# Improved Bid Prices for Choice-Based Network Revenue Management


Joern Meissner
Kuehne Logistics University, Hamburg, Germany
joe@meiss.com

Arne K. Strauss
Warwick Business School, The University of Warwick, United Kingdom



October 2010
Revision: May 2011, September 2011



One of the latest developments in network revenue management (RM) is the incorporation of customer purchase behavior via discrete choice models. Many authors presented control policies for the booking process that are expressed in terms of which combination of products to offer at a given point in time and given resource inventories. However, in many implemented RM systems—most notably in the hotel industry—bid price control is being used, and this entails the problem that the recommended combination of products as identified by these policies might not be representable through bid price control. If demand were independent from available product alternatives, an optimal choice of bid prices is to use the marginal value of capacity for each resource in the network. But under dependent demand, this is not necessarily the case. In fact, it seems that these bid prices are typically not restrictive enough and result in buy-down effects.

We propose (1) a simple and fast heuristic that iteratively improves on an initial guess for the bid price vector; this first guess could be, for example, dynamic estimates of the marginal value of capacity. Moreover, (2) we demonstrate that using these dynamic marginal capacity values directly as bid prices can lead to significant revenue loss as compared to using our heuristic to improve them. Finally, (3) we investigate numerically how much revenue performance is lost due to the confinement to product combinations that can be represented by a bid price.

The heuristic is not restricted to a particular choice model and can be combined with any method that provides us with estimates of the marginal values of capacity. In our numerical experiments, we test the heuristic on some popular networks examples taken from peer literature. We use a multinomial logit choice model which allows customers from different segments to have products in common that they consider to purchase. In most problem instances, our heuristic policy results in significant revenue gains over some currently available alternatives at low computational cost.



*Key words*: revenue management, network, bid prices, choice


---

## 1. Introduction

Network revenue management (RM) is concerned with managing demand for products that require capacity on one or several resources (e.g. a seat on each flight leg of a specific itinerary), with the objective to maximize revenue. A particular form of such demand management decisions is to control product availability over time. This sort of problem appears in the hotel, railway, car rental, tour operator business, but the air travel is perhaps the most well-known source for such problems and therefore we will stick to the terminology or the airline application. A product in this setting is composed of seats on one or more flight legs, potentially some fare rules and an associated fare. The firm faces stochastic demand and capacity of the resources is limited. In the context of this paper, we treat capacities as fixed, in particular, we do not consider overbooking.

The incorporation of choice behavior into network RM has increasingly gained attention in recent years as the means of segmentation erode in many markets. Traditional models for revenue management have worked under what is known as *independent demand* assumption. This assumption postulates that every customer is interested only in a single product and makes a purchase or no-purchase decision independent of other offers available by the firm or competitors. This assumption is reasonable if products are perfectly fenced, but with the severe cuts of fare restrictions that traditional airlines made in response to low-cost competition, it cannot be upheld in many markets. We still assume that customers' purchase behavior is myopic so that demand at any point in time does not depend on previous or anticipated future demand.

The problem can be formulated as a dynamic program, unfortunately one with a computationally intractable state-space even for networks of moderate size. Therefore, significant research has been devoted to approximating the value function to obtain heuristic policies. Many of this recent work proposes policies under the assumption that any combination of products can in principle be made available. However, in practice, this is not the case as some product combinations might not be representable via the two dominant methodologies, i.e., virtual nesting controls and bid prices. Adoption of one or the other method seems to be driven by corporate history rather than an informed choice, but it is noteworthy that in both cases bid prices have to be computed. For a detailed discussion see Chaneton and Vulcano (2009). These authors have recently proposed a stochastic gradient method based on simulated sample paths. The authors show that their algorithm converges under mild assumptions to a stationary point and improves previous methods to calculate bid prices.

In this paper, we present a simple, yet effective, way to improve bid prices that can be based on any choice-based network RM method that provides estimates of the marginal value of capacity, irrespective of the choice model (as long as it is reasonably fast to calculate purchase probabilities). The basic idea is to start with some initial bid price (based, e.g., on estimated marginal capacity values), and then to raise bid prices in a greedy fashion to exclude products that have a negative impact on overall profits because of buy-down effects. Numerical experiments confirm that this new method performs very well when compared with other available approaches.

## 2. Literature Review

Network Revenue Management (RM) are computationally intensive even without consideration of customer choice behavior, thus research has primarily concentrated on finding good heuristics. A comprehensive description of both scientific and applied RM can be found in the book of Talluri and van Ryzin (2004b), and the reader interested in a general overview of research over the last decades shall be referred to the reviews McGill and van Ryzin (1999) and Chiang et al. (2007). We focus in the following on papers closer related our approach.

Independent demand is a valid assumption in the case that customer segments are well fenced off, and recent work includes for example Adelman (2007) and Topaloglu (2009). Adelman (2007) proposes a time-dependent approximation and shows that upper bounds on the optimal objective value are tightened relative to the standard so-called deterministic linear programming (DLP) approach, and that the obtained policies perform better in a simulation study. Similarly, Topaloglu (2009) improves on the DLP by using Lagrangian relaxation to obtain a time- and inventory-level-dependent approximation. Farias and Van Roy (2007) introduce a linear programming approach to approximate dynamic programming that depends on both time and inventory level. The same approximation was independently proposed by Talluri (2008) who focuses on the relationships of upper bounds on the optimal objective value of the aforementioned approaches by Topaloglu and Adelman, respectively, as well as the DLP and a randomized linear programming model.

The earliest contributions to single leg RM with choice behavior include Brumelle et al. (1990) and Belobaba and Weatherford (1996), amongst others, and for networks the passenger origin and destination simulator studies by Belobaba and Hopperstad (1999). Zhang and Cooper (2005) consider an inventory control problem of a set of parallel flights including a customer choice model yielding a stochastic optimization problem which is being solved by simulation-based methods. Another simulation-based approach is given by van Ryzin and Vulcano (2008), who compute virtual nesting controls by constructing a stochastic steepest ascent algorithm designed to find stationary points of the expected revenue function.

The incorporation of choice behavior into network RM has increasingly gained attention as the means of segmentation erode in many markets by the arrival of competitors employing a low-cost strategy. In this situation, the inclusion of choice behavior becomes a crucial element for any RM system. Among the first approaches with a general model of customer choice is Talluri and van Ryzin (2004a) for a single flight leg problem. Among the techniques that have been proposed for the network context is the so-called choice-based linear program (CDLP) of Gallego and Phillips (2004). Based on this work, Liu and van Ryzin (2008) present an extension of the standard deterministic linear program approach to include choice behavior, albeit with customer segments that do not consider the same products. The result is an indication of the number of time periods out the finite time horizon that an offer set should be available. A dynamic programming decomposition approach is taken to obtain policies from the static solution of the CDLP and applied to the multinomial logit (MNL) choice model with disjoint consideration sets. Furthermore, the solution to the CDLP constitutes an upper bound on the optimal expected revenue. The notion of efficient sets introduced by Talluri and van Ryzin (2004a) for the single leg case is translated into the network context and these authors show that CDLP only uses efficient sets in its optimal solution. Unfortunately, for the network problem the optimal policy does not necessarily only use efficient sets like the single leg case, but Liu and van Ryzin (2008) can show asymptotic optimality of the CDLP which indicates that using efficient sets only might be a good choice. Kunnumkal and Topaloglu (2010) propose an alternative deterministic linear programming approach (ADLP) with very similar structure like the CDLP, but they try to address its shortcoming by calculating time dependent bid prices in contrast to the static ones produced by the CDLP. Although neither CDLP nor ADLP can be proven to be theoretically superior, numerical experiments indicate ADLP results in tighter upper bounds on the optimal expected revenue and better policies as well. Kunnumkal and Topaloglu (2010) also apply their model to the MNL choice model with disjoint consideration sets. Similar results like for the CDLP are presented, including asymptotic optimality, the fact that ADLP provides an upper bound on the objective value and a dynamic programming decomposition approach. The extension though comes at the cost of having significantly more constraints in the arising linear program. A generalization of the CDLP that can also handle the MNL choice model with overlapping consideration sets is presented in Miranda Bront et al. (2009), who employ column generation to solve the arising large linear program. Meissner and Strauss (2009) extend the approach of Adelman (2007) to include bid prices that depend on the remaining inventory level as well as time. Zhang and Cooper (2009) develop a pricing model for substitutable flights where customers choose among the available flights. This work differs from Zhang and Cooper (2005) in that they use pricing instead of availability control, they assume at most one customer arrival per time period as opposed to the previous block demand assumption, and does not assume a pre-determined order of arrivals. Talluri (2010) proposes a generalization of the randomized linear programming method that was developed in Talluri and van Ryzin (1999) to a customer choice framework. The resulting concave program can be solved as a series of linear programs with dynamic generation of cuts. Another mathematical programming approach to network RM is presented by Chen and de Mello (2010), where customer choice is formulated through the concept of preference orders.

Bid Prices in network revenue management have been discussed in Talluri and van Ryzin (1998) who show that the resulting control is not necessarily optimal. Prior simulations based methods include Topaloglu (2008) which builds on van Ryzin and Vulcano (2008), however these papers focus on optimizing virtual nesting rather than bid prices. The work of Chaneton and Vulcano (2009) is closest to ours in that they also focus on the optimization of bid prices under customer choice. They propose a stochastic gradient method to optimize bid prices based on simulated sample paths and show that the resulting algorithm converges under mild conditions.

We propose a greedy heuristic that iteratively improves on an initial bid price. The latter could be based on the estimate of the marginal value of capacity derived by any of aforementioned mathematical programming or decomposition approaches, for example. In our numerical experiments we use the dynamic programming decomposition of Miranda Bront et al. (2009) to obtain initial bid prices under the MNL choice model.

## 3. Problem Formulation

We face a network with $m$ resources—flights legs in the airline application—and $n$ products. A product $j$ is a seat on one or several flight legs and has a fixed fare $f_j$ and potentially some fare rules associated with it. The set of all products is denoted by $N = \{1, \dots, n\}$. Which resources a product requires is defined in a matrix $A \in \{0,1\}^{m,n}$ whose component $a_{ij}$ represents whether product $j$ requires resource $i$, so we assume that there are no group requests. We write $A_j$ for the $j$th column of $A$ and $A^i$ for its $i$th row. The notation $i \in A_j$ ($j \in A^i$) represents resources $i$ that are used by product $j$ (products $j$ that use resource $i$).

Customers arrive continuously over time while decisions on which products to offer are made at discrete points in time such that the time intervals are small enough to have a negligible probability that two or more arrivals occur. A customer arrives in time periods $t$ with probability $\lambda$. For the sake of simplicity, we assume that $\lambda$ is constant over all time periods. However, the extension to the time-heterogeneous case is not difficult. The decision time periods are indexed with $t$ starting at time $t = 1$ until the end of the booking horizon $t = \tau$. All flights depart at time $t = \tau + 1$. The index $t$ can also refer to the time interval between decisions at $t$ and $t + 1$ and will be clear from the context. Given that we offer a set $S \subset N$ of products at time $t$, a customer purchases product $j \in S$ with probability $P_j(S)$ and does not purchase with probability $P_0(S)$. The choice option $j = 0$ stands for the non-purchase alternative and can be used to reflect the attractiveness of competition. The choice probabilities are derived by some choice model such that $\sum_{j\in S} P_j(S) + P_0(S) = 1$ and do not depend on time (the same comments as for $\lambda$ apply concerning time-dependence). All customers show up and do not cancel so that no overbooking is required.

Each resource $i$ initially has a capacity $c_i$ available, and the state vector $x \in \mathbb{N}_0^m$ indicates how much inventory is still available. Although $x$ is clearly time-dependent, we do not use a subscript $t$ because it will be clear from the context. The remaining inventory also affects which products can be offered; since we exclude overbooking, we require that sufficient inventory must be available to provide a product. The set of all feasible products is then $N(x) = \{j \in N : A_j \le x\}$.

Let us denote the optimal expected revenue obtainable from time $t$ until the end of the booking horizon given remaining capacity $x$ by $v_t(x)$, usually referred to as the value function. A common assumption in recent work on this kind of network RM problem is that we can offer any combination of products at any time; subject to sufficient remaining inventory. Under this assumption, $v_t(x)$ can be written as follows:

$$\begin{aligned} v_t(x) &= \max_{S\subset N(x)} \left\{ \sum_{j\in S} \lambda P_j(S) \Big[ f_j + v_{t+1}(x - A_j) \Big] + \Big[ 1 - \lambda + \lambda P_0(S) \Big] v_{t+1}(x) \right\} \\ &= \max_{S\subset N(x)} \left\{ \sum_{j\in S} \lambda P_j(S) \Big[ f_j - \big( v_{t+1}(x) - v_{t+1}(x - A_j) \big) \Big] \right\} + v_{t+1}(x), \qquad \forall\, t, x. \end{aligned} \tag{1}$$

The boundary conditions are given by $v_{\tau+1}(x) = 0$ for all inventory states $x$. Note that the expression $\big(v_{t+1}(x) - v_{t+1}(x - A_j)\big)$ represents the opportunity cost of selling product $j$. If we have a good approximation of $v_t(x)$ for all time periods $t$ and all inventory state vectors $x$, then we can use this as an approximation of the opportunity cost. Various approximations have been proposed, see e.g. Miranda Bront et al. (2009), Liu and van Ryzin (2008), Meissner and Strauss (2009), or Zhang and Adelman (2009). They all construct a policy of the following kind:

$$S^* = \arg\max_{S\subset N(x)} \left\{ \sum_{j\in S} \lambda P_j(S) \Big[ f_j - \big( \text{approx. opportunity cost}(t, x, j) \big) \Big] \right\}. \tag{2}$$

In words, the policy recommends to offer the product set $S^*$ at time period $t$ when we have inventory $x$ still available in the network.

Problem (2) assumes that we are able to offer any combination of products for which we have sufficient inventory left, thus the maximization is over $S \subset N(x)$. However, this is not necessarily true if we are forced to use bid price control (e.g. owing to the restrictions imposed by global distribution systems). In this case, we need to set a bid price $b_i(t,x) \in \mathbb{R}$ at each time $t$ for each resource $i$ given remaining capacity $x$, which is subsequently used in a so-called bid price control:

**Definition 1 (Talluri and van Ryzin (2004b))** *A control $u(t,x,f)$ is a* bid-price control *if there exist real-valued functions $b(t,x) = [b_1(t,x),\dots,b_m(t,x)]$, $t = 1,2,\dots,\tau$, (called* bid prices*), such that*

$$u_j(t,x,f_j) = \begin{cases} 1 & \text{if } f_j > \sum_{i\in A_j} b_i(t,x),\ A_j \le x, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

*The control $u$ maps into a binary $m$-vector that represents which products shall be offered.*

Two central issues arise: Firstly, we need a way to compute bid prices dynamically. If demand were independent from the firm's control, then an optimal choice of bid prices (though not necessarily the optimal policy) would be to use the marginal value of capacity $\delta(t,x,i)$ for each resource $i$ since this ensures that each product with positive contribution $f_j - \sum_{i\in A_j}\delta(t,x,i)$ will be offered, where we approximated the opportunity cost with the sum of marginal capacity values. However, if the customers' choices do depend on the firm's control, it might be better to close some products—even though they might have a positive contribution—if this influences other purchase probabilities so as to improve the overall objective. Closing such a product could induce buy-up effects. We emphasize that bid prices $b$ and marginal values of capacity $\delta$ are not necessarily the same thing: the latter can be estimated by various methods and represents an estimate of the value of an additional unit of capacity for a resource, whereas bid prices are used only as a "product availability control" mechanism, and may differ significantly from $\delta$.

Secondly, if we are required to use a bid price control due to technical or other reasons, the formulation of the dynamic program (1) is incorrect in as far as some sets $S \subset N(x)$ might not be feasible under bid price control. Intuitively, the value function $v_t(x)$ in (1) therefore overestimates the "true" value function that we would obtain when accounting for bid price control because we are only able to offer combinations of products that can be represented by a bid price.

In the following, we elaborate on these two research questions. Our numerical experiments are based on the example of the Choice-based Deterministic Linear Program (CDLP) with dynamic programming decomposition as described in Miranda Bront et al. (2009)—we refer the interested reader to the appendix for details concerning CDLP. We keep the discussion general since our bid price policy below can be combined with other existing methods as well. Essentially, we only require a good estimate of the marginal value of capacity of each resource, and the ability to quickly evaluate choice probabilities under given bid prices.

## 4. Heuristic Bid Price Improvement

Let us begin with the problem of improving bid prices dynamically. We assume we used some solution approach to solve the dynamic program (1) approximately so that we have estimates of the marginal values of capacity $\delta(t,x,i)$. This assumption is justified by the availability of many methods that provide such estimates and that are working well in industry practice; for example, dynamic programming decomposition including customer choice is used by the optimizer of Lufthansa Systems as reported in Kemmer et al. (2010). These estimates we can use to approximate opportunity cost and to obtain a policy of the form (2). We seek to develop a heuristic that attempts to maximize the objective $\phi(b)$ over bid price vectors $b \in \mathbb{R}^m$; note that any bid price vector can be mapped into a corresponding offer set of products $S(b)$ using Definition 1. The objective function is given by

$$\phi(b) := \sum_{j \in S(b)} \lambda P_j(S(b)) \big(f_j - \sum_{i \in A_j} \delta(t,x,i)\big), \tag{4}$$

for given time period $t$ and remaining inventory $x$.

The underlying idea of the heuristic is to start with the estimate of the marginal values of capacity as bid price vector and then to iteratively improve it in a greedy fashion. This initial choice is motivated by the fact that this bid price vector offers exactly all products with positive displacement-adjusted revenues $f_j - \sum_{i \in A_j} \delta(t,x,i)$. If we use the Multinomial Logit (MNL) choice model with non-negative purchase preference values to define the choice probabilities $P_j$ (for an introduction to the MNL model see appendix), it is easy to see that adding a product to a certain product set $S$ cannot increase the purchase probability for any product in $S$. Therefore we would never offer a product with negative displacement-adjusted revenues since this negative contribution to the objective cannot be offset by an increase in purchase probability of the products with positive contribution. This means that we are in this situation only interested in increasing bid prices so as to gain potential improvements in the objective by closing products. Of course, we could also use a different choice model and/or start with any other vector $b \in \mathbb{R}^m$ as bid prices, and then should also consider decreasing bid prices. Our numerical tests (using MNL) showed no policy improvement from using zeros as initial bid price vector instead of using the best available marginal capacity value estimate; on the contrary, policy performance was worse. Note that the algorithm is guaranteed to stop after a finite number of iterations since the objective function has only a finite number of distinct values (at most as many as there are feasible offer sets).

We ensure that $S(b) \subset N(x)$ by setting the initial bid price for any resource $i$ with $x_i = 0$ sufficiently high so that no product using this leg is offered. The offer set $S(b)$ associated with $b_i(t,x) = \delta(t,x,i)$ is the set of all products whose contribution in terms of revenue minus estimated opportunity cost is positive, that is, $f_j - \sum_{i \in A_j} \delta(t,x,i) > 0$. Intuitively, this is probably not restrictive enough as this corresponds to the optimal bid price under independent demand assumption that

**Algorithm 1** Bid Price Heuristic under a General Choice Model for fixed $t$ and $x$

1: for all resources $i$ set bid price $b_i \leftarrow$ estimated marginal capacity value $\delta(t,x,i)$

2: **repeat**

3: **for all** $i = 1:m$ **do**

4: $\Delta b_i^+ = \min_{j \in A^i \cap S(b)} \{f_j - \sum_{k \in A_j} b_k\}$

5: $\Delta b_i^- = \min_{j \in A^i \cap \bar{S}(b)} \{\sum_{k \in A_j} b_k - f_j\} + \epsilon$

6: $\tilde{b}_+^i \leftarrow [b_1, \ldots, b_i + \Delta b_i^+, \ldots, b_m]$

7: $\tilde{b}_-^i \leftarrow [b_1, \ldots, b_i - \Delta b_i^-, \ldots, b_m]$

8: **end for**

9: $\hat{b} \leftarrow \arg\max\{\phi(\tilde{b}_+^1), \phi(\tilde{b}_-^1), \phi(\tilde{b}_+^2), \phi(\tilde{b}_-^2), \ldots, \phi(\tilde{b}_-^m)\}$

10: **if** $\phi(\hat{b}) > \phi(b)$ **then**

11: $b \leftarrow \hat{b}$

12: **end if**

13: **until** no further improvements

14: **return** bid price vector $b$

ignores buy-down effects. Hence, we wonder which products should be closed in order to achieve a higher objective. Closing a product can be accomplished by increasing a bid price; if we steadily increase $b_i(t,x)$, then we must reach a threshold where one (or more) products $j \in A^i$ are closed for the first time (unless all products using this leg are closed already, of course, in which case we do not change $b_i(t,x)$). We perform this smallest possible change of $b$ on every resource separately, obtain accordingly $m$ candidates, and select the one that yields the largest improvement in the objective. The same logic applies to reduction of a bid price so as to offer a product that is closed under the current bid price vector $b$. Note that we need to add a small positive number $\epsilon$ in line 5 to $\Delta b_i^-$ because at equality of fare and sum of bid prices, the corresponding product is still closed. The set $\bar{S}(b)$ denotes the set of products closed under the bid price vector $b$.

The heuristic is more formally given in Algorithm 1, where we suppress the dependence of the bid price vector $b$ on $t$ and $x$ to improve readability. Note that the objective function $\phi(b)$ that we attempt to maximize with the heuristic is based on revenue contributions derived from the estimated marginal values of capacity $\delta$, whereas the bid price $b$ is solely used as control on the availability of products as represented by the set $S(b)$.

We would solve Algorithm 1 at each time step to obtain improved bid prices that are used to control opening and closing of products, and thereby obtain a dynamic bid price policy. Also the

| Product | Flight legs | Fare $f_j$ |
|---|---|---|
| 1 | 1 | 25 |
| 2 | 2 | 20 |
| 3 | 1,2 | 30 |
| 4 | 1,2 | 60 |

**Table 1 Product definitions.**

| Segment $l$ | Consideration set $C_l$ | Preference vector $v_l$ |
|---|---|---|
| 1 | {1} | 1 |
| 2 | {2} | 1 |
| 3 | {3,4} | [1,2] |

**Table 2 Segment definitions.**

second central issue concerning the requirement of a feasible bid price control—as outlined in the previous section—is resolved since we optimize in the decision space of feasible bid prices rather than over all feasible product sets $S \subset N(x)$.

The heuristic can be applied to any choice-based method that provides estimates of marginal values of capacity, disregarding the choice model being used. Naturally, we have to restrict ourselves to choice models that allow quick evaluation of the purchase probabilities $P_j(S)$ since otherwise the evaluation of the objective function might be too expensive. Under the MNL choice model, offering (closing) another product cannot increase (decrease) the purchase probabilities of other available products. Therefore, it can never be optimal to offer a product with negative contribution $(f_j - \sum_{i\in A_j} \delta(x,i,t))$. However, for a different choice model this could be possible (because offering a product with negative contribution induces an increase in purchase for some other product with positive contribution, for whatever reason). However, this situation seems fairly unlikely to us, thus we confined ourselves to the consideration of closing products. The heuristic provides optimal bid prices under independent demand if the marginal capacity value estimates are sufficiently accurate since, in that case, the initial bid price vector is already optimal (though not necessarily the optimal policy).

Example 1. In this example we show that the objective function is in general not concave under the MNL choice model, and discuss a shortcoming of Algorithm 1. Consider a network with two flight legs, the first from A to B, the second from B to C. Each flight has plenty of remaining capacity left. The products are defined in Table 1. We use the MNL choice model as described in the appendix with parameters defined in Table 2 for three customer segments, arrival rates $\lambda = [1/3, 1/3, 1/3]$, and non-purchase preference $v_0 = [1,1,1]$. Note that product 4 is higher priced than product 3, yet customers from segment 3 prefer it over the cheaper alternative. Such situations might arise from the restrictions (or their absence) that are associated with the product. Let us assume that the current estimates of the marginal value of capacity are all zero. The objective function that should be maximized over $b \in \mathbb{R}^2$ is therefore

$$\phi(b) = \sum_{j\in S(b)} \sum_{l=1}^{3} \lambda_l P_{lj}(S(b)) f_j,$$

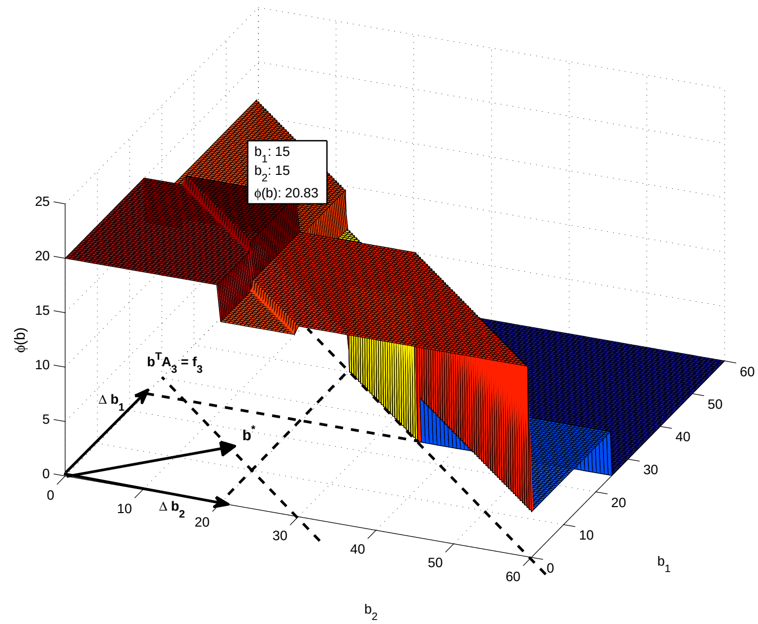


**Figure 1 Objective function for Example 1.**

and we depict it in Figure 1. The dotted lines show where the objective changes and are given by $b^T A_j = f_j$ for all products $j$. First, note that this example shows that the objective $\phi(b)$ is unfortunately in general not concave for the MNL model. Second, the lines $b^T A_j = f_j$ only intersect in the positive quadrant, hence there are no other function values in the three other quadrants than those depicted. Finally, we can observe a shortcoming of Algorithm 1: it would start from [0,0] and try out increases of the bid prices in directions $\Delta b_1$ and $\Delta b_2$, respectively. The objective function is lower in both directions than at the initial bid price vector, hence the algorithm would stop with bid prices unchanged at zero. However, the optimal solution is clearly to offer $\{1, 2, 4\}$ and could be reached by projecting the current bid price vector $b$ onto the space $b^T A_3 = f_3$, resulting in an optimal bid price vector $b^* = [15, 15]^T$. Algorithm 1 would be locally convergent if we would also check bid price moves along projections onto each $b^T A_j = f_j$ for all products $j$ that constitute facets on the set of bid prices that describe the current offer set. In our numerical experiments, we do not use this idea because it can be expected to not scale very well with increasing number of products. Still, for application to smaller networks this might very well be a useful addition.

## 5. Numerical Results

We tested various policies on three groups of test problems that are frequently being used in peer publications, see Liu and van Ryzin (2008), Miranda Bront et al. (2009), Chaneton and Vulcano (2009). These policies are:

1. *BP-MCV*: We solve the CDLP and use the optimal dual solution for the dynamic programming decomposition as it was proposed by Liu and van Ryzin (2008) and Miranda Bront et al. (2009) as outlined in the appendix. We obtain leg-level value function approximations $v^i$ for all legs $i$ as described in the appendix. Bid prices can be obtained by setting each bid price equal to the estimated leg-level marginal capacity value (MCV) unless there is no inventory left of resource $i$; in this case, we set the bid price sufficiently high so that the product cannot be offered:

$$b_i(t,x) = \begin{cases} v^i_{t+1}(x_i) - v^i_{t+1}(x_i - 1) & \text{for } x_i \geq 1, \\ \max_j f_j & \text{for } x_i = 0. \end{cases}$$

2. *BP-SG*: Stochastic gradient method with 4 re-optimizations as proposed by Chaneton and Vulcano (2009). This method is of interest particularly because it accounts for customer choice behavior, uses bid price control and is applicable to problems where the considered product sets of customer segments overlap.

3. *BP-Heu*: Our proposed heuristic method; bid prices are computed by Algorithm 1 using the dynamic marginal capacity value estimates from CDLP-based dynamic programming decomposition as outlined in the appendix. The heuristic is not being used in the dynamic programs arising from the decomposition, but only applied in implementing the policy.

4. *GOS*: This "General Offer Set" policy is also based on approximating the opportunity cost in the general policy (2) using CDLP-based dynamic programming decomposition as outlined in the appendix, however, we assume here that *any* combination of products can be offered. Therefore, GOS is *not* necessarily compatible with bid price control. We obtain a solution in terms of a combination of products by solving the following problem, using the heuristic from Miranda Bront et al. (2009):

$$\max_{S \subset N(x)} \sum_{j \in S} \lambda P_j(S) \left[ f_j - \sum_{i \in A_j} \left( v^i_{t+1}(x_i) - v^i_{t+1}(x_i - 1) \right) \right]. \tag{5}$$

This is the type of policy that was frequently used in recent work and serves us as an "upper bound" on policy performance because its feasible set is not constrained by the requirement that the offer set must be representable with a bid price. We write "upper bound" in quotation marks because the policies' revenue performances are measured by simulation and because (5) is being solved by a heuristic. Hence, the average revenues of GOS may occasionally be less than those of a bid price policy.

We consider GOS in order to address the question as to how much improvement is achievable if we could step away from bid price control in favor of product-level control where any combination of products can be offered. Computing optimal bid prices (optimal in the sense of a bid price vector

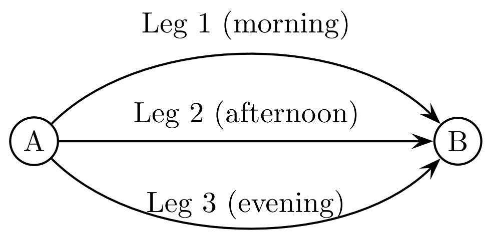


**Figure 2** **Parallel Flights Example.**

**Table 3** **Parallel Flights**

| Product | Leg | Class | Fare |
|---|---|---|---|
| 1 | 1 | L | 400 |
| 2 | 1 | H | 800 |
| 3 | 2 | L | 500 |
| 4 | 2 | H | 1000 |
| 5 | 3 | L | 300 |
| 6 | 3 | H | 600 |

Product definitions.

$b^*$ that maximizes $\phi(b)$ as defined in (4)) is difficult, so instead we compare the bid price policies defined above with GOS to illustrate the potential of improvement.

We test the different policies on three example networks that were used by Miranda Bront et al. (2009) and Chaneton and Vulcano (2009). All examples use the multinomial logit choice model with overlapping consideration sets which we specify in the appendix. For each test network, we compare the revenue performance of the four policies by varying leg capacities via a scaling factor $\alpha \in \{0.4, 0.6, 0.8, 1\}$ and by varying customers' no-purchase preferences $v_0$.

### 5.1. Parallel Flights Example

The first network example consists of three parallel flight legs as depicted in Figure 2 with initial leg capacity 30, 50 and 40, respectively. On each flight there is a low and a high fare class L and H, respectively, with fares as specified in Table 3. We define four customer segments in Table 4; note that we do not give the preference values for the no-purchase option at this point. This is because we consider various scenarios of this network by varying both the vector of no-purchase preferences and the network capacity. The sales horizon consists of 300 time periods.

We ran 2000 simulations with each bid price policy and report in Table 5 the average revenue results, load factors and percentage gaps of the average revenue achieved by BP-Heu relative to the other policies. Negative percentage deviations are enclosed by parentheses. The relative percentage

**Table 4 Parallel Flights**

| Segment | Consideration set | Pref. vector | $\lambda_l$ | Description |
|---|---|---|---|---|
| 1 | {2,4,6} | [5,10,1] | 0.1 | Price insensitive, afternoon preference |
| 2 | {1,3,5} | [5,1,10] | 0.15 | Price sensitive, evening preference |
| 3 | {1,2,3,4,5,6} | [10,8,6,4,3,1] | 0.2 | Early preference, price sensitive |
| 4 | {1,2,3,4,5,6} | [8,10,4,6,1,3] | 0.05 | Price insensitive, early preference |

Segment definitions.

error of these results was less than 0.5% with 95% confidence as reported in Table 6. BP-Heu yields an average improvement of 2.2% over BP-MCV and 0.5% over BP-SG. In fact, BP-Heu reaches in all cases the benchmark of the GOS. Note that the cases where BP-Heu slightly improves on GOS are not statistically significant. This behavior is caused by solving the dynamic policy problem (5) for GOS with the heuristic proposed in Miranda Bront et al. (2009), so that sometimes BP-Heu might yield better results). This implies that bid price control does not necessarily deteriorate revenues in this example; as long as the bid prices are being optimized. Apparently, this network has no products that use more than one resource. Therefore, the only combinations of products that cannot be offered under bid price control would be sets where we offer class L but close class H on some flight leg. This clearly would not be optimal, hence it is not surprising to see that bid price controls reach the revenue levels of GOS. On the other hand, choosing poor bid prices (for example, according to BP-MCV) has a potentially dramatic effect of up to almost 8% revenue loss as compared to BP-Heu.

The load factors in Table 5 indicate that BP-MCV is not restrictive enough as it results in the highest load factors but with, in some cases, very poor average revenues as compared to BP-Heu. This justifies our approach of closing some of the products offered under BP-MCV in order to induce buy-up behavior. As expected, the benchmark GOS results in the lowest load factors in combination with the highest revenues, which suggests that capacity utilization could be improved if we could do away with the bid price control structure. Similar observations can be made in the following examples.

### 5.2. Small Network Example

Next, we test the policies on a network with seven flight legs as depicted in Figure 3. In total, 22 products are defined in Table 7 and the network capacity is $c = [100, 150, 150, 150, 150, 80, 80]$, where $c_i$ is the initial seat capacity of flight leg $i$. In Table 8, we summarize the segment definitions according to desired origin-destination (O-D), price sensitivity and preference for earlier flights. The booking horizon has $\tau = 1000$ time periods.

**Table 5 Policy results for Parallel Flights Example**

| $\alpha$ | $v_0$ | GOS | LF | BP-MCV | LF | BP-SG | LF | BP-Heu | LF | $\Delta\frac{\text{BP-Heu}}{\text{BP-MCV}}$ | $\Delta\frac{\text{BP-Heu}}{\text{BP-SG}}$ | $\Delta\frac{\text{BP-Heu}}{\text{GOS}}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.4 | [1,5,5,1] | 38,974 | 0.99 | 39,158 | 1.00 | 39,115 | 1.00 | 38,974 | 0.99 | (0.47) | (0.36) | 0.00 |
| | [1,10,5,1] | 38,972 | 0.99 | 39,157 | 1.00 | 39,167 | 1.00 | 38,973 | 0.99 | (0.47) | (0.5) | 0.00 |
| | [5,20,10,5] | 36,955 | 0.99 | 36,534 | 0.99 | 37,027 | 0.99 | 36,957 | 0.99 | 1.16 | (0.19) | 0.01 |
| 0.6 | [1,5,5,1] | 55,967 | 0.98 | 53,957 | 0.99 | 55,964 | 0.99 | 55,965 | 0.99 | 3.72 | 0.00 | (0.00) |
| | [1,10,5,1] | 55,841 | 0.98 | 53,932 | 0.99 | 55,683 | 0.98 | 55,886 | 0.98 | 3.62 | 0.36 | 0.08 |
| | [5,20,10,5] | 51,360 | 0.95 | 52,395 | 0.99 | 52,029 | 0.98 | 51,402 | 0.95 | (1.89) | (1.21) | 0.08 |
| 0.8 | [1,5,5,1] | 69,573 | 0.96 | 69,804 | 0.99 | 69,024 | 0.99 | 69,673 | 0.96 | (0.19) | 0.94 | 0.14 |
| | [1,10,5,1] | 69,124 | 0.95 | 69,563 | 0.99 | 69,337 | 0.98 | 69,210 | 0.96 | (0.51) | (0.18) | 0.12 |
| | [5,20,10,5] | 60,056 | 0.90 | 59,167 | 0.92 | 59,132 | 0.92 | 60,056 | 0.90 | 1.50 | 1.56 | 0.00 |
| 1.0 | [1,5,5,1] | 76,979 | 0.95 | 71,268 | 0.97 | 75,073 | 0.96 | 76,746 | 0.96 | 7.69 | 2.23 | (0.30) |
| | [1,10,5,1] | 75,695 | 0.90 | 70,549 | 0.94 | 74,064 | 0.92 | 75,605 | 0.92 | 7.17 | 2.08 | (0.12) |
| | [5,20,10,5] | 62,599 | 0.77 | 59,850 | 0.79 | 62,105 | 0.76 | 62,603 | 0.77 | 4.60 | 0.80 | 0.01 |

LF: load factor. $\Delta\frac{a}{b} \equiv 100 * a/b - 100$: percentage gap. Results for BP-SG taken from Chaneton and Vulcano (2009).

**Table 6 Relative percentage errors for Parallel Flights Example assuming Bid Price Control**

| $\alpha$ | $v_0$ | GOS | BP-MCV | BP-Heu |
|---|---|---|---|---|
| 0.4 | [1,5,5,1] | 0.11 | 0.02 | 0.11 |
| | [1,10,5,1] | 0.11 | 0.02 | 0.11 |
| | [5,20,10,5] | 0.14 | 0.08 | 0.12 |
| 0.6 | [1,5,5,1] | 0.14 | 0.06 | 0.11 |
| | [1,10,5,1] | 0.15 | 0.07 | 0.12 |
| | [5,20,10,5] | 0.29 | 0.17 | 0.29 |
| 0.8 | [1,5,5,1] | 0.23 | 0.15 | 0.23 |
| | [1,10,5,1] | 0.25 | 0.17 | 0.24 |
| | [5,20,10,5] | 0.34 | 0.28 | 0.34 |
| 1.0 | [1,5,5,1] | 0.25 | 0.20 | 0.25 |
| | [1,10,5,1] | 0.30 | 0.24 | 0.31 |
| | [5,20,10,5] | 0.40 | 0.39 | 0.40 |

Relative percentage errors with 95% confidence based on 2000 simulations.

Table 9 contains the policy results of BP-MCV, BP-SG, BP-Heu and GOS with relative errors less than 0.5% with 95% confidence as reported in Table 10. BP-Heu yields an average improvement of 6.7% over BP-MCV and 1.1% over BP-SG. We observe that BP-Heu again achieves in most cases the revenue levels of GOS, that means, bid prices are also in this example mostly able to capture

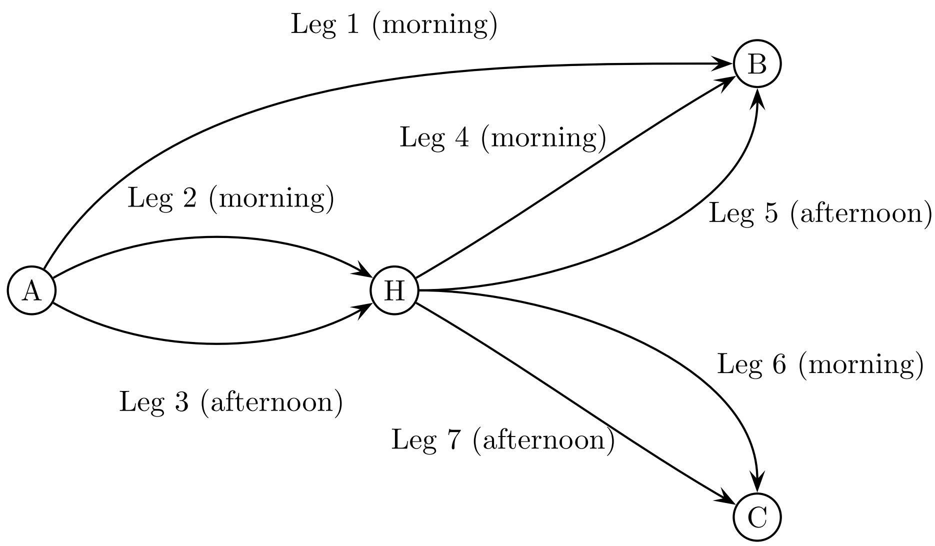


**Figure 3** **Small Network example.**

the best product combinations if they are properly optimized. However, the importance of further bid price optimization over simply choosing the marginal capacity values as in BP-MCV becomes apparent. The policies deliver similar results when the network is highly congested ($\alpha = 0.4$) since the optimal policy becomes simple then; we usually just offer the products with highest revenue. If capacity is less tight, the gains of BP-Heu over BP-MCV become very large, particularly for the cases of non-purchase preferences $[1,5]$ and $\alpha$ being 0.8 and 1. This illustrates that BP-MCV is not restrictive enough so that our heuristic can produce dramatic improvements, even over BP-SG. The last test instance, on the contrary, has high non-purchase preferences and large network capacity so that the capacity constraints are often not binding; accordingly, offering the unconstrained revenue maximizing set of fares is here usually optimal, and the policies do not differ much.

### 5.3. Hub & Spoke Network Example

Consider the Hub & Spoke network in Figure 4. It has eight flight legs, one hub and four spokes. Each flight $i$ has initial capacity $c_i = 200$ and the booking horizon is divided into $\tau = 2000$ time periods. There are 80 products in total which we define in Table 11 in the following way: products 1, 2, 3 and 4 correspond to the trip ATL-BOS using leg 3 in class Y, M, B and Q, respectively, product 5, 6, 7, 8 are BOS-ATL using leg 4 in class Y, M, B and Q, respectively, products 9, 10, 11 and 12 are ATL-LAX using leg 2 in class Y, M, B and Q, respectively, and so on. Definitions of the 20 customer segments for this example can be found in Table 12.

All simulation results in Table 13 have a relative error of less than 0.5% with 95% confidence as reported in Table 14. BP-Heu yields an average improvement of 3.8% over BP-MCV and 1.8% over

**Table 7 Small Network example**

| Product | Legs | Class | Fare | Product | Legs | Class | Fare |
|---|---|---|---|---|---|---|---|
| 1 | 1 | H | 1000 | 12 | 1 | L | 500 |
| 2 | 2 | H | 400 | 13 | 2 | L | 200 |
| 3 | 3 | H | 400 | 14 | 3 | L | 200 |
| 4 | 4 | H | 300 | 15 | 4 | L | 150 |
| 5 | 5 | H | 300 | 16 | 5 | L | 150 |
| 6 | 6 | H | 500 | 17 | 6 | L | 250 |
| 7 | 7 | H | 500 | 18 | 7 | L | 250 |
| 8 | 2,4 | H | 600 | 19 | 2,4 | L | 300 |
| 9 | 3,5 | H | 600 | 20 | 3,5 | L | 300 |
| 10 | 2,6 | H | 700 | 21 | 2,6 | L | 350 |
| 11 | 3,7 | H | 700 | 22 | 3,7 | L | 350 |

Product definitions.

**Table 8 Small Network example**

| Segment | O-D | Consideration set | Pref. vector | $\lambda_l$ | Description |
|---|---|---|---|---|---|
| 1 | A→B | {1,8,9,12,19,20} | (10,8,8,6,4,4) | 0.08 | less price sensitive, early pref. |
| 2 | A→B | {1,8,9,12,19,20} | (1,2,2,8,10,10) | 0.2 | price sensitive |
| 3 | A→H | {2,3,13,14} | (10,10,5,5) | 0.05 | less price sensitive |
| 4 | A→H | {2,3,13,14} | (2,2,10,10) | 0.2 | price sensitive |
| 5 | H→B | {4,5,15,16} | (10,10,5,5) | 0.1 | less price sensitive |
| 6 | H→B | {4,5,15,16} | (2,2,10,8) | 0.15 | price sensitive, slight early pref. |
| 7 | H→C | {6,7,17,18} | (10,8,5,5) | 0.02 | less price sensitive, slight early pref. |
| 8 | H→C | {6,7,17,18} | (2,2,10,8) | 0.05 | price sensitive |
| 9 | A→C | {10,11,21,22} | (10,8,5,5) | 0.02 | less price sensitive, slight early pref. |
| 10 | A→C | {10,11,21,22} | (2,2,10,10) | 0.04 | price sensitive |

Segment definitions.

BP-SG. The policy results show stable improvements of BP-Heu versus both BP-MCV and BP-SG in all instances, however, BP-Heu does not achieve the revenue performance of GOS in most cases. The fact that the Hub and Spoke Example has more multi-resource products than single-resource products might be a reason for these results, as it makes on the one hand the calibration of the best bid price vector more difficult, and might cause often situations where there is no bid price vector that would yield the offer set that GOS would recommend. However, it is not clear whether the under-performance of BP-Heu with respect to GOS particularly for the difficult instances corresponding to $[1,5]$ can be attributed to faults inherent to the heuristic itself, or to the fact that there exists no bid price that can represent the offer set recommended by GOS. In the last problem

**Table 9 Policy results for Small Network Example**

| $\alpha$ | $v_0$ | GOS | LF | BP-MCV | LF | BP-SG | LF | BP-Heu | LF | $\Delta\frac{\text{BP-Heu}}{\text{BP-MCV}}$ | $\Delta\frac{\text{BP-Heu}}{\text{BP-SG}}$ | $\Delta\frac{\text{BP-Heu}}{\text{GOS}}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | [1,5] | 149,300 | 0.98 | 149,287 | 0.99 | 149,693 | 0.99 | 149,287 | 0.98 | 0.00 | (0.27) | (0.00) |
| 0.4 | [5,10] | 144,193 | 0.98 | 142,572 | 0.99 | 145,010 | 0.98 | 144,218 | 0.98 | 1.15 | (0.55) | 0.02 |
| | [10,20] | 134,370 | 0.96 | 134,001 | 0.98 | 135,252 | 0.96 | 134,450 | 0.97 | 0.34 | (0.59) | 0.06 |
| | [1,5] | 213,237 | 0.95 | 211,041 | 0.98 | 212,459 | 0.97 | 213,071 | 0.95 | 0.96 | 0.29 | 0.00 |
| 0.6 | [5,10] | 193,402 | 0.94 | 179,728 | 0.97 | 195,037 | 0.95 | 193,260 | 0.95 | 7.53 | (0.91) | (0.07) |
| | [10,20] | 167,909 | 0.94 | 158,007 | 0.95 | 165,638 | 0.94 | 167,712 | 0.94 | 6.14 | 1.25 | (0.12) |
| | [1,5] | 262,421 | 0.90 | 217,577 | 0.96 | 238,041 | 0.94 | 262,408 | 0.90 | 20.60 | 10.24 | (0.01) |
| 0.8 | [5,10] | 220,631 | 0.93 | 198,690 | 0.94 | 214,847 | 0.92 | 220,112 | 0.93 | 10.78 | 2.45 | (0.24) |
| | [10,20] | 185,943 | 0.88 | 178,364 | 0.90 | 185,150 | 0.88 | 184,118 | 0.90 | 3.23 | (0.56) | (0.98) |
| | [1,5] | 278,927 | 0.85 | 225,061 | 0.92 | 272,569 | 0.86 | 278,930 | 0.92 | 23.94 | 2.33 | 0.00 |
| 1.0 | [5,10] | 233,700 | 0.87 | 218,324 | 0.90 | 231,094 | 0.88 | 229,178 | 0.90 | 4.97 | (0.83) | (1.93) |
| | [10,20] | 191,421 | 0.79 | 188,778 | 0.82 | 189,349 | 0.82 | 190,254 | 0.82 | 0.78 | 0.48 | (0.61) |

LF: load factor. $\Delta\frac{a}{b} \equiv 100 * a/b - 100$: percentage gap. Results of BP-SG taken from Chaneton and Vulcano (2009). The non-purchase preference vectors are abbreviated: e.g., [1,5] stands for [1,5,1,5,1,5,1,5,1,5].

**Table 10 Relative percentage errors for Small Network Example**

| $\alpha$ | $v_0$ | GOS | nSim | BP-MCV | nSim | BP-Heu | nSim |
|---|---|---|---|---|---|---|---|
| | [1,5] | 0.07 | 2000 | 0.06 | 2000 | 0.07 | 2000 |
| 0.4 | [5,10] | 0.07 | 2000 | 0.05 | 2000 | 0.20 | 200 |
| | [10,20] | 0.12 | 2000 | 0.10 | 2000 | 0.35 | 200 |
| | [1,5] | 0.25 | 200 | 0.05 | 2000 | 0.25 | 200 |
| 0.6 | [5,10] | 0.38 | 200 | 0.08 | 2000 | 0.36 | 200 |
| | [10,20] | 0.41 | 200 | 0.10 | 2000 | 0.41 | 200 |
| | [1,5] | 0.33 | 200 | 0.08 | 2000 | 0.31 | 200 |
| 0.8 | [5,10] | 0.37 | 200 | 0.09 | 2000 | 0.37 | 200 |
| | [10,20] | 0.45 | 200 | 0.14 | 2000 | 0.47 | 200 |
| | [1,5] | 0.35 | 200 | 0.09 | 2000 | 0.35 | 200 |
| 1.0 | [5,10] | 0.41 | 200 | 0.13 | 2000 | 0.42 | 200 |
| | [10,20] | 0.54 | 200 | 0.15 | 2000 | 0.50 | 200 |

Relative percentage error with 95% confidence and number of simulations (nSim) for each policy.

instance in Table 13 there is a slightly positive gap between BP-Heu and GOS. This is because we solved the dynamic policy problem (5) with the heuristic proposed in Miranda Bront et al. (2009), so that sometimes BP-Heu might yield better results. The difference is not statistically significant.

Algorithm 1 runs very quickly for the considered network examples; in the worst case, it took

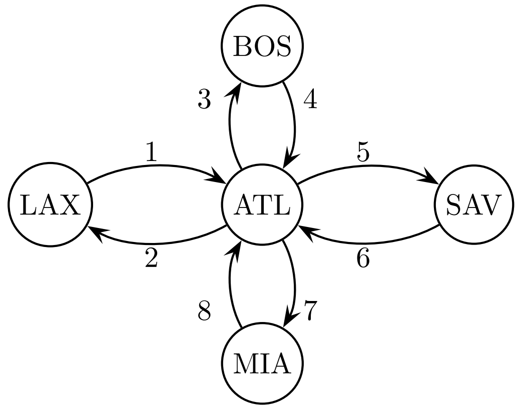


**Figure 4** **Hub & Spoke Network example.**

**Table 11** **Hub & Spoke Network Example**

| O-D Market | Legs | Revenue | | | |
|---|---|---|---|---|---|
| | | Y | M | B | Q |
| ATLBOS/BOSATL | 3/4 | 310 | 290 | 95 | 69 |
| ATLLAX/LAXATL | 2/1 | 455 | 391 | 142 | 122 |
| ATLMIA/MIAATL | 7/8 | 280 | 209 | 94 | 59 |
| ATLSAV/SAVATL | 5/6 | 159 | 140 | 64 | 49 |
| BOSLAX/LAXBOS | 4,2/1,3 | 575 | 380 | 159 | 139 |
| BOSMIA/MIABOS | 4,7/8,3 | 403 | 314 | 124 | 89 |
| BOSSAV/SAVBOS | 4,5/6,3 | 319 | 250 | 109 | 69 |
| LAXMIA/MIALAX | 1,7/8,2 | 477 | 239 | 139 | 119 |
| LAXSAV/SAVLAX | 1,5/6,2 | 502 | 450 | 154 | 134 |
| MIASAV/SAVMIA | 8,5/6,7 | 226 | 168 | 84 | 59 |

Product definitions.

about 0.1 seconds in Matlab on a 3GHz PC.

**Table 12 Hub & Spoke Network Example.**

| Segment | $C_l$ | $v_l$ | $\lambda_l$ | Segment | $C_l$ | $v_l$ | $\lambda_l$ |
|---|---|---|---|---|---|---|---|
| ATL/BOS H | {1,2,3,4} | {6,7,9,10} | 0.015 | BOS/MIA H | {41,42,43,44} | {6,7,10,10} | 0.008 |
| ATL/BOS L | {3,4} | {8,10} | 0.035 | BOS/MIA L | {43,44} | {8,10} | 0.03 |
| BOS/AT H | {5,6,7,8} | {6,7,9,10} | 0.015 | MIA/BOS H | {45,46,47,48} | {6,7,10,10} | 0.008 |
| BOS/ATL L | {7,8} | {8,10} | 0.035 | MIA/BOS L | {47,48} | {8,10} | 0.03 |
| ATL/LAX H | {9,10,11,12} | {5,6,9,10} | 0.01 | BOS/SAV H | {49,50,51,52} | {5,6,9,10} | 0.01 |
| ATL/LAX L | {11,12} | {10,10} | 0.04 | BOS/SAV L | {51,52} | {8,10} | 0.035 |
| LAX/ATL H | {13,14,15,16} | {5,6,9,10} | 0.01 | SAV/BOS H | {53,54,55,56} | {5,6,9,10} | 0.01 |
| LAX/ATL L | {15,16} | {10,10} | 0.04 | SAV/BOS L | {55,56} | {8,10} | 0.035 |
| ATL/MIA H | {17,18,19,20} | {5,5,10,10} | 0.012 | LAX/MIA H | {57,58,59,60} | {5,6,10,10} | 0.012 |
| ATL/MIA L | {19,20} | {8,10} | 0.035 | LAX/MIA L | {59,60} | {9,10} | 0.028 |
| MIA/ATL H | {21,22,23,24} | {5,5,10,10} | 0.012 | MIA/LAX H | {61,62,63,64} | {5,6,10,10} | 0.012 |
| MIA/ATL L | {23,24} | {8,10} | 0.035 | MIA/LAX L | {63,64} | {9,10} | 0.028 |
| ATL/SAV H | {25,26,27,28} | {4,5,8,9} | 0.01 | LAX/SAV H | {65,66,67,68} | {6,7,10,10} | 0.016 |
| ATL/SAV L | {27,28} | {7,10} | 0.03 | LAX/SAV L | {67,68} | {10,10} | 0.03 |
| SAV/ATL H | {29,30,31,32} | {4,5,8,9} | 0.01 | SAV/LAX H | {69,70,71,72} | {6,7,10,10} | 0.016 |
| SAV/ATL L | {31,32} | {7,10} | 0.03 | SAV/LAX L | {71,72} | {10,10} | 0.03 |
| BOS/LAX H | {33,34,35,36} | {5,5,7,10} | 0.01 | MIA/SAV H | {73,74,75,76} | {6,7,8,10} | 0.01 |
| BOS/LAX L | {35,36} | {9,10} | 0.032 | MIA/SAV L | {75,76} | {9,10} | 0.025 |
| LAX/BOS H | {37,38,39,40} | {5,5,7,10} | 0.01 | MIA/SAV H | {77,78,79,80} | {6,7,8,10} | 0.01 |
| LAX/BOS L | {39,40} | {9,10} | 0.032 | MIA/SAV L | {79,80} | {9,10} | 0.025 |

Segment definitions for Hub and Spoke Example.

**Table 13 Policy results for Hub & Spoke Example**

| $\alpha$ | $v_0$ | GOS | LF | BP-MCV | LF | BP-SG | LF | BP-Heu | LF | $\Delta\frac{\text{BP-Heu}}{\text{BP-MCV}}$ | $\Delta\frac{\text{BP-Heu}}{\text{BP-SG}}$ | $\Delta\frac{\text{BP-Heu}}{\text{GOS}}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.4 | [1,5] | 139,453 | 0.97 | 130,613 | 0.98 | 135,452 | 0.95 | 136,180 | 0.98 | 4.26 | 0.54 | (2.34) |
| | [5,10] | 112,730 | 0.97 | 109,532 | 0.98 | 110,287 | 0.93 | 112,322 | 0.97 | 2.55 | 1.85 | (0.37) |
| | [10,20] | 94,869 | 0.97 | 91,780 | 0.98 | 93,274 | 0.94 | 94,506 | 0.97 | 2.97 | 1.32 | (0.38) |
| 0.6 | [1,5] | 160,613 | 0.96 | 147,114 | 0.98 | 152,212 | 0.93 | 155,794 | 0.98 | 5.90 | 2.35 | (3.00) |
| | [5,10] | 130,483 | 0.97 | 123,394 | 0.98 | 126,816 | 0.95 | 128,726 | 0.97 | 4.32 | 1.51 | (1.35) |
| | [10,20] | 110,167 | 0.97 | 105,206 | 0.98 | 107,408 | 0.96 | 108,763 | 0.97 | 3.38 | 1.26 | (1.27) |
| 0.8 | [1,5] | 174,469 | 0.96 | 156,306 | 0.98 | 160,586 | 0.95 | 166,640 | 0.98 | 6.61 | 4.00 | (4.49) |
| | [5,10] | 144,039 | 0.97 | 135,892 | 0.98 | 139,038 | 0.96 | 140,972 | 0.97 | 3.74 | 1.39 | (2.13) |
| | [10,20] | 120,699 | 0.96 | 117,624 | 0.97 | 118,618 | 0.97 | 120,275 | 0.96 | 2.25 | 1.40 | (0.35) |
| 1.0 | [1,5] | 183,682 | 0.95 | 167,345 | 0.98 | 170,349 | 0.89 | 176,414 | 0.97 | 5.42 | 3.56 | (3.96) |
| | [5,10] | 153,932 | 0.94 | 147,014 | 0.97 | 150,021 | 0.97 | 152,049 | 0.96 | 3.42 | 1.35 | (1.22) |
| | [10,20] | 126,782 | 0.90 | 126,383 | 0.91 | 125,795 | 0.92 | 126,865 | 0.91 | 0.38 | 0.85 | 0.07 |

LF: load factor. $\Delta\frac{a}{b} \equiv 100 * a/b - 100$: percentage gap. Results for BP-SG taken from Chaneton and Vulcano (2009). The no-purchase preference vector are abbreviated; e.g., [1,5] represents the vector $[1,5,1,5,\ldots,1,5] \in \mathbb{R}^{40}$.

**Table 14** **Relative percentage errors for Hub and Spoke Example**

| $\alpha$ | $v_0$ | GOS | nSim | BP-MCV | nSim | BP-Heu | nSim |
|---|---|---|---|---|---|---|---|
| | [1,5] | 0.33 | 200 | 0.31 | 200 | 0.32 | 200 |
| 0.4 | [5,10] | 0.47 | 200 | 0.43 | 200 | 0.48 | 200 |
| | [10,20] | 0.52 | 200 | 0.47 | 200 | 0.52 | 200 |
| | [1,5] | 0.50 | 162 | 0.50 | 99 | 0.50 | 143 |
| 0.6 | [5,10] | 0.50 | 173 | 0.50 | 108 | 0.50 | 169 |
| | [10,20] | 0.50 | 174 | 0.50 | 133 | 0.50 | 166 |
| | [1,5] | 0.50 | 136 | 0.50 | 69 | 0.50 | 117 |
| 0.8 | [5,10] | 0.50 | 131 | 0.50 | 90 | 0.50 | 121 |
| | [10,20] | 0.50 | 154 | 0.50 | 111 | 0.50 | 142 |
| | [1,5] | 0.50 | 111 | 0.50 | 75 | 0.50 | 101 |
| 1.0 | [5,10] | 0.50 | 111 | 0.50 | 81 | 0.50 | 105 |
| | [10,20] | 0.50 | 146 | 0.50 | 127 | 0.50 | 138 |

Relative percentage errors with 95% confidence and number of simulations (nSim) for each policy.

## 6. Conclusion

We presented a simple and fast heuristic policy that can be used to iteratively improve bid prices based on an initial estimate of marginal values of capacity that need to be supplied by any of the available solution techniques. It can in principle be used with any choice model that allows for fast evaluation of the objective function of the dynamic policy problem and yields promising results for the considered test scenarios using the choice-based linear program (CDLP) with dynamic programming decomposition with the multinomial logit choice model.

The simplicity and flexibility of the approach make it attractive for practical application. It exploits that estimates of the marginal value of capacity are usually not sufficiently restrictive when demand is dependent on availability of alternative products, and accordingly tries to increase bid prices to avoid buy-down effects.

## References

Adelman, D. 2007. Dynamic bid prices in revenue management. *Operations Research* **55** 647–661.

Belobaba, P. P., C. Hopperstad. 1999. Boeing/MIT simulation study: PODS results update. *1999 AGIFORS Reservations and Yield Management Study Group Symposium Proceedings*.

Belobaba, P. P., L. R. Weatherford. 1996. Comparing decision rules that incorporate customer diversion in perishable asset revenue management situations. *Decision Sciences* **27**(2) 343–363.

Brumelle, S.L., J.I. McGill, T.H. Oum, K. Sawaki, M.W. Tretheway. 1990. Allocation of airline seats between stochastically dependent demands. *Transportation Science* **24** 183–192.

Chaneton, J.M., G. Vulcano. 2009. Computing bid-prices for revenue management under customer choice behavior. URL `http://pages.stern.nyu.edu/\~{}gvulcano/BidPricesChoice.pdf`. Working Paper.

Chen, L., T. Homem de Mello. 2010. Mathematical programming models for revenue management under customer choice. *European Journal of Operational Research* **203** 294–305.

Chiang, W.-C., J. C. H. Chen, X. Xu. 2007. An overview of research revenue management: current issues and future research. *International Journal of Revenue Management* **1** 97–128.

Farias, V. F., B. Van Roy. 2007. An approximate dynamic programming approach to network revenue management. Working paper.

Gallego, G., R. Phillips. 2004. Revenue management of flexible products. *Manufacturing & Service Operations Management* **6** 321–337.

Kemmer, P., T. Winter, A.K. Strauss. 2010. Decomposition techniques for market sensitive revenue optimization. *AGIFORS Annual Symposium*.

Kunnumkal, S., H. Topaloglu. 2010. A new dynamic programming decomposition method for the network revenue management problem with customer choice behavior. *Production and Operations Management* **19** 575–590.

Liu, Q., G. J. van Ryzin. 2008. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management* **10** 288–311.

McGill, J., G.J. van Ryzin. 1999. Revenue management: Research overview and prospects. *Transportation Science* **33** 233–256.

Meissner, J., A.K. Strauss. 2009. Network revenue management with inventory-sensitive bid prices and customer choice. Working Paper, Lancaster University Management School.

Miranda Bront, J. J., I. Méndez-Díaz, G. Vulcano. 2009. A column generation algorithm for choice-based network revenue management. *Operations Research* **57** 769–784. URL `http://pages.stern.nyu.edu/\~{}gvulcano/ColGenChoiceRM-Rev2.pdf`.

Talluri, K. 2008. On bounds for network revenue management. Working Paper, Universitat Pompeu Fabra.

Talluri, K. 2010. A randomized concave programming method for choice network revenue management. Working Paper.

Talluri, K., G. van Ryzin. 1999. A randomized linear programming method for computing network bid prices. *Transportation Science* **33** 207–216.

Talluri, K., G. J. van Ryzin. 1998. An analysis of bid-price controls for network revenue management. *Management Science* **44** 1577–1593.

Talluri, K., G. J. van Ryzin. 2004a. Revenue management under a general discrete choice model of consumer behavior. *Management Science* **50** 15–33.

Talluri, K., G. J. van Ryzin. 2004b. *The Theory and Practice of Revenue Management*. Springer, New York.

Topaloglu, H. 2008. Stochastic approximation method to compute bid prices. *INFORMS Journal of Computing* **20** 596–610.

Topaloglu, H. 2009. Using Lagrangian relaxation to compute capacity-dependent bid prices in network revenue management. *Operations Research* **57** 637–649.

van Ryzin, G. J., G. Vulcano. 2008. Computing virtual nesting controls for network revenue management under customer choice behavior. *Manufacturing & Service Operations Management* **10** 448–467.

Zhang, D., D. Adelman. 2009. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science* **43** 381–394.

Zhang, D., W. Cooper. 2005. Revenue management for parallel flights with customer choice behavior. *Operations Research* **53** 415–431.

Zhang, D., W.L. Cooper. 2009. Pricing substitutable flights in airline revenue management. *European Journal of Operational Research* **197** 848–861.

# Appendix
## CDLP with Dynamic Programming Decomposition

Denote the *expected total revenue* from offering $S$ by

$$R(S) = \sum_{j \in S} P_j(S) f_j,$$

and the *expected total consumption of resource* $i$ from offering $S$ by

$$Q_i(S) = \sum_{j \in S} P_j(S) a_{ij}, \qquad \forall i \in \{1, \dots, m\}.$$

Then the choice-based deterministic linear program (CDLP) is given by

$$\begin{aligned} z_{\text{CDLP}} = \max_h & \sum_{S \subset N} \lambda R(S) h(S) \\ & \sum_{S \subset N} \lambda Q_i(S) h(S) \le c_i, && \forall i \in \{1, \dots, m\}, \\ & \sum_{S \subset N} h(S) = \tau, \\ & h(S) \ge 0, && \forall S \in N. \end{aligned}$$

CDLP has $2^n - 1$ decision variables corresponding to the all possible offer sets, however, only $m+1$ constraints. Therefore, column generation can be used to solve (CDLP): We start from a pool of columns that allows a feasible solution (e.g., using $h(\emptyset) = \tau$ and $h(S) = 0$ for all other $S \subset N$), and solve that reduced master problem. The dual solution can be used to compute the reduced profit of any other column, and we solve a small maximization problem to identify the column with highest "reduced profit" that we subsequently add to the master problem. This process is repeated until no further column can be found with positive reduced profit, in which case an optimal solution has been identified. Let us denote the dual solution corresponding to the capacity and time constraints by the vector $\eta \in \mathbb{R}_+^m$ and $\sigma \in \mathbb{R}_+$, respectively. The reduced profit maximization is then

$$\lambda \max_{S \subset N} \{R(S) - \sum_{i=1}^{m} Q_i(S) \eta_i\} - \sigma.$$

### Dynamic Programming Decomposition

The optimal dual variables of the capacity constraints in the CDLP can be used to estimate the marginal value of capacity on each resource, however, they suffer from the static nature of the model, namely, that there is no dependency on the time or inventory. Liu and van Ryzin (2008) proposed to introduce time- and inventory dependence by using a techniques called dynamic programming decomposition. After solving CDLP and obtaining an optimal dual solution $(\eta^*, \sigma^*)$, we decompose the network by the resource and approximate the value function $v_t(x) \approx v_t^i(x_i) + \sum_{k \neq i} \eta_k^* x_k$ for each resource $i$. When we substitute this approximation into the optimal dynamic programming formulation we obtain one-dimensional resource-level dynamic programs:

$$v_t^i(x_i) = \max_{S \subset N} \left\{ \sum_{j \in S} \lambda P_j(S) \left( f_j - \sum_{k \in A_j, k \neq i} \eta_k^* - \left( v_{t+1}^i(x_i) - v_{t+1}^i(x_i - 1) \right) \right) \right\} + v_{t+1}^i(x_i), \qquad (6)$$

with boundary condition $v_{\tau+1}^i(x_i) = 0$ for all $x_i$ and $v_t^i(0) = 0$ for all $t$. Having computed $v^i(\cdot)$ for all resources $i$, we can approximate the network value function by $v_t(x) \approx \sum_i v_t^i(x_i)$.

**Multinomial Logit Choice Model**

Multinomial Logit (MNL) is a random utility model for a finite number of alternatives: a customer is assumed to have some utility valuation $U_j$ for each product $j$ that can be decomposed into a deterministic component $\nu_j$ and a random component $\xi_j$ so that $U_j = \nu_j + \xi_j$. Usually, one assumes $\nu_j$ to be linear in a number of attributes such as price, service quality etc. The random variables $\xi_j$ are independent identically distributed random variables with a Gumbel distribution with zero mean and variance $(\mu\pi)^2/6$ for some scaling parameter $\mu$, and with $\pi$ denoting the constant 3.1415... Under this assumption, the purchase probability for product $j$ is given by

$$P_j(S) = \frac{\exp(\nu_j/\mu)}{\sum_{k \in S} \exp(\nu_k/\mu) + \exp(\nu_0/\mu)} = \frac{v_j}{\sum_{k \in S} v_k + v_0},$$

where $v_j := \exp(\nu_j/\mu)$ represents the preference of the customer for product $j$ and $j = 0$ stands for the non-purchase option. We remark that the quantity $v_0$ can also be used to include the influence of competition on the decision in that it may reflect the attractiveness of competitive products.

We divide customers into $L$ segments, where customers within a given segment $l \in \{1, \ldots, L\}$ are assumed to be homogenous in that they all consider the same set of products $C_l \subset N$ for purchase—the so-called consideration set—and product preferences $v_{lj}$ for all products $j \in C_l$ in their consideration set. The means of segmentation are left unspecified; they could be based, for example, on itinerary and departure time (early morning, midday etc). The probability that a customer in segment $l$ purchases product $j \in S$ when we offer the fare set $S$ is given by $P_{lj}(S) = v_{lj}/(\sum_{k \in C_l \cap S} v_{lk} + v_{l0})$ for $S \subseteq N$, where $v_{l0}$ is the preference for not buying anything. An arriving customer belongs to segment $l$ with probability $p_l$ such that $\sum_l p_l = 1$, hence we can define arrival probabilities $\lambda_l := p_l \lambda$ for every segment where $\lambda$ is the probability that a customer arrives in a given time period. Taken together we have $\lambda = \sum_l \lambda_l$. For a given segment $l$, let the vector $u_l$ describe the product availability such that $u_{lj} = 1$ if product $j \in C_l$ is available and $u_{lj} = 0$ otherwise. Accordingly, the probability that a customer from segment $l$ purchases product $j$ can be rewritten in the following form:

$$P_{lj}(u_l) = \frac{u_{lj} v_{lj}}{\sum_{k \in C_l} u_{lk} v_{lk} + v_{l0}}.$$

If the consideration sets are allowed to *overlap*, then the firm cannot distinguish with certainty between different segments, and the purchase probability for product $j$ given the offer set $S$ and the arrival of a customer is defined by

$$P_j(S) = \sum_{l=1}^{L} p_l P_{lj}(u_l(S)).$$