Int. J. Production Economics 145 (2013) 584–598

Contents lists available at ScienceDirect

Int. J. Production Economics

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

A simulation-based approach to price optimisation of the mixed bundling problem with capacity constraints

Stefan Mayer *, Robert Klein, Stephanie Seiermann

Chair for Analytics and Optimization, University of Augsburg, Universitätsstraße 16, D-86159 Augsburg, Germany

ARTICLE INFO

Article history:
Received 16 April 2012
Accepted 6 May 2013
Available online 23 May 2013

Keywords:
Pricing
Bundling
Mixed-integer programming
Metaheuristics
VNS

ABSTRACT

In this paper, we analyse a service provider's mixed bundling problem for services such as sporting events or holiday packages. Pursuing the objective of maximising total revenue, the service provider has to determine static prices for each single product at the beginning of the selling period. Additionally, an optimal package price has to be chosen for the bundle that comprises one unit of each single product. Because of capacity constraints, the availability of products can change over time such that consumers are forced to switch from their preferred subset of products to an alternative following dynamic substitution.

We propose two mixed-integer linear programmes based on reservation prices that appropriately model the consumer choice process to address the bundling problem. It becomes evident that the determination of the optimal prices is computationally expensive even for small problem classes. Therefore, we develop metaheuristics using variable neighbourhoods. To evaluate their performance, we propose the following new approach: our extensive computational study is performed using especially generated scenarios for which the optimal product prices are known. For this purpose, we present a set of conditions for the generation of reservation prices that guarantee the optimality of the predefined prices. Based on our computational results, managerial insights are derived.

1. Introduction

Over the past years, the practice of selling two or more goods or services in a package at a normally discounted price has proven to be an efficient method to achieve a company's main business objectives. The application of price bundling is used as a strategic marketing concept in many different industry sectors, such as the travel industry, catering, trading of durable goods, information technologies, and financial services. Bundling is primarily motivated by the opportunity to increase revenues. It can be used as an instrument for price discrimination to extract consumer surplus or attract new customers (cf., e.g., Nalebuff and Majerus, 2003). In addition, bundling can help preserve or increase market power or help establish market entry barriers for potential competitors (cf., e.g., Matutes and Regibeau, 1988, 1992).

The term *price bundling* typically refers to the strategies of selling products either additionally (i.e., *mixed bundling*) or solely (i.e., *pure bundling*) in a package, rather than selling the products individually, which is referred to as *unbundling* or *pure components*. In contrast, *product bundling* denotes the integration of several products into one new product that usually creates additional value for a consumer. As we only focus on the first strategy in this paper, the terms price bundling and bundling are used synonymously. For a detailed definition of terminology, see Stremersch and Tellis (2002).

Over the past few decades, many research teams studying the field of *product design and pricing* have addressed the problem of identifying an optimal bundling strategy and have analysed the necessary assumptions and requirements for its successful implementation. Most of this research is based on the early works of Stigler (1963) and Adams and Yellen (1976), taking a monopolist's point of view. Stigler (1963) demonstrates that selling two products as a bundle can extract additional consumer surplus if the consumers' preferences are negatively correlated. Adams and Yellen (1976) also consider the (symmetry of the) distribution of reservation prices to be essential for the profitability of bundling but account for the structure of variable costs as well. Over the ensuing years, the problems of designing and pricing bundles have been studied with both continuously more realistic and more general assumptions. Among others, the most prominent publications include Schmalensee (1984), McAfee et al. (1989), Salinger (1995), and Olderog and Skiera (2000).

In a second research stream, authors have investigated the market forms of a duopoly and oligopoly. Applying game theoretical concepts, researchers often pose the question of whether

* Corresponding author. Tel.: +49 821 598 4347; fax: +49 821 598 4226.
E-mail addresses: stefan.mayer@wiwi.uni-augsburg.de (S. Mayer), robert.klein@wiwi.uni-augsburg.de (R. Klein), stephanie.seiermann@googlemail.com (S. Seiermann).

0925-5273/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ijpe.2013.05.014

a dominant bundling strategy exists and which circumstances would yield market equilibrium. See, for instance, Anderson and Leruth (1993), Chen (1997), Kopalle et al. (1999), and Liao and Tauman (2002).

Publications that are more closely related to our research examine price bundling from an operations research perspective. Here, the focus is primarily on the optimal *bundle design*, i.e., the composition of single products in a package, and the optimal *bundle pricing*, i.e., the determination of revenue-maximising product prices. Hanson and Martin (1990) are the first to formulate a mixed-integer linear programme (MILP) to address these two problems. Assuming consumer segments with heterogeneous valuations, they present an algorithm to solve large problem classes. In general, the solution space of the bundle design and the pricing problem increases exponentially with the number of single products that are offered by the seller. One option to counteract this problem is *customised bundling* (cf., e.g., Hitt and Chen, 2005; Wu et al., 2008). Here, the consumers may decide on the specific composition of the bundle that they purchase. Thus, the seller must only determine the size of the bundle, i.e., how many single products compose each bundle and the corresponding price. Thereafter, the cardinality of the set of solutions only increases linearly with the number of single products. However, an important assumption for the implementation of customised bundling is that marginal costs are low, which is especially true for information goods. Bakos and Brynjolfsson (1999) are the first to focus on this application.

Regardless of the research focus, for the most part, all publications assume utility-maximising consumers who either self-select a subset of the single products or one of the bundles that are offered for sale. In most cases, consumer choice behaviour is represented by the max-surplus choice rule, which is based on reservation prices. The decision rule ensures that each consumer purchases the subset of products that yields the highest nonnegative surplus. The consumers' surplus is defined as the difference between the reservation price for an alternative (i.e., the willingness-to-pay) and the alternative's price. Some authors include this deterministic decision rule in the context of a mathematical programming formulation (e.g., Wu et al., 2008), whereas others apply an analytical approach (e.g., Venkatesh and Mahajan, 1993; Venkatesh and Kamakura, 2003). Another common concept to model consumer choices is the use of discrete choice models (cf., e.g., Bitran and Ferrer, 2007). Here, for each alternative, choice probabilities are derived instead of directly modelling each individual's choice outcome. Mayer and Gönsch (2012) illustrate that these two concepts are equivalent to each other under certain assumptions.

Although many different aspects including competition, varying cost structures, and the correlation of reservation prices, have been considered recently in the bundling literature, surprisingly, the restricted capacity of supply has not been considered yet. Two exceptions are the papers by Bulut et al. (2009) and Gürler et al. (2009). Assuming a similar setting, the researchers analytically examine the optimality of bundling strategies and various influencing factors for a two product case with the assumption that capacity is limited. In many real-life situations, such capacity constraints are a critical success factor, which is especially true for the service industry, where products cannot be produced for stock. When determining revenue-maximising prices, one consideration is that the availability of the products cannot be adapted on short notice to meet the uncertain demand later on. For example, for services such as sporting or cultural events, the service provider has to define static prices at the beginning of a selling period for every seat category of the stadium or theatre so as to accommodate for a limited number of spectators. Here, the bundle corresponds to a seasonal ticket that allows a consumer to attend multiple events within the current season. A similar setting can be observed in the tourism sector; for example, a cruise line company has to determine static brochure prices for a limited number of cabins for a specific tour. The purchase of several consecutive nights in one cabin type can be viewed as a bundle. The same is true for online newspapers or broadcasting companies offering different types of advertisement, such as wallpapers, banners, or skyscrapers on web pages or commercials on TV, where space and time, respectively, are limited.

Therefore, in this paper, we analyse the bundle pricing problem of a service provider who faces restricted capacity of his services. We assume that the provider (seller) is a monopolist, who offers his services both individually as single products and in a package as a bundle (mixed bundling). The service provider's task is to select optimal prices for all products out of a set of discrete price points that are predefined as a result of marketing considerations. As the selling process takes place over time, the availability of services is successively reduced with each purchase. Moreover, as the bundle comprises one unit of each single product, it is only available as long as no single product is sold out. Each consumer has individual and product-specific reservation prices that reflect different valuations for the services in monetary terms. Evaluating only available services for which the corresponding reservation price exceeds the price chosen by the service provider, the consumers decide on one of the following three options depending on which option yields the highest surplus: they either purchase the bundle, the subset of products that provide a nonnegative surplus, or they leave without buying anything. Hence, the consumer choice behaviour does not solely depend on individual preferences, but is also influenced by a dynamically changing choice set. The resulting effects of this influence are known as spill and recapture or dynamic substitution (Burkart et al., 2012). To appropriately cope with these effects, the selling process has to be modelled in a disaggregated fashion, which explicitly simulates consumer choice. To gain basic insight into the computational tractability of the problem, we assume a deterministic case in which prices have to be optimised for a given stream of consumers whose individual reservation prices are known.

The rest of this paper is organised as follows: in Section 2, we describe the problem and our assumptions in more detail. Then, we present two novel alternative MILP formulations in Section 3. As an exact solution of the problem is computationally intractable even for small problem classes, we provide heuristic solution procedures based on variable neighbourhoods in Section 4. To test the performance of these metaheuristics, we propose a new methodology in Section 5, which allows for generating random scenarios in such a way that the optimal solutions are known. In Section 6, the computational results of our experimental study based on these scenarios are presented. Finally, Section 7 provides a summary and our conclusions.

2. Problem description and demand modelling

In this section, we describe the bundle pricing problem in the mixed bundling setting with constrained capacity in more detail. At the beginning of a selling period, the service provider has to determine static product-specific prices p_j for J single products $(j=1,\ldots,J)$ and for the bundle $(j=0)$, which comprises one unit of each single product. The service provider's aim is to select prices in such a way that the total revenue is maximised by selling the products to I consumers $(i=1,\ldots,I)$, who consecutively arrive during the period. Note that in the remainder of this paper, the terms 'product' and 'service' are used synonymously and that these terms refer to the service provider's complete initial offer set, i.e., to all of the single products and to the bundle $(j=0,\ldots,J)$. Furthermore, the consumer is referred to as *she* with the sole purpose to distinguish him or her from the seller, who is correspondingly referred to as *he*.

In many industry sectors it can be observed that due to marketing considerations or psychological aspects, prices are either kept below a certain threshold (e.g., \$9.99) or they are consciously chosen to be integer (e.g., \$75). Hence, the service provider has to choose a price p_j for each product j out of a set of discrete price points p_{jn} $(n=1,\dots,N_j)$, i.e., $p_j \in \{p_{j1},\dots,p_{jN_j}\}$. Treating the price as a discrete variable allows for using neighbourhood-based solution algorithms (cf., Section 4), and, as a consequence, these algorithms identify non-continuous solutions that do not require any rounding during implementation. As the services are assumed to be standard, the bundle price is not superadditive: $p_0 \le \sum_{j=1}^{J} p_j$. In general, a discount is granted for buying the complete set of single products.

Each single product $j=1,\dots,J$ has an initial capacity of c_{0j} that is reduced each time a unit or the bundle is sold to a consumer i. The initial capacity of the bundle c_{00} equates to the minimum initial capacity of all of the single products: $c_{00}=\min_{j=1,\dots,J}\{c_{0j}\}$. The remaining capacity after handling consumer i is denoted by c_{ij}. Hence, we say that product j is *available* for consumer i, if $c_{i-1,j}>0$.

The I consumers $(i=1,\dots,I)$ appear in the sequence of their numbering and, following the concept of dynamic substitution, only evaluate the products that are available at the time of their arrival. Following the majority of publications in the area of product-line and bundle pricing (see also Section 3.3), the monetary valuation of consumer i for product j is represented by a reservation price that is denoted by v_{ij} with v_{i0} referring to the bundle. The degree of contingency θ that is defined as $\theta=(v_{i0}-\sum_{j=1}^{J} v_{ij})/\sum_{j=1}^{J} v_{ij}\ \forall\, i$ represents the consumers' relative perception of the services (cf., e.g., Venkatesh and Kamakura, 2003). θ is a predefined parameter that indicates whether consumers consider the single products to be substitutes $(\theta<0)$, complements $(\theta>0)$, or independent from each other $(\theta=0)$.

Following the concept of dynamic substitution, each consumer i self-selects the subset of available and acquirable single products, which provides her the highest positive consumer surplus, and she then purchases one unit of each. A product j is called *acquirable* for consumer i $(q_{ij}=1)$ when the corresponding surplus, which is defined as the difference between the reservation price and the product price, is nonnegative, i.e., if $v_{ij}-p_j \ge 0$. Therefore, a consumer i selects one of the following three alternatives:

- selection of the bundle provided $c_{i-1,j}>0\ \forall j=1,\dots,J$, if:

$$v_{i0}-p_0 \ge \sum_{j=1}^{J}\{v_{ij}-p_j \mid q_{ij}=1\} \tag{1}$$

- selection of all available and acquirable single products (up to a total of $J-1$ products), if:

$$v_{i0}-p_0 < \sum_{j=1}^{J}\{v_{ij}-p_j \mid q_{ij}=1\} \quad \text{provided } c_{i-1,j}>0\ \forall j=1,\dots,J \tag{2}$$

$$\text{or if } c_{i-1,0}=0 \tag{3}$$

- selection of the no-purchase option, if:

$$q_{ij}=0 \vee c_{ij}=0 \quad \forall j=0,\dots,J \tag{4}$$

Formulae (1)–(4) represent a modified form of the well-known max-surplus choice rule to account for the rational consumer behaviour in the context of mixed bundling with capacity constraints. The decision rule is based on the following two assumptions (cf., e.g., Talluri and van Ryzin, 2004, Ch. 8.3.3.3): First, a product j may only be bought if the participation constraint (PC) is satisfied, i.e., if the product is acquirable $(q_{ij}=1)$. Second, the incentive compatibility constraint (IC) claims that there is no alternative, which yields a higher surplus than the subset of products that is selected. In the event that no available product is acquirable, consumer i leaves without buying anything.

To illustrate our assumptions, we provide a little example representing the sales process of $I=5$ consumers. The service provider offers $J=2$ single products with an initial capacity of $c_{01}=c_{02}=2$ and the corresponding bundle. Trying to maximise his revenues, he can choose prices p_j for all of the products $j\in\{0,1,2\}$ out of a set of discrete price points $p_{jn}\in\{5,10,\dots,100\}$. Table 1 presents the consumers' reservation prices who value the two products independently $(\theta=0)$, i.e., $v_{i0}=v_{i1}+v_{i2}$ for all i.

In this small example, it is easy to retrace that the optimal price vector is $\boldsymbol{p}=(70,50,45)$, which generates a total revenue of $1\cdot 70+1\cdot 50+1\cdot 45=165$. The checkmarks indicate whether the PC and IC are satisfied (✓) or not (✗) for each consumer i; furthermore, the products' availability is indicated after each purchase decision. Note that if the reservation price of consumer $i=4$ for the single product $j=2$ dropped slightly to $v'_{42}=19$, a different solution would be optimal. In this case, due to (2) and $v'_{40}=v_{41}+v'_{42}=72$, she no longer buys the bundle; rather, she decides to buy the single product $j=1$ with $v_{41}-p_1=3>v_{40}-p_0=2$. As a result, the total revenue with $\boldsymbol{p}=(70,50,45)$ declines to 145, and the price vector $\boldsymbol{p}=(65,50,45)$ yields the optimal total revenue of 160.

3. Model formulations

In the following, we present two novel MILP formulations for the previously described bundle pricing problem with capacity constraints. The first model represents the consumers' choice options directly by means of the available products (p-MILP), whereas the second formulation is based on the definition of alternatives (a-MILP).

3.1. Product-based MILP

The notation for the p-MILP formulation is shown in Table 2, and the model itself follows below:

$$\text{Max } TR(\boldsymbol{x},\boldsymbol{\pi},\boldsymbol{\gamma},\boldsymbol{\omega},\mathbf{c})=\sum_{i=1}^{I}\sum_{j=0}^{J}\sum_{n=1}^{N_j} x_{ijn}p_{jn} \tag{5}$$

$$\text{s.t. } \textstyle\sum_{n=1}^{N_j}\pi_{jn}=1 \qquad \text{for all } j \tag{6}$$

$$\textstyle\sum_{n=1}^{N_0}p_{0n}\cdot\pi_{0n} \le \sum_{j=1}^{J}\sum_{n=1}^{N_j}p_{jn}\cdot\pi_{jn} \tag{7}$$

$$c_{i0} \le c_{ij} \qquad \text{for all } i; j=1,\dots,J \tag{8}$$

$$c_{ij}-c_{i0} \le M(1-\omega_{ij}) \qquad \text{for all } i; j=1,\dots,J \tag{9}$$

$$\textstyle\sum_{j=1}^{J}\omega_{ij}=1 \qquad \text{for all } i \tag{10}$$

$$c_{ij}=c_{i-1,j}-\textstyle\sum_{n=1}^{N_j}x_{ijn}-\sum_{n=1}^{N_0}x_{i0n} \qquad \text{for all } i; j=1,\dots,J \tag{11}$$

Table 1
Exemplary sales process.

Consumer i	$j=0$ $(p_0=p_{0,14}=70)$				$j=1$ $(p_1=p_{1,10}=50)$				$j=2$ $(p_2=p_{2,9}=45)$			
	v_{i0}	PC	IC	c_{i0}	v_{i1}	PC	IC	c_{i1}	v_{i2}	PC	IC	c_{i2}
1	22	✗	–	2	22	✗	–	2	0	✗	–	2
2	64	✗	–	1	51	✓	✓	1	13	✗	–	2
3	51	✗	–	1	28	✗	–	1	23	✗	–	2
4	74	✓	✓	0	53	✓	✗	0	21	✗	–	1
5	53	✗	–	0	4	✗	–	0	49	✓	✓	0

Table 2
Notation summary for the p-MILP.

Indices			Parameters	
i	$=1,\ldots,I$	Index of the consumers	p_{jn}	n^{th} price point of product j
j,k	$=0,\ldots,J$	Indices of the products (bundle is indexed by 0)	v_{ij}	Reservation price of the consumer i for the product j
n,m	$=1,\ldots,N_j$	Indices of the price points for the product j	c_{0j}	Capacity limit of the product j (integer)
			q_{ijn}	$=1$, if the PC of the consumer i for the product j at the price point p_{jn} is satisfied, i.e., $(v_{ij}-p_{jn})\geq 0$, else 0
			M	Sufficiently large number (big M)
Decision variables				
x_{ijn}	$=1$, if the consumer i chooses the product j at the price point p_{jn}, else 0			
π_{jn}	$=1$, if the product j is offered at the price point p_{jn}, else 0			
γ_{ijn}	$=1$, if $\pi_{jn}=1$, $q_{ijn}=1$, and $c_{i-1,j}>0$, else 0			
ω_{ij}	$=1$, if the remaining capacity of the product j after handling the consumer i is the minimum capacity of all of the single products and, hence, defines the remaining capacity of the bundle, else 0			
c_{ij}	Remaining capacity of the product j after handling the consumer i			

$$\textstyle\sum_{n=1}^{N_j}\gamma_{ijn}\leq c_{i-1,j} \quad \text{for all } i,j \tag{12}$$

$$\gamma_{ijn}\leq \pi_{jn}\cdot q_{ijn} \quad \text{for all } i,\ j,\ n \tag{13}$$

$$\gamma_{ijn}\geq q_{ijn}+\pi_{jn}+\frac{1}{M}\cdot c_{i-1,j}-2 \quad \text{for all } i,j,n \tag{14}$$

$$x_{ijn}\leq \gamma_{ijn} \quad \text{for all } i,j,n \tag{15}$$

$$(J-1)\cdot\textstyle\sum_{n=1}^{N_0}x_{i0n}+\sum_{j=1}^{J}\sum_{n=1}^{N_j}x_{ijn}\leq J-1 \quad \text{for all } i \tag{16}$$

$$\frac{1}{M}\left[\textstyle\sum_{n=1}^{N_0}(v_{i0}-p_{0n})\cdot\gamma_{i0n}+\frac{1}{M}-\sum_{j=1}^{J}\sum_{n=1}^{N_j}(v_{ij}-p_{jn})\cdot\gamma_{ijn}\right]\leq\textstyle\sum_{n=1}^{N_0}x_{i0n} \quad \text{for all } i \tag{17}$$

$$\frac{1}{M}\left[\sum_{k=1}^{J}\sum_{n=1}^{N_k}(v_{ik}-p_{kn})\cdot\gamma_{ikn}-\sum_{n=1}^{N_0}(v_{i0}-p_{0n})\cdot\gamma_{i0n}\right]+\sum_{n=1}^{N_j}\gamma_{ijn}-1\leq\sum_{n=1}^{N_j}x_{ijn} \quad \text{for all } i;j=1,\ldots,J \tag{18}$$

$$c_{ij}\geq 0 \quad \text{for all } i,j \tag{19}$$

$$x_{ijn},\ \pi_{jn},\ \gamma_{ijn},\omega_{ij}\in\{0,\ 1\} \quad \text{for all } i,j,n \tag{20}$$

The linear objective function (5) maximises the service provider's total revenue (TR) by aggregating the purchase decisions of the consumers $i=1,\ldots,I$. In the event that the demand of a consumer i leads to a purchase of product j at price point p_{jn}, i.e., where $x_{ijn}=1$, the corresponding revenue is added to the objective function value.

Constraints (6) ensure that exactly one price point p_{jn} is selected for each product j. Due to constraint (7), the sum of the prices of all single products represents an upper bound for the price of the bundle.

For each consumer i, constraints (8)–(10) force the remaining capacity of the bundle to the minimum of the remaining capacity of all single products. To avoid a violation of these constraints, the auxiliary variable ω_{ij} takes on the value 1 exactly for the single product that has the least remaining capacity (or for one of the products of which the least remaining capacity is identical. In this case, constraints (9) are not binding for the other products). In constraints (11), the capacity of each single product $j=1,\ldots,J$ is reduced by 1 if consumer i has bought the respective single product or the bundle.

Constraints (12)–(14) control the value of the auxiliary variable γ_{ijn}, which is forced to 0 if product j is not available (12), the corresponding price point p_{jn} is not chosen, or the product is not acquirable for consumer i (13). In each case, constraints (15) prevent a purchase. If all of these conditions are satisfied, it is possible that consumer i selects product j and γ_{ijn} is set to 1 in constraints (14). Note that M can be replaced by c_{oj} for tightening constraints (14). Furthermore, constraints (16) ensure both that at most $(J-1)$ single products are bought individually and that no single product is bought in addition to the purchase of the bundle.

Finally, constraints (17) and (18) assure compliance with the IC and force a consumer i into purchasing, if she can buy at least one product j, i.e., if $\sum_{j=0}^{J}\sum_{n=1}^{N_j}\gamma_{ijn}\geq 1$. If the aggregated consumer surplus of all of the available and acquirable single products $(\sum_{k=1}^{J}\sum_{n=1}^{N_k}(v_{ik}-p_{kn})\cdot\gamma_{ikn})$ is greater than the surplus of the bundle $(\sum_{n=1}^{N_0}(v_{i0}-p_{0n})\cdot\gamma_{i0n})$, all of these are bought according to constraints (18). In this case, for each product j with $\sum_{n=1}^{N_j}\gamma_{ijn}=1$ the value of the left hand side of constraints (18) is in the interval $(0,1)$ and, hence, the corresponding variable x_{ijn} is forced to 1. Otherwise, according to constraints (17), the bundle is chosen.

3.2. Alternative-based MILP

The idea of the a-MILP is to explicitly model each possible choice outcome as an alternative including the bundle, the no-purchase option, and every subset of the single products. The model can be viewed as a modified version of the MILP presented by Hanson and Martin (1990) that is extended to account for dynamic substitution. Although, in this approach, the number of the main decision variables x_{ik} that represent the consumers' decisions to purchase alternative k increases exponentially, the constraints for the IC can be formulated far more efficiently. Thus, the performance of solvers can be enhanced, and an optimal solution can be identified in less time for scenarios of small problem classes compared with the p-MILP presented in Section 3.1 (cf., Table 4). Similar to the p-MILP, alternative $k=0$ represents the bundle, and the alternatives $k=1,\ldots,J$ correspond to the J single products. The real subsets of single products correspond to the alternatives $k=J+1,\ldots,\ 2^J-2$, and alternative $k=2^J-1$ represents the no-purchase option. For each alternative k, the parameter y_{jk} indicates whether the alternative comprises the single product $(y_{jk}=1)$ or not $(y_{jk}=0)$. An overview of the modified notation, additional variables, and parameters is presented in Table 3.

Again, the objective function maximises the service provider's total revenue that, in contrast to (5), is calculated by aggregating the auxiliary variables τ_{ik}: max $TR'(\boldsymbol{x},\boldsymbol{p},\boldsymbol{c},\boldsymbol{\kappa},\ \boldsymbol{\pi},\boldsymbol{\tau})=\sum_{i=1}^{I}\sum_{k=0}^{2^J-1}\tau_{ik}$.

Table 3
Additional/modified notation summary for the a-MILP.

Indices			Parameters	
k,l	$=0,\ldots,2^J-1$	Indices of the alternatives	y_{jk}	$=1$, if the alternative k comprises the single product j, else 0
n	$=1,\ldots,N_k$	Indices of the price points for the bundle $k=0$ and for the single products $k=1,\ldots,J$		
Decision variables				
x_{ik}	$=1$, if the consumer i chooses the alternative k, else 0			
κ_{ik}	$=1$, if capacity of all products $j=1,\ldots,J$ with $y_{jk}=1$ is available for the consumer i, else 0			
π_{kn}	$=1$, if the bundle $k=0$ or the single product $k=1,\ldots,J$ is offered at the price point p_{kn}, else 0			
p_k	Price of the alternative $k=0,\ldots,2^J-2$			
τ_{ik}	$=p_k$, if the consumer i chooses the alternative k (if $x_{ik}=1$), else 0			
c_{ik}	Remaining capacity of the alternative k after handling the consumer i			

If consumer i purchases alternative k, these variables are equal to the price p_k of this alternative according to the following constraints (21)–(23):

$$\tau_{ik} \le p_k \quad \text{for all } i,k \tag{21}$$

$$\tau_{ik} \le v_{ik}\cdot x_{ik} \quad \text{for all } i,k \tag{22}$$

$$\tau_{ik} \ge p_k - M\cdot(1-x_{ik}) \quad \text{for all } i,k \tag{23}$$

According to the chosen price point π_{kn}, a simple linear equation allocates the price p_k to each alternative $k=0,\ldots,J$ ($p_k=\sum_{n=1}^{N_k}\pi_{kn}p_{kn}$), and for each composed alternative $k=J+1,\ldots,2^J-2$ consisting of more than 1 but less than J elements, the prices of all of the included single products are aggregated: $p_k=\sum_{j=1}^{J}p_j y_{jk}$. Obviously, the price for the no-purchase option $k=2^J-1$ is equal to 0. The capacity c_{ij} is reduced for each single product $j=1,\ldots,J$ analogously to (11), and, in addition, the binary variable κ_{ik} indicates whether the capacity of all single products being contained in alternative k is available ($\kappa_{ik}=1$) or not ($\kappa_{ik}=0$). Then, each consumer i is forced to choose exactly one alternative: $\sum_{k=0}^{2^J-1}x_{ik}=1$. Finally, the IC can be formulated according to constraints (24)

$$\sum_{k=0}^{2^J-1}(v_{ik}\cdot x_{ik}-\tau_{ik}) \ge v_{il}\cdot\kappa_{il}-p_l \quad \text{for all } i,l \tag{24}$$

3.3. Determination of consumers' reservation prices

Both the p-MILP and the a-MILP are based on deterministic values of the consumers' willingness-to-pay. Following the majority of publications in the area of product-line and bundle pricing (cf., e.g., Adams and Yellen, 1976; Schmalensee, 1984; Hanson and Martin, 1990; Venkatesh and Kamakura, 2003; Wu et al., 2008; Burkart et al., 2012) and also some revenue management applications (Li and Tang, 2012), the monetary valuation of consumer i for alternative k is represented by a reservation price that is denoted by v_{ik}. In case alternative k provides no utility to the consumer and, hence, she is not willing to pay anything, $v_{ik}=0$. This concept of using reservation prices to express a consumer's upper threshold of her price acceptance for a set of products is one of the most widely used methods to express willingness-to-pay in the literature.

One of the main reasons for its high degree of acceptance may be attributed to the fact that during the past decades, the marketing community provided a large collection of methods to measure consumer preferences, willingness-to-pay intervals, and realistic distributions of reservation prices of consumer segments (see, e.g., Jedidi and Zhang, 2002; Jedidi et al., 2003). Among the most common methods are conjoint analysis and contingent valuation (Voelckner, 2006). Moreover, the estimation procedures are continuously developed further. Recently, Gensler et al. (2012) presented a method that is based on choice-based conjoint analysis to improve the accuracy of the estimation of consumers' willingness-to-pay and Schlereth et al. (2012) illustrated how to estimate reservation price intervals.

Note that both model formulations represent the sales process of a single scenario, i.e., prices are optimised given a known stream of consumers with deterministic reservation prices, which can be thought of random variates of the underlying distribution. Using efficient solution algorithms (such as the metaheuristics presented in Section 4) it is easily possible to optimise over the stochastic counterpart of $S \ge \sigma$ scenarios with σ being a sufficiently large number in terms of a predefined level of accuracy (please cf. our tests in Section 6.4). In this method, also known as sample average approximation, the arithmetic mean (or any other statistic depending on the risk aversion of the service provider) of the revenues of all S scenarios is optimised (Kleywegt et al., 2002). Thereby, the underlying distribution of the consumers' reservation prices can adequately be addressed.

3.4. Comparison of the p-MILP and the a-MILP

As constraints (20) suggest, the p-MILP is characterised by a large number of binary variables. In total, $\sum_{j=0}^{J}N_j$ variables π_{jn} are necessary for the allocation of price points to products and $I\cdot J$ variables ω_{ij} control the availability of the bundle. In addition, the definition of $2\cdot I\cdot\sum_{j=0}^{J}N_j$ variables x_{ijn} and γ_{ijn} are required to model the consumer choice behaviour.

In contrast, the a-MILP does not require any decision variables with more than two indices due to the introduction of the variable p_k, which inherently comprises the selection of the corresponding price point p_{kn} (cf. the definitions of the variables in Table 3). Like the p-MILP, the definition of $\sum_{k=0}^{J}N_k$ variables π_{kn} is necessary to allocate the price points to the bundle and the single products, respectively. Furthermore, an exponentially increasing number of $2\cdot I\cdot 2^J$ variables x_{ik} and κ_{ik} are required to model the consumer choice behaviour and to track the availability of each alternative.

Taking a model-independent point of view, it becomes apparent that the number of possible solutions, i.e., the number of feasible price combinations, is determined by both the number of single products J and the number of price points N_j. This number grows exponentially with J and is given by $\Pi_{j=0}^{J}N_j$. For each feasible price combination and each consumer $i=1,\ldots,I$, $(J+1)$ consumer preferences have to be evaluated. Hence, for the determination of the objective function values for all solutions by means of a complete enumeration a computational effort of $O((J+1)\cdot I\cdot\Pi_{j=0}^{J}N_j)$ results.

As preliminary tests have shown, both model formulations can be solved exactly with standard solvers for small problem instances. We have implemented the models with ILOG CPLEX Optimization Studio 12.2 and used the corresponding solver CPLEX 12.2 with default settings performing a depth-first search for their solutions. (For details on the parameter settings and the test computer, the reader is referred to Section 6.1). However, the results shown in Table 4 reveal that the solution of problem classes

Table 4
The average solution runtimes of CPLEX 12.2 in seconds (o.o.m.=out of memory).

Single products	price points	p-MILP	a-MILP	improvement of the a-MILP
$J=2$	$N^{'}=7$	2.47	3.12	−17.5%
	$N^{'}=31$	42.10	4.42	81.8%
	$N^{'}=61$	231.59	5.19	93.2%
$J=4$	$N^{'}=7$	136.79	o.o.m.	
	$N^{'}=31$	16,390.85	o.o.m.	
	$N^{'}=61$	112,512.93	o.o.m.	

with $J>2$ either requires enormous runtimes depending on the number of price points $N^{'}=N_0=\ldots=N_J$ or is not possible at all. Therefore, in the following, we present several heuristic solution procedures based on variable neighbourhoods to efficiently solve medium to large instances of the bundle pricing problem as well.

4. Metaheuristics

This section describes various metaheuristics, which are based on Variable Neighbourhood Search (VNS) and have been adapted to the bundle pricing problem on hand. VNS is a single-solution based metaheuristic that was first introduced by Mladenović and Hansen (1997). Since then, VNS has been further developed and applied to a wide field of applications (cf., e.g., Hansen et al., 2010a, 2010b). The basic idea of VNS is to dynamically expand the neighbourhood in case a local optimum is reached with the purpose to resolve it. Depending on the specific definition of the algorithm, the heuristic is either performed in a deterministic way or contains stochastic elements.

4.1. VNS-based solution procedures

To find near optimal solutions for the bundle pricing problem, we have implemented three different methods of VNS: Variable Neighbourhood Descent (VND), which, because of our objective of maximising the service provider's revenue, is an ascent type procedure; Reduced VNS (RVNS); and Basic VNS (BVNS). Given a predefined set of neighbourhood structures $\mathcal{N}_k$ with $k=1,\ldots,k_{max}$ and the corresponding definition of the k^th^ neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ of a current solution $\boldsymbol{z}^0$, the VNS procedures differ in the modality of how the selection of a neighbour solution $\boldsymbol{z}'\in\mathcal{N}_k(\boldsymbol{z}^0)$ is conducted. In all of the procedures, solutions are represented by a vector of price indices $\boldsymbol{z}=(z_0,\ldots,z_J)$ with $z_j\in\{1,\ldots,N_j\}$ for all j. The set of all of the feasible vectors $\boldsymbol{z}$, i.e., for which constraint (7) holds, is denoted by Z.

Starting with an initial solution $\boldsymbol{z}^0\in Z$, VND performs a deterministic best improvement (BI) or first improvement (FI) local search within the first neighbourhood structure $\mathcal{N}_1$ until a local optimum is reached. After each move, the current solution is updated and the best neighbour solution $\boldsymbol{z}'\in\mathcal{N}_1(\boldsymbol{z}^0)$ with $\mathcal{N}_1(\boldsymbol{z}^0)\subseteq Z$ becomes the basis for the next move ($\boldsymbol{z}^0\leftarrow\boldsymbol{z}'$). If no further improvement is possible, the neighbourhood structure $\mathcal{N}_k$ is successively extended with $k=2,\ldots,k_{max}$ and a local search is performed again. In the event that an improved solution is identified, the index of the neighbourhood structure is reset to $k=1$, and the procedure starts from the beginning; otherwise, if $k=k_{max}$, VND stops and returns the best known solution $\boldsymbol{z}^0$.

In contrast, in RVNS, the selection of a neighbour solution $\boldsymbol{z}'\in Z$ from the current neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ is performed completely at random. If its objective function value is greater than the incumbent, an update of the current solution takes place ($\boldsymbol{z}^0\leftarrow\boldsymbol{z}'$) and the search continues with $k=1$. Otherwise, the neighbourhood structure is increased ($k\leftarrow k+1$) to choose the next random solution. To appropriately account for the stochastic nature of RVNS, this procedure does not necessarily stop if $k=k_{max}$. In fact, this procedure continues with the current solution using $k=1$ again, until a predefined stopping criterion, such as the maximum solution time or the maximum number of iterations it_{max}, is reached.

Finally, BVNS combines deterministic and stochastic elements. After selecting a first neighbour solution $\boldsymbol{z}'\in\mathcal{N}_1(\boldsymbol{z}^0)$ at random, a deterministic BI or FI local search is performed analogously to VND. Then, the objective function values of the local optimum $\boldsymbol{z}''\in Z$ and the starting solution $\boldsymbol{z}^0\in Z$ are compared. Depending on whether $\boldsymbol{z}^0$ could have been improved, either an update of the incumbent takes place ($\boldsymbol{z}^0\leftarrow\boldsymbol{z}''$) keeping the current neighbourhood structure $\mathcal{N}_1$ or the process starts again using an expanded set of neighbour solutions ($k\leftarrow k+1$). Similar to RVNS, this procedure iterates until a stopping criterion is reached.

4.2. Problem-specific neighbourhood schemes

A crucial factor to the success of VNS-based metaheuristics is the definition of problem specific neighbourhood structures $\mathcal{N}_k$ ($k=1,\ldots,k_{max}$). Thereby, the solutions $\boldsymbol{z}'\in Z$ that compose the k^th^ neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)\subseteq Z$ of the current solution $\boldsymbol{z}^0\in Z$ are defined according to a pre-specified transformation rule, referred to as *neighbourhood scheme*. In what follows, we present three different neighbourhood schemes that are all based on a common parameter s measuring the distance between two solutions $\boldsymbol{z}^1$ and $\boldsymbol{z}^2$. The parameter is defined as the sum of the product-specific distances s_j that are calculated for each product $j=0,\ldots,J$ by taking the absolute value of the difference between the index of the price point of the first solution z_j^1 and the corresponding index of the price point of the second solution z_j^2:

$$s=\sum_{j=0}^{J}s_j=\sum_{j=0}^{J}|z_j^1-z_j^2| \tag{25}$$

In the first neighbourhood scheme, the neighbour solutions $\boldsymbol{z}'\in Z$ belonging to a neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ of the current solution $\boldsymbol{z}^0\in Z$ are obtained by increasing or decreasing the index of the price points of k products by 1. Thus, a neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ contains all of the price vectors $\boldsymbol{z}^{'}\in Z$ having a distance $s=k$ to the current solution $\boldsymbol{z}^0\in Z$ with exactly k product-specific distances s_j being 1. The transformation rule of the first neighbourhood scheme is formally given by:

$$\mathcal{N}_k(\boldsymbol{z}^0)=\left\{\boldsymbol{z}'\in Z\middle|\, s=k\wedge\left[\sum_{j=0}^{J}(1||s_j|=1)\right]=k\right\}\quad\text{for all } k \tag{26}$$

The second neighbourhood scheme only allows for altering the index of the price point of one product. As the distance between any neighbour solution $\boldsymbol{z}'\in Z$ and the current solution $\boldsymbol{z}^0\in Z$ is again defined as $s=k$, the neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ comprises all of the solutions that result from increasing or decreasing the index of one price point by k. That is, exactly one product-specific distance s_j equates to k, whereas $s_j=0$ holds for all other products:

$$\mathcal{N}_k(\boldsymbol{z}^0)=\left\{\boldsymbol{z}'\in Z\middle|\, s=k\wedge\left[\sum_{j=0}^{J}(1||s_j|=k)\right]=1\right\}\quad\text{for all } k \tag{27}$$

The third neighbourhood scheme defines a neighbourhood $\mathcal{N}_k(\boldsymbol{z}^0)$ as the set of solutions $\boldsymbol{z}'\in Z$ with distance $s=k$ to the current solution $\boldsymbol{z}^0\in Z$ without any further restrictions. Thus, up to k price points of a neighbour solution can be changed compared with $\boldsymbol{z}^0\in Z$. Therefore, the amount of each modification is not predetermined and depends on the actual number of altered price points. Thus, the third neighbourhood scheme is a combination and an extension of the first and the second scheme and is defined according to the following transformation rule:

$$\mathcal{N}_k(\boldsymbol{z}^0) = \{\, \boldsymbol{z}' \in Z |\; s = k \,\} \quad \text{for all } k \tag{28}$$

Obviously, the proposed neighbourhood schemes specify an identical set of neighbour solutions with respect to the first neighbourhood $\mathcal{N}_1(\boldsymbol{z}^0)$. However, they differ significantly for $k = 2, \ldots, k_{max}$. For all of the schemes, we can determine an upper bound of the cardinality of the set of neighbour solutions that is reached if all neighbour solutions specified by the transformation rule are feasible:

- scheme 1 : $|\mathcal{N}_k(\boldsymbol{z}^0)| \le \binom{J+1}{k} \cdot 2^k \quad \forall\, k = 1, \ldots, k_{max}$
- scheme 2 : $|\mathcal{N}_k(\boldsymbol{z}^0)| \le (J+1) \cdot 2 \quad \forall\, k = 1, \ldots, k_{max}$
- scheme 3 : $|\mathcal{N}_k(\boldsymbol{z}^0)| \le \sum_{k'=1}^{\min\{J+1,k\}} \binom{J+1}{k'} \cdot \binom{k-1}{k'-1} \cdot 2^{k'} \quad \forall\, k = 1, \ldots, k_{max}$

To illustrate the differences between the three neighbourhood schemes, Fig. 1 displays a possible alteration of the current solution $\boldsymbol{z}^0 \in Z$, which is represented by the zero line in each case. Assuming $J = 2$ single products and $k = 3$, this illustration reveals that a neighbour solution $\boldsymbol{z}^1 \in Z$ is obtained by altering the price of each product when using scheme 1. With scheme 2, only one product price is altered, whereas any combination is possible with scheme 3.

4.3. Construction heuristics

In addition to the definition of the set of neighbourhood structures $\mathcal{N}_k$, it is necessary to specify how an initial feasible solution $\boldsymbol{z} \in Z$ is generated regardless of which VNS-based method is applied. For this purpose, we present two construction heuristics (CHs) along with the possibility to randomly create an initial solution. For each CH, price points $p_{jn'}$ are separately determined for each product $j = 0, \ldots, J$, and the corresponding indices are stored for representing the initial solution ($z_j := n'$). If the resulting price vector violates constraint (7), i.e., if $p_{0,z_0} > \sum_{j=1}^{J} p_{j,z_j}$, a repair rule is applied to guarantee feasibility. In this case, we set $z_0 = \max\{n \in \{1, \ldots, N_0\} |\; p_{0n} \le \sum_{j=1}^{J} p_{j,z_j}\}$.

Within the first CH (CH 1), we begin with calculating the arithmetic mean of the reservation prices $\overline{v}_j = \frac{1}{I}\sum_{i=1}^{I} v_{ij}$ for each product j. To compose an initial solution, we then choose the price point $p_{jn'}$ ($z_j := n'$) that is the closest to $\overline{v}_j$, i.e., for which $|p_{jn'} - \overline{v}_j| = \min_{n \in \{1,\ldots,N_j\}} \{|p_{jn} - \overline{v}_j|\}$ holds.

The idea of the second CH (CH 2) is to separate consumers, who buy single products, from those who purchase the bundle. As the seller prefers selling products individually to bundle sales with respect to his revenue when neglecting any side effects, we first consider only single products. We start with calculating so-called achievable revenues a_{jn} for each price point p_{jn} with $n = 1, \ldots, N_j$ by $a_{jn} = p_{jn} \cdot \min\{\, c_{0j}; |B_{jn}|\}$ with $B_{jn} = \{i | v_{ij} \ge p_{jn}\}$. Here, B_{jn} defines the set of consumers for each product j for whom the PC is satisfied. Then, the index n' of the price point yielding the highest a_{jn} is stored in the initial solution $\boldsymbol{z}$. As a consumer who buys j' single products with $1 \le j' < J$ does not choose the bundle, we eliminate up to $\sum_{j=1}^{J} c_{0j}$ consumers with $v_{ij} \ge p_{jn'}$ (in the sequence of their arrival). Subsequently, we determine $z_0 = n'$ that yields the highest achievable revenue for the bundle $a_{0n'}$.

5. Generation of special benchmark scenarios

To appropriately evaluate the performance of our metaheuristics, a realistic test-bed with well-defined problem classes has to be defined. Each problem class C is characterised by a set of deterministic parameters, such as the number of products J, the number of consumers I, the degree of contingency θ, and the initial capacity of each product c_{0j}. By fixing all of the stochastic variables, such as the reservation prices v_{ij} at the problem on hand, to a certain feasible value, a scenario of the particular problem class arises. Then, given a set of scenarios, which is also referred to as the deterministic equivalent of the stochastic problem, a solution $\boldsymbol{z} \in Z$ can be evaluated with respect to the average objective function value for the current set.

For a single scenario, the performance of heuristic solution procedures in terms of the solution quality can either be evaluated by comparing the corresponding objective function value with an optimal one or with an upper or lower bound on the objective function value (depending on the orientation of the objective function). However, due to the difficulty of the bundle pricing problem, it is computationally expensive to determine an optimal solution. Furthermore, the determination of tight upper bounds is difficult, as the problem (5)–(20) does not hold any constraints of which the structure could be favourably exploited. The majority of constraints model the consumer behaviour. For these reasons, we present an innovative approach to measure the performance of our metaheuristics. The idea is to generate random scenarios of all problem classes in our test-bed in such a way that the optimal solution and the corresponding total revenue are known in advance and, as a consequence, serve as a benchmark for the solution quality of the various heuristics.

Similar to the generation of standard scenarios in a simulation study, the first step is to specify the deterministic parameters, which characterise the problem class on hand. Next, we determine a feasible price vector $\hat{\boldsymbol{p}}$ with $\hat{p}_0 < \sum_{j=1}^{J} \hat{p}_j$ and sales volumes $\hat{x}_j$ for each product j, which define the optimal revenue $\hat{r} = \sum_{j=0}^{J} \hat{p}_j \hat{x}_j$. Subsequently, a random matrix of purchase decisions $\hat{\boldsymbol{x}} = (\hat{x}_{ij})$ can be established by defining exactly which consumer i selects which product j ($\hat{x}_{ij} = 1$) such that $\sum_{i=1}^{I} \hat{x}_{ij} = \hat{x}_j$. As a consequence, we can classify all consumers $i = 1, \ldots, I$ into three segments, i.e., purchasers of the bundle ($\mathcal{J}^B$), purchasers of single products ($\mathcal{J}^{SP}$), and

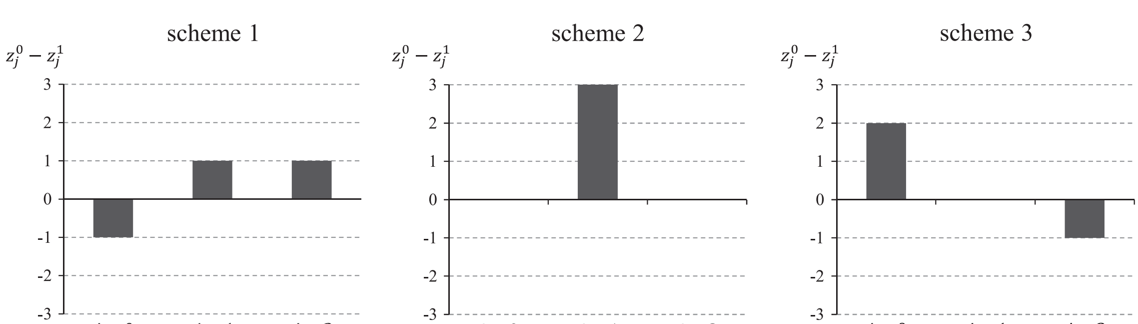

Fig. 1. Exemplary outcomes of the neighbourhood schemes.

consumers who do not buy anything ($\mathcal{J}^{NP}$):

$$i \in \begin{cases} \mathcal{J}^{B}, & \text{if } \hat{x}_{i0} = 1 \\ \mathcal{J}^{SP}, & \text{if} \sum_{j=1}^{J} \hat{x}_{ij} \geq 1 \\ \mathcal{J}^{NP}, & \text{if } \sum_{j=0}^{J} \hat{x}_{ij} = 0 \end{cases} \tag{29}$$

In the last step, we generate reservation prices v_{ij} of each consumer $i=1,\ldots,I$ for all of the products $j=0,\ldots,J$ in such a way that the following two requirements are satisfied:

(1) the predefined purchase decisions $\hat{\boldsymbol{x}}$ and the price vector $\hat{\boldsymbol{p}}$ represent a feasible solution to (5)–(20)
(2) the price vector $\hat{\boldsymbol{p}}$ is optimal, i.e., there is no other price vector $\boldsymbol{p}'$ with feasible purchase decisions $\boldsymbol{x}'$ and $r' = \sum_{j=0}^{J} p_j' x_j' > \hat{r} = \sum_{j=0}^{J} \hat{p}_j \hat{x}_j$.

In the following subsections, the necessary conditions for the generation of the reservation prices that satisfy these requirements are derived.

5.1. Feasibility conditions

First, we want to develop the necessary conditions for the requirement (1). These conditions specify how the reservation prices v_{ij} have to be determined such that the predefined purchase decisions $\hat{\boldsymbol{x}}$ result from the given price vector $\hat{\boldsymbol{p}}$, leading $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{p}}$ to represent a feasible solution to (4)–(19). For this purpose, we discriminate between the case that all products are still available and the case in which at least one single product and, hence, the bundle, is sold out. Analysing each consumer segment separately, we start with the consideration of the case that $c_{i-1,j} > 0\ \forall j$:

- Obviously, for each purchaser of single products $i \in \mathcal{J}^{SP}$, it must hold that $v_{ij} \geq \hat{p}_j$ for each product j with $\hat{x}_{ij} = 1$ and $v_{ij} < \hat{p}_j$ for each product j with $\hat{x}_{ij} = 0$. Furthermore, the following condition that represents the IC and, hence, guarantees that a consumer receives a higher utility from buying products individually than from buying the bundle has to be satisfied: $\sum_{j=1}^{J} (v_{ij} - \hat{p}_j) q_{ij} > v_{i0} - \hat{p}_0$ with q_{ij} indicating whether the PC of consumer i for product j is satisfied ($q_{ij} = 1$) or not ($q_{ij} = 0$). Substituting $v_{i0} = (1+\theta) \sum_{j=1}^{J} v_{ij} > 0$ (cf. the definition of θ in Section 2) we obtain, after some minor rearrangements of terms, a necessary condition for the sum of the reservations prices for the products that are not acquirable: $\sum_{j=1}^{J} v_{ij}(1-q_{ij}) < \frac{1}{1+\theta} \left[\hat{p}_0 - \sum_{j=1}^{J} \hat{p}_j q_{ij} - \theta \sum_{j=1}^{J} v_{ij} q_{ij} \right]$. This inequality can easily be interpreted, if $\theta = 0$: a consumer selects products individually, if and only if the premium she had to pay for the bundle compared to the aggregated prices of the single products that she buys $\left(\hat{p}_0 - \sum_{j=1}^{J} \hat{p}_j q_{ij} \right)$ is greater than what she is willing to pay in total for the products that she does not buy individually, i.e., for which $q_{ij} = 0$.
- For each purchaser of the bundle $i \in \mathcal{J}^{B}$, the conditions $v_{i0} \geq \hat{p}_0$ and $v_{i0} - \hat{p}_0 \geq \sum_{j=1}^{J} (v_{ij} - \hat{p}_j) q_{ij}$ must hold. Applying the same substitution as in the previous paragraph, we obtain the necessary condition that $\sum_{j=1}^{J} v_{ij}(1 - q_{ij} + \theta) \geq \hat{p}_0 - \sum_{j=1}^{J} \hat{p}_j q_{ij}$ after some algebraic transformations. The interpretation is identical to the single product case. If $\theta = 0$, it is obvious that the condition again constrains the reservation prices for the products that are not acquirable ($q_{ij} = 0$) compared with the premium for the bundle.
- For all consumers $i \in \mathcal{J}^{NP}$, who do not purchase anything, the reservation prices must be bound by the product price, i.e., $v_{ij} < \hat{p}_j\ \forall\, j = 0,\ldots,J$.

Once one or several single products j are sold out, the bundle is not available anymore as well. In this case, no restriction for the reservation prices of any purchaser of single products $i \in \mathcal{J}^{SP}$ or any consumer $i \in \mathcal{J}^{NP}$ are necessary for each product j with $c_{i-1,j} = 0$ (besides $v_{ij} \geq 0$). For the reservation prices of the single products with $c_{i-1,j} > 0$, the aforementioned conditions persist, i.e., $v_{ij} < \hat{p}_j$ for each single product j with $\hat{x}_{ij} = 0$ and $v_{ij} \geq \hat{p}_j$ for each single product j with $\hat{x}_{ij} = 1$.

5.2. Optimality conditions

In this subsection, we derive sufficient and necessary conditions for the requirement (2), which further limit the set of possible reservation prices. Observing the conditions of Section 5.1, the reservation prices v_{ij} will guarantee that the solution given by the set of prices $\hat{\boldsymbol{p}}$ and the purchase decisions $\hat{\boldsymbol{x}}$ is feasible. However, the conditions do not prevent that eventually other feasible solutions exist, which lead to a higher total revenue. The simplest rule to avoid this eventuality is to force the purchasers' reservation prices for each product j with $\hat{x}_{ij} = 1$ to the interval $v_{ij} \in [p_{jn} = \hat{p}_j;\ p_{j,n+1})$. This approach, however, results in quite unrealistic scenarios with uncommon distributions of reservation prices. Therefore, we want to derive optimality conditions that allow reservation prices to be as general and smoothly distributed as possible. For this purpose, we introduce further assumptions for the definition of our problem classes that, in general, do not constitute significant limitations regarding the applicability to real world problems in the light of capacity constraints:

- The initial capacity c_{0j} of each single product $j = 1,\ldots,J$ is sold out during the sales period: $\sum_{i=1}^{I} (\hat{x}_{ij} + \hat{x}_{i0}) = \hat{x}_j + \hat{x}_0 = c_{0j}\ \forall\, j = 1,\ldots,J$.
- For convenience and without loss of generality, the same set of N' price points p_{jn} is given for all of the single products j. The price points are equidistant with d^{SP} defining the difference between two adjacent price points: $d^{SP} = p_{j,n+1} - p_{jn}\quad \forall j = 1,\ldots,J;\quad \forall n = 1,\ldots,N'-1$. Correspondingly, the distance between two price points p_{0n} for the bundle is referred to as d^{B}.

Given the assumption that no products are available at the end of the selling period, ceteris paribus, choosing a lower price $p_{jn} < \hat{p}_j$ for any product $j = 0,\ldots,J$ is not reasonable.

However, it has to be guaranteed that there is no incentive to choose a product price $p_{jn} > \hat{p}_j$ and generate more revenue despite a possible lower sales volume $(\sum_{i=1}^{I} x_{ij} \leq \hat{x}_j)$. Therefore, we calculate for each product j the optimal product-specific revenue $\overline{r}_j$ that is to be expected at the pre-specified price $\hat{p}_j$ and sales volume $\hat{x}_j$ by $\overline{r}_j = \hat{p}_j \cdot \hat{x}_j$. Then, to guarantee that the price vector $\hat{\boldsymbol{p}}$ is optimal, necessary conditions for the reservation prices v_{ij} of all consumers $i \in \mathcal{J}^{SP}$ and $i \in \mathcal{J}^{NP}$ are:

$$\sum_{i=1}^{I} \{1 \mid i \in \{\mathcal{J}^{SP} \cup \mathcal{J}^{NP}\} \wedge v_{ij} \geq p_{jn}\} \leq \left\lfloor \frac{\overline{r}_j}{p_{jn}} \right\rfloor \quad \text{for all } j = 1,\ldots,J;\ n \middle| p_{jn} > \hat{p}_j \tag{30}$$

Constraints (30) restrain the number of consumers with reservation prices v_{ij} above each price point $p_{jn} > \hat{p}_j$, thereby ensuring that the selection of a price point higher than the pre-specified price $\hat{p}_j$ results in lower product-specific revenues for each product $j = 1,\ldots,J$. To avoid multiple optimal solutions, the right hand side of constraints (30) can be altered to $\left\lfloor \frac{\overline{r}_j}{p_{jn}} \right\rfloor - 1$ if $\overline{r}_j \bmod p_{jn} = 0$. In this context, purchasers of the bundle $i \in \mathcal{J}^{B}$ do not have to be considered, as they never choose single products $j = 1,\ldots,J$ instead of the bundle at any price point $p_{jn} > \hat{p}_j$, as a result of the conditions derived in Section 5.1.

To further illustrate the optimality conditions, we revisit our example from Section 2. For the sales volumes $\hat{x}_0 = \hat{x}_1 = \hat{x}_2 = 1$ and the optimal price vector $\hat{\boldsymbol{p}} = \overline{\boldsymbol{r}} = (70,\ 50,\ 45)$, the consumer segmentation $\mathcal{J}^{SP}=\{2, 5\}$, $\mathcal{J}^{B}=\{4\}$, and $\mathcal{J}^{NP}=\{1, 3\}$ results. Comparing the data presented in Table 1, we observe that the reservation prices for the single products $j\in\{1, 2\}$ of the consumers $i\in\{1, 2, 3, 5\}$ satisfy constraints (30). As for each $p_{1n}>50$ and $p_{2n}>45$, the right hand side of (30) equals 0, it must hold that $v_{ij}<p_{jn}\ \forall\ n|p_{jn}>\hat{p}_j$.

Thus far, the optimality conditions ensure that the consumers' reservation prices offer no incentive to change the price of a single product, ceteris paribus, or to choose a lower bundle price. Yet, in some scenarios, the total revenue could be increased by changing several prices simultaneously. For example, the total revenue could be enhanced by selecting a higher bundle price $p_{0n}>\hat{p}_0$, which, at first, leads to a lower sales volume of bundles. As a result, however, the single products originally intended to be sold as a bundle could now be sold separately. This in turn might allow for increasing the price of some of the single products, which might yield a higher total revenue in the end. To rule out this possibility and guarantee optimality despite these complex interdependencies, we need to formulate rather restrictive conditions for the reservation prices of purchasers of the bundle $i\in\mathcal{J}^B$:

$$v_{i0}\in\left[\hat{p}_0;\ \hat{p}_0+d^B\right) \quad \text{for all } i\in\mathcal{J}^B \tag{31}$$

$$v_{ij}\in\left[0;\ \hat{p}_j+d^{SP}\right) \quad \text{for all } i\in\mathcal{J}^B;\quad j=1,\dots,J \tag{32}$$

Constraints (31) guarantee that choosing a bundle price $p_{0n}>\hat{p}_0$ results in no bundle sales at all. At the same time, the reservation prices of the consumers $i\in\mathcal{J}^B$ who were intended to buy the bundle do not exceed the pre-specified single product prices $\hat{p}_j$ by more than d^{SP} according to (32). Therefore, and due to constraints (30), it is not possible to generate more revenue by simultaneously charging higher prices for the single products. Looking at the example again, the reservation prices of consumer $i\in\mathcal{J}^B=\{4\}$ both comply with constraints (31) and (32), as $v_{40}\in[70;\ 75)$, $v_{41}\in[0;\ 55)$, and $v_{42}\in[0;\ 50)$.

Additionally, constraints (30) and (32) ensure that it might not be possible to increase revenues by charging a lower price $p_{j'}n<\hat{p}_{j'}$ for a single product j', thereby leading to an early clearance sale of that single product. Besides a lower product-specific revenue $\overline{r}_{j'}$, as a consequence, the bundle would be unavailable. Then, for the same reason as previously discussed, choosing higher prices for the other single products $j\in\{1,\dots,J\}\backslash\{j'\}$ does not pay off.

Finally, the option to enhance the total revenue by choosing a bundle price $p_{0n}>\hat{p}_0$, which rules out bundle sales due to (31), and selling up to J single products individually to the consumers $i\in\mathcal{J}^B$ at the unchanged prices $\hat{p}_j$ with $\sum_{j=1}^{J}\hat{p}_j>\hat{p}_0$ can be obviated by the following condition. Depending on the price vector $\hat{\boldsymbol{p}}$, constraints (33) with ϵ representing a sufficiently small positive number limit the number of single products $j=1,\dots,J$ that are acquirable for each purchaser of the bundle $i\in\mathcal{J}^B$ and, therefore, her reservation prices v_{ij}:

$$\sum_{j=1}^{J}\hat{p}_j q_{ij}=\sum_{j=1}^{J}\hat{p}_j\left\lceil\frac{v_{ij}-\hat{p}_j+\epsilon}{M}\right\rceil\le\hat{p}_0 \quad \text{for all } i\in\mathcal{J}^B \tag{33}$$

Beyond that, we observe the following causal relations in this context assuming the degree of contingency $\theta=0$, i.e., $v_{i0}=\sum_{j=1}^{J}v_{ij}$:

- First, if $\hat{p}_0\le\sum_{j=1}^{J}\hat{p}_j-d^B$, then it is not possible that $q_{ij}=1$ for all single products $j=1,\dots,J$ as this would imply that $\sum_{j=1}^{J}v_{ij}=v_{i0}\ge\sum_{j=1}^{J}\hat{p}_j$; however, $v_{i0}<\hat{p}_0+d^B\le\sum_{j=1}^{J}\hat{p}_j$ due to (31). As a consequence, at most $J-1$ single products may be sold individually to a consumer $i\in\mathcal{J}^B$. Hence, $\sum_{j=1}^{J}\hat{p}_j-\min_{j\in\{1,\dots,J\}}\{\hat{p}_j\}$ represents an upper bound on the achievable revenue from each consumer $i\in\mathcal{J}^B$. As long as $\hat{p}_0$ exceeds this upper bound, selling products individually does not pay off as constraints (33) are satisfied accordingly.

  This is the case in our example: as $\hat{p}_0=70\le90=\sum_{j=1}^{J}\hat{p}_j-d^B$, there is at least one reservation price with $v_{ij}<\hat{p}_j$, i.e., $v_{42}=21<\hat{p}_2=45$. At the same time, $\hat{p}_0=70$ is greater than the upper bound of 50.
- Second, if $\hat{p}_0\in\left(\sum_{j=1}^{J}\hat{p}_j-d^B;\ \sum_{j=1}^{J}\hat{p}_j\right]$, then it might be possible that $v_{ij}\in[\hat{p}_j;\ \hat{p}_j+d^{SP})\ \forall j=1,\dots,J$. However, choosing a higher bundle price $p_{0n}>\hat{p}_0$ leads to an infeasible price vector $\boldsymbol{p}$ with $\hat{p}_0>\sum_{j=1}^{J}\hat{p}_j$, and thus, the latter problem is not an issue.

If we create scenarios of a given problem class C by generating the consumers' reservation prices according to all of the assumptions and conditions previously outlined, on the one hand, the predefined purchase decisions $\hat{\boldsymbol{x}}$ and the price vector $\hat{\boldsymbol{p}}$ represent a feasible solution to the MILP presented in Section 3.1 (requirement 1). On the other hand, the chosen price vector $\hat{\boldsymbol{p}}$ is optimal (requirement 2). For illustration purposes, we have shown that the consumers' reservation prices in the introductory example of Section 2 characterise this type of special benchmark scenario. We have developed an algorithm that maintains all of the conditions outlined above and thus generates random scenarios of which the optimal price vector $\hat{\boldsymbol{p}}$ and the resulting total revenue $\hat{r}$ are known.

6. Experimental study

In this section, we present the details of our simulation study. On the basis of the results of our computational tests, we evaluate the performance of our heuristic solution procedures that were described in Section 4 with respect to their solution qualities and runtimes and derive managerial insights.

6.1. Data sets and experimental conditions

First, we introduce the problem classes by their distinctive deterministic parameters. As mentioned previously, each problem class C is characterised by the number of single products J, their initial capacity c_0, the set of identical price points $p_{jn}\ \forall j=1,\dots,J;\ n=1,\dots,N'$, and the number of consumers I. Furthermore, each class is distinguished by the pre-selected optimal price vector $\hat{\boldsymbol{p}}$ and the sales volumes $\hat{x}_j$ for each product j. The degree of contingency $\theta=0$.

Applying a full factorial design, we generate $\Pi_{i=1}^{7}|C_i|=432$ problem classes by combining each element out of 7 different sets of components $C_1,\dots,C_7$:

- $C_1=\{2, 4, 10\}$: the number of single products $J\in C_1$.
- $C_2=\{100\}$: the initial capacity $c_0\in C_2$.
- $C_3=\{7, 31, 61\}$: the number of predefined price points $N'\in C_3$ with $N'=N_0=\dots=N_J$. As the lowest (highest) price point for each single product p_{j1} ($p_{jN'}$) is fixed to 50 (80), the difference between two adjacent price points corresponds to $d^{SP}\in\{0.5, 1, 5\}$, depending on the number of price points N'. Note that the price range can be rescaled arbitrarily.
- $C_4=\{2, 5\}$: the scaling factor $sf\in C_4$ that determines the number of consumers $I=sf\cdot I^{min}$. The minimum number of consumers I^{min} that is necessary to sell the entire initial capacity $\sum_{j=1}^{J}c_0$ is given by $I^{min}=\hat{x}_0+2\cdot\hat{x}_1$, as we assume that $1\le\sum_{j=1}^{J}\hat{x}_{ij}\le\frac{J}{2}\ \forall i\in\mathcal{J}^{SP}$ and $\hat{x}_1=\dots=\hat{x}_J=c_0-\hat{x}_0$.
- $C_5=\{1, 2, 3, 4\}$: the numbering $\eta\in C_5$ of the optimal price vectors $\hat{\boldsymbol{p}}$ that represent the degree of homogeneity of the

single products:

- if $\eta=1$, then $\hat{p}_j=65$ for all $j=1,\ldots,J$
- if $\eta=2$, then $\hat{p}_1=50$, $\hat{p}_J=80$, and $\hat{p}_j=65$ for all $j=2,\ldots,J-1$
- if $\eta=3$, then $\hat{p}_j=55$ for all $j=1,\ldots,\frac{J}{2}$, $\hat{p}_j=75$ for all $j=J/2+1,\ldots,J$
- if $\eta=4$, then $\hat{p}_1=60, \hat{p}_2=70$, if $J=2$, $\hat{p}_j=55+(j-1)\cdot 5 \quad \forall j=1,\ldots,J$, if $J=4$, and $\hat{p}_1=50$, $\hat{p}_j=\hat{p}_{j-1}+\{5|j \bmod 3\neq 0\}$ $\forall j=2,\ldots,J$, if $J=10$

- $C_6=\{1,3\}$: the number of price points $\delta\in C_6$ by which the optimal bundle price $\hat{p}_0$ is lower than the sum of the optimal prices of the single products. If that bundle price is not defined, the next higher price point is chosen: $\hat{p}_0=\min_{n\in\{1,\ldots,N_0\}}\{p_{0n}|p_{0n} \geq \sum_{j=1}^{J}\hat{p}_j-\delta\cdot d^B\}$. This may be the case in our tests as the price points of the bundle are derived by aggregating the corresponding price points of all of the single products $p_{0n}=\sum_{j=1}^{J}p_{jn}\forall n$, which implies that $d^B=J\cdot d^{SP}$.
- $C_7=\{0.25,\ 0.5,\ 0.75\}$: the percentage of the initial product-specific capacity $\rho\in C_7$ that is sold as a bundle: $\hat{x}_0=\rho\cdot c_0$. Consequently, $(1-\rho)$ corresponds to the percentage of the initial capacity that is sold individually.

Using an algorithm that adheres to all of the optimality constraints presented in Section 5, we generate $\tilde{Q}=5$ scenarios $(q=1,\ldots,\tilde{Q})$ with randomly distributed purchase decisions $\hat{\boldsymbol{x}}$ according to $\hat{x}_j$ and random reservation prices v_{ij} for each problem class C. Hence, our experimental study consists of 2160 scenarios.

Each scenario is solved using a variety of VNS specifications, as shown in Table 5. Each VNS method is performed in combination with the three neighbourhood schemes presented in Section 4.2 and the two construction heuristics (CH 1 and CH 2) described in Section 4.3; we also create initial solutions at random. For the maximum neighbourhood structure, values $k_{max}\in\{2,\ 3\}$ are used, and for RVNS (BVNS), stopping criterions are defined by $it_{max}\in\{10^2,\ 10^4\}$ ($it_{max}\in\{5,\ 20\}$). Additionally, for VND and BVNS, the local search is performed in the modes BI and FI, respectively. In total, we obtain 144 VNS specifications, and hence, the experimental study consists of 311040 data sets.

The entire study is performed on a desktop PC with an Intel Core i7 CPU at 2.80 GHz, 8 GB RAM, and Windows 7 Professional. The algorithms generating the scenarios and the VNS algorithms are coded in VB.NET using the .NET Framework 4.0.

Table 5
The VNS specifications.

Method	Scheme	Initial solution	k_{max}	it_{max}	Local search
VND	1	CH 1	2	–	BI
	2	CH 2	3		FI
	3	Random			
RVNS	1	CH 1	2	10^2	–
	2	CH 2	3	10^4	
	3	Random			
BVNS	1	CH 1	2	5	BI
	2	CH 2	3	20	FI
	3	Random			

6.2. Performance of the different VNS specifications

To evaluate the performance of the different VNS specifications, the following measures are used:

- av.dev: *average relative deviation* from optimality across Q scenarios, which is computed by $\text{av.dev} := \frac{1}{Q}\sum_{q=1}^{Q}\frac{|r_q-\hat{r}_q|}{\hat{r}_q}$ (quoted in %). Here, r_q denotes the objective function value for scenario q obtained by a certain VNS specification, whereas its optimal objective function value is given by $\hat{r}_q=\sum_{j=0}^{J}\hat{p}_j\hat{x}_j$ depending on the problem class C.
- max.dev: *maximum relative deviation* from optimality across Q scenarios, which is calculated by $\text{max.dev}:=\max_{q\in\{1,\ldots,Q\}}\left\{\frac{|r_q-\hat{r}_q|}{\hat{r}_q}\right\}$ (quoted in %). Analogously, the *minimum relative deviation* (min.dev) is defined.
- av.cpu: *average runtime* for solving Q scenarios (quoted in seconds).
- frac.opt: fraction of Q scenarios that is solved to optimality (i.e., where $r_q=\hat{r}_q$), which is computed by $\text{frac.opt} := \frac{1}{Q}\sum_{q=1}^{Q}\{1|r_q=\hat{r}_q\}$ (quoted in %).

Table 6 depicts the values of av.dev obtained by each VNS specification. The max.dev is shown in squared brackets, whereas the min.dev is omitted as it equals 0.0 in each case. The bold numbers indicate the best five VNS specifications in terms of the av.dev. In the following, we analyse the impact of the initial solutions, the iteration limit it_{max}, the maximum neighbourhood structure k_{max}, the neighbourhood schemes, and the mode of the local search with regard to the performance of VND, RVNS, and BVNS, respectively.

First, the results reveal that the way the initial solutions are generated has a significant impact on the performance of each VNS method. It is obvious that using problem-specific CHs pays off. The high values of max.dev, especially with randomly generated initial solutions, are evidence that the objective function is characterised by a multitude of local optima that in some cases cannot be overcome even with an enlarged neighbourhood using $k_{max}=3$, given the settings that we analyse. With the exception of RVNS with a low maximum number of iterations ($it_{max}=10^2$), in most of the cases, the methods yield the best av.dev using CH 1.

For VNS methods incorporating a stochastic component, an increase of it_{max} reduces the values of av.dev significantly. In particular, for BVNS (RVNS) the av.dev can be decreased on average by 19.3% (40.5%). Similar results can be derived by comparing the av.dev with regard to the different values of k_{max}. In all but one of the cases, a greater neighbourhood structure yields a reduction in the av.dev. Specifically, the av.dev can be reduced on average by 30.6% with VND, by 40.5% with RVNS, and by 27.2% with BVNS. Again, these results demonstrate that it is necessary to conduct large ‘jumps’ in local optima to search the non-smooth solution space. This also shows that VNS-based metaheuristics are able to perform considerably well in the bundle pricing context provided large values for it_{max} and k_{max} are selected.

In contrast, we cannot observe a clear dominance of one of the local search modes. The advantageousness of BI or FI depends on both the VNS method and the generation of the initial solutions. With VND, neither BI nor FI is dominant in the sense that neither has values of av.dev that are consistently lower compared with the other mode. The statement is also true for BVNS in combination with randomly created initial solutions. However, when applying BVNS together with CH 1 or CH 2, the BI local search always yields better values of av.dev, which are 51.5% lower compared with FI.

After comparing the effects of the different neighbourhood schemes, it is apparent that scheme 3 is always best for VND,

Table 6
The av.dev [max.dev] for each VNS specification ($Q = 2160$ scenarios each).

Initial solution	k_{max}	Scheme	VND – BI	VND – FI	RVNS $it_{max}=10^2$ -	RVNS $it_{max}=10^4$ -	BVNS $it_{max}=5$ BI	BVNS $it_{max}=5$ FI	BVNS $it_{max}=20$ BI	BVNS $it_{max}=20$ FI
CH 1	2	1	0.37 [28.97]	0.44 [28.97]	3.15 [28.97]	0.37 [28.97]	0.25 [27.93]	0.58 [28.97]	0.19 [26.88]	0.46 [28.97]
		2	0.53 [27.24]	0.44 [27.24]	1.51 [27.24]	0.37 [27.24]	0.28 [26.88]	0.48 [28.97]	0.21 [26.88]	0.36 [27.24]
		3	0.31 [27.24]	0.30 [27.24]	2.05 [27.10]	0.27 [27.10]	0.26 [27.10]	0.47 [28.97]	0.20 [26.88]	0.33 [27.24]
	3	1	0.22 [26.88]	0.29 [26.88]	1.55 [28.97]	0.24 [26.88]	**0.15 [26.88]**	0.42 [28.97]	**0.10 [25.63]**	0.30 [27.93]
		2	0.48 [27.24]	0.38 [27.24]	0.53 [27.24]	0.32 [27.29]	0.24 [25.73]	0.39 [28.73]	0.17 [26.88]	0.28 [27.24]
		3	0.16 [26.88]	0.15 [26.88]	0.68 [26.88]	0.16 [26.88]	0.18 [26.88]	0.38 [27.93]	**0.11 [25.63]**	0.22 [27.10]
CH 2	2	1	0.38 [28.97]	0.49 [28.97]	1.01 [28.97]	0.41 [28.97]	0.29 [28.97]	0.61 [28.97]	0.22 [26.88]	0.58 [28.97]
		2	0.53 [27.24]	0.43 [27.24]	1.08 [27.29]	0.42 [27.10]	0.27 [27.24]	0.54 [28.97]	0.25 [27.10]	0.41 [27.24]
		3	0.31 [27.24]	0.30 [27.24]	0.89 [27.10]	0.30 [27.24]	0.28 [27.24]	0.52 [28.97]	0.23 [26.88]	0.49 [28.97]
	3	1	0.23 [26.88]	0.35 [26.88]	0.43 [28.97]	0.33 [26.88]	**0.13 [25.63]**	0.46 [28.60]	0.15 [26.88]	0.40 [27.93]
		2	0.47 [27.24]	0.38 [27.24]	0.45 [27.24]	0.36 [27.10]	0.21 [27.10]	0.40 [28.97]	0.17 [25.73]	0.31 [27.24]
		3	0.16 [26.88]	0.16 [26.88]	0.27 [27.10]	0.17 [26.88]	0.20 [27.24]	0.39 [27.24]	**0.11 [25.63]**	0.31 [27.93]
Random	2	1	5.13 [77.10]	5.14 [78.29]	5.59 [83.02]	4.72 [83.75]	3.53 [76.25]	3.37 [76.33]	3.29 [75.23]	2.89 [77.49]
		2	4.22 [88.39]	4.57 [94.12]	4.98 [93.74]	4.49 [87.18]	3.04 [75.82]	3.35 [75.67]	2.47 [75.38]	2.78 [75.23]
		3	3.53 [79.25]	3.25 [75.53]	4.35 [78.37]	3.02 [75.47]	3.88 [77.31]	3.62 [76.79]	2.74 [75.34]	2.88 [77.16]
	3	1	4.35 [76.66]	5.01 [83.02]	4.76 [79.08]	3.87 [82.41]	3.05 [76.57]	3.01 [79.38]	3.12 [75.82]	3.10 [76.72]
		2	2.51 [76.61]	2.65 [79.35]	2.83 [90.20]	2.48 [95.43]	2.21 [75.35]	2.63 [75.72]	2.09 [75.00]	2.08 [75.18]
		3	2.10 [75.42]	2.41 [75.65]	2.35 [75.35]	2.11 [75.07]	2.36 [75.15]	2.43 [75.19]	2.07 [75.51]	2.11 [75.21]

Table 7
The av.cpu for each VNS specifications ($Q = 2160$ scenarios each).

Initial solution	k_{max}	Scheme	VND – BI	VND – FI	RVNS $it_{max}=10^2$ -	RVNS $it_{max}=10^4$ -	BVNS $it_{max}=5$ BI	BVNS $it_{max}=5$ FI	BVNS $it_{max}=20$ BI	BVNS $it_{max}=20$ FI
CH 1	2	1	0.21	0.14	0.03	6.25	0.24	0.16	0.36	0.25
		2	0.20	0.13	0.04	6.23	0.24	0.17	0.36	0.27
		3	0.21	0.14	0.04	6.25	0.24	0.16	0.36	0.26
	3	1	0.27	0.20	0.05	12.60	0.27	0.18	0.48	0.33
		2	0.20	0.13	0.05	12.49	0.27	0.20	0.46	0.37
		3	0.34	0.26	0.05	12.48	0.27	0.19	0.47	0.35
CH 2	2	1	0.11	0.08	0.03	6.39	0.14	0.10	0.18	0.13
		2	0.10	0.07	0.03	6.42	0.14	0.11	0.18	0.14
		3	0.13	0.10	0.03	6.43	0.14	0.10	0.18	0.13
	3	1	0.18	0.15	0.04	13.06	0.18	0.13	0.25	0.18
		2	0.11	0.08	0.04	13.00	0.18	0.13	0.25	0.20
		3	0.33	0.24	0.04	12.98	0.18	0.13	0.25	0.19
Random	2	1	0.06	0.06	0.03	6.44	0.09	0.08	0.13	0.11
		2	0.05	0.05	0.03	6.42	0.09	0.08	0.13	0.12
		3	0.07	0.07	0.03	6.42	0.09	0.08	0.13	0.11
	3	1	0.13	0.13	0.04	12.97	0.13	0.10	0.19	0.15
		2	0.05	0.05	0.04	12.86	0.12	0.11	0.19	0.17
		3	0.20	0.18	0.04	12.85	0.13	0.11	0.19	0.16

regardless of any other attributes. The same is true for almost all of the specifications of RVNS, except when $it_{\max} = 10^2$ in combination with CH 1. The use of the most complex neighbourhood scheme (scheme 3) allows for considering a greater set of neighbour solutions in each iteration compared with the two other schemes, which seems to pay off in this context. For BVNS, we observe that none of the schemes is superior in combination with all of the specifications. The solution quality of BVNS appears to not be heavily dependent on the underlying transformation rule that is used to define a neighbourhood structure, i.e., on the neighbourhood scheme. However, Table 6 also reveals that the differences in av.dev are relatively small and that, in most cases, scheme 3 yields almost the same average deviations as the best of the three schemes.

Table 7 summarises the values of av.cpu for each VNS specification across $Q = 2160$ scenarios. First, in most cases, the av.cpu is slightly lower if the initial solutions are generated at random. In the

Table 8
The statistics with regard to the five best VNS specifications ($Q = 2160$ scenarios each).

Method	Initial solution	it_{max}	k_{max}	Local search	Scheme	av. dev	99% quantile	max. dev	frac. opt
BVNS	CH 1	20	3	BI	1	0.10	2.04	25.63	97.08
BVNS	CH 1	20	3	BI	3	0.11	1.48	25.63	98.56
BVNS	CH 2	20	3	BI	3	0.11	2.04	25.63	97.31
BVNS	CH 2	5	3	BI	1	0.13	2.27	25.63	96.02
BVNS	CH 1	5	3	BI	1	0.15	2.34	26.88	96.30

event that the incumbent solution cannot be improved any more within the maximum number of iterations, the VNS method is terminated. On average, this occurs earlier when no problem-specific knowledge is incorporated into creating the first solution. This finding is consistent with the analysis of the av.dev (cf. Table 6). Unsurprisingly, the av.cpu increases with the enhancement of it_{max} and k_{max}. Raising the latter from $k_{max} = 2$ to 3 increases the av.cpu by 52.9% on average. With respect to it_{max}, the av.cpu increases above average with RVNS (by a factor of 2.51 on average), whereas the opposite is true with BVNS (by a factor of 0.37 on average). An interesting observation is that the differences in the av.cpu that result from the three neighbourhood schemes are almost negligible. Hence, we can conclude that the number of neighbours in a neighbourhood does not significantly influence the solution time. However, this result might change if it_{max} or k_{max} are further incremented. Additionally, the use of BI instead of FI increases the av.cpu by up to 51.2% (on average, 29.8%). In sum, with the exception of RVNS with a high maximum iteration limit, each VNS specification has acceptable solution runtimes. As the differences are relatively low in terms of the absolute values of av.cpu, a decision for a certain VNS specification can mainly be made based on the expected av.dev.

Therefore, in the following subsection, we analyse the influence of the different deterministic parameters of the scenarios with regard to the solution quality of the five VNS specifications that perform best on average across all problem classes in terms of av.dev (depicted in bold in Table 6). All of these specifications have the use of BVNS in combination with the maximum neighbourhood structure $k_{max} = 3$ and a BI local search in common.

6.3. Solution quality of the five best VNS specifications

First, we consider the more detailed statistics of the five best VNS specifications shown in Table 8. In addition to the av.dev and the max.dev, the 99% quantile of the av.dev and the frac.opt are depicted. Each of the five specifications is able to solve at least 96.0% of the 2160 scenarios to optimality, which implies that the positive value of av.dev is caused by only 4.0% of the scenarios at most. Furthermore, we can observe that out of these scenarios, 31–75% can be solved with an average deviation from optimality that is lower than 1.48–2.34% depending on the specification. These values are exceeded in only 1.0% of all of the scenarios. Thus, the overall results are very satisfying.

To analyse in more detail which problem classes are likely to cause a deviation of the heuristic solutions from optimality, the av.dev and the frac.opt (in squared brackets) are reported as a function of sf and $\hat{x}_0$ in Table 9. The results demonstrate that the av.dev increases with the number of bundles that are sold, i.e., the higher $\hat{x}_0$ is. Although each scenario with $sf = 2$ and $\hat{x}_0 \leq 50$ can either be solved optimally or with an av.dev that is almost zero, this result does not hold for $\hat{x}_0 = 75$. If $sf = 5$ this effect intensifies as only for scenarios with $\hat{x}_0 = 25$ the solutions show almost no deviation from optimality. Simultaneously, the results demonstrate that in most cases a rising number of consumers have a negative impact on the values of av.dev and frac.opt. In summary, the values of these two parameters, sf and $\hat{x}_0$, influence the solution quality in the same way; the effect is enforced if they change simultaneously.

Assuming a direct relationship between the perception of the products' quality and the corresponding price, we account for various degrees of homogeneity of the single products by analysing different sets of optimal price vectors. Although depending on the number of products, the parameter $\eta = 1$ represents homogeneous single products, and $\eta \neq 1$ represents the more heterogeneous single products. Table 10 displays the av.dev and the frac.opt (in squared brackets) as a function of the degree of homogeneity. On the basis of these results, we can conclude that we tend to obtain better solutions with our heuristics the more heterogeneous the single products are. While the av.dev in problem classes with $\eta = 1$ averages up to 0.35%, the results of the problem classes with $\eta \in \{2, 3, 4\}$ are significantly better and close to optimality. However, the values of frac.opt indicate that altogether the values of η have little effect on the solution quality of the VNS methods. For each of the five best VNS specifications, some scenarios cannot be optimally solved independent of the degree of homogeneity.

Finally, Table 11 shows the values of av.dev and frac.opt (in squared brackets) as a function of J and N'. Regardless of N', each scenario with $J \in \{2, 4\}$ can either be solved optimally or be solved with an av.dev close to zero. In problem classes with $J = 10$, this is not possible for up to 16.25% of the scenarios. Here, a larger value of k_{max} probably helps to overcome the local optima. Furthermore, in these cases, it is apparent that the av.dev decreases as N' becomes larger. However, this result is not surprising as the differences between two price points are greater when N' is low; hence, a deviation from the pre-selected optimal price vector has a greater impact.

6.4. Managerial insights to the bundling problem with capacity constraints

To gain managerial insights and analyse the general benefit of the different standard sales strategies mixed bundling (MB), pure bundling (PB), and unbundling (UB) in the context of price optimisation facing capacity constraints further computational tests have been conducted. For this purpose, we have generated random scenarios of which the optimal solutions are not known in advance. Following Burkart et al. (2012), consumers' reservation prices are Beta (α, β)-distributed in the interval [0, 100] with $j = 1$ representing the high value product ($Beta(4, 5)$), $j = J$ representing the low value product ($Beta(2, 7)$), and $j = 2, \ldots, J-1$ are valued in between ($Beta(4, 7)$). Given the excellent solution quality of our metaheuristics, we optimise over the stochastic counterpart (SC) of $S = 30$ scenarios for each problem class using the best three VNS specifications with $it_{max} = 20$ as indicated in Table 8 (the set of VNS specifications is referred to as C_{VNS}). The problem classes are characterised by the following parameters: $J \in C'_1 = \{2, 4, 20\}$, $c_0 = c_{01} = \ldots = c_{0J} \in C'_2 = \{80, 100, 120\}$, $I \in C'_3 = \{80, 150, 200\}$, and $N' \in C'_4 = \{10, 20, 40\}$. In addition to the figures introduced in Section 6.2, we use the following measures:

- av.uti: *average relative utilisation* across Q' SCs each optimised by a certain VNS specification, which is computed by av.uti$:= \frac{1}{Q'} \sum_{q'=1}^{Q'} \frac{\bar{x}_{q'}}{J c_0^{q'}}$ (quoted in %). Here, $\bar{x}_{q'}$ denotes the average number of sold products of SC q' ($\bar{x}_{q'} = \frac{1}{S} \sum_{s'=1}^{S} \{ \sum_{i=1}^{I} \sum_{n=1}^{N'} J x_{i0n}^{s'q'} + \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{n=1}^{N'} x_{ijn}^{s'q'} \}$ with $x_{ijn}^{s'q'} = 1$ if consumer i chooses product j at price point p_{jn} in scenario s' of SC q'),

Table 9
The av.dev [frac.opt] as a function of sf and $\hat{x}_0$ ($Q=360$ scenarios each).

Method	Initial solution	it_{max}	k_{max}	Local search	Scheme	$sf=2$			$sf=5$		
						$\hat{x}_0=25$	$\hat{x}_0=50$	$\hat{x}_0=75$	$\hat{x}_0=25$	$\hat{x}_0=50$	$\hat{x}_0=75$
BVNS	CH 1	20	3	BI	1	0.00 [99.72]	0.00 [100.0]	0.31 [95.28]	0.00 [99.72]	0.09 [94.44]	0.21 [93.33]
BVNS	CH 1	20	3	BI	3	0.00 [100.0]	0.00 [100.0]	0.36 [96.39]	0.00 [100.0]	0.05 [97.78]	0.23 [97.22]
BVNS	CH 2	20	3	BI	3	0.00 [99.44]	0.00 [100.0]	0.37 [95.56]	0.00 [99.44]	0.08 [95.00]	0.20 [94.44]
BVNS	CH 2	5	3	BI	1	0.00 [97.50]	0.00 [100.0]	0.39 [94.44]	0.00 [98.61]	0.12 [93.63]	0.29 [92.50]
BVNS	CH 1	5	3	BI	1	0.00 [99.72]	0.00 [100.0]	0.42 [93.33]	0.00 [99.17]	0.14 [92.50]	0.34 [93.06]

Table 10
The av.dev [frac.opt] as a function of η ($Q=540$ scenarios each).

Method	Initial solution	it_{max}	k_{max}	Local search	Scheme	$\eta=1$	$\eta=2$	$\eta=3$	$\eta=4$
BVNS	CH 1	20	3	BI	1	0.26 [96.30]	0.07 [98.15]	0.05 [96.67]	0.03 [97.22]
BVNS	CH 1	20	3	BI	3	0.31 [97.78]	0.07 [98.52]	0.02 [98.52]	0.01 [99.44]
BVNS	CH 2	20	3	BI	3	0.30 [96.11]	0.07 [97.59]	0.03 [97.59]	0.03 [97.96]
BVNS	CH 2	5	3	BI	1	0.35 [95.19]	0.08 [97.22]	0.05 [95.56]	0.06 [96.11]
BVNS	CH 1	5	3	BI	1	0.35 [95.93]	0.10 [97.04]	0.05 [96.30]	0.11 [95.03]

Table 11
The av.dev [frac.opt] as a function of J and N' ($Q=240$ scenarios each).

Method	Initial solution	it_{max}	k_{max}	Local search	Scheme		$N'=7$	$N'=31$	$N'=61$
BVNS	CH 1	20	3	BI	1	$J=2$	0.00 [100.0]	0.00 [100.0]	0.00 [99.17]
						$J=4$	0.00 [100.0]	0.00 [100.0]	0.00 [100.0]
						$J=10$	0.68 [93.33]	0.20 [86.25]	0.06 [95.00]
BVNS	CH 1	20	3	BI	3	$J=2$	0.00 [100.0]	0.00 [100.0]	0.00 [100.0]
						$J=4$	0.00 [100.0]	0.00 [100.0]	0.00 [100.0]
						$J=10$	0.88 [92.08]	0.06 [95.83]	0.01 [99.17]
BVNS	CH 2	20	3	BI	3	$J=2$	0.00 [100.0]	0.00 [100.0]	0.00 [98.33]
						$J=4$	0.00 [100.0]	0.00 [100.0]	0.00 [100.0]
						$J=10$	0.80 [92.08]	0.14 [89.58]	0.04 [95.83]
BVNS	CH 2	5	3	BI	1	$J=2$	0.00 [100.0]	0.00 [100.0]	0.00 [94.17]
						$J=4$	0.00 [100.0]	0.00 [100.0]	0.00 [100.0]
						$J=10$	0.90 [91.67]	0.24 [83.75]	0.06 [94.58]
BVNS	CH 1	5	3	BI	1	$J=2$	0.00 [100.0]	0.00 [100.0]	0.00 [98.75]
						$J=4$	0.00 [100.0]	0.01 [99.58]	0.00 [100.0]
						$J=10$	1.04 [90.42]	0.23 [85.75]	0.07 [94.17]

Table 12
The av.uti and the av.uti^B as a function of I/c_0 for the mixed bundling strategy. ($Q'=27$ SCs each).

Measure	$\frac{I}{c_0}=0.67$	$\frac{I}{c_0}=0.80$	$\frac{I}{c_0}=1.00$	$\frac{I}{c_0}=1.25$	$\frac{I}{c_0}=1.50$	$\frac{I}{c_0}=1.67$	$\frac{I}{c_0}=1.88$	$\frac{I}{c_0}=2.00$	$\frac{I}{c_0}=2.50$
av.uti	56.4	67.8	84.7	94.4	96.2	98.3	98.0	98.1	94.1
av.uti^B	56.2	67.6	84.4	84.5	76.0	73.5	68.6	65.2	36.9

whereas the total initial capacity in SC q' is given by $Jc_0^{q'}$. Analogously, av.uti^B describes the average relative utilisation caused by bundle sales, i.e., $\overline{x}_{q'}^B = \frac{1}{S}\sum_{s'=1}^{S}\left\{\sum_{i=1}^{I}\sum_{n=1}^{N'} Jx_{i0n}^{s'q'}\right\}$.

- av.rev: *average revenue* across Q' SCs, which is calculated by $\text{av.rev} := \frac{1}{Q'}\sum_{q'=1}^{Q'} TR_{q'}$; $TR_{q'}$ denotes the average revenue of the S scenarios in SC q'.
- *av.p$_j$: average price for product j*$\in\{1,\ldots,J\}$ across Q' SCs, which is computed by $\text{av.}p_j := \frac{1}{Q'}\sum_{q'=1}^{Q'} p_j^{q'}$ with $p_j^{q'}$ representing the (unique) price for product j in SC q'.

Table 12 depicts the values of av.uti and av.uti^B as a function of the relative number of consumers I/c_0 for the mixed bundling case. Each value is calculated as an average across $Q'=|C_{VNS}|\cdot|C_1'|\cdot|C_4'|=27$ SCs. Unsurprisingly, the total utilisation ratio tends to increase with a rising number of consumers who are interested in the services that are offered. However, it is noticeable that the fraction of products that are sold as a bundle is highly dependent on the relative number of consumers. As long as capacity is not limited ($I/c_0 \leq 1.0$) the optimal prices are chosen in such a way that almost no single products are sold. Yet, the more consumers appear relative to the initial capacity of the services the more single products and the less bundles are sold — both in absolute terms and relatively to c_0.

Besides applying the MB strategy we also analysed the impact of implementing a PB or UB strategy for the same problem classes

Table 13
The av.rev, $av.p_j$, and av.uti as a function of c_0 for each sales strategy. ($Q' = 81$ SCs each).

Strategy	c_0	av.$\frac{I}{c_0}$	Revenue			Prices			Utilisation	
			av.rev	Δ rel. to MB	Δ rel. to $c_0 = 100$	av.p_0	av.p_1	av.p_2	av. uti	av. uti^B
MB	80	1.8	22235	0.0	−10.8	273.8	53.0	41.3	92.3	63.3
	100	1.4	24643	0.0	0.0	260.5	51.7	41.6	87.4	69.6
	120	1.2	26941	0.0	9.3	251.4	51.5	41.5	83.1	71.4
PB	80	1.8	21619	−2.8	−15.1	276.5			94.1	94.1
	100	1.4	24893	1.0	0.0	274.6			87.4	87.4
	120	1.2	27973	3.8	12.4	267.0			83.4	83.4
UB	80	1.8	20176	−9.3	−8.7		40.9	29.4	85.5	0.0
	100	1.4	21922	−11.0	0.0		38.3	27.6	79.2	0.0
	120	1.2	22985	−14.7	4.8		35.6	26.5	72.4	0.0

(cf. Table 13). Here, each value represents the average across $Q' = |C_{VNS}| \cdot |C'_1| \cdot |C'_3| \cdot |C'_4| = 81$ SCs. The results demonstrate that in terms of av.rev UB is always worse compared to MB (cf. the column Δ rel. to MB, which displays the difference of the values of av.rev relative to the respective value of the MB strategy for each c_0 in %). The strategy of selling the products only as a bundle (PB), however, can pay off in case the relative number of consumers is low compared to the initial capacity, i.e., as long as $I/c_0 \le 1.5$. This finding is in accordance with the results of Table 12. The greater the importance of capacity constraints is, the more profitable it is to sell the products both individually and as a bundle.

Looking at the optimal prices, it becomes apparent that in the PB strategy p_0 can be increased by up to 6.2% on average compared to MB, while the values of av.uti remain almost constant. In contrast, the implementation of a UB strategy requires to reduce the prices by up to 36.1% on average to maximise revenue (Table 13 exemplarily presents the values of av.p_j for $j = 1$ and $j = 2$). Here, a significant reduction of av.uti is inevitable.

Finally, increasing the capacity by 20% to 120 while keeping the number of consumers unchanged results in an average growth of revenues by 9.3% in MB, 12.4% in PB, and 4.8% in UB in our tests (cf., the column Δ rel. to $c_0 = 100$, which displays the difference of the values of av.rev relative to the respective values of the problem classes with $c_0 = 100$ for each strategy in %). A reduction of the capacity by 20%, in contrast, causes an economic slump by 10.8% in MB, 15.1% in PB, and 8.7% in UB.

7. Summary and conclusion

In this paper, we have addressed the bundle pricing problem for services with constrained capacity. The seller's objective is to determine revenue-maximising static prices for both the single products and the corresponding bundle. To adequately address this problem, we have presented two novel mixed-integer linear programmes that are based on products and alternatives, respectively. Both models explicitly account for dynamic substitution effects within the consumer choice behaviour that are caused by a successive reduction of the available capacity. To the best of our knowledge, this problem has not been studied in this context by any other research to date.

As the exact solution of the models is computationally expensive except for with regard to small problem classes, we have developed several metaheuristics that are based on variable neighbourhoods to solve the problem. To evaluate the performance of our heuristics with respect to their solution quality and runtimes, we have proposed a new approach: We have generated scenarios in such a way that the optimal solutions are known and, hence, can serve as a benchmark. For this purpose, we have derived a set of necessary conditions for the generation of the consumers' reservation prices that guarantee optimality. On the basis of these scenarios, extensive computational experiments have been conducted.

The best results have been obtained using metaheuristics, which combine deterministic and stochastic components. We have discovered that more than 98.5% of all scenarios, which represent a wide variety of real world problem classes, can be solved to optimality. However, in single scenarios of various problem classes, the solution quality of our heuristics still deviates from optimality despite using a successively enlarged neighbourhood during the search. Hence, future research should identify the specifications of the metaheuristics that also allow these local optima to be overcome. Apart from that, tests using general scenarios have revealed that the ratio of consumers to the initial capacity accounts for the fraction of products that is sold as a bundle. A higher total demand causes fewer bundle sales, whereas, only in case the demand is significantly low, pure bundling can be an alternative to mixed bundling.

References

Adams, W.J., Yellen, J.L., 1976. Commodity bundling and the burden of monopoly. The Quarterly Journal of Economics 90 (3), 475–498.
Anderson, S.P., Leruth, L., 1993. Why firms may prefer not to price discriminate via mixed bundling. International Journal of Industrial Organization 11 (1), 49–61.
Bakos, Y., Brynjolfsson, E., 1999. Bundling information goods: Pricing, profits, and efficiency. Management Science 45 (12), 1613–1630.
Bitran, G.R., Ferrer, J.–C., 2007. On pricing and composition of bundles. Production and Operations Management 16 (1), 93–108.
Bulut, Z., Gürler, Ü., Sen, A., 2009. Bundle pricing of inventories with stochastic demand. European Journal of Operational Research 197 (3), 897–911.
Burkart, W.R., Klein, R., Mayer, S., 2012. Product line pricing for services with capacity constraints and dynamic substitution. European Journal of Operational Research 219 (2), 347–359.
Chen, Y., 1997. Equilibrium product bundling. The Journal of Business 70 (1), 85–103.
Gensler, S., Hinz, O., Skiera, B., Theysohn, S., 2012. Willingness-to-pay estimation with choice-based conjoint analysis: addressing extreme response behavior with individually adapted designs. European Journal of Operational Research 219 (2), 368–378.
Gürler, Ü., Öztop, S., Sen, A., 2009. Optimal bundle formation and pricing of two products with limited stock. International Journal of Production Economics 118 (2), 442–462.
Hansen, P., Mladenović, N., Brimberg, J., Moreno Pérez, J.A., 2010a. Variable neighborhood search. In: Gendreau, M., Potvin, J.-Y. (Eds.), Handbook of metaheuristics. Springer, New York, pp. 61–85.
Hansen, P., Mladenović, N., Moreno Pérez, J.A., 2010b. Variable neighbourhood search: methods and applications. Annals of Operations Research 175 (1), 367–407.
Hanson, W., Martin, R., 1990. Optimal bundle pricing. Management Science 36 (2), 155–174.
Hitt, L.M., Chen, P., 2005. Bundling with customer self-selection: a simple approach to bundling low-marginal-cost goods. Management Science 51 (10), 1481–1493.
Jedidi, K., Zhang, Z.J., 2002. Augmenting conjoint analysis to estimate consumer reservation price. Management Science 48 (10), 1350–1368.
Jedidi, K., Jagpal, S., Manchanda, P., 2003. Measuring heterogeneous reservation prices for product bundles. Marketing Science 22 (1), 107–130.
Kleywegt, A.J., Shapiro, A., Homem-De-Mello, T., 2002. The sample average approximation method for stochastic discrete optimization. SIAM Journal of Optimization 12 (2), 479–502.
Kopalle, P.K., Krishna, A., Ao, J.L.A., 1999. The role of market expansion on equilibrium strategies. Managerial and Decision Economics 20 (7), 365–377.
Li, J., Tang, O., 2012. Capacity and pricing policies with consumer overflow behavior. International Journal of Production Economics 140 (2), 825–832.
Liao, C.-H., Tauman, Y., 2002. The role of bundling in price competition. International Journal of Industrial Organization 20 (3), 365–389.
Matutes, C., Regibeau, P., 1988. 'Mix and match': product compatibility without network externalities. The RAND Journal of Economics 19 (2), 221–234.
Matutes, C., Regibeau, P., 1992. Compatibility and bundling of complementary goods in a duopoly. The Journal of Industrial Economics 40 (1), 37–54.
Mayer, S., Gönsch, J., 2012. Consumer choice modelling in product line pricing: reservation prices and discrete choice theory. In: Klatte, D., Lüthi, H.-J., Schmedders, K. (Eds.), Operations Research Proceedings 2011. Springer, Berlin, pp. 61–85.
McAfee, R.P., McMillan, J., Whinston, M.D., 1989. Multiproduct monopoly, commodity bundling, and correlation of values. The Quarterly Journal of Economics 104 (2), 371–383.

Mladenović, N., Hansen, P., 1997. Variable neighborhood search. Computers & Operations Research 24 (11), 1097–1100.

Nalebuff, B., Majerus, D., 2003. Bundling, tying, and portfolio effects—part 1: conceptual issues. DTI Economics Paper No. 1, Technical report. Department of Trade and Industry, London, UK.

Olderog, T., Skiera, B., 2000. The benefits of bundling strategies. Schmalenbach Business Review 52 (2), 137–159.

Salinger, M.A., 1995. A graphical analysis of bundling. The Journal of Business 68 (1), 85–98.

Schlereth, C., Eckert, C., Skiera, B., 2012. Using discrete choice experiments to estimate willingness-to-pay intervals. Marketing Letters 23 (3), 761–776.

Schmalensee, R., 1984. Gaussian demand and commodity bundling. The Journal of Business 57 (1), S211–S230.

Stigler, G.J., 1963. United States vs. Loew's Inc: a note on block-booking. Supreme Court Review, 152–157.

Stremersch, S., Tellis, G.J., 2002. Strategic bundling of products and prices: a new synthesis for marketing. Journal of Marketing 66 (1), 55–72.

Talluri, K.T., van Ryzin, G.J., 2004. The theory and practice of revenue management. Springer, New York.

Venkatesh, R., Kamakura, W., 2003. Optimal bundling and pricing under a monopoly: contrasting complements and substitutes from independently valued products. Journal of Business 76 (2), 211–231.

Venkatesh, R., Mahajan, V., 1993. A probabilistic approach to pricing a bundle of products or services. Journal of Marketing Research 40 (4), 494–508.

Voelckner, F., 2006. An empirical comparison of methods for measuring consumers' willingness to pay. Marketing Letters 17 (2), 137–149.

Wu, S., Hitt, L.M., Chen, P., Anandalingam, G., 2008. Customized bundle pricing for information goods: a nonlinear mixed-integer programming approach. Management Science 54 (3), 608–622.