text{(1b)}\\
& \sum_{n\in N^k}\sum_{m\in M_{S_n^k}} y_n^{km} \le 1 \qquad \forall k \in K, && \text{(1c)}\\
& y_n^{km} \le \sum_{i\in S_n^k} v_i^m \qquad \forall k \in K, n \in N^k, m \in M_{S_n^k}, && \text{(1d)}\\
& \sum_{\substack{m'\in M_i^k:\\ m'\le m}} v_i^{m'} + \sum_{\substack{m'\in M_{S_n^k}:\\ m'>m}} y_n^{km'} + \sum_{n'=n+1}^{n^k}\sum_{m'\in M_{S_{n'}^k}} y_{n'}^{km'} \le 1 \\
& \qquad \forall k \in K, n \in N^k, i \in S_n^k, m \in M_i^k, && \text{(1e)}\\
& v_i^m \in \{0,1\} \quad \forall i \in I, m \in M_i, && \text{(1f)}\\
& y_n^{km} \in \{0,1\} \quad \forall k \in K, n \in N^k, m \in M_{S_n^k}. && \text{(1g)}
\end{aligned}
$$

Constraints (1b) ensure that each product price is unique. Constraints (1c) guarantee that each customer purchases at most one product. Constraints (1d) state that if a customer k purchases a product from class S_n^k at price b^m, then there exists $i \in S_n^k$ at price b^m. And constraints (1e) are the *preference constraints*, and they ensure that the preferences of the customers are satisfied in any feasible solution. Thus, if the first sum $\sum_{\substack{m'\in M_i^k:\\ m'\le m}} v_i^{m'}$ is equal to 1, then k can purchase i at a price smaller than or equal to $b^{m'}$. So the second and third sums of the LHS of (1e) are equal to 0, ensuring that k does not purchase either a product from a class $S_{n'}^k$ with $n' > n$, or any product from S_n^k at a higher price $b^{m'}$, $m' > m$.

Remark 4.1. *Formulation* (3IM) *is also valid for RPP.*

Now we prove that the integrality of the set of y-variables can be relaxed:

Proposition 4.2. *The integrality of variables* y_n^{km}, $\forall k \in K$, $\forall n \in N^k$, $\forall m \in M_{S_n^k}$, *can be relaxed in formulation* (3IM)*. Indeed, family* (1g) *can be replaced with family*

$$y_n^{km} \ge 0 \quad \forall k \in K, n \in N^k, m \in M_{S_n^k}. \tag{2}$$

Furthermore, for a given fixed feasible vector $(\bar{v}_i^m) \in \{0,1\}^{I\times M_i}$ *and a fixed customer* k*, the optimal values of variables* y_n^{km} *for* (3IM) *with* (2) *instead of* (1g) *are as follows.*

1. *If* $\sum_{i\in I^k}\sum_{m\in M_i^k} \bar{v}_i^m = 0$, *then* $y_n^{km} = 0$ $\forall n \in N^k, m \in M_{S_n^k}$.

2. *Otherwise, let* $n^* := \min\left\{n \in N^k : \sum_{i \in S_n^k} \sum_{m \in M_i^k} \bar{v}_i^m \geq 1\right\}$, $m^* := \min\left\{m \in M_{S_{n*}^k} : \sum_{i \in S_{n*}^k} \bar{v}_i^m \geq 1\right\}$. *Then,* $y_{n^*}^{km^*} = 1$, $y_n^{km} = 0$ *for* $(n, m) \neq (n^*, m^*)$.

Proof. In Appendix A. □

Example 4.3. *Let us describe the variables used to solve the instance given in Table 1 with formulation* (3IM)*. First, we define the v-variables associated with each product. For instance, for product 5 we define variables* v_5^m *for* $m \in M_5 = \{1, 2, 3, 5, 7\}$*. Regarding the y-variables, for customer 2 we have that* $n_2 = 2$*, and* $S_1^2 = \{2, 3\}$*,* $S_2^2 = \{1\}$*. For the products in* S_1^2*,* $M_{S_1^2} = \{1, 2, 3, 4, 5, 6\}$*, so we define variables* y_1^{2m} *for* $m \in M_{S_1^2}$*. As for* S_2^2*, we define* y_2^{2m} *for* $m \in M_{S_2^2} = \{1, 2, 3, 5, 6\}$ *(there are no customers with budget* $b^4 = 74$ *interested in product 1, so 1 will not have price 74 in an optimal solution). In the optimal solution, customer 2 purchases* $2 \in S_1^2$ *at price* b^4*, so* $y_1^{24} = 1$*.*

Formulation (3IM) yields very good linear relaxation bounds. The main drawback of this formulation is that it has a large number of variables and constraints, and therefore it is not suitable for instances with a large number of customers or dense matrices of preferences.

5 Projecting the customer decision variables on the Reduced Model

In this section, we discuss how to project out formulation (3IM) on a formulation of a smaller size, the Reduced Model (RM). The projection results in a set of valid inequalities for (RM) for which we develop a separation algorithm.

First, we define the sets of two-indexed variables of (RM). We use variables v_i^m, $\forall i \in I$, $m \in M_i$, that represent, as in (3IM), the price of a product. Considering once again the partition of I^k into equivalence classes $S_1^k, \ldots, S_{n^k}^k$, we define binary variables x_n^k, $\forall k \in K$, $n \in N^k$, as decision variables that take value 1 if customer k purchases some product $i \in S_n^k$, and zero otherwise. And finally, to be able to model the profit of the company, we define continuous variables z_n^k, $\forall k \in K$, $\forall n \in N^k$, that represent the profit associated to a customer k and an equivalence class S_n^k. In a feasible solution, the value of z_n^k is equal to the price of the least expensive product from S_n^k provided that customer k purchases a product from S_n^k, and zero otherwise.

Using these variables, the Reduced Model (RM) for RPPT is:

$$
\begin{aligned}
\text{(RM)} \quad \max_{\mathbf{v},\mathbf{x},\mathbf{z}} \quad & \sum_{k \in K} \sum_{n \in N^k} z_n^k && && \text{(3a)} \\
\text{s.t.} \quad & \sum_{m \in M_i} v_i^m \leq 1 && \forall i \in I, && \text{(3b)} \\
& \sum_{n \in N^k} x_n^k \leq 1 && \forall k \in K, && \text{(3c)}
\end{aligned}
$$

$$x_n^k \leq \sum_{i \in S_n^k} \sum_{m \in M_i^k} v_i^m \qquad \forall k \in K, n \in N^k, \tag{3d}$$

$$\sum_{m \in M_i^k} v_i^m + \sum_{n'=n+1}^{n^k} x_{n'}^k \leq 1 \qquad \forall k \in K, n < n^k, i \in S_n^k, \tag{3e}$$

$$z_n^k \leq b^{\sigma(k)} x_n^k \qquad \forall k \in K, n \in N^k, \tag{3f}$$

$$z_n^k \leq b^{\sigma(k)} + \sum_{m \in M_i^k} \left(b^m - b^{\sigma(k)}\right) v_i^m \qquad \forall k \in K, n \in N^k, i \in S_n^k, \tag{3g}$$

$$v_i^m \in \{0,1\} \qquad \forall i \in I, m \in M_i, \tag{3h}$$

$$x_n^k \in \{0,1\} \qquad \forall k \in K, n \in N^k, \tag{3i}$$

$$z_n^k \geq 0 \qquad \forall k \in K, n \in N^k. \tag{3j}$$

Constraints (3b) ensure that each product price is unique. Constraints (3c) guarantee that each customer purchases at most one product, i.e., that all customers are unit-demand. Constraints (3d) prevent a customer k from purchasing a product $i \in S_n^k$ when he cannot afford it. Constraints (3e) are the *preference constraints*, and they guarantee that if a customer k can afford a product i, that is if $\sum_{m \in M_i^k} v_i^m = 1$, then k does not purchase any other product $j \succ_k i$, i.e. $\sum_{n'=n+1}^{n^k} x_n^k = 0$. The sets of constraints (3f) and (3g) model the profit. Constraints (3f) ensure that if customer k does not purchase any product from S_n^k ($x_n^k = 0$), then $z_n^k = 0$ and the corresponding profit is zero. When customer k can afford a product $j \in I^k$, then constraints (3g) ensure that the profit associated to k and a class S_n^k is the minimum of the prices of the products in S_n^k. Indeed, when $v_i^{m_0} = 1$ for some $m_0 \leq \sigma(k)$, then $\sum_{m \neq m_0} v_i^m = 0$ and the RHS is equal to $b^{\sigma(k)} - (b^{m_0} - b^{\sigma(k)}) = b^{m_0}$. Since z_n^k is bounded by the price of all the products $i \in S_n^k$, it is actually bounded by the price of the cheapest product from the set. Finally, the objective function (3a) represents the profit of the company, that is maximized.

5.1 Comparison of models (RM) and (3IM)

In this subsection, we compare the bounds given by the linear relaxations of models (3IM) and (RM). The proof of Proposition 5.1 is detailed in Appendix A.

Proposition 5.1. *The upper bound given by the linear relaxation of formulation* (3IM) *is always less than or equal to that of formulation* (RM).

Table 2: Preference matrix of a small instance of RPPT

	Prod. 1	Prod. 2	Prod. 3	Budgets
Customer 1	1	3	2	2
Customer 2	-	2	1	4
Customer 3	2	1	1	8

Example 5.2. *Let us show through the small example illustrated by Table 2 how the linear relaxation bound given by model* (3IM) *can be strictly less than that of* (RM)*. An optimal solution of this example is obtained when we assign price* $b^1 = 2$ *to product 1 and price* $b^2 = 4$ *to product 2 (and product 3 remains unsold). For this price vector, customer 1 purchases 1 and customers 2 and 3 purchase 2, so the optimal value is 10.*

The upper bound given by the linear relaxation of model (RM) *is 14. The fractional values of v-variables are* $v_1^1 = 1$, $v_2^2 = v_3^2 = 0.5$ *(and the rest equal to zero). Likewise, the values of x-variables different from zero are* $x_1^1 = 1$, $x_1^2 = x_2^2 = 0.5$, $x_1^3 = 0.75$, $x_2^3 = 0.25$, *and the values of z-variables are* $z_1^1 = 2$, $z_1^2 = z_2^2 = 2$, $z_1^3 = 6$, $z_2^3 = 2$*. However, if we use the same v-values in* (3IM) *and calculate the y-values by means of constraints* (1c)-(1e)*, we obtain* $y_1^{11} = 1$, $y_1^{22} = y_2^{22} = 0.5$, $y_1^{32} = 1$*. This solution yields an objective value of 10 in* (3IM)*. In fact, the upper bound given by the linear relaxation of model* (3IM) *is 12.*

As we will see in the computational experiments of Section 8, the upper bounds given by the linear relaxation of model (3IM) *are usually strictly less than those given by model* (RM)*. Regarding the previous instance from Table 1, the upper relaxation bounds given by the linear relaxation of models* (RM) *and* (3IM) *are, respectively, 640 and 588 (recall that its optimal value is equal to 585).*

5.2 Strengthening the Reduced Model (RM)

The linear relaxation of model (3IM) generally yields a smaller upper bound than that of model (RM). By projecting out variables y_n^{km} in (3IM), we can derive a set of valid inequalities to strengthen model (RM).

We first extend formulation (3IM) adding x-variables and the corresponding constraints from (RM) relating them to the previous variables. By definition, we have $x_n^k = \sum_{m \in M_{S_n^k}} y_n^{km}$ and $z_n^k = \sum_{m \in M_{S_n^k}} b^m y_n^{km}$ for all $k \in K$, $n \in N^k$. Using x- and z-variables in place of y-variables when possible in formulation (3IM) leads to:

$$\text{(3IM+)} \quad \max_{\mathbf{v},\mathbf{y},\mathbf{x},\mathbf{z}} \quad \sum_{k \in K} \sum_{n \in N^k} z_n^k \tag{4a}$$

$$\text{s.t.} \quad \sum_{m \in M_i} v_i^m \le 1 \qquad \forall i \in I, \tag{4b}$$

$$\sum_{n \in N^k} x_n^k \le 1 \qquad \forall k \in K, \tag{4c}$$

$$y_n^{km} \le \sum_{i \in S_n^k} v_i^m \qquad \forall k \in K, n \in N^k, m \in M_{S_n^k}, \tag{4d}$$

$$x_n^k \le \sum_{i \in S_n^k} \sum_{m \in M_i^k} v_i^m \qquad \forall k \in K, n \in N^k, \tag{4e}$$

$$\sum_{\substack{m' \in M_i^k: \\ m' \leq m}} v_i^{m'} + \sum_{\substack{m' \in M_{S_n^k}: \\ m' > m}} y_n^{km'} + \sum_{n'=n+1}^{n^k} x_{n'}^k \leq 1$$

$$\forall k \in K, n \in N^k, m \in M_{S_n^k}, i \in S_n^k : m \in M_i^k, \tag{4f}$$

$$x_n^k \geq \sum_{m \in M_{S_n^k}} y_n^{km} \quad \forall k \in K, n \in N^k, \tag{4g}$$

$$z_n^k \leq \sum_{m \in M_{S_n^k}} b^m y_n^{km} \quad \forall k \in K, n \in N^k, \tag{4h}$$

$$v_i^m \in \{0,1\} \quad \forall i \in I, m \in M_i, \tag{4i}$$

$$y_n^{km} \in \{0,1\} \quad \forall k \in K, n \in N^k, m \in M_{S_n^k}, \tag{4j}$$

$$x_n^k \in \{0,1\} \quad \forall k \in K, n \in N^k, \tag{4k}$$

$$z_n^k \geq 0 \quad \forall k \in K, n \in N^k. \tag{4l}$$

Constraints (1b), (1c), (1d) and (1e) from (3IM) correspond to (4b), (4c), (4d) and (4f)), respectively. As for (RM), constraints (3b), (3c) and (3d) are, respectively, constraints (4b), (4c) and (4e). Constraints (3e) are a subset of (3e), whereas constraints (3f) and (3g) are no longer necessary due to the addition of (4h). Finally, even though (4g) and (4h) are added as inequality constraints, in any optimal solution they will be satisfied as equalities. This is clear for constraints (4h) because of the objective. As for constraints (4g), suppose with the aim of contradiction that there exists an optimal solution $(\bar{v}, \bar{y}, \bar{x}, \bar{z})$ of (3IM+) with $\bar{x}_n^k > \sum_{m \in M_{S_n^k}} \bar{y}_n^{km}$ for some $k \in K$, $n \in N^k$. On the one hand, (4g) imply $\sum_{n \in N^k} \bar{x}_n^k > \sum_{n \in N^k} \sum_{m \in M_{S_n^k}} \bar{y}_n^{km}$, and because of the integrality constraints this leads to $\sum_{n \in N^k} \sum_{m \in M_{S_n^k}} \bar{y}_n^{km} = 0$, so $\sum_{n \in N^k} \bar{z}_n^k = 0$ due to (4h). On the other hand, $\bar{x}_n^k = 1$ and (4e) imply $\sum_{m \in M_{S_n^k}} \bar{v}_i^m = 1$ for some $i \in S_n^k$. As a result, customer k can afford product i but $\sum_{n \in N^k} \bar{z}_n^k = 0$, so the solution is not optimal.

Proposition 5.3. *Consider a fixed customer $k \in K$ and a fixed set of products $S_n^k \in \mathscr{S}^k$. Then the following family of constraints*

$$z_n^k \leq x_n^k \alpha + \sum_{m \in M_{S_n^k}} \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} \left(1 - \sum_{m' \leq m} v_i^{m'} - \sum_{n'=n+1}^{n^k} x_{n'}^k \right) \beta_i^m + \sum_{m \in M_{S_n^k}} \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} v_i^m \gamma^m \tag{5}$$

is valid for (RM) *if for all $\alpha \geq 0$, $\beta_i^m \geq 0$ for $i \in S_n^k$, $m \in M_i^k$, $\gamma^m \geq 0$ for $m \in M_{S_n^k}$, it holds*

$$\alpha + \sum_{\substack{m' \in M_{S_n^k}: \\ m' < m}} \sum_{\substack{i \in S_n^k: \\ m' \in M_i^k}} \beta_i^{m'} + \gamma^m \geq b^m \qquad \forall m \in M_{S_n^k}. \tag{6}$$

Furthermore, the linear relaxation of (RM) *plus the set of valid inequalities* (5) *is exactly the projection of the linear relaxation of* (3IM) *on the space of variables* $(\boldsymbol{v}, \boldsymbol{x}, \boldsymbol{z})$*.*

Proof. For fixed $k \in K$, $n \in N^k$, we are going to project out the y-variables of formulation

(3IM+) to obtain (5) and prove the statement. We make use of (4g) and (4h) from (3IM+), and we associate dual variables α, β_i^m, γ^m, δ to the corresponding constraints (4g), (4f), (4d), (4h), respectively. By Farkas' Lemma, we have the following result: given a feasible solution $(\boldsymbol{v}, \boldsymbol{x}, \boldsymbol{z})$ of the linear relaxation of (RM), there exists a vector $\mathbf{y}$ satisfying (4d)-(4f) if and only if it holds

$$z_n^k \delta \leq x_n^k \alpha + \sum_{m \in M_{S_n^k}} \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} \left(1 - \sum_{m' \leq m} v_i^{m'} - \sum_{n'=n+1}^{n^k} x_{n'}^k\right) \beta_i^m + \sum_{m \in M_{S_n^k}} \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} v_i^m \gamma^m \qquad (7)$$

$\forall k \in K$, $n \in N^k$, and $\forall (\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}, \delta) \geq \mathbf{0}$ such that

$$\alpha + \sum_{\substack{m' \in M_{S_n^k}: \\ m' < m}} \sum_{\substack{i \in S_n^k: \\ m' \in M_i^k}} \beta_i^{m'} + \gamma^m \geq b^m \delta \qquad \forall m \in M_{S_n^k}. \qquad (8)$$

If $\delta > 0$, We obtain (5) if we normalize by setting $\delta = 1$.

If $\delta = 0$, the obtained inequality is dominated by (3e) and the nonnegativity constraints on varibles v_i^m and x_n^k. It is indeed easy to see that for any feasible solution of (RM), the RHS of (5.2) is nonnegative. □

Proposition 5.3 provides a family of valid inequalities for (RM) of infinite size. Therefore, their inclusion in the model requires the election of a subset of them following a separation procedure. Below, we formally determine the separation problem and show that it is equivalent to a minimum cost flow problem (MCFP).

Let us assume we are given a fractional optimal solution $(\bar{v}_i^m, \bar{x}_n^k, \bar{z}_n^k)$ of the linear relaxation of (RM) or a current solution at a given node of the search tree. We solve a separation problem for each customer k and equivalence class $S_n^k \in \mathscr{S}^k$.

First of all, the special structure of conditions (6) implies that to minimize of RHS in (5), we can set, for each m, at most one β_i^m to a positive value. More precisely, for each m, we define $i_m \in \arg\min_{i \in S_n^k : m \in M_i^k} \left\{1 - \sum_{m' \leq m} v_i^{m'} - \sum_{n'=n+1}^{n^k} x_{n'}^k\right\}$ and then set $\beta_i^m = 0$ $\forall i \in S_n^k$, $i \neq i_m$. Hence, the separation problem (SP_n^k) can be stated as:

$$(\mathrm{SP}_n^k) \quad \min_{\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}} \quad \bar{x}_n^k \alpha + \sum_{m \in M_{S_n^k}} \left(1 - \sum_{\substack{m' \in M_{i_m}^k: \\ m' \leq m}} \bar{v}_{i_m}^{m'} - \sum_{n'=n+1}^{n^k} \bar{x}_{n'}^k\right) \beta_{i_m}^m + \sum_{m \in M_{S_n^k}} \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} \bar{v}_i^m \gamma^m \qquad (9a)$$

$$\text{s.t.} \quad \alpha + \sum_{m' < m} \beta_{i_{m'}}^{m'} + \gamma^m \geq b^m \qquad \forall m \in M_{S_n^k}, \qquad (9b)$$

$$\alpha, \beta_{i_m}^m, \gamma^m \geq 0 \qquad \forall m \in M_{S_n^k}. \qquad (9c)$$

Problem (SP_n^k) is linear and the matrix associated to constraints (9b) is binary and possesses the Consecutive Ones Property: the elements equal to 1 in each column appear consecutively. This property permits to solve the problem as a MCFP, see e.g. page 304 in Ahuja et al. [2]. We now describe how to derive this MCFP.

To begin with, we sort the budgets $b^m, m \in M_{S_n^k}$ by increasing order of their values. Then, we transform the constraints in (9b) into equalities by introducing slack variables δ^m for each row m in (9b). We also add the row $0 \cdot \alpha + 0 \cdot \sum_{m \in M_n^k} \beta_{i_m}^m + 0 \cdot \sum_{m \in M_n^k} \gamma^m + 0 \cdot \sum_{m \in M_n^k} \delta^m = 0$. These modifications lead to an equivalent formulation with the same objective function (9a) and the following constraints:

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & -1 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \boldsymbol{\beta_{i_m}^m} \\ \\ \boldsymbol{\gamma^m} \\ \\ \boldsymbol{\delta^m} \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^{\sigma(k)} \\ 0 \end{bmatrix},$$

$$\alpha, \beta_{i_m}^m, \gamma^m \geq 0 \qquad \forall m \in M_{S_n^k}.$$

To finish the transformation, we carry out a row operation for each $m = \sigma(k), \sigma(k)-1, \ldots, 1$ in this order: we subtract the m-th constraint to the $(m+1)$-th one. The equivalent linear formulation (SP-MCFP$_n^k$) obtained is:

$$\min_{\alpha,\boldsymbol{\beta},\boldsymbol{\gamma}} \quad \alpha \bar{x}_n^k + \sum_{m \in M_{S_n^k}} \beta_{i_m}^m \left(1 - \sum_{\substack{m' \in M_{i_m}^k: \\ m' \leq m}} \bar{v}_{i_m}^{m'} - \sum_{n'=n+1}^{n^k} \bar{x}_{n'}^k \right) + \sum_{m \in M_{S_n^k}} \gamma^m \sum_{\substack{i \in S_n^k: \\ m \in M_i^k}} \bar{v}_i^m \tag{10a}$$

$$\text{s.t.} \quad \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & -1 & 1 & \cdots & 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & -1 \\ -1 & -1 & \cdots & -1 & 0 & 0 & \cdots & -1 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \boldsymbol{\beta_{i_m}^m} \\ \\ \boldsymbol{\gamma^m} \\ \\ \boldsymbol{\delta^m} \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 - b^1 \\ \vdots \\ b^{\sigma(k)} - b^{\sigma(k)-1} \\ -b^{\sigma(k)} \end{bmatrix}, \tag{10b}$$

$$\alpha, \beta_{i_m}^m, \gamma^m \geq 0 \qquad \forall m \in M_{S_n^k}. \tag{10c}$$

The constraint matrix in (10b) is the incidence matrix of a graph $G = (N, A)$. Each row corresponds to a node in $N = M_{S_n^k}$ whose supply/demand is given by the corresponding RHS of (10b) and each column corresponds to an arc. Hence, the variables represent uncapacitated flows on the arcs and the objective function consists in minimizing the total cost of the flow. The node corresponding to the last row is the unique sink with demand $b^{\sigma(k)}$ and all other nodes are sources with offer equal to the difference of two consecutive budget values in $M_{S_n^k}$. The MCFP corresponding to problem (SP-MCFP$_n^k$) is illustrated in Figure 1. Given that there is no capacity on the arcs and there is only one sink, the problem can be solved in $M_{S_n^k}$ steps, by

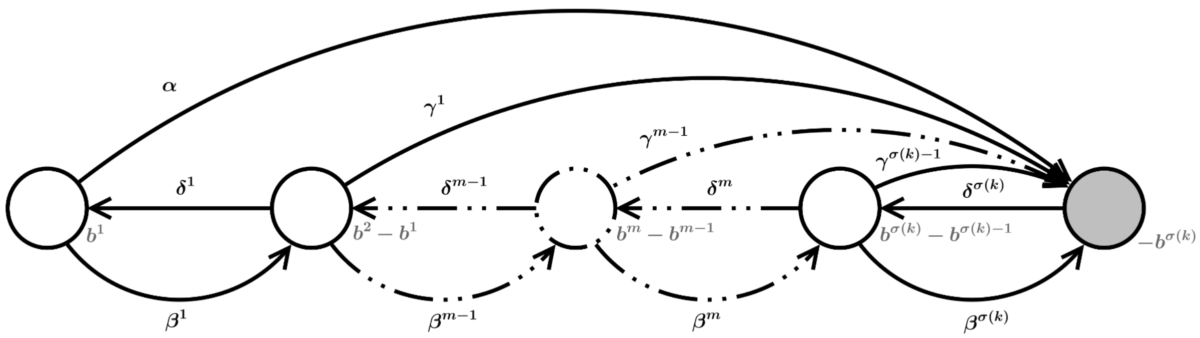

Figure 1: MCFP corresponding to (SP-MCFP^k_n). Next to each node we have its supply/demand, and variables $(\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta})$ are associated to the flow of the corresponding arc

computing one shortest path from each source to the sink.

To solve RPPT with formulation (RM) we thus use a branch and cut algorithm that adds violated inequalities from (5) at the root node as well as at every node of the branch and bound tree of depth less than 4. Algorithm 1 details the different steps of the separation procedure.

Algorithm 1 Resolution of the separation problems (SP^k_n)

Let $(\bar{v}^m_i, \bar{x}^k_n, \bar{z}^k_n)$ be an optimal fractional solution of the linear relaxation of (RM) or a solution found in a node of the search tree of depth less than 4.
For every customer $k \in K$ and integer $n \in N^k$, do

Step 1. Obtain $\bar{i}_m \in \arg\min_{i \in S^k_n} \left\{ 1 - \sum_{m' \leq m} \bar{v}^{m'}_i - \sum_{n'=n+1}^{n^k} \bar{x}^k_{n'} \right\} \; \forall m \in M_{S^k_n}$.
Step 2. Transform the instance of (SP^k_n) into an instance of the MCFP.
Step 3. Compute an optimal flow on the corresponding graph of the instance of MCFP, obtaining $\bar{\alpha}$, $\bar{\beta}^m_{i_m}$, $\bar{\gamma}^m \; \forall m \in M_{S^k_n}$.
Step 4. Incorporate constraint

$$z^k_n \leq \bar{\alpha} x^k_n + \sum_{m \in M_{S^k_n}} \bar{\beta}^m_{i_m} \left(1 - \sum_{\substack{m' \in M^k_{i_m}: \\ m' \leq m}} v^{m'}_{i_m} - \sum_{n'=n+1}^{n^k} x^k_{n'} \right) + \sum_{m \in M_{S^k_n}} \bar{\gamma}^m \sum_{\substack{i \in S^k_n: \\ m \in M^k_i}} v^m_i$$

to (RM) provided that it is violated.

---

6 Solution of (3IM) via Benders Decomposition: the Benders Model

Formulation (3IM) yields very good linear relaxation bounds but it has a large number of variables and constraints. However, as shown in this section, its structure allows for its resolution by means of a Benders decomposition.

First, we introduce the Benders Model (BM). To reformulate (3IM), we need to be able to relax the integrality on the set of y-variables. However, this result is shown in Proposition (4.2). We address the Benders reformulation of (3IM) and relate it to the Benders Model in the following subsections.

We define continuous variables z^k, $\forall k \in K$, that represent the profit from customer k. With this set of variables and the set of v-variables used for (3IM) and (RM), we present the Benders Model (BM) for RPPT:

$$
\begin{aligned}
\text{(BM)} \quad \max_{\mathbf{v},\mathbf{z}} \quad & \sum_{k \in K} z^k && \text{(11a)}\\
\text{s.t.} \quad & \sum_{m \in M_i} v_i^m \leq 1 \quad \forall i \in I, && \text{(11b)}\\
& z^k \leq \sum_{i \in I^k} \sum_{m \in M_i^k} b^m v_i^m \quad \forall k \in K, && \text{(11c)}\\
& z^k \leq b^{\sigma(k)} + \sum_{m \in M_i^k} \left(b^m - b^{\sigma(k)}\right) v_i^m + \sum_{\substack{j \in I^k:\\ j \prec_k i}} \sum_{m \in M_j^k} b^m v_j^m \\
& \qquad \forall k \in K, i \in I^k, && \text{(11d)}\\
& v_i^m \in \{0,1\} \quad \forall i \in I, m \in M_i, && \text{(11e)}\\
& z^k \geq 0 \quad \forall k \in K. && \text{(11f)}
\end{aligned}
$$

Proposition 6.1. *Model* (BM) *is valid for* RPPT.

The proof of Proposition 6.1 is fully detailed in Appendix A.

6.1 Benders Reformulation

We can proceed with a Benders reformulation of (3IM):

$$
\begin{aligned}
(\text{BR}_{\text{MAS}}) \quad \max_{\mathbf{v},\mathbf{z}} \quad & \sum_{k \in K} z^k && \text{(12a)}\\
\text{s.t.} \quad & \sum_{m \in M_i} v_i^m \leq 1 \quad \forall i \in I, && \text{(12b)}\\
& z^k \leq P^k(v), \quad \forall k \in K && \text{(12c)}\\
& v_i^m \in \{0,1\} \quad \forall i \in I, m \in M_i, && \text{(12d)}\\
& z^k \geq 0 \quad \forall k \in K, && \text{(12e)}
\end{aligned}
$$

where $\forall k \in K$, $P^k(v)$ is defined as the optimal value of

$$
\begin{aligned}
(\text{BR}_{\text{SUB}^k}) \quad \max_{\mathbf{y}} \quad & \sum_{n \in N^k} \sum_{m \in M_{S_n^k}} b^m y_n^m && \text{(13a)}\\
\text{s.t.} \quad & \sum_{n \in N^k} \sum_{m \in M_{S_n^k}} y_n^m \leq 1, && \text{(13b)}
\end{aligned}
$$

$$y_n^m \leq \sum_{i \in S_n^k} v_i^m \qquad \forall n \in N^k, m \in M_{S_n^k}, \tag{13c}$$

$$\sum_{\substack{m' \in M_{S_n^k}: \\ m' \geq m}} y_n^{m'} + \sum_{n'=n+1}^{n^k} \sum_{m' \in M_{S_{n'}^k}} y_{n'}^{m'} \leq 1 - \sum_{\substack{m' \in M_i^k: \\ m' < m}} v_i^{m'}$$

$$\forall n \in N^k, i \in S_n^k, m \in M_i^k, \tag{13d}$$

$$y_n^m \geq 0 \quad \forall n \in N^k, m \in M_{S_n^k}. \tag{13e}$$

In (BR_{SUB^k}), we drop the upper index k of the y-variables for the sake of notation. Constraints (12b) ensure that every product price is unique. This guarantees the feasibility in problem (BR_{SUB^k}) for a given integer solution (v_i^m) of (3IM), since the RHS of constraints (13b)-(13d) is always nonnegative. Furthermore, constraint (13b) ensures that (BR_{SUB^k}) is bounded. Therefore, by linear optimization strong duality, the optimal value of problem (BR_{SUB^k}) is equal to the optimal value of its dual problem, ($\text{BR}_{\text{SUBD}^k}$). Associating variables $\alpha, \beta_i^m, \gamma_n^m$ to the corresponding constraint from sets (13b), (13d), (13c), respectively, ($\text{BR}_{\text{SUBD}^k}$) can be stated as

$$(\text{BR}_{\text{SUBD}^k}) \quad \min_{\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}} \quad \alpha + \sum_{n \in N^k} \sum_{i \in S_n^k} \sum_{m \in M_i^k} \left(1 - \sum_{m' \in M_i^k : m' < m} v_i^{m'} \right) \beta_i^m + \sum_{n \in N^k} \sum_{i \in S_n^k} \sum_{m \in M_i^k} v_i^m \gamma_n^m \tag{14a}$$

$$\text{s.t.} \quad \alpha + \sum_{n'=1}^{n-1} \sum_{i \in S_{n'}^k} \sum_{m' \in M_i^k} \beta_i^{m'} + \sum_{i \in S_n^k} \sum_{\substack{m' \in M_i^k: \\ m' \leq m}} \beta_i^{m'} + \gamma_n^m \geq b^m$$

$$\forall n \in N^k, m \in M_{S_n^k}, \tag{14b}$$

$$\alpha, \beta_i^m, \gamma_n^m \geq 0 \qquad \forall n \in N^k, m \in M_{S_n^k}, i \in S_n^k. \tag{14c}$$

Now, we can rewrite problem (BR_{MAS}) making use of subproblems ($\text{BR}_{\text{SUBD}^k}$). Thus, defining D^k for each $k \in K$ as the set of feasible solutions $(\alpha^k, \beta_i^{km}, \gamma_n^{km})$ for the dual subproblem ($\text{BR}_{\text{SUBD}^k}$), we have:

$$(\text{BR}_{\text{MAS}}) \quad \max_{\mathbf{v}, \mathbf{z}} \quad \sum_{k \in K} z^k \tag{15a}$$

$$\text{s.t.} \quad \sum_{m \in M_i} v_i^m \leq 1 \quad \forall i \in I, \tag{15b}$$

$$z^k \leq \alpha^k + \sum_{n \in N^k} \sum_{i \in S_n^k} \sum_{m \in M_i^k} \left(1 - \sum_{m' \in M_i^k : m' < m} v_i^{m'} \right) \beta_i^{km} + \sum_{n \in N^k} \sum_{i \in S_n^k} \sum_{m \in M_i^k} v_i^m \gamma_n^{km}, \quad \forall k \in K, (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \in D^k, \tag{15c}$$

$$v_i^m \in \{0,1\} \quad \forall i \in I, m \in M_i, \tag{15d}$$
$$z^k \geq 0 \quad \forall k \in K. \tag{15e}$$

Finally, we state (the proof is in Appendix A) that model (BR_{MAS}) obtained by means of a Benders reformulation is in fact a reinforcement of the previous Benders Model (BM):

Proposition 6.2. *The sets of constraints* (11c) *and* (11d) *are included in* (15c).

6.2 Resolution Approach

The classical Benders resolution approach begins by solving to optimality the master problem (BR_{MAS}) without constraints (15c). Then, a subset of constraints from (15c) is obtained by solving problems ($\text{BR}_{\text{SUBD}^k}$) for all $k \in K$, and the violated constraints are added to the master problem, which is again solved to optimality. This process is done iteratively until none of the constraints from (15c) is violated, and thus the solution is optimal for (BR_{MAS}). The drawback of this method is that (BR_{MAS}), that is an IP, is solved many times, which can take a considerable amount of time.

In the lazy approach, however, the resolution starts by solving the linear relaxation of (BR_{MAS}) without the set (15c), obtaining a fractional solution and an upper bound on the optimal value. In order to decrease this bound, the subproblems ($\text{BR}_{\text{SUBD}^k}$) are solved for each customer using the fractional solution of the master problem, and a set of constraints is added to the problem. Constraints are added at this phase until the bound is no longer improved. The second step of the resolution is to solve the integer problem with the usual branch-and-bound algorithm. In this phase, constraints are added in the so-called lazy fashion, i.e. only checking for them when the resolution of a node in the search tree leads to an integer solution. In such case, if a constraint is violated, the cut is pulled into the active node and the solution is discarded. Otherwise, the solution is feasible for (BR_{MAS}). At this step, constraints from (15c) may also be added at a current fractional node of the branching tree. The interested reader may find the advantages of this method thoroughly explained in Naoum-Sawaya and Elhedhli [24].

In this work, we solve the Benders Model (BM) instead of (BR_{MAS}). The advantage is that, since (BM) gives feasible solutions for RPPT, we no longer need to solve ($\text{BR}_{\text{SUBD}^k}$) in order to check the validity of an integer solution of the master. Nevertheless, we can still add valid inequalities from (15c) to cut off fractional solutions of (BM), thus strengthening the model.

Resolution of the dual subproblem ($\text{BR}_{\text{SUBD}^k}$))

Solving problem ($\text{BR}_{\text{SUBD}^k}$) for fractional solutions of (BM) is interesting because it allows for the incorporation of valid inequalities in the linear relaxation phase, thus helping to decrease the upper bound before solving the integer phase. An analogous procedure to that of the resolution of the separation problem (SP_n^k) in Subsection 5.2 can be applied to ($\mathrm{BR}_{\mathrm{SUBD}^k}$). In this case, it suffices to consider the lexicographical order in the rows of matrix (14b) (that is, $(n, m) < (n', m')$ if $n < n'$ or $n = n'$ and $m < m'$) in order to state that it also satisfies the Consecutive Ones Property. Thus, ($\mathrm{BR}_{\mathrm{SUBD}^k}$) can be transformed into a MCFP and solved by means of an efficient implementation of an existing algorithm.

For our implementation, we have selected the Successive Shortest Path (SSP) Algorithm to obtain the solution of the MCFP. In each iteration, this algorithm selects a shortest path between a supply and a demand node and increases the flow along the path (it also modifies the reduced costs of the arcs used to compute the shortest path and the residual network in each iteration). Since our graph has $\sum_{n\in N^k} |M_{S_n^k}| + 1$ nodes, solving the problem for a given customer k can take at most $\sum_{n\in N^k} |M_{S_n^k}|$ iterations. When the preference matrix is dense, this amounts to $|N^k|\sigma(k)$.

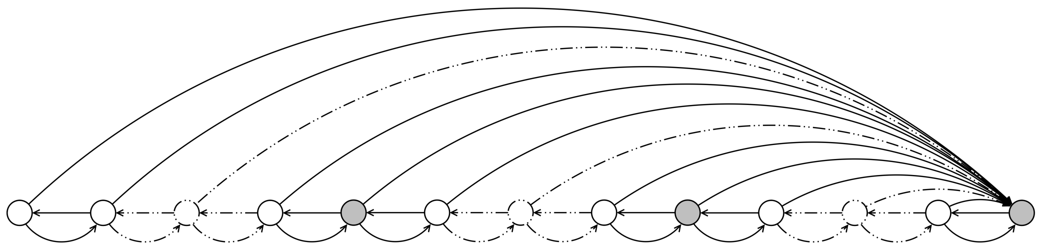

Figure 2: Structure of the MCFP graph corresponding to ($\mathrm{BR}_{\mathrm{SUBD}^k}$). Source nodes appear in white and sink nodes appear in gray

Leveraging the special structure of our MCFP, we have reduced the number of iterations in which a shortest path is computed. The structure of the graph associated to our MCFP is depicted in Figure 2. As in the graph from Figure 1, the white nodes represent sources, the gray ones represent sinks and sending flow through δ-arcs (the arcs from a node to the previous one) has cost equal to zero. Hence, we need not compute the shortest path between a node with excess supply (n, m) and a node with unfulfilled demand (n', m') whenever $(n, m) > (n', m')$. In the first phase of the algorithm, we select a source (n, m) and a sink (n', m') with $(n, m) > (n', m')$, and then apply the SSP algorithm without computing the shortest path. Then, when for all supply node (n, m) and demand node (n', m') it holds $(n, m) < (n', m')$, we continue with the second phase, where we apply the SSP algorithm in the standard way. This preprocessing of the MCFP reduces the number of iterations in which an algorithm to obtain a shortest path is executed to at most $\sigma(k)$ iterations. Thus, the amount of computational time saved during the first phase is significant.

Finally, note that the transformation of the subproblems into a MCFP can also be used to solve the subproblems of the Benders decomposition proposed by Bertsimas and Mišić [5] for the resolution of PLD. Indeed, the Consecutive Ones Property holds in this case as well.

In-out stabilization method and overall resolution approach

In this subsection, we present our resolution strategy to solve model (BM) as well as an in-out stabilization method implemented to speed up the linear relaxation phase of the resolution.

The procedure is divided in two phases:

1. **Linear relaxation phase.** The linear relaxation of (BM) is solved, obtaining a fractional solution and an upper bound on the optimal value. In order to decrease this bound, the corresponding MCFP of subproblems ($\mathrm{BR}_{\mathrm{SUBD}^k}$) are solved for each customer and for the fixed fractional solution of the master, and a set of valid inequalities from (15c) is derived and added to the formulation. Valid inequalities are added at this phase until the upper bound is no longer improved.

2. **Integer phase.** The integer problem with the subset of constraints derived in the previous phase is solved to optimality by means of a branch-and-cut. Due to the fact that it is very time consuming, no more valid inequalities from (15c) are added in this phase.

As we have proved, the SSP algorithm used to solve the transformation of subproblems ($\mathrm{BR}_{\mathrm{SUBD}^k}$) into a MCFP constitutes an exact algorithm of separation. In this sense, it finds at least one violated constraint for any solution of (BM) which is infeasible for ($\mathrm{BR}_{\mathrm{MAS}}$). On the other hand, when the problem size is large, computing these inequalities is time consuming, and frequently the upper bound decreases very slowly and many cuts are generated in the process. In order to speed up this cutting phase, we implemented an in-out stabilization method with the aim of generating less cuts of better quality. The steps of the cutting plane in-out algorithm are detailed in Ben-Ameur and Neto [4] and Bonami et al. [6].

Let D represent the domain given by all the constraints of problem ($\mathrm{BR}_{\mathrm{MAS}}$), and $P \supseteq D$ the domain given by the constraints from (BM). Then the in-out stabilization method is based on the election of good separation points. Specifically, at each loop iteration of the linear relaxation phase three points are considered: a point $(v_{out}, z_{out}) \in P \setminus D$ given by the optimal solution of the linear relaxation of the current reduced master problem (BM), a feasible interior point $(v_{in}, z_{in}) \in D$, and a separation point (v_{sep}, z_{sep}), which is a convex combination of the previous two: $(v_{sep}, z_{sep}) := \lambda(v_{out}, z_{out}) + (1-\lambda)(v_{in}, z_{in})$ with $\lambda \in (0, 1]$. At each iteration, two possibilities can occur. If $(v_{sep}, z_{sep}) \notin D$, then we use it instead of (v_{out}, z_{out}) as a separation point to solve the dual subproblem ($\mathrm{BR}_{\mathrm{SUBD}^k}$), since the inequalities provided by this point are expected to be more efficient. We finish the iteration by solving the new optimization problem and obtaining a new point (v_{out}, z_{out}). Otherwise, $(v_{sep}, z_{sep}) \in D$, and in this case solving the dual subproblem does not provide new violated cuts. Therefore, in this iteration no constraints are added but (v_{in}, z_{in}) is replaced with (v_{sep}, z_{sep}), which is a feasible point with greater objective value. As we can see, at each iteration either (v_{in}, z_{in}) or (v_{out}, z_{out}) are updated, until convergence is obtained because the relative difference between the two points is lower than a fixed tolerance ϵ. Although λ is a scalar that can change in every iteration, preliminary testing led us to set $\lambda = 0.99$ for all iterations. As for the interior point (v_{in}, z_{in}), it is frequently obtained using the barrier algorithm with crossover.

In our case, an interior point $(v_{in}, z_{in}) \in D$ can be very easily derived by exploiting the particular structure of the problem. To do so, it suffices to build a non-degenerate convex combination of $|I \times M^I| + |K| + 1$ points of the polytope and then compute the centroid. Point (i, m) of the first feasible set of $|I \times M^I|$ points was created taking $v_i^m = 1$, $v_{i'}^{m'} = 0$ for $(i', m') \neq (i, m)$, $\boldsymbol{z} = \mathbf{0}$. Point k of the next $|K|$ points is $z^k = b^{\sigma(k)}$, $z^{k'} = 0$ for $k' \neq k$, $v_i^m = 1$ for $i = \min\{i \in S_1^k\}$, $m = \sigma(k)$, $v_{i'}^{m'} = 0$ for $(i', m') \neq (i, m)$. Finally, we used $(\boldsymbol{v}, \boldsymbol{z}) = \mathbf{0}$.

7 Preprocessing

In this section, we present a preprocessing procedure with the aim of reducing the size of the problem by fixing variables to zero. Note that, even though the results are stated for models (RM) and (3IM), they also apply to subproblems ($\text{BR}_{\text{SUBD}^k}$) during the resolution of model (BM). This preprocessing is based on the one described in Calvete et al. [8] for RPP problem. We define a recursive function $u' : K \to \mathscr{S}^K$ that assigns the index n of an equivalence class $S_n^k \in \mathscr{S}^k$ to each customer $k \in K$. Function u' is defined as follows, for the set of customers ordered according to their budgets in decreasing order:

1. If $\sigma(k) = |M|$, then $u'(k) := 1$.
2. If $\sigma(k) < |M|$ and it holds $I^k \nsubseteq \left(\cup_{\substack{k' \in K: \\ \sigma(k') > \sigma(k)}} S_{u'(k')}^{k'} \right)$, then
$u'(k) := \min \left\{ n \in N^k : S_n^k \nsubseteq \left(\cup_{\substack{k' \in K: \\ \sigma(k') > \sigma(k)}} S_{u'(k')}^{k'} \right) \right\}$.
3. If $\sigma(k) < |M|$ and it holds $I^k \subseteq \left(\cup_{\substack{k' \in K: \\ \sigma(k') > \sigma(k)}} S_{u'(k')}^{k'} \right)$, then $u'(k) := n^k$.

Proposition 7.1. *For* (RM) *(resp.* (3IM)*), there exists an optimal solution* $(\bar{v}_i^m, \bar{x}_n^k, \bar{z}_n^k)$ *(resp.* $(\bar{v}_i^m, \bar{y}_n^{km})$*) such that* $\bar{x}_n^k = 0$ *(resp.* $\bar{y}_n^{km} = 0$*) for all* $k \in K$, $n > u'(k)$, $m \in M_{S_n^k}$.

Let C_r, $r = \{1, 2, 3\}$, be such that $k \in C_r$ if and only if $u'(k)$ has been defined for k making use of item r of the definition of u'. Finally, we give a condition under which an optimal solution can be found by inspection.

Corollary 7.2. *If* $C_3 = \emptyset$*, an optimal solution of* (RM) *(resp.* (3IM)*) can be found by inspection.*

The proofs of Proposition 7.1 and Corollary 7.2 can be found in Appendix A.

8 Computational results

Extensive computational experiments were carried out to compare the performance of (RM) and (BM) in terms of the number of nodes of the branching tree, computational time and integrality gap, as well as the performance of the valid inequalities derived for both models and the preprocessing techniques. We implemented both models by means of Mosel version 4.0.3 of Xpress-MP, Optimizer version 29.01.10, running on a Dell PowerEdge T110 II Server (Intel Xeon E3-1270, 3.40 GHz) with 16 GB of RAM.

Regarding the instances, we modified those proposed in Calvete et al. [8]. This set was designed following a model based on the Characteristics Model proposed by Fernandes et al. [14]. Calvete et al. generated instances for $|K| = 50$, $|K| = 100$ and $|K| = 150$ customers and $0.1|K|$, $0.5|K|$ and $|K|$ products. For each size, they generated four instances modifying $|I^k|$. Out of the four, we consider the instances with three sizes, namely $|I^k| = \lceil 0.2|I| \rceil$, $|I^k| = \lceil 0.5|I| \rceil$ and $|I^k| = |I|$. The budgets of the customers and their ranked lists of preferences were randomly generated between 1 and $2|K|$.

These instances were proposed in [8] for RPP, so we modified them by adding ties in the ranked lists of preferences of the customers. Thus, for each size we generated three instances varying the number of ties in the list of preferences (denoted *Ties* in Tables 4-6 and in the following), with 1, 2, 3, 5 or 10 ties depending on the instance. This parameter establishes the relationship between $|I^k|$ and $n^k = |\mathscr{S}^k|$ in the following way: $|I^k| - \text{Ties} = n^k$. We generated 5 instances of each size, 365 in total. The time limit was set to 3600 seconds, and the default setting of Xpress was used.

For completeness, we report the results of the computational experiments in three tables grouped in Appendix B. Tables 4, 5 and 6 contain all the data concerning the instances of sizes $|K| = 50$, $|K| = 100$ and $|K| = 150$, respectively. In the remaining of the section, the most significant information from those tables is summarized by means of several figures. Models (RM) and (BM), as well as models (RM) and (BM) with the corresponding branch-and-cut procedures and preprocessing techniques, are shown in the legends of the figures as RM, BM, RM+VI+prepro and BM+VI+prepro, respectively.

Figure 3 is a performance profile that shows the percentage of instances having an integrality gap less than or equal to the value on the x-axis. For models (RM) and (BM), the integrality gap is $\text{RLGap} = 100\frac{\text{UB}-\text{BV}}{\text{OV}}$, where UB represents the upper bound given by the linear relaxation, BV is the best value found by any of the models for such instance and OV is the best objective value found by any of the models (the optimal value in most cases). As for models (RM) and (BM) with the branch-and-cut procedure and the preprocessing techniques, the integrality gap represented corresponds to: $\text{RGap} = 100\frac{\text{UBC}-\text{OV}}{\text{OV}}$, where UBC is the upper bound obtained after adding the cuts in the root node. Figure 3 shows that the linear relaxation bound given by model (BM) is in general much smaller than that of (RM), which in some cases goes up to a gap of 50%. Moreover, the cuts added in the root node are very efficient in both cases in reducing the gap. Adding these cuts leads to gaps 2-3% in 80% of the instances, and gaps smaller than 14% in all the instances. As we explained throughout the paper, the upper bound in this case is in fact the bound provided by formulation (3IM), and this is why the integrality gap is roughly the same for both models (since the value BV used is the same in all cases). Hence, Figure 3 illustrates the decisive role of the valid inequalities derived in Sections 5 and 6 when reducing

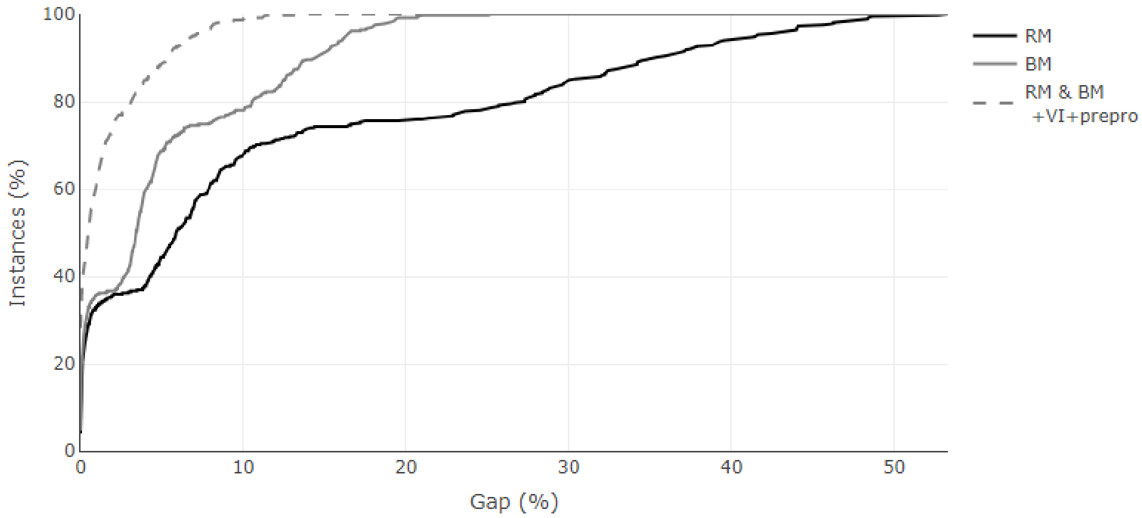

Figure 3: In the y-axis, the percentage of instances with an integrality gap less than or equal to that of the x-value is represented for models (RM), (BM) and (RM) and (BM) with the branch-and-cut procedures and the preprocessing techniques

the upper bounds to close the integrality gap and reach optimality.

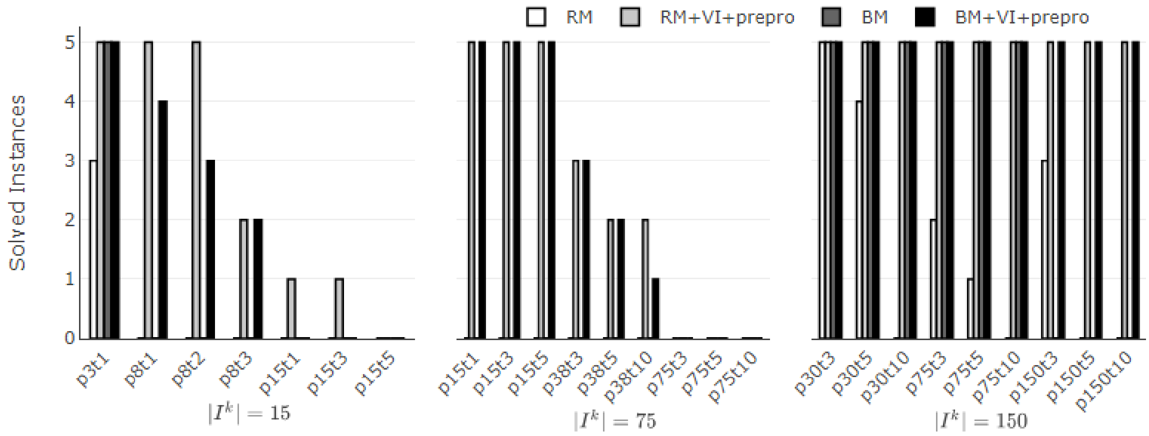

Figure 4: Percentage of solved instances with $|K| = 150$, depending on their size. The size of the set of products is included at the bottom of the corresponding group of bars, the number of products in the list of preference of any customer ($|I^k|$) appears after the letter p in the notation of the instances, and the number of Ties of every customer is shown after the letter t

Models (RM) and (BM) solved to optimality the majority of the instances with 50 customers, and the same models including the branch-and-cut and the preprocessing solved all of them. As for the biggest instances, Figure 4 shows the number of instances with 150 customers solved by each of the four models, depending on their size. As we can see, the relationship between the number of customers $|K|$ and products $|I|$ determines the difficulty of the instance: the instances with $|I| = 0.5|K|$ (the ones *in the middle* of the table) are generally the most difficult ones. Only (RM) and (BM) with the branch-and-cut and preprocessing are able to solve some of the instances with $|K| = 150$ and $|I| = 75$. The fact that they are more difficult than those with $|I| = 0.1|K|$ is explained because the preference matrices of the latter ones are less dense and they have a much smaller number of variables and constraints, so the branch-and-cut and the branching procedures are faster. As for the instances with $|K| = |I|$ (the ones *at the right* of the table), they are easier due to the preprocessing techniques, which eliminate a great number of customer decision variables when the number of products is big compared to the number of customers. Within the instances with the same amount of customers and products, the increase in the numbers of products in the list of preferences of each customer ($|I^k|$) also increases the difficulty of the instance, as well as the growth in the number of Ties.

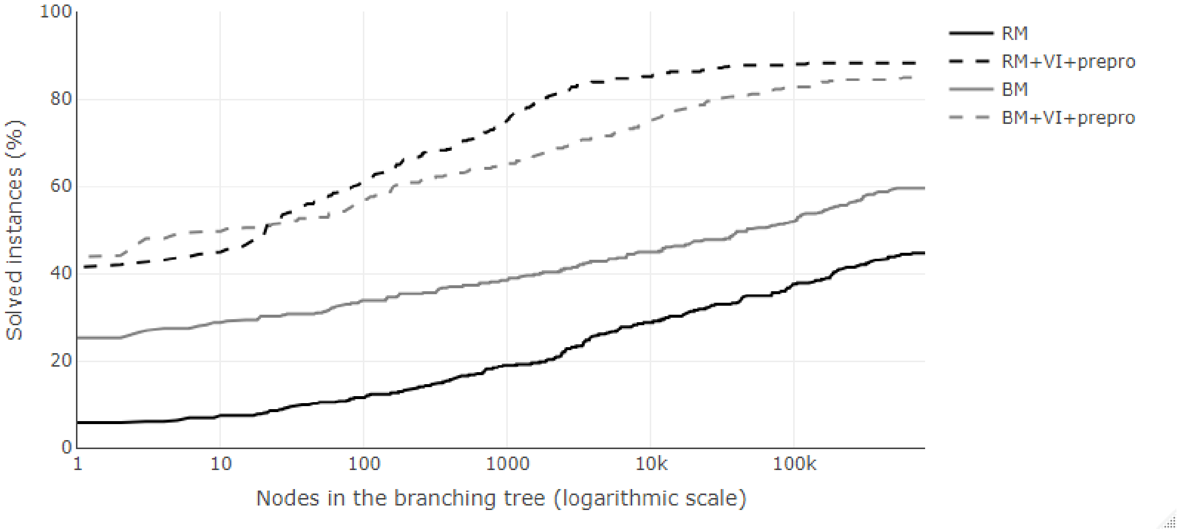

Figure 5: Percentage of solved instances depending on the number of nodes explored in the branching tree by models (RM), (BM), and (RM) and (BM) with the corresponding branch-and-cut procedures and the preprocessing techniques

We also compared the performance of the four models in terms of the number of nodes explored during the branching process. Figure 5 shows the percentage of solved instances depending on the number of nodes explored in the branching tree by models (RM), (BM), and (RM) and (BM) with the corresponding branch-and-cut procedures and the preprocessing techniques. It is clear that (BM) outperforms (RM), solving a greater percentage of instances by exploring the same amount of nodes, and that the models with the branch-and-cut and preprocessing explore far less nodes than without these improvements. It is not so straightforward to compare the performance in terms of number of nodes between models (RM) and (BM) with the valid inequalities. However, we can see that for greater number of nodes explored, (RM) slightly outperforms (BM), since the former solves around 3% more instances than the latter.

Finally, the percentage of solved instances with respect to the time (up to a time limit of one hour) by the four models is illustrated in Figure 6. This figure shows results coherent with the previous ones, in the sense that it shows that model (BM) outperforms (RM), but the opposite occurs if we consider the models with the valid inequalities and the preprocessing. It is remarkable how model (RM) solves 44% of the instances in less than 3600 seconds, whereas the same model with the improvements solves twice as many.

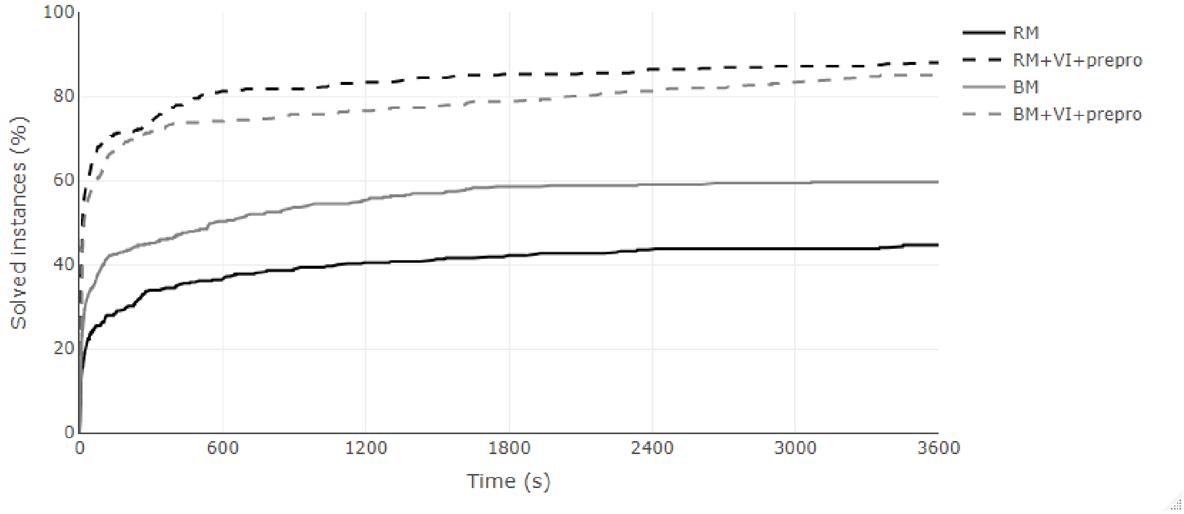

Figure 6: Percentage of instances solved (with a time limit of 3600 seconds) by models (RM) and (BM), with and without the corresponding branch-and-cut procedures and the preprocessing techniques

Overall, it is clear that the branch-and-cut and the preprocessing techniques applied constitute a major improvement in the performance of both (RM) and (BM). Comparing the two formulations with the upgrades, it can be seen that the linear relaxation gap is always smaller for model (BM) than for (RM). However, the cuts added in the root node are very efficient in both cases in reducing the gap, and after adding them the gap is the same for both models. From the number of nodes explored in the branching tree, the average time and the number of instances solved, it is clear that model (RM) slightly outperforms model (BM). The reason is that computing the valid inequalities for model (BM) is harder and time consuming. Indeed, we compute one inequality for each customer for (BM), but we obtain one inequality per customer and product in the case of model (RM). The fact that valid inequalities added to (RM) can be separated by products makes the processes of computing the inequalities and branching a lot more efficient.

Motivated by the results obtained by Bertsimas and Mišić [5] with a Benders decomposition procedure to tackle PLD, we decided to test the performance of our models using some large-scale instances. In [5], they use a real data set with 3584 candidate products and 330 customer rankings, and vary the number of products available in the product line creating instances with a line of up to 2, 3, 4, 5, 10, 20 and 50 products. We generated two instances of RPPT of similar size, that is, with 350 customers, all with different budgets, and 10 products. As explained in Section 3, this is equivalent to having 3500 different *products* (if we consider a product with its candidate price for PLD). And setting $|I| = 10$ also implies that the product line will have up to 10 products. We tested both instances with models (RM) and (RM) including the corresponding branch-and-cut procedures and the preprocessing techniques, and the results are shown in Table 3.

Ins	(RM)+VIs+prepro								
	LR phase		**Cuts phase**		**MIO phase**			**Nodes**	**Total time (s)**
	Best Bound	Time (s)	Best Bound	Time (s)	Obj	Best Bound	Time (s)		
1	217071	0.5	155889	3534.5	148414	148414	8575.4	21113	12109.9
2	216549	0.6	154642	5482.7	143469	143483	72255.2	276914	77737.9
	(BM)+ VIs+prepro								
Ins	**LR phase**		**Cuts phase**		**MIO phase**			**Nodes**	**Total time (s)**
	Best Bound	Time (s)	Best Bound	Time (s)	Obj	Best Bound	Time (s)		
1	172006	1.1	155889	21540.3	148414	148414	2103.7	23279	23644
2	170810	0.9	154642	21805.8	143469	143481	3487.1	120415	25292.9

Table 3: Results of two large-scale instances ($|K| = 350$, $|I| = 10$) given by models (RM) and (BM) including the branch-and-cut method and the preprocessing techniques. The *LR phase* of the table shows the bound and time of the linear relaxation phase. The *Cuts phase* includes the bound after the cuts in the root node and the time to generate them. And the *MIO phase* shows the best solution (Obj), the best bound and the time. We set a final integrality gap of 0.01% or lower for this integer phase. Finally, the table shows the number of nodes explored in the branching tree and the total time in seconds

The results show that the time needed to solve the Cuts phase is much smaller for formulation (RM), with times of around an hour for the first instance and an hour and a half for the second. Model (BM), on the contrary, takes nearly six hours to add the cuts in the Cuts phase. These results are consistent with the ones obtained in the previous experiment.

Nonetheless, we can see a different performance in the MIO phase. Model (BM) takes less than an hour to close the gap and reach optimality for both instances. Regarding instance 1, the MIO phase for model (RM) takes two hours and a half. But for instance 2, this phase takes 72255 seconds, i.e. more than 20 hours. Comparing the number of nodes explored during the MIO phase with the time taken to solve instance 1, we see that both models explore a similar amount of nodes, but model (RM) takes four times longer. We observe a similar pattern for instance 2. Therefore, it is clear that exploring a node is much faster for model (BM) than for (RM), and this is decisive in the reduction of the MIO phase time.

9 Conclusions

In this work, we presented a three-indexed integer formulation for RPPT, a problem which consists in setting the prices of a set of products to maximize the profit of a company, taking into account the customers' choice. We then developed two resolution approaches. The first one started with a smaller formulation (RM) of the problem which in general yields worse upper bounds. To strengthen it, we projected out the customer decision variables of smaller size, obtaining a set of valid inequalities. An ulterior transformation of the linear separation problem into a MCFP was developed to take advantage of its features. The second resolution approach is based on a Benders decomposition. We first reformulated the problem into a master problem and a series of subproblems. Then we derived a set of constraints from the subproblems to make the master problem feasible and a separation procedure to include them dynamically. We also proved that a very small set of them can be included to make the master feasible while the rest of them are still separated, thus linking the Benders reformulation with model (BM). We completed the paper with preprocessing techniques designed to reduce the size of the instances and extensive computational experiments to test the overall performance of both methods.

Computational experiments show that the valid inequalities and the preprocessing techniques highly improve the performance of models (RM) and (BM). In particular, the valid inequalities significantly reduce the upper linear relaxation bound and the preprocessing techniques reduce the size of the instance, making the linear relaxation and the branching phases faster. Together they allow for the resolution of up to 40% more of the instances proposed within the same time limit. When comparing both models, (BM) generally yiels better linear relaxation bounds, but (RM) slightly outperforms (BM) when we consider both with the valid inequalities and the preprocessing techniques due to the amount of time the generation of the valid inequalities takes for model (BM). Regarding the two instances with 350 customers proposed, the performance of the models is consistent with that obtained for the smaller instances. In this case, we can clearly see how model (BM) takes more time when computing the valid inequalities than model (RM) but less time when exploring each node of the branch-and-bound tree, thus reducing the linear relaxation bound faster than (RM). All in all, the theoretical study of a novel three-indexed model with very tight upper bounds results in the development of two different exact resolution approaches including models of a much smaller size that maintain the linear relaxation bounds of the former model through the addition of valid inequalities.

Acknowledgements

The research of Concepción Domínguez has been funded by the Fonds de la Recherche Scientifique - FNRS under a FRIA grant. Concepción Domínguez and Alfredo Marín have been partially supported by the research project MTM2015-65915-R (MINECO, Spain). Martine Labbé has been partially supported by the Fonds de la Recherche Scientifique -FNRS under Grant PDR T0098.18 .

References

[1] Aggarwal, G., Feder, T., Motwani, R. & Zhu, A. (2004). Algorithms for multi-product pricing. In *International Colloquium on Automata, Languages, and Programming*, Springer, Berlin, Heidelberg, 72-83.

[2] Ahuja, R.K., Magnanti, T.L. & Orlin, J.B. (1993). *Network flows: Theory, Algorithms and Applications*, Prentice-Hall, Inc, New Jersey.

[3] Belloni, A., Freund, R., Selove, M. & Simester, D. (2008). Optimizing product line designs: Efficient methods and comparisons. *Management Science*, 54(9), 1544–1552.

[4] Ben-Ameur, W. & Neto, J. (2007). Acceleration of cutting-plane and column generation algorithms: Applications to network design. *Networks: An International Journal*, 49(1), 3-17.

[5] Bertsimas, D. & Mišić, V.V. (2019). Exact first-choice product line optimization. *Operations Research*, 67(3), 651-670.

[6] Bonami, P., Salvagnin, D. & Tramontani, A. Implementing Automatic Benders Decomposition in a Modern MIP Solver. e-print: http://www.optimization-online.org/DB_HTML/2019/12/7506.html

[7] Briest, P., & Krysta, P. (2011). Buying cheap is expensive: Hardness of non-parametric multi-product pricing. *SIAM Journal on Computing*, 40(6), 1554-1586.

[8] Calvete, H.I., Domínguez, C., Galé, C., Labbé, M. & Marín, A. (2019). The Rank Pricing Problem: models and branch-and-cut algorithms. *Computers & Operations Research*, 105, 12-31.

[9] Cánovas, L., García, S., Labbé, M. & Marín, A. (2007). A strengthened formulation for the simple plant location problem with order. *Operations Research Letters*, 35(2), 141-150.

[10] Chen, K.D. & Hausman, W.H. (2000). Mathematical properties of the optimal product line selection problem using choice-based conjoint analysis. *Management Science*, 46(2), 327–332.

[11] Delorme, M., García, S., Gondzio, J., Kalcsics, J., Manlove, D. & Pettersson, W. (2019). Mathematical models for stable matching problems with ties and incomplete lists. *European Journal of Operational Research*, 277(2), 426-441.

[12] Dobson, G. & Kalish, S. (1988). Positioning and pricing a product line. *Marketing Science*, 7(2), 107–125.

[13] Espejo, I., Marín, A. & Rodríguez-Chía, A.M. (2012). Closest assignment constraints in discrete location problems. *European Journal of Operational Research*, 219(1), 49-58.

[14] Fernandes, C.G., Ferreira, C.E., Franco, A.J. & Schouery, R.C. (2016). The envy-free pricing problem, unit-demand markets and connections with the network pricing problem. *Discrete Optimization*, 22, 141-161.

[15] Green, P.E. & Krieger, A.M. (1985). Models and heuristics for product line selection. *Marketing Science*, 4(1), 1–19.

[16] Green, P.E. & Krieger, A.M. (1993). Conjoint analysis with product positioning applications. *Handbooks in Operations Research and Management Science*, 5, 467–515.

[17] Gusfield, D. & Irving, R.W. (1989). *The stable marriage problem: Structure and algorithms.* MIT Press.

[18] Hanjoul, P. & Peeters, D. (1987). A facility location problem with clients' preference orderings. *Regional Science and Urban Economics*, 17(3), 451–473.

[19] Hansen, P., Kochetov, Y. & Mladenović, N. (2004). *Lower bounds for the uncapacitated facility location problem with user preferences.* Les Cahiers du GERAD G-2004-24, HEC Montréal.

[20] Kraus, U.G. & Yano, C.A. (2003). Product line selection and pricing under a share-of-surplus choice model. *European Journal of Operational Research*, 150(3), 653–671.

[21] Kwanashie, A. & Manlove, D.F. (2014). An Integer Programming Approach to the Hospitals/Residents problem with Ties. In *Operations Research Proceedings 2013*, Springer, 263–269.

[22] Manlove, D.F. (2013). *Algorithmics of matching under preferences* (Vol. 2). World Scientific Publishing Co, Singapore.

[23] McBride, R.D. & Zufryden, F.S. (1988). An integer programming approach to the optimal product line selection problem. *Marketing Science*, 7(2), 126–140.

[24] Naoum-Sawaya, J. & Elhedhli, S. (2013). An interior-point Benders based branch-and-cut algorithm for mixed integer programs. *Annals of Operations Research*, 210(1), 33-55.

[25] Nemhauser, G.L. & Wolsey, L.A. (1988). *Integer and Combinatorial Optimization.* John Wiley & Sons.

[26] Rothblum, U.G. (1992). Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54, 57–67 .

[27] Rusmevichientong, P., Van Roy, B. & Glynn, P.W. (2006). A nonparametric approach to multiproduct pricing. *Operations Research*, 54(1), 82-98.

[28] Schön, C. (2010). On the optimal product line selection problem with price discrimination. *Management Science*, 56(5), 896–902.

[29] Schön, C. (2010). On the product line selection problem under attraction choice models of consumer behavior. *European Journal of Operational Research*, 206(1), 260–264.

[30] Vande Vate, J.V. (1989). Linear programming brings marital bliss. *Operations Research Letters*, 8(3), 147–153 .

[31] Vasilyev, I.L. & Klimentova, K.B. (2010). The branch and cut method for the facility location problem with client's preferences. *Journal of Applied and Industrial Mathematics*, 4(3), 441-454.

A Complementary proofs

Proof of Proposition 4.2

1. This is a direct consequence of (1d).
2. If $\sum_{i\in I^k}\sum_{m\in M_i^k} \bar{v}_i^m > 0$, let n^*, m^* be as stated, and $i^* \in \{i \in S_{n^*}^k : \bar{v}_i^{m^*} = 1\}$. Then for all $n \in N^k$, $m \in M_{S_n^k}$, $(n,m) \neq (n^*,m^*)$, it holds

   - If $(n,m) < (n^*,m^*)$ (with the lexicographic order), then by the corresponding constraint from (1d) we obtain $y_n^{km} \leq \sum_{i\in S_n^k} \bar{v}_i^m = 0$.
   - If $(n,m) > (n^*,m^*)$, then we turn to the constraint from (1e) given by $(n,i,m) = (n^*,i^*,m^*)$:

   $$\sum_{\substack{m'\in M_{i^*}^k:\\ m'\leq m^*}} \bar{v}_{i^*}^{m'} + \sum_{\substack{m'\in M_{S_{n^*}^k}:\\ m'>m^*}} y_{n^*}^{km'} + \sum_{n'=n^*+1}^{n^k}\sum_{m'\in M_{S_{n'}^k}} y_{n'}^{km'}$$
   $$= 1 + \sum_{\substack{m'\in M_{S_{n^*}^k}:\\ m'>m^*}} y_{n^*}^{km'} + \sum_{n'=n^*+1}^{n^k}\sum_{m'\in M_{S_{n'}^k}} y_{n'}^{km'} \leq 1.$$

   We distinguish two cases:
     - If $n = n^*$, then $m > m^*$ and y_n^{km} belongs to the sum $\sum_{m'\in M_{S_{n^*}^k}:m'>m^*} y_{n^*}^{km'}$.
     - If $n > n^*$, then y_n^{km} belongs to $\sum_{n'=n^*+1}^{n^k}\sum_{m'\in M_{S_{n'}^k}} y_{n'}^{km'}$.

   Hence, in both cases the constraint implies $y_n^{km} = 0$.

   We just proved that $y_n^{km} = 0\,\forall(n,m) \neq (n^*,m^*)$. Finally, for $y_{m^*}^{kn^*}$, we have that constraints (1b) and (1c) reduce to $y_{m^*}^{kn^*} \leq d$ with $d \geq 1$. As for constraints (1e), $y_{m^*}^{kn^*}$ may belong to the second or third sum of the LHS for a given k. If $y_{m^*}^{kn^*}$ belongs to the second sum, then $m^* > m$ and hence the sum of v-variables $\sum_{m'\in M_i^k:m'\leq m} v_i^{m'}$ is equal to 0. Otherwise, $y_{m^*}^{kn^*}$ belongs to the third sum, so $n < n^*$ and the way n^* is defined once again implies $\sum_{m'\in M_i^k:m'\leq m} v_i^{m'} = 0$ for such $i \in S_n^k$. Therefore, $y_{m^*}^{kn^*}$ is free, and it will take value 1 in the optimal solution because its coefficient in the objective function is positive.

Proof of Proposition 5.1

Consider a feasible fractional solution $(\bar{v},\bar{y})$ of the linear relaxation of (3IM) that yields an objective value $v(\bar{v},\bar{y})$. We build a fractional solution $(\hat{v},\hat{x},\hat{z})$ of (RM) with an objective value $v(\hat{v},\hat{x},\hat{z}) \geq v(\bar{v},\bar{y})$. In this way, if $v(\bar{v},\bar{y})$ is an optimal solution of the linear relaxation of (3IM), we obtain $v(\text{RM}) \geq v(\hat{v},\hat{x},\hat{z}) \geq v(\bar{v},\bar{y}) = v(\text{3IM})$,where $v(\text{RM})$ (resp. $v(\text{3IM})$) is the optimal value of the linear relaxation of (RM) (resp. (3IM)).

We define $\hat{v}_i^m := \bar{v}_i^m$, $\hat{x}_n^k := \sum_{m\in M_{S_n^k}} \bar{y}_n^{km}$, $\hat{z}_n^k := \sum_{m\in M_{S_n^k}} b^m \bar{y}_n^{km}$ $\forall k \in K$, $m \in M_i^k$, $n \in N^k$, $i \in S_n^k$.

First, we prove that this solution is feasible for the linear relaxation of (RM). Constraints (3b) hold because (1b) hold. Fixing $\bar{v}$ and $\hat{v}$, the problems are decomposable by customers, so we assume a fixed customer k in the following, and we prove that the associated constraints from sets (3c)-(3g) hold. As for the corresponding constraint from (3c), using the above we have $\sum_{n\in N^k} \hat{x}_n^k = \sum_{n\in N^k}\sum_{m\in M_{S_n^k}} \bar{y}_n^{km}$, and the last sum is less than or equal to 1 because of (1c). As for the constraint from (3d), it translates to $\sum_{m\in M_{S_n^k}} \bar{y}_n^{km} \leq \sum_{i\in S_n^k}\sum_{m\in M_i^k} \bar{v}_i^m$, which holds because of constraints (1d) (summing up on m). Regarding the constraint from (3e) $\sum_{m\in M_i^k} \hat{v}_i^k + \sum_{n'=n+1}^{n^k} \hat{x}_{n'}^k \leq 1$, it translates to $\sum_{m\in M_i^k} \bar{v}_i^k + \sum_{n'=n+1}^{n^k}\sum_{m\in M_{S_{n'}^k}} \bar{y}_{n'}^{km} \leq 1$, which is exactly the inequality from set (1e) for such k and $m = \sigma(k)$, so it also holds. Constraint $\hat{z}_n^k \leq b^{\sigma(k)}\hat{x}_n^k$ from set (3f) holds trivially using the definition of $\hat{x}$ and $\hat{z}$, since $\hat{z}_n^k = \sum_{m\in M_{S_n^k}} b^m \bar{y}_n^{km} \leq b^{\sigma(k)} \sum_{m\in M_{S_n^k}} \bar{y}_n^{km} = b^{\sigma(k)}\hat{x}_n^k$. And finally let us prove the feasibility of the corresponding constraint from (3g). To begin with, we know that for a given customer k and product $i \in I^k$, (1c) and (1e) imply $\sum_{m'\in M_i^k : m'<m} \bar{v}_i^{m'} + \sum_{m'\in M_{S_n^k}: m'\geq m} \bar{y}_n^{km'} \leq 1$ $\forall$ m such that $m-1 \in M_i^k$. Let us suppose $M_i^k := \{1, 2, \ldots, \sigma(k)\}$. Then, multiplying the previous constraint m such that $m-1 \in M_i^k$ by $b^m - b^{m-1}$ (where $b^0 = 0$) and adding together all the constraints, we obtain:

$$\sum_{m=1}^{\sigma(k)}\sum_{m'=1}^{m-1}\left(b^m - b^{m-1}\right)\bar{v}_i^{m'} + \sum_{m=1}^{\sigma(k)}\sum_{m'=m}^{\sigma(k)}\left(b^m - b^{m-1}\right)\bar{y}_n^{km'} \leq \sum_{m=1}^{\sigma(k)}\left(b^m - b^{m-1}\right) = b^{\sigma(k)}. \tag{16}$$

The LHS of (16) is equal to

$$\begin{aligned}
&\sum_{m=1}^{\sigma(k)}\sum_{m'=1}^{m-1}\left(b^m - b^{m-1}\right)\bar{v}_i^{m'} + \sum_{m=1}^{\sigma(k)}\sum_{m'=m}^{\sigma(k)}\left(b^m - b^{m-1}\right)\bar{y}_n^{km'} \\
&= \sum_{m'=1}^{\sigma(k)-1}\sum_{m=m'}^{\sigma(k)}\left(b^m - b^{m-1}\right)\bar{v}_i^{m'} + \sum_{m'=1}^{\sigma(k)}\sum_{m=1}^{m'-1}\left(b^m - b^{m-1}\right)\bar{y}_n^{km'} \\
&= \sum_{m'=1}^{\sigma(k)-1}\left(b^{\sigma(k)} - b^{m'}\right)\bar{v}_i^{m'} + \sum_{m'=1}^{\sigma(k)} b^{m'}\bar{y}_n^{km'} = \sum_{m'\in M_i^k}\left(b^{\sigma(k)} - b^{m'}\right)\bar{v}_i^{m'} + \hat{z}_n^k.
\end{aligned}$$

All in all, we obtain that constraint $\hat{z}_n^k + \sum_{m'\in M_i^k}\left(b^{\sigma(k)} - b^{m'}\right)\bar{v}_i^{m'} \leq b^{\sigma(k)}$ is satisfied, so the corresponding constraint from (3g) holds. On the other hand, if $M_i^k \subsetneq \{1, 2, \ldots, \sigma(k)\}$ it suffices to multiply each constraint associated to $m \in M_i^k$ by $b^m - b^{m'}$, where $m' = \{\max\{m'' \in \{0\} \cup M_i^k : m'' < m\}\}$ instead, and the same result is obtained applying the previous procedure.

Finally, we need to prove that $v(\hat{v}, \hat{x}, \hat{z}) \geq v(\bar{v}, \bar{y})$. But this is straightforward by definition of $\hat{z}$, since $v(\hat{v}, \hat{x}, \hat{z}) = \sum_{k\in K}\sum_{n\in N^k} \hat{z}_n^k = \sum_{k\in K}\sum_{n\in N^k}\sum_{m\in M_{S_n^k}} b^m \bar{y}_n^{km} = v(\bar{v}, \bar{y})$.

Proof of Proposition 6.1

Constraints (11c) guarantee that if customer k cannot afford any product, then $z^k = 0$. When k can afford several products, the RHS of (11c) is an upper bound on the value of z^k.

Constraints (11d) model the preferences and ensure that k purchases his most preferred product (at the cheapest price in case of ties). Indeed, given an integer feasible solution $(\bar{v}, \bar{z})$, let $n^* := \min\{n \in N^k : \sum_{i \in S_n^k} \sum_{m \in M_i^k} \bar{v}_i^k \geq 1\}$, and $i^* \in \arg\min_{i \in S_{n^*}^k} \{\sum_{m \in M_i^k} b^m \bar{v}_i^m\}$. Clearly, $S_{n^*}^k$ is the first class (according to the ranking) from which k is able to afford a product, whereas i^* is one of the cheapest products from $S_{n^*}^k$. So assuming $\bar{v}_{i^*}^{m^*} = 1$, we need to prove that it holds $z^k = \sum_{m \in M_{i^*}^k} b^m \bar{v}_{i^*}^m = b^{m^*}$. Since we are maximizing the objective, it suffices to prove that all the RHSs of (11c) and (11d) for such k are all greater than or equal to b^{m^*}, and that at least one is equal to b^{m^*}. We have one constraint per product $i \in I^k$, so to begin with we distinguish two cases:

- $i \preceq i^*$. In this case, the last sum of the corresponding constraint from (11d) $\sum_{j \in I^k : j \prec_k i} \sum_{m \in M_j^k} b^m v_j^m = 0$. We have three subcases to consider:
  - $i \prec_k i^*$. Then k cannot afford i or any $j \prec_k i$, so the RHS of (11d) is equal to $b^{\sigma(k)}$, an upper bound on the profit from k.
  - $i = i^*$. In this case, the RHS of (11d) is equal to b^{m^*}:
  $$b^{\sigma(k)} + \sum_{m \in M_{i^*}^k} \left(b^m - b^{\sigma(k)}\right) \bar{v}_{i^*}^m = b^{\sigma(k)} + \left(b^{m^*} - b^{\sigma(k)}\right) = b^{m^*}.$$
  - $i \sim_k i^*$, $i \neq i^*$. In this case, by definition of i^* we know that $\bar{v}_i^{\hat{m}} = 1$ for some $\hat{m} \geq m^*$. If $\hat{m} > \sigma(k)$, then the RHS of (11d) is equal to $b^{\sigma(k)}$. Otherwise, we have $\sum_{m \in M_i^k} b^m \bar{v}_i^m = b^{\hat{m}} \geq b^{m^*}$ and it holds
  $$b^{\sigma(k)} + \sum_{m \in M_i^k} \left(b^m - b^{\sigma(k)}\right) \bar{v}_i^m = b^{\sigma(k)} + \left(b^{\hat{m}} - b^{\sigma(k)}\right) = b^{\hat{m}}.$$
- $i \succ_k i^*$. Then it holds
$$b^{\sigma(k)} + \sum_{m \in M_i^k} \left(b^m - b^{\sigma(k)}\right) \bar{v}_i^m + \sum_{\substack{j \in I^k: \\ j \prec_k i}} \sum_{m \in M_j^k} b^m \bar{v}_j^m \geq \sum_{\substack{j \in I^k: \\ j \prec_k i}} \sum_{m \in M_j^k} b^m \bar{v}_j^m \geq b^{m^*},$$
where the last inequality holds because $v_{i^*}^{m^*} = 1$ belongs to the previous sum.

Proof of Proposition 6.2

We drop the k index from the variables for the sake of notation. Constraints (11c) are obtained, for a fixed customer k, when fixing $\alpha := 0$, $\beta_i^m := 0$ $\forall i \in I^k$, $m \in M_i^k$, $\gamma_n^m := b^m$ $\forall n \in N^k, m \in M_{S_n^k}$. The described $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ belongs to D^k because constraints (14b) are trivially satisfied, since for each $m \in M_{S_n^k}$ it holds $\alpha + \sum_{n'=1}^{n-1} \sum_{i \in S_{n'}^k} \sum_{m' \in M_i^k} \beta_i^{m'} + \sum_{i \in S_n^k} \sum_{\substack{m' \in M_i^k: \\ m' < m}} \beta_i^{m'} + \gamma_n^m \geq \gamma_n^m = b^m$.

As for constraints (11d), consider for fixed $k \in K$, $n^* \in N^k$ and $i^* \in S^k_{n^*} \subset I^k$, and assume $M^k_{i^*} = \{1, \ldots, \sigma(k)\}$. Let us set the values $\alpha := b^1$, $\beta^m_{i^*} := b^{m+1} - b^m$ for $m \in M^k_{i^*}$: $m < \sigma(k)$, $\beta^{\sigma(k)}_{i^*} := 0$, $\beta^m_i := 0 \ \forall i \neq i^*$, $m \in M^k_i$, $\gamma^m_n := b^m \ \forall n < n^*$, $m \in M_{S^k_n}$, $\gamma^m_n := 0$ for $n \geq n^*$, $m \in M_{S^k_n}$. Then it follows $\sum_{m \in M^k_{i^*}} \beta^m_{i^*} = \sum_{m=1}^{\sigma(k)-1} \left(b^{m+1} - b^m\right) = b^{\sigma(k)} - b^1$ and $\sum_{m \in M^k_{i^*}: m \geq m'} \beta^m_{i^*} = b^{\sigma(k)} - b^{m'}$ for $m' \in M^k_{i^*}$. Therefore, we have that the RHS of the corresponding constraint from (15c) is

$$\alpha + \sum_{n \in N^k} \sum_{i \in S^k_n} \sum_{m \in M^k_i} \left(1 - \sum_{\substack{m' \in M^k_i: \\ m' \leq m}} v^{m'}_i \right) \beta^m_i + \sum_{n \in N^k} \sum_{i \in S^k_n} \sum_{m \in M^k_i} v^m_i \gamma^m_n$$

$$= b^1 + \sum_{m \in M^k_{i^*}} \left(1 - \sum_{\substack{m' \in M^k_{i^*}: \\ m' \leq m}} v^{m'}_{i^*} \right) \beta^m_{i^*} + \sum_{n < n^*} \sum_{i \in S^k_n} \sum_{m \in M^k_i} v^m_i b^m$$

$$= b^1 + \sum_{m \in M^k_{i^*}} \beta^m_{i^*} - \sum_{m' \in M^k_{i^*}} \left(\sum_{\substack{m \in M^k_{i^*}: \\ m \geq m'}} \beta^m_{i^*} \right) v^{m'}_{i^*} + \sum_{\substack{i \in I^k: \\ i \prec i^*}} \sum_{m \in M^k_i} v^m_i b^m$$

$$= b^1 + \left(b^{\sigma(k)} - b^1\right) - \sum_{m' \in M^k_{i^*}} \left(b^{\sigma(k)} - b^{m'}\right) v^{m'}_{i^*} + \sum_{\substack{i \in I^k: \\ i \prec i^*}} \sum_{m \in M^k_i} v^m_i b^m,$$

which is equal to the RHS of (11d) for customer k and product $i^* \in S^k_{n^*}$.

To check whether $(\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})$ belongs to D^k, and knowing that the vectors are nonnegative by definition, it is left to prove that (14b) hold $\forall n \in N^k$, $m \in M_{S^k_n}$, $i \in S^k_n$. To do so, we study three cases depending on $n \in N^k$:

- $n < n^*$. Then for given $m \in M_{S^k_n}$, we have the LHS of (14b) equal to

$$\alpha + \sum_{n'=1}^{n-1} \sum_{i \in S^k_{n'}} \sum_{m' \in M^k_i} \beta^{m'}_i + \sum_{i \in S^k_n} \sum_{\substack{m' \in M^k_i: \\ m' < m}} \beta^{m'}_i + \gamma^m_n \geq \gamma^m_n = b^m.$$

- $n = n^*$. Then it holds for $m \in M_{S^k_n}$:

$$\alpha + \sum_{n'=1}^{n^*-1} \sum_{i \in S^k_{n'}} \sum_{m' \in M^k_i} \beta^{m'}_i + \sum_{i \in S^k_{n^*}} \sum_{\substack{m' \in M^k_i: \\ m' < m}} \beta^{m'}_i + \gamma^m_{n^*} = \alpha + \sum_{\substack{m' \in M^k_{i^*}: \\ m' < m}} \beta^{m'}_{i^*} = b^1 + \left(b^m - b^1\right) = b^m.$$

- $n > n^*$. Then for given $m \in M_{S_n^k}$, we have the LHS of (14b) equal to

$$\alpha + \sum_{n'=1}^{n^*-1} \sum_{i \in S_{n'}^k} \sum_{m' \in M_i^k} \beta_i^{m'} + \sum_{i \in S_{n^*}^k} \sum_{\substack{m' \in M_i^k: \\ m' < m}} \beta_i^{m'} + \gamma_{n^*}^m = \alpha + \sum_{m' \in M_{i^*}^k} \beta_{i^*}^{m'} = b^{\sigma(k)}.$$

In the three cases, the LHS of (14b) is greater than or equal to b^m, so the given $(\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})$ satisfies (14b) and thus it belongs to D^k.

If $M_{i^*}^k \subsetneq \{1, 2, \ldots, \sigma(k)\}$, the proof follows analogously applying the previous procedure to the same α and $\boldsymbol{\gamma}$, but defining $\beta_{i^*}^m := b^{m'} - b^m$, where $m' = \min\{m'' \in M_{i^*}^k : m'' > m\}$, for $m \in M_i^k : m < \sigma(k)$, $\beta_{i^*}^{\sigma(k)} := 0$, $\beta_i^m := 0\ \forall i \neq i^*$, $m \in M_i^k$.

Proof of Proposition 7.1

We shall prove the statement for model (RM), since the proof for model (3IM) is analogous. Thus, suppose we have an optimal solution $(\hat{v}_i^m, \hat{x}_n^k, \hat{z}_n^k)$ not satisfying the statement conditions. Our aim is to build another one which does satisfy them. We will proceed by induction on k.

To begin with, it is clear that the statement holds for all customers k with budget $b^{|M|}$. Indeed, since these customers can afford any product, they always get one of their favorite ones, so one in the set S_1^k, and $\hat{x}_n^k = 0$ for $n > 1 = u'(k)$. Now, let $k_0 \in K$ be such that the statement holds $\forall k \in K$ with $\sigma(k) > \sigma(k_0)$ but $\hat{x}_n^{k_0} = 1$ for some $n > u'(k_0)$. Then it is clear that $k_0 \notin C_3$. Besides, from the definition of u' we know there is a product $i_0 \in S_{u'(k_0)}^{k_0} \setminus \cup_{\substack{k \in K: \\ \sigma(k) > \sigma(k_0)}} S_{u'(k)}^k$, and we also know that i_0 remains unsold in this solution.

Hence, consider the vector of prices $\bar{v}_i^m$ obtained by modifying the price of i_0: $\bar{v}_i^m = \hat{v}_i^m$ $\forall i \neq i_0, m \in M_i^k$, $\bar{v}_{i_0}^{\sigma(k_0)} = 1$, $\bar{v}_{i_0}^m = 0\ \forall m \neq \sigma(k_0)$. Given this vector of prices, customers k with $\sigma(k) < \sigma(k_0)$ can afford the same products than in solution $(\hat{v}_i^m, \hat{x}_n^k, \hat{z}_n^k)$, so they make the same purchase. Customers k with $\sigma(k) > \sigma(k_0)$ were already purchasing in the previous solution a product that they liked better than i_0. And customers k with $\sigma(k) = \sigma(k_0)$ might purchase product i_0 in the new solution, but in this case, since they pay their whole budget, the objective value does not decrease with respect to the previous solution. Therefore, $(\hat{v}_i^m, \hat{x}_n^k, \hat{z}_n^k)$ is an optimal solution that meets the statement requirements for customer k_0. Applying the procedure iteratively, we can obtain an optimal solution satisfying the statement.

Proof of Corollary 7.2

We will prove the statement for formulation (RM), and the proof for (3IM) is analogous. Let us define a solution $(\bar{v}_i^m, \bar{x}_n^k, \bar{z}_n^k)$ of (RM) and prove its optimality.

We begin by defining the vector $\bar{v}$ of prices in the following way: $\forall i \in I : i \in S_{u'(k)}^k$ for some $k \in K$, then $\bar{v}_i^{m_0} = 1$ for $m_0 := \max\{m \in M_i : \exists k \in K \text{ with } \sigma(k) = m_0, i \in S_{u'(k)}^k\}$, $\bar{v}_i^m = 0$ $\forall m \neq m_0$; and $\forall i \in I$ such that $\{k \in K : i \in S_{u'(k)}^k\} = \emptyset$, then $\bar{v}_i^{m_0} = 1$ for $m_0 := \max\{m \in M_i\}$, $\bar{v}_i^m = 0\ \forall m \neq m_0$.

Now let us see the customers' purchasing decision based on vector $\bar{v}$. Thus, given $k \in K$

we have that $\forall i \in S_n^k$ with $n < u'(k)$, it holds by definition of u' that $i \in S_{u'(k')}^k$ for some $k' : \sigma(k') > \sigma(k)$, and therefore $\bar{v}_i^m = 1$ for some $m > \sigma(k)$ and thus k cannot afford i. Hence, we have $\bar{x}_n^k = 0 \; \forall n < u'(k)$. Moreover, since $k \notin C_3$, there exists $i_0 \in S_{u'(k)}^k$ such that $\bar{v}_{i_0}^{\sigma(k)} = 1$. This combined with the fact that $\forall i \in S_{u'(k)}^k$ it holds $\bar{v}_i^m = 1$ for some $m \geq \sigma(k)$ by definition of $\bar{v}$, implies that customer k purchases i_0, so $\bar{x}_{u'(k)}^k = 1$ and $\bar{z}_{u'(k)}^k = b^{\sigma(k)}$.

Given that the objective value of this above derived feasible solution is $\sum_{k \in K} b^{\sigma(k)}$, which is an upper bound on the profit the company can obtain, solution $(\bar{v}_i^m, \bar{x}_n^k, \bar{z}_n^k)$ is optimal.

B Tables with the computational results

$\|I\|$	$\|I^k\|$	Ties	(RM) LRGap	(RM) Nodes	(RM) t(s)	(RM) Sol.	(RM)+VIs+prepro Cuts	(RM)+VIs+prepro RGap	(RM)+VIs+prepro Nodes	(RM)+VIs+prepro t(s)	(RM)+VIs+prepro Sol.	(BM) LRGap	(BM) Nodes	(BM) t(s)	(BM) Sol.	(BM)+ VIs+prepro Cuts	(BM)+ VIs+prepro RGap	(BM)+ VIs+prepro Nodes	(BM)+ VIs+prepro t(s)	(BM)+ VIs+prepro Sol.
5	2	1	14.3	575	**1**	5	226	8.2	73	3	5	14	**57**	**1**	5	74	8.2	66	**1**	5
5	3	1	48.3	722	**3**	5	346	3.9	**29**	5	5	11	640	1	5	92	3.9	33	5	5
5	5	1	36.3	3090	5	5	549	2.3	19	**10**	5	11	13115	**4**	5	119	2.4	94	24	5
5	5	2	43.1	4430	**6**	5	583	6.1	**212**	11	5	16	10558	7	5	134	6.1	643	24	5
5	5	3	49.2	3170	**4**	5	498	10.1	**1131**	10	5	20	10246	6	5	155	10.1	1200	15	5
25	5	1	12.3	156	2	5	168	0.2	**1**	**1**	5	3	102	2	5	51	0.2	**1**	**1**	5
25	5	2	16.8	879	5	5	232	0.4	6	1	5	3	169	3	5	82	0.4	**2**	**2**	5
25	5	3	23.1	3758	10	5	335	1.7	74	4	5	5	1305	5	5	122	1.7	**73**	**2**	5
25	13	1	7.1	46268	119	5	502	0.3	13	**5**	5	3	18367	45	5	93	0.3	**2**	9	5
25	13	3	8.9	127533	316	5	591	0.3	**9**	**5**	5	4	33133	89	5	128	0.3	13	14	5
25	13	5	10.8	370924	977	5	653	0.7	43	**10**	5	5	83401	138	5	160	0.7	**38**	12	5
25	25	3	7.0	477374	2999	2	1278	0.7	**60**	**24**	5	4	215833	1914	4	146	0.7	386	122	5
25	25	5	6.9	364368	2366	3	1326	0.7	**28**	**32**	5	4	446999	1681	4	163	0.7	34	72	5
25	25	10	9.8	725923	3600	0	1397	1.2	**445**	**43**	5	6	717883	2661	2	194	1.2	7501	88	5
50	10	1	0.5	22	3	5	28	0.0	1	0	5	0	1	0	5	11	0.0	1	0	5
50	10	3	1.8	15	5	5	77	0.1	1	0	5	0	1	0	5	20	0.1	1	0	5
50	10	5	4.3	3569	13	5	169	0.2	2	1	5	2	114	2	5	54	0.2	1	1	5
50	25	3	0.4	1370	17	5	45	0.0	1	0	5	0	1	2	5	12	0.0	1	0	5
50	25	5	0.4	502	16	5	61	0.0	1	0	5	0	1	3	5	16	0.0	1	1	5
50	25	10	1.5	82853	144	5	201	0.1	7	3	5	1	102	7	5	51	0.1	1	3	5
50	50	3	0.0	161813	758	4	12	0.0	1	0	5	0	1	14	5	2	0.0	1	0	5
50	50	5	0.1	245928	751	4	23	0.0	1	0	5	0	1	14	5	4	0.0	1	0	5
50	50	10	0.4	955965	2891	1	180	0.0	2	2	5	0	15	48	5	35	0.0	1	2	5

Table 4: Comparison of models (RM) and (BM) with models (RM) and (BM) including the branch-and-cut method and the preprocessing techniques described in Section 7 to all the instances (5 instances averaged per line). All instances have $|K| = 50$ customers, and the table shows the number of products ($|I|$), the number of products in which every customer is interested ($|I^k|$) and the number of ties (Ties). Depending on the model, it also includes the integrality gap of the linear relaxation (LRGap), the average number of valid inequalities added in total (Cuts), the integrality gap of the linear relaxation after the cuts in the root node (RGap), the number of nodes of the branching tree (Nodes), and the average time needed to optimally solve the instances (t(s)), and the number of instances solved to optimality in less than the time limit of 3600 seconds. Notice that, although the integrality gap (LRGap) only appears in model (RM) (resp. (BM)), it is the same for model (RM) (resp. (BM)) with the valid inequalities and preprocessing.

$\lvert I\rvert$	$\lvert I^k\rvert$	Ties	**(RM)**				**(RM)+VIs+prepro**					**(BM)**				**(BM)+ VIs+prepro**				
			LRGap	Nodes	t(s)	Sol.	Cuts	RGap	Nodes	t(s)	Sol.	LRGap	Nodes	t(s)	Sol.	Cuts	RGap	Nodes	t(s)	Sol.
10	2	1	7.8	3699	3	5	364	3.3	153	6	5	7.8	156	**2**	5	131	3.3	**145**	4	5
10	5	1	37.9	159008	569	5	1167	2.8	**171**	51	5	9.9	141037	147	5	270	2.8	2024	**115**	5
10	5	2	42.5	223376	929	5	1154	4.4	**1223**	71	5	12.6	176716	232	5	310	4.4	8046	**228**	5
10	5	3	48.4	171011	475	5	1269	7.2	**8582**	138	5	16.9	206363	193	5	426	7.3	24671	**109**	5
10	10	1	28.4	618809	3600	0	3002	3.1	**596**	**309**	5	12.6	1180601	3600	0	354	3.1	201456	2179	4
10	10	3	31.3	262796	3600	0	2742	4.0	**1754**	**372**	5	13.9	1252562	3600	0	424	4.0	198660	2385	5
10	10	5	36.6	323288	3600	0	2165	7.0	**55411**	**1014**	5	17.5	1082764	3415	1	407	7.0	550294	2705	3
50	10	1	7.9	565948	3130	2	900	0.1	21	**14**	5	2.9	428422	1823	4	236	0.1	**1**	**14**	5
50	10	3	10.1	769607	3600	0	1003	0.4	**44**	**17**	5	3.7	578994	2530	2	284	0.4	389	20	5
50	10	5	14.5	563317	3600	0	1297	1.1	**574**	41	5	5.4	494869	3153	1	423	1.1	1807	**39**	5
50	25	3	6.1	195457	3600	0	2667	0.6	**437**	**179**	5	3.6	103666	3600	0	406	0.6	3332	269	5
50	25	5	6.9	198695	3600	0	2911	0.9	**878**	**300**	5	4.0	169278	3600	0	460	0.9	10074	485	5
50	25	10	9.4	193551	3600	0	3034	1.4	**6028**	1310	4	5.4	314906	3600	0	570	1.4	102590	**1295**	4
50	50	3	4.6	63323	3600	0	6337	0.6	**500**	**828**	5	3.3	19146	3600	0	475	0.6	6938	2276	5
50	50	5	4.9	69643	3600	0	6192	0.7	**2028**	**1642**	4	3.6	26615	3600	0	471	0.7	6523	2762	3
50	50	10	5.5	55020	3600	0	6181	0.9	**2564**	**2158**	3	3.9	58328	3600	0	508	0.9	8344	3150	2
100	20	1	0.2	206	31	5	92	0.0	1	1	5	0.1	1	4	5	23	0.0	1	1	5
100	20	3	0.3	17506	198	5	166	0.0	1	2	5	0.1	1	3	5	48	0.0	1	3	5
100	20	5	0.8	203085	1712	4	293	0.0	1	2	5	0.4	480	18	5	88	0.0	1	6	5
100	50	3	0.0	152676	1573	3	131	0.0	1	2	5	0.0	1	85	5	30	0.0	1	4	5
100	50	5	0.1	217922	2263	2	214	0.0	2	3	5	0.1	3	65	5	56	0.0	1	7	5
100	50	10	0.4	345287	3600	0	447	0.0	5	8	5	0.3	347	136	5	94	0.0	1	21	5
100	100	3	0.0	19712	1764	3	114	0.0	1	2	5	0.0	1	741	5	24	0.0	1	5	5
100	100	5	0.0	49335	3063	1	144	0.0	1	2	5	0.0	1	1031	5	28	0.0	1	5	5
100	100	10	0.1	33501	3178	1	406	0.0	1	9	5	0.1	318	1188	5	72	0.0	1	43	5

Table 5: Comparison of models (RM) and (BM) with models (RM) and (BM) including the branch-and-cut method and the preprocessing techniques described in Section 7 to all the instances (5 instances averaged per line). All instances have $|K| = 100$ customers, and the table shows the number of products ($|I|$), the number of products in which every customer is interested ($|I^k|$) and the number of ties (Ties). Depending on the model, it also includes the integrality gap of the linear relaxation (LRGap), the average number of valid inequalities added in total (Cuts), the integrality gap of the linear relaxation after the cuts in the root node (RGap), the number of nodes of the branching tree (Nodes), and the average time needed to optimally solve the instances (t(s)), and the number of instances solved to optimality in less than the time limit of 3600 seconds. Notice that, although the integrality gap (LRGap) only appears in model (RM) (resp. (BM)), it is the same for model (RM) (resp. (BM)) with the valid inequalities and preprocessing.

$\|I\|$	$\|I^k\|$	Ties	**(RM)**				**(RM)+VIs+prepro**					**(BM)**				**(BM)+ VIs+prepro**				
			LRGap	Nodes	t(s)	Sol.	Cuts	RGap	Nodes	t(s)	Sol.	LRGap	Nodes	t(s)	Sol.	Cuts	RGap	Nodes	t(s)	Sol.
15	3	1	46.2	558241	1609	3	1029	2.3	**771**	**37**	5	8.5	46215	72	5	335	2.3	3020	54	5
15	8	1	29.5	449124	3600	0	3415	3.1	**4607**	**770**	5	11.5	900227	3600	0	583	3.1	206063	1647	4
15	8	2	31.5	203008	3600	0	3275	3.8	**15601**	**1983**	5	12.6	738058	3600	0	643	3.8	202375	2321	3
15	8	3	34.3	238364	3600	0	3098	5.0	**22502**	**2570**	2	14.4	615985	3600	0	725	5.0	197529	2949	2
15	15	1	24.1	155967	3600	0	7822	3.9	**3952**	3600	1	13.2	235058	3600	0	414	4.9	31246	3600	0
15	15	3	26.0	51748	3600	0	7258	4.8	**7594**	**3403**	1	14.5	170589	3600	0	484	5.3	38046	3600	0
15	15	5	27.6	42113	3600	0	6563	5.2	**8586**	3600	0	15.3	193678	3600	0	550	5.4	30173	3600	0
75	15	1	6.2	194129	3600	0	2004	0.4	**174**	**67**	5	3.0	159600	3600	0	395	0.4	1098	206	5
75	15	3	7.4	264952	3600	0	2236	0.4	**250**	**87**	5	3.4	158060	3600	0	474	0.4	1473	113	5
75	15	5	9.1	262027	3600	0	2470	0.7	**599**	**184**	5	4.4	160915	3600	0	578	0.7	5956	185	5
75	38	3	4.7	70717	3600	0	5992	0.7	**2409**	**2243**	3	3.1	48331	3600	0	638	0.7	43386	2940	3
75	38	5	5.3	70146	3600	0	6522	0.9	**3516**	**2841**	2	3.5	52101	3600	0	697	0.9	26709	3078	2
75	38	10	6.0	78166	3600	0	6526	0.9	**2948**	**2811**	2	3.9	55390	3600	0	776	0.9	18824	3519	1
75	75	3	4.7	17219	3600	0	14382	1.6	474	3600	0	3.8	1113	4309	0	276	2.4	5	3600	0
75	75	5	4.1	13893	3600	0	13787	1.0	506	3600	0	3.2	1351	4274	0	301	1.8	34	3600	0
75	75	10	5.0	15908	3600	0	14346	1.5	504	3600	0	4.0	1352	4289	0	294	2.4	4	3600	0
150	30	3	0.1	9414	172	5	165	0.0	1	4	5	0.1	1	25	5	45	0.0	1	4	5
150	30	5	0.2	90355	1248	4	249	0.0	1	5	5	0.1	6	32	5	60	0.0	1	7	5
150	30	10	0.7	260290	3600	0	638	0.0	4	12	5	0.4	10209	343	5	167	0.0	2	20	5
150	75	3	0.0	63340	2480	2	167	0.0	1	5	5	0.0	0	495	5	37	0.0	1	7	5
150	75	5	0.0	89260	3136	1	190	0.0	1	6	5	0.0	2	536	5	47	0.0	1	7	5
150	75	10	0.1	100169	3600	0	410	0.0	1	11	5	0.1	163	859	5	86	0.0	1	17	5
150	150	3	0.0	2438	2997	3	113	0.0	1	5	5	0.0	0	3600	0	24	0.0	1	6	5
150	150	5	0.0	8389	3600	0	164	0.0	1	8	5	0.0	0	3600	0	28	0.0	0	8	5
150	150	10	0.0	7753	3600	0	401	0.0	1	18	5	0.0	0	3600	0	71	0.0	0	29	5

Table 6: Comparison of models (RM) and (BM) with models (RM) and (BM) including the branch-and-cut method and the preprocessing techniques described in Section 7 to all the instances (5 instances averaged per line). All instances have $|K| = 150$ customers, and the table shows the number of products ($|I|$), the number of products in which every customer is interested ($|I^k|$) and the number of ties (Ties). Depending on the model, it also includes the integrality gap of the linear relaxation (LRGap), the average number of valid inequalities added in total (Cuts), the integrality gap of the linear relaxation after the cuts in the root node (RGap), the number of nodes of the branching tree (Nodes), and the average time needed to optimally solve the instances (t(s)), and the number of instances solved to optimality in less than the time limit of 3600 seconds. Notice that, although the integrality gap (LRGap) only appears in model (RM) (resp. (BM)), it is the same for model (RM) (resp. (BM)) with the valid inequalities and preprocessing.