Using Dynamic Programming Decomposition for Revenue Management with Opaque Products

Jochen Gönsch, Department of Analytics & Optimization, University of Augsburg, Germany, E-mail: jochen.goensch@wiwi.uni-augsburg.de

Claudius Steinhardt, Department of Analytics & Optimization, University of Augsburg, Germany, E-mail: claudius.steinhardt@wiwi.uni-augsburg.de

Abstract

Opaque products enable service providers to hide specific characteristics of their service fulfillment from the customer until after purchase. Prominent examples include internet-based service providers selling airline tickets without defining details, such as departure time or operating airline, until the booking has been made. Owing to the resulting flexibility in resource utilization, the traditional revenue management process needs to be modified. In this paper, we extend dynamic programming decomposition techniques widely used for traditional revenue management to develop an intuitive capacity control approach that allows for the incorporation of opaque products. In a simulation study, we show that the developed approach significantly outperforms other well-known capacity control approaches adapted to the opaque product setting. Based on the approach, we also provide computational examples of how the share of opaque products as well as the degree of opacity can influence the results.

1   Introduction

Over the past decade, the rapid development of e-commerce has opened up new possibilities, which have led to dramatic changes in how firms design, place, and price their products. In this paper, we focus on a sales strategy called opaque selling, which has become quite popular in service industries – especially in the travel industry. When selling an opaque product, the provider conceals some aspects of the offered service until the transaction has been completed. From a monopolistic point of view, opaque products can basically be seen as an additional instrument of price discrimination that allows the supplier to provide a discount in order to attract additional low-value customers, but without excessively cannibalizing or diluting existing demand for fully specified products. As an alternative to this

market expansion, the provider could offer only a few discounts and raise the regular products' prices in order to enhance the price discrimination of the existing customer base (e.g., Fay 2008).

The usage of opaque products has been shown to be successful in many practical applications. Specifically in markets in which business travelers often make late purchases and the provider usually has to resort to some kind of temporal segmentation with advanced purchase restrictions, opaque products can help effectively segment the market until the last minute before service provision without too much buy-down behavior. Segmentation is possible due to business travelers' reluctance to risk accepting the "randomness" inherent in opaque products (e.g., Jiang 2007). Indeed, opaque products are often regarded as a more efficient alternative to last-

minute selling (Jerath, Netessine, and Veeraraghavan 2010). Examples of opaque products in practice include airlines that allow customers to book a flight from A to B on a specific date, but conceal schedule information, such as the exact departure and arrival times, connections, transfers, and layover durations; hotel chains that conceal the specific hotel in which the customer will stay; and cruise lines that hide the cabin type or even the itinerary until after purchase.

Opaque products' popularity has been heavily driven by offers from intermediaries, such as Hotwire and Priceline, which multiple providers share. In addition to the concealed information described above, intermediaries usually hide the service providers' brand names (e.g., those of the airlines, rental firms, hotels, or cruise lines) when selling an opaque product. Many service providers currently tend to move customers of their traditional products back to company-managed direct distribution channels as well as to streamline prices and create price parity across these channels. Thus, the additional usage of a completely separate, anonymous channel via intermediaries – which hide the provider's identity by combining different providers' products into a single opaque product – allows providers to effectively post different prices simultaneously while maintaining price parity and control of full-information-posted price channels (e.g., Anderson 2009). Furthermore, a provider that does not produce multiple, horizontally differentiated regular products even needs to utilize an intermediary in order to successfully introduce an opaque product (Fay and Xie 2008).

There has been increasing interest in opaque products in the academic literature in the last few years. Most of the research on opaque selling with posted prices has concentrated on analyzing the general economic benefits of offering opaque products in a static setting and discussed how prices should be set compared to those of traditional products, mostly on the basis of stylized economic models. In this paper, we tackle the topic from a revenue management perspective in a dynamic setting based on capacity control, which is a key component of modern revenue management. Basically, capacity control is concerned with the task of optimally selling a network of perishable resources with a fixed capacity over time by dynamically making products defined on this network available or unavailable to customers (e.g., Talluri and van Ryzin 2004). As

soon as opaque products are introduced, traditional optimization models, which are used to perform automated capacity control, can no longer be applied due to the supplier-driven substitution possibilities inherent in opaque products.

Therefore, the contribution of this paper is the development of a new approach to capacity control, enabling the service provider to simultaneously offer both, opaque products and traditional ones. We show that this approach outperforms other well-known capacity control approaches adapted to the opaque product setting. Our approach builds on the well-known dynamic programming decomposition procedure; this procedure is not only state-of-the-art from a theoretical point of view, but also widely used for standard capacity control in commercial revenue management systems. Therefore, the results are relevant to both research and practice. Moreover, service providers who already apply advanced revenue management techniques and are considering introducing opaque products can implement this approach very easily. Based on the new approach, we are able to investigate some specific effects that arise in a dynamic setting when opaque products are introduced. In particular, we give examples of how the degree of opacity can influence the revenue performance, and that the share of opaque products obtained from total sales is a critical aspect with respect to overall revenue performance.

The paper is structured as follows: In section 2, we begin with a brief discussion of the existing literature on and related to our topic. We then present the basic network dynamic programming formulation for the capacity control problem, including opaque products (section 3). In section 4, we analytically show how to use dynamic programming decomposition in order to decompose the given capacity control problem. Moreover, we prove that important results known from standard capacity control also hold in the opaque setting. In section 5, the control mechanism resulting from the decomposition is extensively investigated in a computational study. We consider typical airline revenue management scenarios in order to show its practical applicability and its relative performance compared to other potential control approaches. In addition, we analyze some specific effects arising from the introduction of opaque products. In section 6, we conclude with a summary of the paper's main results.

2 Related literature

In the scientific literature, opaque products are usually discussed from a perspective that could best be described as situated at the interface of economics, marketing, and operations management. Most of the work is concerned with the task of finding optimal selling strategies, especially optimal prices, in scenarios that allow for opaque selling. In most cases, static settings with an unlimited capacity are considered and stylized economic models are developed.

Jiang (2007) investigated a monopolist's different strategies by comparing the settings solely with regular products, solely with opaque products, and both product types simultaneously. He showed that opaque selling can be pareto-improving for both, customers and provider if customers' valuations are quite differentiated. Furthermore, Fay and Xie (2008) incorporated capacity constraints and demand uncertainty. Fay (2008) investigated how product opacity affects the market in a competitive environment and analyzed a common intermediary's impact. Whereas he assumed there are two service providers and one intermediary, Shapiro and Shi (2008) extended his setting to an arbitrary number of competing providers; the total number of providers can be regarded as a proxy for the opacity level of the opaque product that the common intermediary offers. The authors showed that although the opaque product's introduction increases competition for the low-price segment, competition for the market's more lucrative segment – which the providers serve directly by means of regular products – decreases. Jerath, Netessine, and Veeraraghavan (2009, 2010) investigated a related setting but considered two subsequent periods. They specifically addressed the question of whether opaque products should be offered via the intermediary or whether the direct last-minute selling of regular products should be used in the last period. Post (2010) as well as Post and Spann (2011) investigated variable opaque products, which allow the customer to configure the amount of opaqueness to a certain extent. However, there are only a few papers from the traditional revenue management stream of research that consider the impact that the introduction of opaque products has on established capacity control mechanisms. Talluri (2001) investigated an airline company whose customers are indifferent to the various itineraries serving the same market, as long as they are similar with regard to arrival/departure times

and price. The author suggested a deterministic model formulation of the problem and derived a bid price policy, assigning customers to specific itineraries immediately after booking. Chen, Günther, and Johnson (2003) investigated various approaches to capacity control in air cargo revenue management, where flexibility also emerges from different routing options.

The literature on revenue management with flexible products is also related to our work: Flexible products can be seen as a generalization of opaque products in the sense that the provider does not have to determine a flexible product's full specification immediately after the sale, but can postpone this decision even further, if necessary until shortly before service provision. Gallego and Phillips (2004) and Gallego, Iyengar, Phillips, and Dubey (2004) first introduced and investigated the concept of flexible products. Petrick, Steinhardt, Gönsch, and Klein (2012) further analyzed flexible products' potential to compensate for imprecise demand forecasts. Petrick, Gönsch, Steinhardt, and Klein (2010) developed several dynamic control mechanisms that allow for flexible products' practical integration into existing capacity control systems. The idea of revenue management with flexible products has been adapted to several fields of application (e.g., Bartodziej and Derigs 2004; Bartodziej, Derigs, and Zils 2007; Spengler, Rehkopf, and Volling 2007; Müller-Bungart 2007; Kimms and Müller-Bungart 2007).

Regarding the optimization technique used in this paper, the literature on dynamic programming decomposition approaches for revenue management is of particular interest. Talluri and van Ryzin (2004) described the standard decomposition approach for the traditional revenue management setting, which is used to split the full network dynamic program into a number of single-leg problems that are usually much easier to solve. This technique serves as a basis for the approach incorporating opaque selling, which is developed in this paper. A number of recent publications, for example, Liu and van Ryzin (2008), Miranda Bront, Méndez-Díaz, and Vulcano (2009), Zhang and Adelman (2009), Erdelyi and Topaloglu (2010), Kunnumkal and Topaloglu (2010), and Meissner and Strauss (2012) adapted dynamic programming decomposition to other settings, such as the choice-based network revenue management setting or overbooking settings. Finally, for a general introduction to the field of

revenue management, see the textbooks by Talluri and van Ryzin (2004) and Phillips (2005), as well as the surveys by Weatherford and Bodily (1992), McGill and van Ryzin (1999), as well as Chiang, Chen, and Xu (2007).

3   General notation and dynamic programming formulation

We rely on the standard revenue management setting as introduced, for example, by Talluri and van Ryzin (2004), which can be stated as follows: There are l resources indexed by $\mathcal{H} = \left\{1, \dots, l\right\}$ with c_h denoting the total remaining capacity for each $h \in \mathcal{H}$. The corresponding vector is denoted by $\mathbf{c} = \left(c_1, \dots, c_l\right)$. Furthermore, there are n products defined on $\mathcal{H}$ that are indexed by $\mathcal{J} = \left\{1, \dots, n\right\}$. The revenue obtained when selling one unit of product $j \in \mathcal{J}$ is denoted by r_j. Requests for the products arrive throughout a common booking horizon, with all units of capacity remaining after its end being worthless. The booking horizon can be sufficiently discretized into T periods, so that there is at most one incoming customer request for each period $t = 1, \dots, T$. The probability of an arrival of a request for a product indexed by j in period t is denoted by $p_j\left(t\right)$ and $p_0\left(t\right) = 1 - \sum_{j \in \mathcal{J}} p_j\left(t\right)$ is the probability of there being no incoming request. The periods are numbered backwards in time. Thus, $D_{jt} = \sum_{\tau=1}^{t} p_j\left(\tau\right)$ is the expected aggregated demand-to-come at the point in time t for a product indexed by j.

The difference resulting from the introduction of opaque products compared to the standard case lies in the definition of the resource consumption. There are now potentially several specification options $\mathcal{M}_j \subseteq \mathcal{I}$ the provider can choose from after selling an opaque product indexed by $j \in \mathcal{J}$. $\mathcal{I} \subseteq \mathbb{N}^+$ is the index set of all products' specification options, where the consumption of an option with index $i \in \mathcal{I}$ on a resource with index $h \in \mathcal{H}$ is expressed by the parameter a_{hi}. The corresponding vector is denoted by $\mathbf{a}_i = \left(a_{1i}, \dots, a_{li}\right)$. Note that any regular product indexed by j' can be modeled as a special case of an opaque product with $\left|\mathcal{M}_{j'}\right| = 1$. Subsequently, for the sake of simplicity, we refer to a resource indexed by h as "resource h". The same abbreviation is used for the other index sets.

Now, let $V\left(\mathbf{c}, t\right)$ be the maximum expected revenue-to-go from period t onward, assuming that resource capacity $\mathbf{c}$ is left over. Then, the dynamic programming formulation of the capacity control problem with opaque products can be stated as the following Bellman equation (**DP-op**):

(1)
$$
\begin{aligned}
V\left(\mathbf{c}, t\right) = \sum_{j \in \mathcal{J}} p_j\left(t\right) \max\Big(& V\left(\mathbf{c}, t-1\right), \\
& r_j + \max_{m \in \mathcal{M}_j} V\left(\mathbf{c} - \mathbf{a}_m, t-1\right)\Big) \\
& + p_0\left(t\right) V\left(\mathbf{c}, t-1\right)
\end{aligned}
$$

The boundary conditions are $V\left(\mathbf{c}, 0\right) = 0$ for $\mathbf{c} \geq 0$ and $V\left(\mathbf{c}, 0\right) = -\infty$ otherwise, because any remaining capacity at service provision is worthless and negative values of the remaining capacity are not allowed. The extension to the standard dynamic programming formulation (e.g., Talluri and van Ryzin 2004) is that we replace the revenue-to-go resulting from the acceptance of request j with the maximization term $\max_{m \in \mathcal{M}_j} V\left(\mathbf{c} - \mathbf{a}_m, t-1\right)$, because a decision has to be made regarding how to resolve opacity. That is, one of the specification options included in the set $\mathcal{M}_j$ is selected in a revenue-maximizing way. The corresponding decision rule directly follows from the model by rearranging terms. An incoming request for an opaque product j is accepted if and only if

(2) $r_j \geq \min_{m \in \mathcal{M}_j} \left\{ V\left(\mathbf{c}, t-1\right) - V\left(\mathbf{c} - \mathbf{a}_m, t-1\right) \right\},$

which means that there must be at least one specification option for which the opportunity cost does not exceed the revenue. In the case of acceptance, a potential specification $m*$ with minimal opportunity cost $V\left(\mathbf{c}, t-1\right) - V\left(\mathbf{c} - \mathbf{a}_{m*}, t-1\right)$ is selected and capacity is reduced accordingly. The traditional model and its related decision rule are obviously special cases of (1) and (2), because, if $\left|\mathcal{M}_j\right| = 1$, the maximization and the minimization can be omitted in (1) and (2), respectively.

Note that formulation (1) is more closely related to the standard formulation without opaque products than to the dynamic programming formulation proposed by Gallego, Iyengar, Phillips, and Dubey (2004) for flexible products. In particular, from a technical perspective, (1) is not a special case of these authors' model. This is because, with flexible products, the state space is modeled completely different. It does not include current remaining

capacities of the resources, but only stores the commitments that have been made by selling products. Regarding these commitments, capacity is allocated only in the last stage of the dynamic program by solving a linear feasibility problem. This inhibits the direct application of resource-based decomposition approaches, which is possible in our model for opaque products, as we demonstrate in the next section.

4 Dynamic programming decomposition

Similar to the dynamic program in the traditional setting, DP-op – as given by (1) – is not solvable for most realistic resource networks due to the curse of dimensionality. Therefore, in this section, we show that the well-established idea of dynamic programming decomposition used in traditional network revenue management (e.g., Talluri and van Ryzin 2004: Chap. 3) can be transferred to the setting with opaque products. The main idea of dynamic programming decomposition can be summarized as follows: the network dynamic programming model is decomposed into a collection of single-resource dynamic programs, each of which is only one-dimensional and, therefore, avoids the curse of dimensionality. Using these easy-to-solve dynamic programs, dynamic opportunity costs, which vary as a function of time and capacity, can be obtained and used for a price-based control policy. To account for the network structure, the decomposition uses static information from the optimal solution of an easier-to-solve network model, which is usually the well-known Deterministic Linear Program (DLP) assuming deterministic demand (e.g., Talluri and van Ryzin 2004: Chap. 3).

4.1 Derivation

Following the generic idea of dynamic programming decomposition (e.g., Liu and van Ryzin 2008), we approximate DP-op at a given resource h' by:

$$(3) \quad V(\mathbf{c}, t) \approx V_{h'}(c_{h'}, t) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h$$

$V_{h'}(c_{h'}, t)$ is a dynamic, time- and capacity-dependent approximation of the value of capacity of resource h' and $\sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h$ is a static linear approximation of the other resources' capacity. In our setting, we propose to obtain the values of π_h from

a generalization of the DLP that, to the best of our knowledge, was first presented by Talluri (2001) in the context of passenger routing (see section 2). In this model, as opposed to the standard DLP, the decision variables x_{jm} have two indices, as they denote the number of expected future requests for opaque products that should be assigned to each of the specification options $m \in \mathcal{M}_j$. The optimization problem can then be formulated as follows (**DLP-op**):

$$(4) \quad V^{DLP}(\mathbf{c}, t) = \max \sum_{j \in \mathcal{J}} \sum_{m \in \mathcal{M}_j} r_j x_{jm}$$

subject to

$$(5) \quad \sum_{j \in \mathcal{J}} \sum_{m \in \mathcal{M}_j} a_{hm} x_{jm} \leq c_h \qquad \text{for all } h \in \mathcal{H}$$

$$(6) \quad \sum_{m \in \mathcal{M}_j} x_{jm} \leq D_{jt} \qquad \text{for all } j \in \mathcal{J}$$

$$(7) \quad x_{jm} \geq 0 \qquad \text{for all } j \in \mathcal{J}, \, m \in \mathcal{M}_j$$

Compared to the traditional DLP model, in the objective function (4), in the capacity constraints (5), and in the demand constraints (6) it is now necessary to additionally sum up over all specification options for each product. Again, the traditional setting is obviously a special case. Similar to the traditional decomposition approach, each π_h is taken from the optimal solution of DLP-op and corresponds to the dual variable, that is, the bid price associated with the capacity constraint for each resource h.

Substituting the approximation (3) into the Bellman equation (1), we obtain

$$
\begin{aligned}
&V_{h'}(c_{h'}, t) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h \\
&= \sum_{j \in \mathcal{J}} p_j(t) \max \Big(V_{h'}(c_{h'}, t-1) \\
(8) \quad &+ \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h, \max_{m \in \mathcal{M}_j} \Big(r_j + V_{h'}(c_{h'} - a_{h'm}, t-1) \\
&+ \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h \cdot (c_h - a_{hm}) \Big) \Big) \\
&+ p_0(t) \Big(V_{h'}(c_{h'}, t-1) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h \Big)
\end{aligned}
$$

with the boundary conditions $V_{h'}\left(c_{h'},0\right)=0$ for $c_{h'}\geq 0$ and $V\left(c_{h'},0\right)=-\infty$ otherwise. After some minor rearrangements, we have the following dynamic program, which is one-dimensional with respect to capacity:

$$
\begin{aligned}
(9)\quad V_{h'}&\left(c_{h'},t\right)= \\
&\sum_{j\in\mathcal{J}}p_j\left(t\right)\max\Big(V_{h'}\left(c_{h'},t-1\right), \\
&\max_{m\in\mathcal{M}_j}\bigg(r_j-\sum_{h\in\mathcal{H}\backslash\{h'\}}\pi_h a_{hm}+V_{h'}\left(c_{h'}-a_{h'm},t-1\right)\bigg)\Big) \\
&+p_0\left(t\right)V_{h'}\left(c_{h'},t-1\right)
\end{aligned}
$$

with the same boundary conditions as before.

It can be shown that two of the most prominent results known from standard capacity control immediately carry over to the extended setting. In particular, we have

Proposition 1.

$$
V\left(\mathbf{c},t\right)\leq V_{h'}\left(c_{h'},t\right)+\sum_{h\in\mathcal{H}\backslash\{h'\}}\pi_h c_h
$$

for all $\mathbf{c}\geq 0, t=1,\dots,T, h'\in\mathcal{H}$

as well as

Proposition 2.

$$
V_{h'}\left(c_{h'},t\right)+\sum_{h\in\mathcal{H}\backslash\{h'\}}\pi_h c_h\leq V^{DLP}\left(\mathbf{c},t\right)
$$

for all $\mathbf{c}\geq 0, t=1,\dots,T, h'\in\mathcal{H}$

The proofs for Propositions 1 and 2 are given in Appendix A1 and A2, respectively. Proposition 1 means that the dynamic programming decomposition – applied at a single resource h' while the values of the remaining resources are approximated by the duals of DLP-op – provides an upper bound on the network dynamic programming model. This result has been shown analytically for other specific revenue management models (e.g., Zhang and Adelman (2009) for the derivation in choice-based revenue management) and holds in settings with opaque products as well. From Propositions 1 and 2, it follows that the dynamic programming decomposition leads to a tighter upper bound than the one obtained when simply using DLP-op.

4.2 Discussion

The result of the derivation of the dynamic programming decomposition approach for opaque products shows two specific and intuitive aspects of how opacity has to be taken into consideration.

First, similar to the standard approach, which only considers regular products, dual values from the solution of the corresponding deterministic model are used to capture network effects; they are incorporated by subtracting the value of capacity used elsewhere in the network from each product's revenue. What is new in the setting with opaque products is that the dual values from the DLP-op solution are also used to obtain specification-dependent adjusted revenues, which is intuitive because the specification, which is chosen by the provider after sale, determines the capacity used elsewhere in the network.

Second, interestingly, it turns out that the decision problem on how to resolve opacity remains included in each of the resulting one-dimensional dynamic programs. Even though only a single resource h' is considered explicitly, all specification options are taken into consideration, whether they require capacity on h' or not. This is reflected by the inner term

$$
\max_{m\in\mathcal{M}_j}\bigg(r_j-\sum_{h\in\mathcal{H}\backslash\{h'\}}\pi_h a_{hm}+V_{h'}\left(c_{h'}-a_{h'm},t-1\right)\bigg)\quad\text{in}
$$

(9). More specifically, there can even be specification options for opaque products that do not require the current resource at all. Consequently, their selection in the single-resource dynamic program will not lead to any changes in remaining capacity $c_{h'}$. Instead, the required resources' static bid prices π_h capture the capacity consumption implied by the selection of such an option. Thus, in contrast to the standard decomposition, the static bid prices from the deterministic network model are not only incorporated in an "AND"-fashion, reflecting the usage of multiple resources by a product, but also in an "OR"-fashion, reflecting the different specification options a product can use. This is illustrated in detail by the example given in Appendix A3.

4.3 Control mechanism

In line with the mechanism for the standard case, a control mechanism can be constructed based on the dynamic programming decomposition outlined above. Bid prices π_h are derived by solving DLP-op up-front. The approximation is then repeated at each resource h' of the network, obtaining a set of

one-dimensional value functions $V_{h'}(c_{h'}, t), h' \in \mathcal{H}$ (see (9)). In the value function at resource h', the products' revenues are reduced by the bid prices π_h of resources $h \neq h'$ used elsewhere in the network, depending on the specification option. The resource-specific value functions are then aggregated to form a dynamic approximation of the network value function $V(\mathbf{c}, t) \approx \sum_{h' \in \mathcal{H}} V_{h'}(c_{h'}, t)$. This implies that the opportunity cost of accepting a request for the opaque product j and selecting a specification option m is approximated by $\sum_{h \in \mathcal{H}}(V_h(c_h, t-1) - V_h(c_h - a_{hm}, t-1))$. In this context, the used resource-specific value functions (9) can be further simplified by only considering those products that need capacity from the resource h' in at least one specification option. This leads to the following reformulation:

$$
\begin{aligned}
(10) \quad V_{h'}^{res}(c_{h'}, t) = & \sum_{j \in \mathcal{J} \mid \exists m \in \mathcal{M}_j : a_{h'm} > 0} p_j(t) \max\Big(V_{h'}^{res}(c_{h'}, t-1), \\
& r_j - \max_{m \in \mathcal{M}_j}\Big(\sum_{h \in \mathcal{H} \setminus \{h'\}} a_{hm} \pi_h + V_{h'}^{res}(c_{h'} - a_{h'm}, t-1) \Big) \Big) \\
& + \Big(1 - \sum_{j \in \mathcal{J} \mid \exists m \in \mathcal{M}_j : a_{h'm} > 0} p_j(t) \Big) V_{h'}^{res}(c_{h'}, t-1)
\end{aligned}
$$

Considering the definition of opportunity cost based on (9) and (10), respectively, it is easy to see that $V_h^{res}(c_h, t-1) - V_h^{res}(c_h - a_{hm}, t-1) = V_h(c_h, t-1) - V_h(c_h - a_{hm}, t-1) \; \forall j \in \mathcal{J}, m \in \mathcal{M}_j$ holds. Then, throughout the booking horizon, the decision to accept requests is made in a similar way as outlined in section 3 for the original dynamic program by condition (2). The only modification is that opportunity costs are obtained from the value functions $V_h^{res}(c_h, t)$ instead of the original dynamic program, leading to

$$
(11) \quad r_j \geq \min_{m \in \mathcal{M}_j} \sum_{h \in \mathcal{H}} \big(V_h^{res}(c_h, t-1) - V_h^{res}(c_h - a_{hm}, t-1) \big).
$$

In the case of acceptance, a potential specification $m^* \in \mathcal{M}_j$ with minimal opportunity cost $\sum_{h \in \mathcal{H}}(V_h^{res}(c_h, t-1) - V_h^{res}(c_h - a_{hm}, t-1))$ is selected and capacity is reduced accordingly.

5 Computational results

In this section, we perform a simulation study to investigate revenue management settings that include opaque products. Our study is based on two basic network structures that are introduced in section 5.1. Structures of this kind occur directly or as a substructure in common airline networks. The structure of the second network has been adapted from the literature. In section 5.2, we investigate the performance of the dynamic programming decomposition approach proposed in section 4. Having shown that the approach is a reasonable method of capacity control, we are able to investigate several specific aspects and effects that result from integrating opaque products into a capacity control setting in detail in sections 5.3 and 5.4. In particular, in section 5.3, we analyze the effect of systematically varying the share of opaque products. In section 5.4, we describe how the inherent flexibility, that is, the degree of opacity drives the obtainable revenue performance.
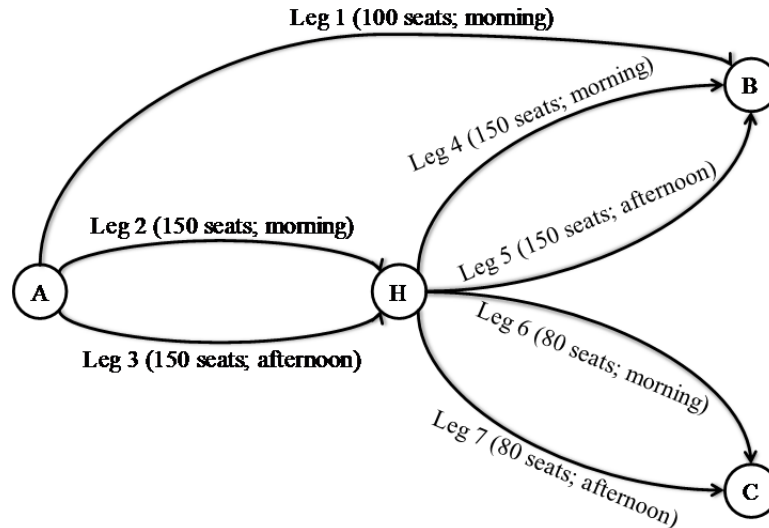
5.1 Simulation experiment design

The numerical experiments we conduct are based on two example airline networks.

Network 1 consists of three parallel flights with capacities of 40, 30, and 35 seats each. There are four booking classes on each flight (Y, M, B, and Q) and one opaque product, producing 13 products in total. The fares are 500, 417.50, 335, and 250, depending on the booking class. Any of the three flights can be assigned to a customer of the opaque product, but he pays only 150. On each flight, 10%, 20%, 20%, and 30% of expected demand is for booking class Y, M, B, and Q, respectively; the remaining 20% of customers request the opaque product.

Network 2 is taken from the literature and was originally proposed by Liu and van Ryzin (2008: §7.2). There are four cities A, B, C, and H, which are connected by seven flight legs with capacities of between 80 and 150 seats (see Figure 1). We consider two booking classes Y and Q, producing a total of 27 regular and opaque products. The long-haul leg A-B can be booked for 900 and 540 in class Y and Q, respectively. The other single-leg flights are considered short-haul and cost 500 in Y and 300 in Q. Four connecting itineraries are available from A via the hub H, priced at 800 and 480 in the two book-

Figure 1: Network 2

ing classes. In addition, five opaque products are offered in this network. There are three short-haul opaque products at a cost of 220 from A to H, H to B, and H to C, guaranteeing transportation on one of two possible itineraries, either in the morning or afternoon. Furthermore, there are two long-haul opaque products offering transportation from A to B and A to C that can be bought for 352. With respect to opacity in these products' definition, besides the two connecting itineraries, customers flying from A to B can also be assigned to the direct flight. In case that, in total, expected demand equals the capacity on all flights, expected demand is calculated as follows: demand for each connecting itinerary equals 30% of its total capacity and demand for the direct flights is set equal to the remaining capacity. On each itinerary, 30% of demand is for class Y, 40% for class Q, and 30% for an opaque product with the demand distributed evenly if there are multiple opaque products. Demand information is additionally given in tabular form in Appendix A4.

In our study, we assume the booking process to be time-homogeneous and the arrival rate is calculated accordingly from the expected demand value. Furthermore, in line with our model assumptions from section 3, the booking process is discretized into T periods, so that there is at most one incoming customer request in each period (e.g., Subramanian, Stidham Jr., and Lautenbacher 1999: section 3.2.1 for the discretization procedure). The resulting number of periods is 250 and 1500 in Network 1 and Network 2, respectively. We fix the number of
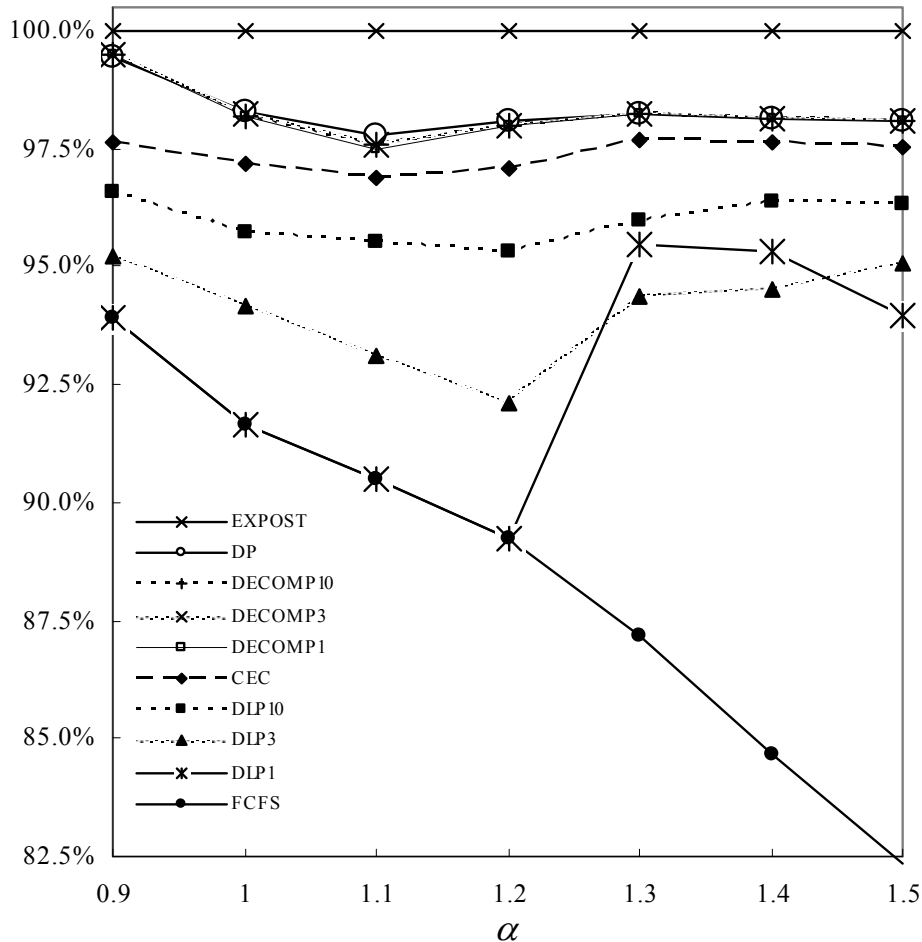
simulation runs, each of them representing a stream of product requests, for all considered scenarios to 100, and, depending on the matter of interest, report resulting aggregated performance indicators along with the corresponding confidence levels. If different control methods are compared, we use the same set of 100 streams of product requests for all of them. Furthermore, in both networks, we generate additional scenarios by varying the demand intensity in advance in order to simulate different load factors. Therefore, we scale the expected demand using a parameter $\alpha \in \{0.9, 1.0, ..., 1.5\}$, where $\alpha = 1$ corresponds to the case that expected demand equals capacity on all flights, as described above.

5.2 Performance evaluation of the dynamic programming decomposition

The purpose of this subsection is to indicate whether the practical application of the proposed dynamic programming decomposition is reasonable in terms of relative performance compared to other typical model-based capacity control methods when generalized to the opaque product setting, as well as in terms of computational runtime.

For Network 1, Figure 2 shows the revenue performance of all control methods tested relative to the perfect hindsight optimal revenue obtained with full information on demand (EXPOST). DP is the control mechanism given by condition (2) based on the network dynamic programming model (1). The

Figure 2: Performance of the control methods relative to EXPOST (Network 1)
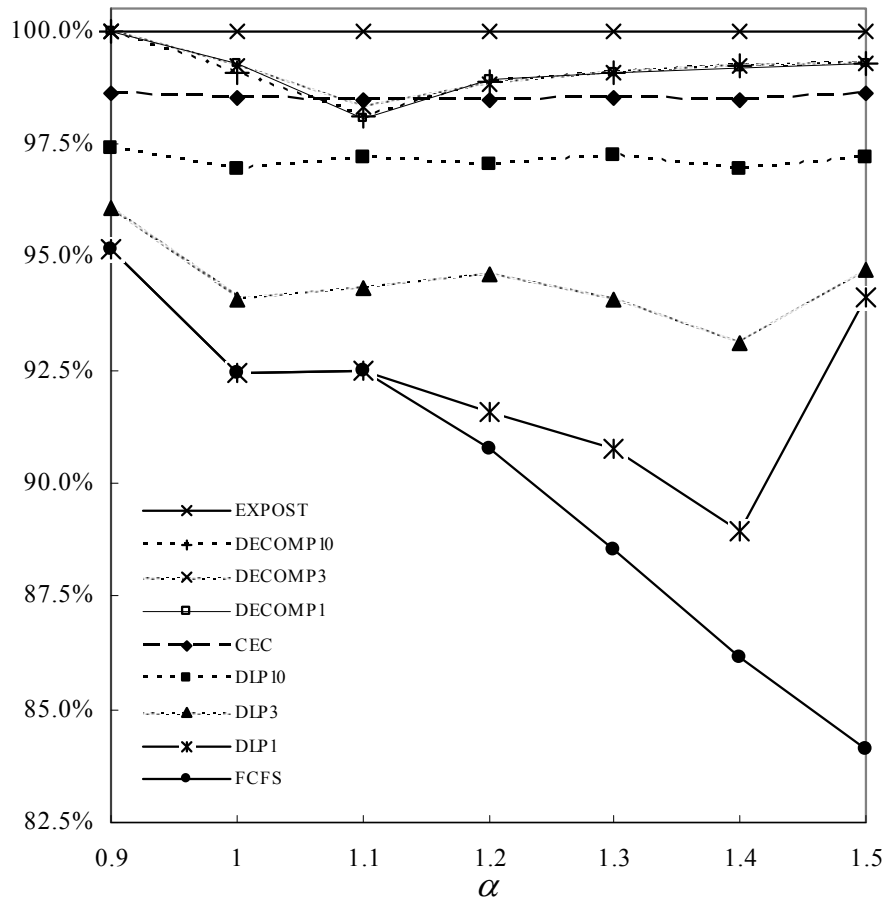

DECOMP#-variants implement the dynamic programming decomposition approach developed in section 4. The next four methods are immediate applications of DLP-op given by (4)-(7). CEC bases the acceptance of a request on the comparison of two optimal solutions of the model – one accepting the request and one rejecting it (see Bertsimas and Popescu (2003) for CEC in the standard setting). DLP# uses the dual variables from DLP-op as bid prices to implement standard bid price controls (e.g., Talluri and van Ryzin 2004: Chap. 3.3.1). A request is accepted if and only if its revenue is not less than the sum of the relevant bid prices in at least one specification option. FCFS is a simple first-come-first-served control, which accepts requests as long as they can be served with the remaining capacity. In DECOMP# and DLP#, the number #

indicates how often the linear program is solved throughout the booking horizon. Within the control methods DECOMP1 and DLP1, the deterministic linear program is solved only once at the beginning. The dual variables associated with the capacity constraints are then used to estimate the marginal value of capacity throughout the entire booking horizon. In methods DECOMP3 and DLP3, we partition the booking horizon into three evenly split periods and resolve the problem at the beginning of each period, using the remaining time, capacity, and estimated demand-to-come as input parameters. In DECOMP10 and DLP10, 10 equal-sized periods are used.

First of all, Figure 2 clearly shows that the DECOMP approaches consistently produce almost the same revenues as DP. The number of reoptimizations has

Figure 3: Performance of the control methods relative to EXPOST (Network 2)

no visible effect and the lines representing DP, DE-COMP1, DECOMP3, and DECOMP10 are so close that it is hard to distinguish between them. This is quite an impressive result since DP's control decisions are perfect in the sense that they maximize expected revenue, given the available demand forecast. DECOMP1 is only significantly below DP at the 99% level of confidence for $\alpha = 1.1$; however, the difference is only 0.5 percentage points. Furthermore, it turns out that DECOMP1 performs significantly better than CEC, regardless of the demand intensity considered. Compared to the methods mentioned above, DLP10 trails about 1% behind. DLP3 is much less stable and about 2%-5% behind. DLP1 shows very high fluctuations: If the load factor is low ($\alpha \leq 1.2$), it yields exactly the same low revenue as FCFS in all demand streams. This is because, although the total demand may exceed capacity, there is a positive contingent in the DLP model's

optimal solution, even for low-value products. This leads to bid prices that allow for the acceptance of all requests, although the contingent in the solution may be very small. However, when the demand is stronger, DLP's revenue sharply increases as it begins to reject some requests. Analyzing the benefit of the reoptimizations in detail confirms our observation from Figure 2 that the effect of reoptimizing is negligible for DECOMP, as the confidence intervals comparing DECOMP1, DECOMP3, and DE-COMP10 are almost centered around 0. The corresponding results for $\alpha = 1.4$ and confidence levels are given in Appendix A5.

Likewise, Figure 3 shows the performance of the control methods for Network 2 relative to EXPOST. Note that the values for DP cannot be reported here, because the high dimensionality and size of the state space make its application impossible. The DE-COMP approaches usually produce the highest rev-

enues, yielding over 99% of EXPOST's revenue, and significantly outperform CEC. The impact of reoptimizing for the DLP approaches is much bigger for Network 2 than for Network 1. Regarding DECOMP, there is no significant advantage. However, our results (see also Appendix A5) indicate that reoptimizations can – at least to a very small extent – help further improve the revenue. These figures suggest that, especially for DLP10, the revenue could be increased even more by further raising the number of reoptimizations. Nevertheless, it is important to note that regardless of the number of reoptimizations, DLP will never outperform CEC in expectation, because CEC can be regarded as resolving the linear program for every single customer and directly deciding on the requests without being constrained to additive bid prices. As DECOMP significantly dominates CEC, no reoptimization strategy will ever enable DLP to get close to DECOMP. The only exception is for $\alpha = 1.1$. Here, the initial dual values of the DLP model that are used to calculate the adjusted revenues used in DECOMP1 are quite bad. In this case, the adjusted revenues are valid for only very few units of capacity, which leads to a revenue of 0.4% below that of CEC. However, when the adjusted revenues are updated during the booking horizon by resolving the linear program, revenue increases and the gap between DECOMP and CEC can be reduced to 0.17%.

Overall, our results regarding the relative performance of the approaches are, by and large, in line with what is known from traditional revenue management without opaque products. Even without reoptimizations, our decomposition approach considering opaque products performs quite well. Its revenues are comparable to those obtained by DP

and it considerably outperforms the other methods, including CEC.

Table 1 shows computational time statistics for both networks for $\alpha = 1.4$. With respect to DP-based methods, that is, DP, DECOMP1, DECOMP3, and DECOMP10, the column "V" shows the average time required for calculating the value function for all states. With regard to the latter three, this also includes the time to calculate the displacement-adjusted revenues; that is, the time required to solve the dual of the corresponding DLP-op. In case of CEC, nothing is computed in advance, as DLP-op needs to be resolved twice for every incoming request. For DLP1, DLP3, and DLP10, the column "V" gives total computation times needed to solve DLP-op. For each method, the column "Control" refers to the time necessary to handle all the customer requests from one demand stream.

Table 1 shows that, although DP is still applicable in Network 1, calculating the value function is very time consuming. All other methods are computationally feasible for both networks. The computational behavior of the decomposition approaches is identical to what is known from the standard setting without opaque products. The values from the column "V" show that the DECOMP-variants are of course more time consuming than the DLP-op-based methods. However, as the resulting dynamic programs' state space is one-dimensional with respect to capacity, computation time scales up linearly with the network size. Furthermore, as our investigation from before suggests, it often seems unnecessary to resolve the DECOMP approaches over time with respect to revenue performance. In this case, the value function of the DECOMP approaches could be completely pre-calculated. The times re-

Table 1: Computation times (Network 1 & 2, $\alpha = 1.4$)

Method	Network 1, α=1.4		Network 2, α=1.4	
	V [mm:ss]	Control [s]	V [hh:mm:ss]	Control [mm:ss]
DP	15:15.907	0.312	-	-
DECOMP10	15.024	0.186	1:00:11.438	3.201
DECOMP3	6.720	0.190	24:14.272	3.124
DECOMP1	2.277	0.187	17:08.249	3.135
CEC	-	3.104	-	4:44.764
DLP10	0.233	0.172	3.806	1.120
DLP3	0.115	0.165	2.511	1.082
DLP1	0.064	0.170	2.199	1.090

quired to handle a request are rather similar and negligible for all methods, except, of course, CEC.

Overall, from our investigation in this subsection, we can conclude that, for our example, the proposed decomposition approach performs particularly well in terms of realized revenue while, at the same time, our experiments demonstrate practical feasibility with respect to runtime. Therefore, with the decomposition, we seem to have found a reasonable method to consider opaque products in a revenue management capacity control process. On this basis, we are now able to investigate several specific effects of opaque products in the following subsections. It is of course important to note that the results cannot be generalized with certainty and clearly depend on the setting. However, the results from this section regarding the relative performance of the various approaches are obviously in line with the results known to be valid for their counterparts in the standard setting. Therefore, we think that our results are at least somewhat representative.

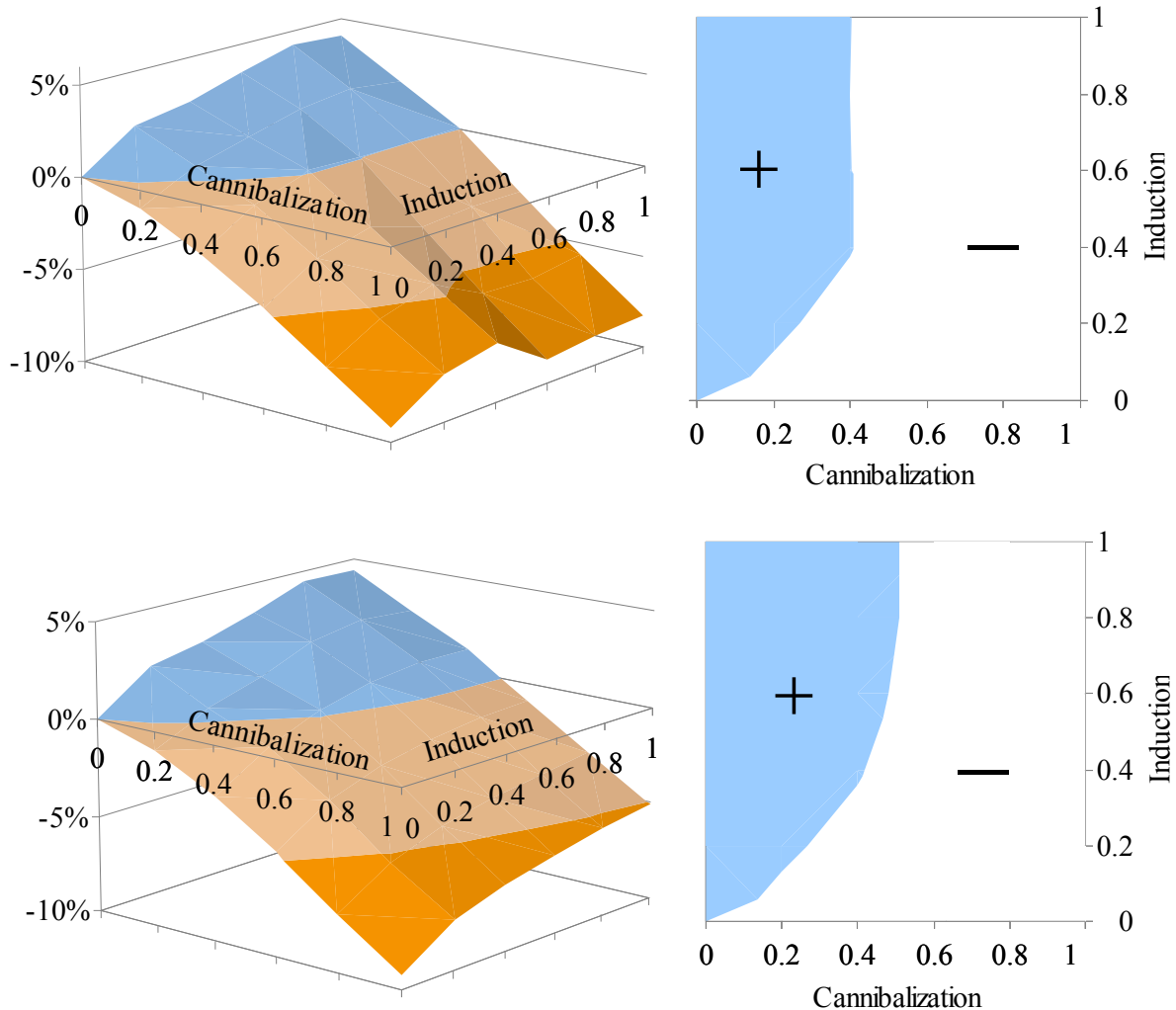### 5.3   Share of opaque products

In this subsection, we analyze the effects of different shares of opaque products. As a point of departure, we use the products, prices, and demand shares from Network 1 with a demand intensity of $\alpha = 1.1$. We use this intermediate demand intensity because, with it, capacity is scarce and, at the same time, it is optimal to accept quite a few opaque products. We now first remove the opaque products in order to construct a base case without any opacity. We then mimic the demand-side effects decision makers would potentially experience when deciding to introduce opaque products. First, the introduction of opaque products can lead to additional low-value demand from customers who had not purchased any of the offered products before (demand induction). Second, customers who originally intended to buy a regular product could change their purchase behaviour and decide to buy the cheaper flexible product (cannibalization). In practice, the impact of demand induction and cannibalization is industry-specific and strongly depends on aspects, such as the products' (relative) prices and the attractiveness of the opaque products.

Figure 4 shows the revenues obtained with DP and DECOMP1 relative to the base case. In the graphs to the left, on one of the axes, we vary the level of demand induction and denote it relative to the number of opaque product requests in the original set-

ting from section 5.1. More precisely, a value of zero refers to the base case without any opaque product requests, while a value of 0.5, for example, results in only half of the original opaque product demand. On the other axis, we vary the degree of cannibalization independently of induction. In this example, we assume that cannibalization always refers to only the cheapest regular product defined on the network, that is, class Q. A factor of zero means that there are no class Q-customers changing their demand behavior, while a value of 1 means that all original class Q-customers switch to the opaque product. The resulting cannibalized demand for the opaque product is added to the demand generated by demand induction. Note that, with a cannibalization of 0 and a demand induction of 1, we have the original setting that was investigated in section 5.2. The graphs on the right side in Figure 4 show the horizontal cross-section of the corresponding graph on the left, indicating the ranges of the values of cannibalization and induction for which the introduction of opaque products leads to a revenue improvement (+) and those for which they lead to a loss (−).

We can observe from the results of DP in Figure 4 that a higher degree of demand induction leads to a better revenue performance, which is intuitive. However, the relationship is concave, as with a high number of induced opaque requests, capacity becomes scarcer. On the other hand, a high cannibalization has a negative effect, as the opaque product is priced at 150 instead of 250 for the regular product. As DP would not be solvable for larger instances, it is important to get an idea of whether the decomposition, which we have identified as the best applicable approach in the previous subsection, leads to comparable results. The results for DECOMP1 in Figure 4 show that the dependency on induction and cannibalization is similar to DP in general. For example, the obtainable gain with a maximum induction rate or for a low induction and a high cannibalization rate are in a comparable range. Moreover, the monotonicity properties are similar, except for the case of high cannibalization, where DECOMP1 is not monotone in induction. This results from a disadvantageous set of bid prices from DLP-op used within DECOMP1, an effect that has already been discussed in section 5.2 and that can be mitigated by applying reoptimizations. Overall, the results from this example show that for a wide range of induction and cannibalization rates leading

Figure 4: Revenue depending on demand induction and cannibalization (upper row: DECOMP1, lower row: DP, Network 1)

to a specific share of opaque products, the introduction of opaque products is advantageous in terms of revenue achieved. The range is similar but slightly smaller for DECOMP1 than for DP. However, these values are of course specific to our example and cannot be generalized.
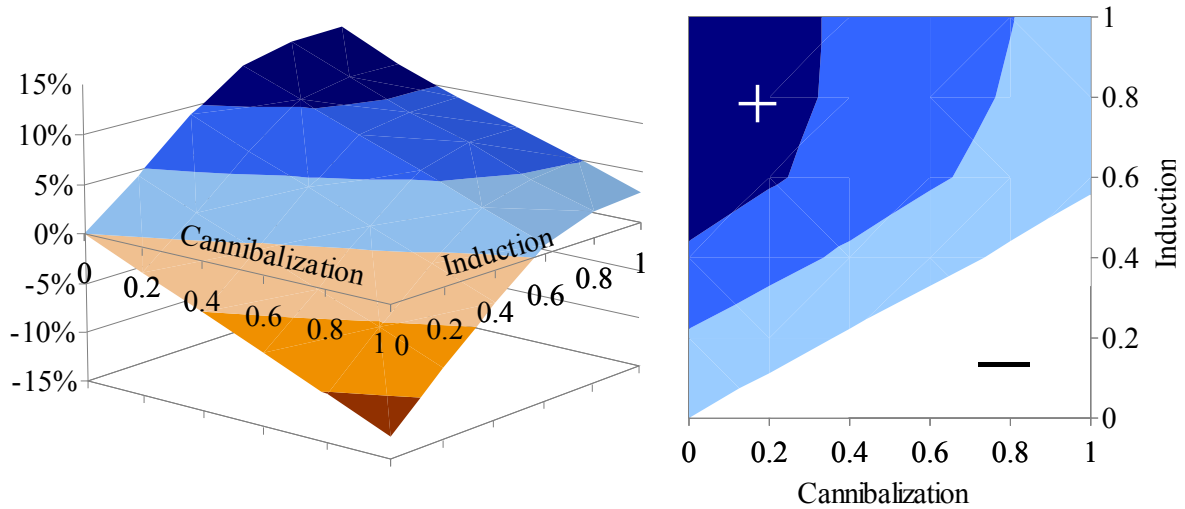
In Figure 5, we perform a similar analysis for Network 2, in which DP is, however, no longer tractable. The results for DECOMP now show strict monotonicity in the parameters, as the influence of individual bid prices from DLP-op seems to be smaller. The possible overall gain of opaque products is bigger than in Network 1. In particular, it is now always positive for high induction levels. As the relative

price difference compared to the cheapest regular product is smaller and the network is more complex, flexibility is obviously more valuable in this setting.

5.4   Degree of opacity
In this subsection, we investigate the degree of opacity's influence on the revenues obtained. Therefore, we compare settings that differ only in the degree of flexibility inherent in their opaque products. To construct the scenarios for this experiment, similar to section 5.3, we use the products, prices, and demand shares from Network 1 with a demand

Figure 5: Revenue depending on demand induction and cannibalization (DECOMP1, Network 2)
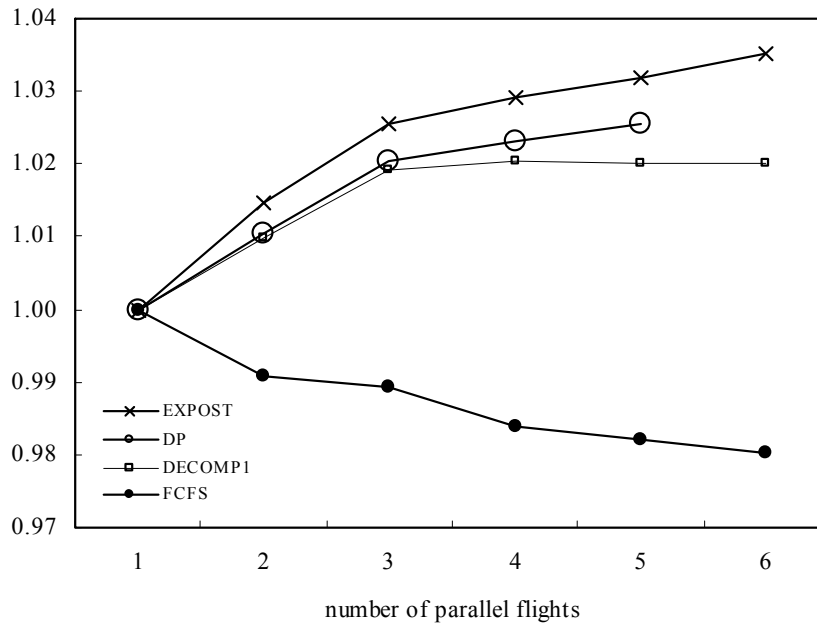

intensity of $\alpha = 1.1$. To consider different degrees of flexibility, we now vary the number of flights between one and six with an identical per-flight capacity of ten seats and a constant number of 100 periods, allowing us to also calculate DP for most of the scenarios. Demand is generated exactly as described for Network 1 for each experiment. For example, when considering two flights, 20% of demand is for the opaque product and can be assigned to either flight. The remaining demand is flight-specific in booking classes Y, M, B, and Q for the two flights. When considering only one flight, demand is still generated as described for Network 1. As overall capacity is half of that compared to two flights, overall demand is half of that, too. From this demand, 80% is still flight-specific in Y, M, B, and Q. A total of 20% is still for the opaque product, which no longer really be considered opaque, as it can only be assigned to the sole flight. Comparing average per-leg revenues, the settings now only differ in the number of legs to which the opaque requests can be assigned.

Figure 6 shows the revenues obtained with DP and DECOMP1 as well as the benchmarks FCFS and EXPOST relative to the revenues obtained with the respective method in the one-flight case. Thus, all methods "start" at 100% (1.00) for one flight. While FCFS's revenue decreases to about 97% for six flights, we observe that the other methods' revenues increase to 103.5% for EXPOST and 102% for DE-

COMP1 for six flights. DP attains about 102.5% for five flights, but is computationally intractable for more. For the latter three methods, increases in revenue are the strongest for one to three flights; DECOMP1 even levels off and stays more or less constant for more than three flights.

These observations can be explained as follows. FCFS is the only method whose revenues decrease with the number of flights, that is, when flexibility increases. This is due to FCFS's general shortcoming of accepting too many low-value requests – in fact, accepting all it can. This shortcoming becomes even more severe with more flights, because the increased flexibility of opaque products means it can, and actually does, accept even more opaque requests. Thus, revenue decreases with increasing flexibility. This is a general property of FCFS; non-increasing revenues in the level of flexibility would show up in every setting where the opaque product is the cheapest one. In contrast, DP as the optimal control is – in expectation – never hurt by additional flexibility and revenues increase with the degree of flexibility as the opaque requests are used to mitigate different levels of scarcity on the legs stemming from stochastic demand. Similarly, EXPOST's revenues increase with the degree of flexibility as well. As this is the perfect hindsight control, this increase is not only in expectation, because no single instance can be constructed where EXPOST would make use of the flexibility although it should not.
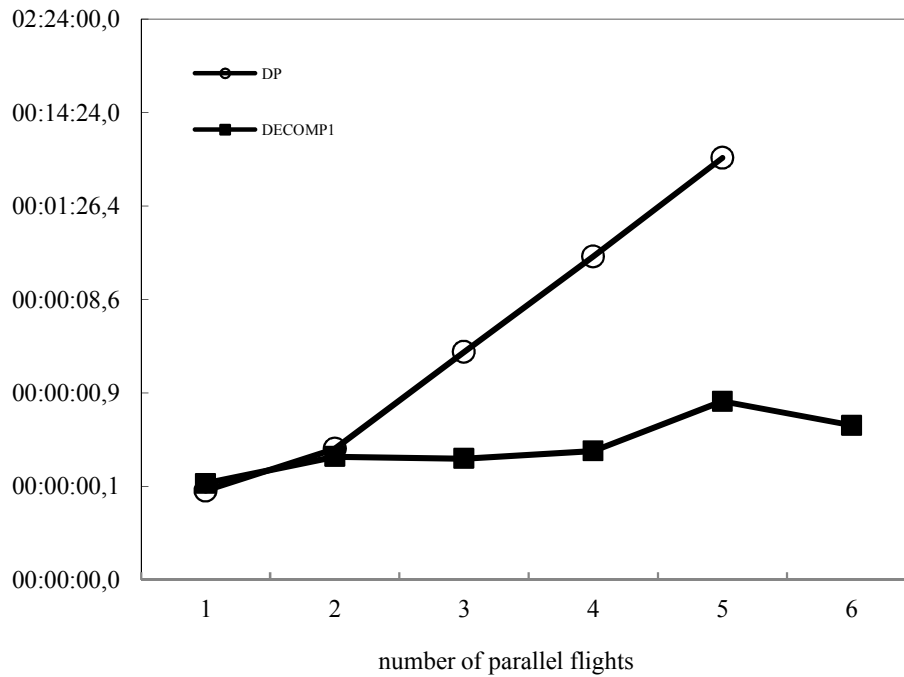
Figure 6: Revenue depending on number of parallel flights (one flight = 1.00)

The picture for DECOMP1 is more differentiated; two overlaying effects have to be distinguished here: On the one hand, an increased flexibility allows for better mitigating stochastic demand as explained above for EXPOST and DP. On the other hand, the more flights and, thus, the higher the degree of flexibility, the more alternative specification options have to be compared in the inner maximization of (9). Most of these specification options are only described approximately with the help of shadow prices from DLP-op. In our example, DECOMP1 is in fact identical to DP in the single-flight scenario. While the first effect increases revenue with flexibility, the latter decreases the quality of the approximation. Which effect prevails clearly depends on the scenario considered. Here, the first one seems to be stronger up to three flights; for more flights they cancel each other out. However, defining such a high degree of opacity would anyway cause a customer's reaction in terms of demand induction and cannibalization as it was analyzed in section 5.3. Such effects have not been considered in this example as they are very much related to the specific industry setting. Nevertheless, it is likely that a very high degree of opacity would lead to a negative impact on revenue due to a much smaller demand induction.

Figure 7 shows the total computation time needed for processing one customer stream with DP and DECOMP1 in this experiment, depending on the number of flights. Note that the vertical axis has a logarithmic scale. In this graph, DP's runtime is a straight line for two to five legs. This result clearly reflects the exponential growth of the DP's number of states, and, thus, of the runtime as the number of flights increases. The slightly higher-than-expected runtime for one flight might be due to overheads, such as deciding on request acceptance. We consider measurement noise less likely, as the runtime closely matches that of DECOMP1 for one flight. Considering more flights, DECOMP1's advantages are obvious: Although its runtime should theoretically increase linearly in the number of flights as more of the single-leg dynamic programs have to be calculated, being less than a second, the runtime is so short that this effect is overshadowed by measurement noise. Therefore, no clear trend can be seen. Note that although DP's runtime would not prevent application to six flights – as can be seen in the figure by extrapolating its line – its memory requirements also grow linearly with the exponentially growing number of states. Thus, the amount of physical RAM available prevents us from using DP with more than five flights.

Figure 7: Computation time [hh:mm:ss] for 1-6 flights, logarithmic scale

6 Conclusions

In this paper, we consider revenue management with opaque products. Opaque selling is an additional instrument of price discrimination that helps segment a market. However, the supplier-driven substitution inherent in these products must be reflected in the revenue management process as well. As the exact dynamic program describing capacity control with opaque products is usually not applicable to problems of real-world size due to its multi-dimensional state space, we propose a quite intuitive approach for capacity control with opaque products. It is based on traditional dynamic programming decomposition heuristics widely used in theory and practice. We formally derive the approach and analyze the resulting single-resource dynamic programs. It turns out that the decision problem of how to resolve opacity is still included in each program. However, the resulting cost of capacity consumption for specification options that do not concern the current resource are completely approximated by the bid prices calculated by the corresponding linear program. We then show that our approximation implies a tighter upper bound on the optimal expected revenue than the well-known DLP model adapted to the opaque product setting – an important result that is also known from the traditional setting without opacity.

In a numerical investigation, we further analyze the performance of our decomposition approach in the context of airline revenue management and compare it with traditional control mechanisms adapted to opaque selling. It shows that the decomposition approach's revenue is comparable to that obtained by the full network dynamic program and significantly outperforms other approaches, such as DLP-based bid prices and certainty equivalent control. While DLP-based bid prices have to be frequently updated by resolving the linear program to adapt to demand and capacity utilization, solving the linear program underlying the decomposition only once up-front is sufficient. Reoptimizations slightly improve the revenue in only a few cases. With respect to memory and computation time, the decomposition is comparable to the traditional decomposition approach without opacity and can thus be used for large networks.

Having identified the decomposition approach as the best applicable capacity control approach in opaque product settings so far, we use the approach

to investigate some aspects that are particularly related to opaque products. In particular, we first investigate how the overall share of opaque products influences the results. When introducing opaque products, this share is mainly driven by demand induction as well as cannibalization effects. The results are meaningful in the sense that they basically correspond to what can be expected. In general, we see a positive influence of induction and a negative one of cannibalization with a few exceptions in our parallel flight settings arising from the heuristic nature of the control procedure. However, in the network setting, these heuristic effects do not show up. Overall, the experiment shows that the benefits of introducing opaque products in a capacity control setting strongly depend on the specific parameters of demand behavior. Second, we investigate how the degree of opacity drives the obtained results. In our examples, it turns out that the degree of opacity – modeled by the number of parallel resources that can be used to fulfill opaque requests – has a strong positive influence on revenue for moderate levels. However, for high levels of opacity, there are settings where the reverse effects arising from the heuristic nature of the decomposition compensate for the positive effect of an additional specification option.

We believe that the results presented in this paper are highly relevant for revenue management in practice. As opaque products are increasingly offered by companies that traditionally make use of revenue management, there is an urgent need to consider these innovative products in the capacity control methods used. This is important because only integrated methods, such as the decomposition approach presented, enable the supplier to fully benefit from the advantages of opaque selling. From our experiments, we conclude that in revenue management settings, the introduction of opaque products in an existing product portfolio followed by the application of the proposed decomposition approach can be quite beneficial. However, it is important to have an idea of what rates of demand induction and cannibalization can be expected as these were shown to be crucial for the success of opaque products. In this context, it is particularly important not to define a degree of opacity that is too high, as this may lead to a loss in revenue compared to a lower degree of opacity. This can be due to both the heuristic nature of the decomposition

approaches as well as to the quite likely and disadvantageous reduction of the demand induction rate when the degree of opacity is too high.

Appendix

A1 Proof of Proposition 1
We make use of a decision vector $\mathbf{u} \in \left\{0, 1, ..., |\mathcal{I}|\right\}^n$ for each remaining capacity $\mathbf{c}$, indicating whether a request arriving in the current period would be accepted. The vector components indicate the provider's acceptance decision as well as the specification to select after sale. That is, a request for product j is denied if $u_j = 0$ and accepted and specified as $u_j \in \mathcal{I}$ if $u_j \geq 1$. To be feasible with remaining capacity $\mathbf{c}$, the vector must satisfy

$$
\begin{aligned}
(12) \quad & \mathbf{u} \in \mathcal{U}(\mathbf{c}) = \left\{ \mathbf{u} \,\middle|\, \mathbf{u} \in \left\{0, 1, ..., |\mathcal{I}|\right\}^n \wedge \right. \\
& \left. \left(u_j = 0 \vee \left(u_j \in \mathcal{M}_j \wedge \mathbf{a}_{u_j} \leq \mathbf{c} \right) \forall j \in \mathcal{J} \right) \right\}.
\end{aligned}
$$

With this set of feasible decision vectors $\mathcal{U}(\mathbf{c})$, the dynamic programming model can be reformulated as a linear program with decision variables $V(\mathbf{c}, t) \geq 0 \ \forall \mathbf{c}, t$ (e.g., Adelman 2007):

$$(13) \quad \min V(\mathbf{c}, t)$$

subject to

$$
\begin{aligned}
(14) \quad & V(\mathbf{c}, t) \geq \\
& \sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j(t) \left(r_j + V\left(\mathbf{c} - \mathbf{a}_{u_j}, t-1\right) \right) + \\
& \left(1 - \sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j(t)\right) V(\mathbf{c}, t-1) \\
& \hspace{3cm} \text{for all } t, \mathbf{c}, \mathbf{u} \in \mathcal{U}(\mathbf{c}).
\end{aligned}
$$

Likewise, the dynamic program that approximates the revenue at resource h' as given by (9) is formulated as a linear program, using decision variables $V_{h'}(c_{h'}, t) \geq 0 \ \forall c_{h'}, t$:

$$(15) \quad \min V_{h'}(c_{h'}, t)$$

subject to

(16)
$$V_{h'}\left(c_{h'}, t\right) \geq$$
$$\sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(t\right)\left(r_j - \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h a_{hu_j}\right.$$
$$+ V_{h'}\left(c_{h'} - a_{h'u_j}, t-1\right)\bigg)$$
$$+ \left(1 - \sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(t\right)\right) V_{h'}\left(c_{h'}, t-1\right)$$
$$\text{for all } t, \mathbf{c}, \mathbf{u} \in \mathcal{U}_{h'}\left(c_{h'}\right).$$

We now perform induction over t. For $t = 1$, we obtain from (16) that

(17) $\quad V_{h'}\left(c_{h'}, 1\right) \geq \sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(1\right)\left(r_j - \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h a_{hu_j}\right)$
$$\text{for all } \mathbf{c}, \mathbf{u} \in \mathcal{U}_{h'}\left(c_{h'}\right).$$

Since this holds for all $\mathbf{u} \in \mathcal{U}_{h'}\left(c_{h'}\right)$ and $\mathcal{U}_{h'}\left(c_{h'}\right) \supseteq \mathcal{U}(\mathbf{c}) \forall \mathbf{c}$, we have

(18)
$$V_{h'}\left(c_{h'}, 1\right) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h \geq$$
$$\max_{\mathbf{u} \in \mathcal{U}(\mathbf{c})}\left\{\sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(1\right)\left(r_j + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h\left(c_h - a_{hu_j}\right)\right)\right\}$$
$$= V\left(\mathbf{c}, 1\right)$$
$$\text{for all } \mathbf{c},$$

where the last equality follows from the fact that there are conditions (14) for all $\mathbf{u} \in \mathcal{U}(\mathbf{c})$ in the model (13)-(14). Thus, the result holds for $t = 1$.
Let us now assume it holds for $t - 1$. We show that it then will hold for t as well. Adding $\sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h$ to (16) and using the induction hypothesis, we obtain

(19)
$$V_{h'}\left(c_{h'}, t\right) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h \geq$$
$$\sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(t\right)\left(r_j + V\left(\mathbf{c} - \mathbf{a}_{u_j}, t-1\right)\right)$$
$$+ \left(1 - \sum_{j \in \mathcal{J} \mid u_j \neq 0} p_j\left(t\right)\right) V\left(\mathbf{c}, t-1\right)$$
$$\text{for all } t, \mathbf{c}, \mathbf{u} \in \mathcal{U}_{h'}\left(c_{h'}\right).$$

Using $\mathcal{U}_{h'}\left(c_{h'}\right) \supseteq \mathcal{U}(\mathbf{c}) \forall \mathbf{c}$ and (13)-(14), we can conclude that the approximation at every resource h' (9) provides an upper bound on the network DP model. In particular, we have

(20) $\quad V\left(\mathbf{c}, t\right) \leq V_{h'}\left(c_{h'}, t\right) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h \quad$ for all $t, \mathbf{c}$.
$\hfill\square$

A2 Proof of Proposition 2
We first consider the optimal solution of DLP-op. We choose a fixed resource h' and relax the constraints (5) for all the other resources $h \in \mathcal{H} \setminus \{h'\}$ by associating the dual multipliers π_h, that is, the values of the dual variables associated with the capacity constraint (5). As a result of linear programming duality, the obtained formulation has the same optimal objective value as DLP-op. After some minor rearrangements, the objective function can be formulated as

(21)
$$V^{DLP}\left(\mathbf{c}, t\right) =$$
$$\max\left(\sum_{j \in \mathcal{J}} \sum_{m \in \mathcal{M}_j}\left(r_j - \sum_{h \in \mathcal{H} \setminus \{h'\}} a_{hm} \pi_h\right) x_{jm}\right.$$
$$+ \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h\bigg).$$

The constraints (5)-(7) remain unchanged, except for the relaxed capacity constraints (5) that are eliminated. Only the constraint for resource h' remains:

(22) $\quad \sum_{j \in \mathcal{J}} \sum_{m \in \mathcal{M}_j} a_{h'm} x_{jm} \leq c_{h'}$

If we ignore the constant term $\sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h$ in the objective function, DLP-op and the formulation (21)-(22) with (6)-(7) are obviously very similar. In particular, the latter can be interpreted as the DLP-op formulation for a modified capacity control problem with revenues $r_j - \sum_{h \in \mathcal{H} \setminus \{h'\}} a_{hm} \pi_h$ and capacity only constrained for resource h' (see Kunnumkal and Topaloglu 2010 for a similar discussion regarding a different problem). With the objective function of this modified problem given by $V^{DLP-\text{mod}}\left(\mathbf{c}, t\right)$, we can rewrite (21) as $V^{DLP}\left(\mathbf{c}, t\right) = V^{DLP-\text{mod}}\left(\mathbf{c}, t\right) + \sum_{h \in \mathcal{H} \setminus \{h'\}} \pi_h c_h$.

For the modified problem, a corresponding dynamic program can be straightforwardly constructed as

$$(23) \quad \begin{aligned} V_{h'}\big(c_{h'},t\big) = \\ \sum_{j\in\mathcal{J}} p_j\big(t\big)\max\Big(V_{h'}\big(\mathbf{c},t-1\big),\, r_j - \\ \max_{m\in\mathcal{M}_j}\bigg(\sum_{h\in\mathcal{H}\setminus\{h'\}} a_{hm}\pi_h + V_{h'}\big(c_{h'}-a_{h'm},t-1\big)\bigg)\Big) \\ + p_0 V_{h'}\big(c_{h'},t-1\big) \end{aligned}$$

with the boundary conditions $V_{h'}\big(\mathbf{c},0\big)=0$ for $\mathbf{c}\geq 0$ and $V_{h'}\big(\mathbf{c},0\big)=-\infty$ otherwise. That is, if a resource other than h' – which is thus considered unrestricted – is to be used by a product, its bid price is subtracted from the product's revenue in both (21)-(22) with (6)-(7) and the Bellman equation (23). Since the DLP formulation is an upper bound for expected revenue, we obtain

$$(24) \quad \begin{aligned} V_{h'}\big(c_{h'},t\big) + \sum_{h\in\mathcal{H}\setminus\{h'\}} \pi_h c_h \leq \\ V^{DLP-\mathrm{mod}}\big(\mathbf{c},t\big) + \sum_{h\in\mathcal{H}\setminus\{h'\}} \pi_h c_h = V^{DLP}\big(\mathbf{c},t\big) \end{aligned}$$

$$\text{for all } t,\mathbf{c}$$

regarding the approximation at every resource h'. This proves that the upper bound obtained by our approach is tighter than the one obtained by simply applying DLP-op. $\quad\square$

A3 Illustrative example of the decomposition approach

We consider a network with four resources indexed by $\mathcal{H}=\{1,\dots,4\}$. There is a product $k\in\mathcal{J}$ with the specification options $\mathcal{M}_k=\{1,2,3\}$, which for the sake of simplicity are denoted by numbers. The first specification option requires one unit of capacity on the resources $h=1$ and $h=2$ (that is, $a_{h1}=1$ for $h=1,2$ and $a_{h1}=0$ for $h=3,4$), the second on the resources $h=2$ and $h=3$ (that is, $a_{h2}=1$ for $h=2,3$ and $a_{h2}=0$ for $h=1,4$), and the third on the resources $h=3$ and $h=4$ (that is, $a_{h3}=1$ for $h=3,4$ and $a_{h3}=0$ for $h=1,2$).
We subsequently consider the single-leg dynamic program (9) for resource $h'=2$ that results from applying the decomposition approach. In this dynamic program, any consumption of a unit from resource 2 is modeled explicitly, that is, c_2-1, while the consumption from the resources $r=1,3,4$ comes at the (static) per unit costs, respectively, π_1,π_3, and π_4 (see Figure 8). Now, if a

request for product k is going to be accepted, the specification is determined by solving $\max\big\{r_j-\pi_1+V_2\big(c_2-1,t-1\big),\, r_j-\pi_3+V_2\big(c_2-1, t-1\big),\, r_j-\pi_3-\pi_4\big\}$, which is the inner maximization term of (9), specifying the obtainable sum of current and future revenue. The corresponding decision rule analogous to (2) is as follows: A request is accepted if $r_j\geq\min\big\{\pi_1+\big(V_2\big(c_2,t-1\big)-V_2\big(c_2-1,t-1\big)\big), \pi_3+\big(V_2\big(c_2,t-1\big)-V_2\big(c_2-1,t-1\big)\big),\, \pi_3+\pi_4\big\}$.
In specification options 1 and 2, the consumption from resource 1 and resource 3 is charged by the corresponding static bid price, respectively (π_1 or π_3), while consumption from resource $h'=2$ is modeled explicitly by calculating the opportunity cost of reducing capacity in the revenue-to-go function, that is, using $V_2\big(c_2,t-1\big)-V_2\big(c_2-1,t-1\big)$. In contrast, in specification option 3, the capacity is not considered at all in order to calculate the total opportunity cost. However, acceptance comes at the cost of the sum of the needed resources' bid prices (π_3 and π_4).

A4 Detailed demand information for Networks 1 & 2
Tables 2 and 3 contain detailed demand information for Network 1 and Network 2, respectively.

Table 2: Demand details (Network 1)

Booking class	Fare	Legs used	Exp. demand ($\alpha = 1.0$)
Y	500.00	1	4.00
Y	500.00	2	3.00
Y	500.00	3	3.50
M	417.50	1	8.00
M	417.50	2	6.00
M	417.50	3	7.00
B	335.00	1	8.00
B	335.00	2	6.00
B	335.00	3	7.00
Q	250.00	1	12.00
Q	250.00	2	9.00
Q	250.00	3	10.50
opaque	150.00	1 or 2 or 3	21.00

Figure 8: Illustration of the single-leg dynamic program for resource $h' = 2$ resulting from the decomposition approach for opaque products
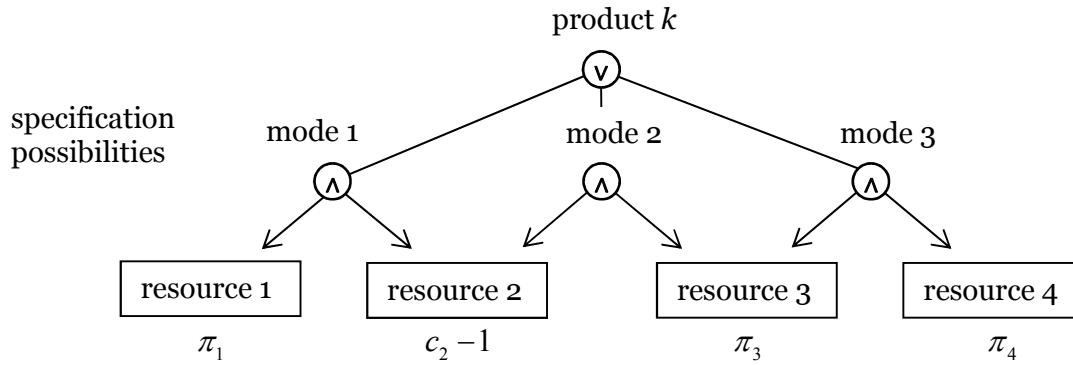

Table 3: Demand details (Network 2)

Booking class	Origin & dest.	Fare	Legs used	Exp. demand ($\alpha = 1.0$)	Booking class	Origin & dest.	Fare	Legs used	Exp. demand ($\alpha = 1.0$)
Y	A-B	900	1	30.00	Q	A-H	300	2	32.40
Q	A-B	540	1	40.00	Y	A-H	500	3	24.30
Y	A-B	800	2 & 4	13.50	Q	A-H	300	3	32.40
Q	A-B	480	2 & 4	18.00	opaque	A-H	220	2 or 3	48.60
Y	A-B	800	3 & 5	13.50	Y	H-B	500	4	31.50
Q	A-B	480	3 & 5	18.00	Q	H-B	300	4	42.00
opaque	A-B	352	1 or 2 & 4 or 3 & 5	57.00	Y	H-B	500	5	31.50
Y	A-C	800	2 & 6	7.20	Q	H-B	300	5	42.00
Q	A-C	480	2 & 6	9.60	opaque	H-B	220	4 or 5	63.00
Y	A-C	800	3 & 7	7.20	Y	H-C	500	6	16.80
Q	A-C	480	3 & 7	9.60	Q	H-C	300	6	22.40
opaque	A-C	352	2 & 6 or 3 & 7	14.40	Y	H-C	500	7	16.80
Y	A-H	500	2	24.30	Q	H-C	300	7	22.40
					opaque	H-C	220	6 or 7	33.60

A5 Detailed performance results

In this Appendix, we present detailed relative performance results of all mechanisms for $\alpha = 1.4$. Table 4 shows the relative performance results for Network 1. For example, the CEC column contains the percentual revenue gain over CEC. To avoid redundancy, only values for mechanisms that are usually superior to CEC are reported here. If, for example, DLP1 and CEC are to be compared, the fifth line of the DLP1 column contains the revenue gain of CEC over DLP1. Compared to Figure 2, the table allows us to gain more precise insights into how the mechanisms perform relative to each other, along with the corresponding 99% confidence intervals related to the resulting gains. Note that, in order to obtain valid confidence intervals, we use the same set of demand streams for all scenarios in the table. Likewise, Table 5 summarizes the relative performance for Network 2 with $\alpha = 1.4$.

Table 4: Relative performance of reoptimizing control methods at the 99% confidence level (Network 1, $\alpha = 1.4$)

% over	FCFS		DLP1		DLP3		DLP10		CEC		DECOMP1		DECOMP3		DECOMP10	
DP	15.90	±0.86	2.95	±0.49	3.86	±0.67	1.82	±0.39	0.53	±0.21	0.02	±0.08	0.01	±0.08	0.01	±0.08
DECOMP10	15.88	±0.86	2.94	±0.49	3.85	±0.68	1.81	±0.39	0.51	±0.22	0.01	±0.02	0.00	±0.01		
DECOMP3	15.88	±0.86	2.94	±0.49	3.84	±0.68	1.81	±0.39	0.51	±0.22	0.01	±0.02				
DECOMP1	15.87	±0.86	2.93	±0.49	3.84	±0.68	1.80	±0.40	0.51	±0.22						
CEC	15.29	±0.81	2.41	±0.51	3.31	±0.66	1.29	±0.36								
DLP10	13.82	±0.74	1.11	±0.56	2.00	±0.55										
DLP3	11.59	±0.85	-0.87	±0.74												
DLP1	12.58	±1.00														

Table 5: Relative performance of reoptimizing control methods at the 99% confidence level (Network 2, $\alpha = 1.4$)

% over	FCFS		DLP1		DLP3		DLP10		CEC		DECOMP1		DECOMP3	
DECOMP10	15.16	±0.41	11.59	±0.33	6.59	±0.37	2.33	±0.19	0.76	±0.11	0.06	±0.05	0.02	±0.03
DECOMP3	15.14	±0.41	11.57	±0.34	6.57	±0.37	2.31	±0.19	0.74	±0.11	0.04	±0.04		
DECOMP1	15.09	±0.41	11.52	±0.33	6.53	±0.37	2.27	±0.19	0.70	±0.11				
CEC	14.29	±0.36	10.74	±0.28	5.79	±0.33	1.56	±0.13						
DLP10	12.54	±0.35	9.05	±0.28	4.16	±0.31								
DLP3	8.04	±0.33	4.69	±0.27										
DLP1	3.21	±0.23												

References

Adelman, Daniel (2007): Dynamic Bid-Prices in Revenue Management, *Operations Research,* 55 (4): 647-661.

Anderson, Chris K. (2009): Setting Prices on Priceline, *Interfaces,* 39 (4): 307-315.

Bartodziej, Paul and Ulrich Derigs (2004): On an Experimental Algorithm for Revenue Management for Cargo Airlines, in: Celso C. Ribeiro and Simone L. Martins (eds.): *Proceedings of the 3rd International Workshop on Experimental and Efficient Algorithms (WEA),* Angra dos Reis, 57-71.

Bartodziej, Paul, Ulrich Derigs, and Markus Zils (2007): O&D Revenue Management in Cargo Airlines – A Mathematical Programming Approach, *OR Spectrum,* 29 (Special Issue 1): 105-121.

Bertsimas, Dimitris and Ioana Popescu (2003): Revenue Management in a Dynamic Network Environment, *Transportation Science,* 37 (3): 257-277.

Chen, Victoria C. P., Dirk Günther, and Ellis L. Johnson (2003): Routing Considerations in Airline Yield Management, in: Titio A. Ciriani, Georgio Fasano, Stefano Gliozzi, and Roberto Tadei (eds.): *Operations Research in Space and Air*, Kluwer: Boston, 333-350.

Chiang, Wen-Chyuan, Jason C. H. Chen, and Xiaojing Xu (2007): An Overview of Research on Revenue Management:Current Issues and Future Research, *International Journal of Revenue Management,* 1 (1): 97-128.

Erdelyi, Alexander and Huseyin Topaloglu (2010): A Dynamic Programming Decomposition Method for Making Overbooking Decisions Over an Airline Network, *INFORMS Journal on Computing,* 22 (3): 443-456.

Fay, Scott (2008): Selling an Opaque Product through an Intermediary: The Case of Disguising One's Product, *Journal of Retailing,* 84 (1): 59-75.

Fay, Scott and Jinhong Xie (2008): Probabilistic Goods: A Creative Way of Selling Products and Services, *Marketing Science,* 27 (4): 674-690.

Gallego, Guillermo, Garud Iyengar, Robert L. Phillips, and Abha Dubey (2004): Managing Flexible Products on a Network, CORC Technical Report TR-2004-01, IEOR Department, Columbia University.

Gallego, Guillermo and Robert Phillips (2004): Revenue Management of Flexible Products, *Manufacturing & Service Operations Management,* 6 (4): 321-337.

Jerath, Kinshuk, Serguei Netessine, and Senthil K. Veeraraghavan (2009): Selling to Strategic Customers: Opaque Selling Strategies, in: Serguei Netessine and Christopher S. Tang (eds.): *Consumer-Driven Demand and Operations Manage-*

ment Models, Springer: Dordrecht, 253-300.

Jerath, Kinshuk, Serguei Netessine, and Senthil K. Veeraraghavan (2010): Revenue Management with Strategic Customers: Last-Minute Selling and Opaque Selling, *Management Science,* 56 (3): 430-448.

Jiang, Yabing (2007): Price Discrimination with Opaque Products, *Journal of Revenue and Pricing Management,* 6 (2): 118-134.

Kimms, Alf and Michael Müller-Bungart (2007): Revenue Management for Broadcasting Commercials: The Channel's Problem of Selecting and Scheduling Advertisements to be Aired, *International Journal of Revenue Management,* 1 (1): 28-44.

Kunnumkal, Sumit and Huseyin Topaloglu (2010): A New Dynamic Programming Decomposition Method for the Network Revenue Management Problem with Customer Choice Behavior, *Production and Operations Management,* 19 (5): 575-590.

Liu, Qian and Garrett J. van Ryzin (2008): On the Choice-Based Linear Programming Model for Network Revenue Management, *Manufacturing & Service Operations Management,* 10 (2): 288-310.

McGill, Jeffrey I. and Garrett J. van Ryzin (1999): Revenue Management: Research Overview and Prospects, *Transportation Science,* 33 (2): 233-256.

Meissner, Jörn and Arne K. Strauss (2012): Network Revenue Management with Inventory-Sensitive Bid Prices and Customer Choice, *European Journal of Operational Research,* 216 (2): 459-468.

Miranda Bront, Juan J., Isabel Méndez-Díaz, and Gustavo Vulcano (2009): A Column Generation Algorithm for Choice-Based Network Revenue Management, *Operations Research,* 57 (3): 769-784.

Müller-Bungart, Michael (2007): *Revenue Management with Flexible Products: Models and Methods for the Broadcasting Industry*, Springer: Berlin.

Petrick, Anita, Claudius Steinhardt, Jochen Gönsch, and Robert Klein (2012): Using Flexible Products to Cope with Demand Uncertainty in Revenue Management, *OR Spectrum*, 34 (1): 215-242.

Petrick, Anita, Jochen Gönsch, Claudius Steinhardt, and Robert Klein (2010): Dynamic Control Mechanisms for Revenue Management with Flexible Products, *Computers & Operations Research,* 37 (11): 2027-2039.

Post, David (2010): Variable Opaque Products in the Airline Industry: A Tool to Fill the Gaps and Increase Revenues. *Journal of Revenue and Pricing Management*, 9 (4): 292-299.

Post, David and Martin Spann (2011): Improving Airline Revenues with Variable Opaque Products: "Blind Booking" at Germanwings. *Interfaces*, 42 (4): 329-338.

Phillips, Robert L. (2005): *Pricing and Revenue Optimization*, Stanford University Press: Stanford.

Shapiro, Dmitry and Xianwen Shi (2008): Market Segmentation: The Role of Opaque Travel Agencies, *Journal of Economics & Management Strategy,* 17 (4): 803-837.

Spengler, Thomas S., Stefan Rehkopf, and Thomas Volling (2007): Revenue Management in Make-to-Order Manufacturing – An Application to the Iron and Steel Industry, *OR Spectrum,* 29 (Special Issue 1): 157-171.

Subramanian, Janakiram, Shaler Stidham Jr., and Conrad J. Lautenbacher (1999): Airline Yield Management with Overbooking, Cancellations, and No-Shows, *Transportation Science,* 33 (2): 147-167.

Talluri, Kalyan T. (2001): Airline Revenue Management with Passenger Routing Control: A New Model with Solution Approaches, *International Journal of Services Technology and Management,* 2 (1/2): 102-115.

Talluri, Kalyan T. and Garrett J. van Ryzin (2004): *The Theory and Practice of Revenue Management*, Kluwer: Boston.

Weatherford, Lawrence R. and Samuel E. Bodily (1992): A Taxonomy and Research Overview of Perishable-Asset Revenue Management: Yield Management, Overbooking, and Pricing, *Operations Research,* 40 (5): 831-844.

Zhang, Dan and Daniel Adelman (2009): An Approximate Dynamic Programming Approach to Network Revenue Management with Customer Choice, *Transportation Science,* 43 (3): 381-394.

Biographies

Jochen Gönsch is currently a postdoc researcher at the University of Augsburg. He holds a diploma degree in computer science and business administration from the Technical University of Darmstadt (2006) and a doctoral degree from the University of Augsburg (2010). His current research focusses on revenue management, dynamic pricing, and approximate dynamic programming. In 2011, he was awarded the Science Award for Supply Chain Management by the German Logistics Association (BVL).

Claudius Steinhardt graduated in computer science and business administration at the Technical University of Darmstadt in 2006 and obtained a doctoral degree in 2010 from the University of Augsburg. Currently, he is a postdoc researcher with the Department of Analytics and Optimization at the University of Augsburg. His research interests include revenue management, pricing, and customer-centric modeling. He is co-author of the first German scientific textbook on revenue management.