Network Revenue Management with Inventory-Sensitive Bid Prices and Customer Choice

Joern Meissner, Arne K. Strauss
Department of Management Science, Lancaster University Management School, Lancaster, LA1 4YX, UK
joe@meiss.com, a.strauss@lancaster.ac.uk

March 19, 2009

We develop a new approximate dynamic programming approach to network revenue management models with customer choice that approximates the value function of the Markov decision process with a concave function which is separable across resource inventory levels. This approach reflects the intuitive interpretation of diminishing marginal utility of inventory levels and allows for significantly improved accuracy compared to currently available methods. The model allows for arbitrary aggregation of inventory units and thereby reduction of computational workload, yields upper bounds on the optimal expected revenue that are provably at least as tight as those obtained from previous approaches, and is asymptotically optimal under fluid scaling. Computational experiments for the multinomial logit choice model with distinct consideration sets show that policies derived from our approach outperform available alternatives, and we demonstrate how aggregation can be used to balance solution quality and runtime.

Key words: revenue management. dynamic programming/optimal control: applications, approximate.

---

1. Introduction

A particular area of revenue management (RM) that currently receives much interest is the approximate solution of the RM network problem including models of customer choice behavior. Network problems arise in many applications such as hospitality or transportation where the managed products might require more than one resource, for example a hotel that sells rooms over several nights. While network models have been around for some time already, only in recent years researchers devoted themselves to advancing discrete choice models where the purchase decisions also depend on the offered product alternatives. The need for such models is heightened by the rise of low cost service providers since they cut many of the traditional restrictions meant to segment the market, leaving the customer with similar products whose essentially only distinguishing feature is the price. Even if there are still some restrictions, customers increasingly tend to ignore them in their purchase decision so that in some business applications demand can only be observed for the product with the lowest available price, as pointed out by Boyd and Kallesen (2004). Such a behavior is in stark contrast to the traditional independent demand setting where it is assumed that demand is associated with a product and does not depend on market conditions such as which other products the firm offers. Therefore it is crucial to incorporate customer choice models into RM; more on the advantages of customer choice in the RM context can be found in van Ryzin (2005) and, for a comprehensive treatment of RM, in Talluri and van Ryzin (2004b).

We base our investigations on the particularly interesting work of Zhang and Adelman (2007) who extend the previous independent demand RM model of Adelman (2007) to incorporate customer choice behavior. Their approach differs from others in that they use an affine function of the state vector to approximate the value function of the exact dynamic programming formulation with a linear program (LP) in a way such that it yields time dependent bid prices. The optimal objective of this LP constitutes an upper bound on the exact optimal expected revenue which is tighter than those obtained by several other currently available methods. Since the LP possesses many variables, solving the problem by column generation is shown for the multinomial logit choice model (MNL) with disjoint consideration sets to reduce essentially to solving smaller mixed integer linear programs and is thus implementable in practice. They construct bid price policies directly from the dual solution as well as through a dynamic programming decomposition scheme and show that both perform very well. The most important reason for the improved performance is that the LP naturally generates time dependent bid prices which gives this approach a cutting edge compared to static bid price methods.

However, bid prices should not only depend on time to departure (for the ease of presentation we will stick to airline terminology), but also on the inventory levels because of the intuitive notion that marginal utilities decrease in inventory. The optimal expected revenue obtainable from having available network capacity x in time t is represented by $v_t(x)$. This value function $v_t(x)$ has the following concavity property: $v_t(x+e) - v_t(x)$ is non-increasing for any vector e with $e_i \in \{0, 1\}$, for all i, as pointed out in Farias and Van Roy (2007), for example. This dependence on intermediate capacity levels of the resources is not captured by current approaches to network RM with choice behavior. In the independent demand setting, a suitable approximation function was recently proposed by Farias and Van Roy (2007). Instead of using constraint generation to deal with the many constraints of the arising linear program they propose using a constraint sampling procedure which is based on the work of de Farias and Van Roy (2003) and de Farias and Van Roy (2004). The same approximation was independently proposed by Talluri (2008) under the name of *strong affine relaxation* and shown to provide tighter upper bounds on the optimal expected revenue than other available methods for the no-choice setting. Also Topaloglu (2007) recently focussed on time- *and* capacity-dependent bid prices: He proposed a network RM approach based on Lagrangian relaxation, but again without inclusion of choice behavior.

Our key contributions are the following:

- We propose a new linear programming approach to approximate dynamic programming that exhibits such a concavity property by approximating the value function with a concave function of the state vector which is separable over arbitrarily chosen ranges of resource inventory levels. As a special case, we can choose this approximation to be separable over each possible inventory level, which then corresponds to the approximation proposed by Farias and Van Roy (2007), but, in contrast to their approach, our model also accounts for customer choice behavior.

- We show that all the linear programs of van Ryzin and Liu (2008), Zhang and Adelman (2007) and Kunnumkal and Topaloglu (2008) can be seen as special cases of our linear programming formulation. In particular, for that reason we obtain tighter upper bounds on the objective value than these other approaches and that are asymptotically optimal as time horizon, demand and capacities are linearly scaled up.

- We prove that column generation essentially reduces to solving small mixed integer linear programs. Policies for the MNL model with disjoint consideration sets are numerically tested and show significantly improved results.

- Due to the larger number of constraints, our approach is considerably more expensive than others if we want bid prices to be able to change from any possible inventory level to another. However, we find that sensitivity to inventory levels is most pronounced only relatively close to the departures: Therefore, in order to cut down computational requirements for large networks without much deterioration of the solution quality, we can exploit the flexibility of our model with respect to arbitrary aggregations of inventory levels to solve it with high inventory aggregation at the beginning of the booking horizon, and later to re-solve it with lower aggregation and thus higher accuracy so that we capture the typically more pronounced concavity in inventory levels of the value function closer to the end of the time horizon.

- A seemingly new upper bound relationship between the approaches of Zhang and Adelman (2007) and Kunnumkal and Topaloglu (2008) is shown, namely that the former provides a tighter upper bound on the objective value than the latter.

The paper at hand is organized as follows: In the next section we briefly review the related literature, then in Section 3 we present our model including the required notation followed by the resulting Markov decision process and its equivalent linear programming form in Section 4. We introduce the linear programming models that we compare our approach with in Sections 4.1, 4.2

and 5. Our own approach is derived in Section 5 as well. We show that the column generation subproblem is reducible to a mixed integer linear program in Section 6 and describe bid price policies in Section 7. Finally, we present the computational results in Section 8 and conclude in Section 9.

2. Literature Review

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 PODS simulation 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 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. More contributions have been made, but we refer at this point to the literature reviews of McGill and van Ryzin (1999) and Chiang et al. (2007) and instead focus on papers closer related our approach. The underlying theory of approximate dynamic programming is presented in the well-written books of Bertsekas and Tsitsiklis (1996) and Powell (2007).

Network problems are computationally intensive even without consideration of customer choice behavior, thus good heuristics need to be found. Among the efficient techniques that have been proposed is the so-called choice-based linear program (CDLP) of Gallego et al. (2004). Based on this work, van Ryzin and Liu (2008) present an extension of the standard deterministic linear program approach to include choice behavior. It returns a vector with as many components as there are possible offer sets, and each component represents the number of time periods out the finite time horizon that the corresponding offer set should be available. The notion of *efficient sets* introduced by Talluri and van Ryzin (2004a) for the single leg case is translated into the network context and the authors show that CDLP only uses efficient sets in its optimal solution. Unfortunately, for the network problem the exact optimal policy does not necessarily only use efficient sets like the single leg case, but van Ryzin and Liu (2008) can show asymptotic optimality of the CDLP which indicates that using efficient sets only might be a good choice. 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. A generalization of the CDLP that can also handle the MNL choice model with overlapping consideration sets is presented in Bront et al. (2007), who employ column generation to solve the arising large linear program.

Kunnumkal and Topaloglu (2008) propose an alternative deterministic linear programming approach (ADLP) that exhibits a very similar structure like the CDLP, but they extend the latter to allow for time dependent bid prices in contrast to the static ones produced by the CDLP. Although no formulation can be proven to dominate the other, their numerical experiments indicate tighter upper bounds on the optimal expected revenue and better policies as well. They 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 comes at the cost of having significantly more constraints in the arising linear program.

3. Model

First, let us make some assumptions that are very common in the literature to facilitate the analysis, that is let there be neither cancellations nor no-shows (therefore no overbooking needed) and no competition. We describe our network RM model in the following, where the notation is geared to that of peer papers in this research area.

Products. Let our network consist of m resources –that means flight legs in the airline application– and n products. A product consists of a seat on one or several flight legs in combination with a fare class and departure date. Each resource i has a fixed capacity of c_i, and the network capacity is given by the corresponding vector $c = [c_1, \ldots, c_m]^T$. The capacity is homogenous, that means all seats are perfectly substitutable and do not differ, hence allowing us to accommodate all kind of requests from the given general capacity on a given flight leg. The set of products is denoted by $N = \{1, \ldots, n\}$. Every product j has an associated revenue f_j. By defining $a_{ij} = 1$ if resource i is used by product j, and $a_{ij} = 0$ otherwise, we obtain the incidence matrix $A = (a_{ij}) \in \{0,1\}^{m \times n}$ whose columns shall be denoted by A^j. We assume that each product uses at most one unit of any resource, so $a_{ij} \leq 1$. Group requests can easily accommodated by allowing a_{ij} to be larger than 1. This does not affect the analysis within this paper, however, we will stick to the assumption $a_{ij} \leq 1$ since it simplifies the notation for our proposed aggregated model in Section 5. Each column A^j gives us information about which resources product j uses, and accordingly we write $i \in A^j$ if resource i is being used by product j. The state of the system is given by the vector of unused capacity $x = [x_1, \ldots, x_m]^T$, and selling product j changes x to $x - A^j$.

Customer Choice. Potential customers usually do not come with a predetermined idea of which product to purchase. Rather, they only know some particular features that the product should possess and compare several alternatives that have these features in common before coming

Table 1 Notation.

m	Quantity of resources in the network
$c=[c_1,\dots,c_m]^T$	Vector of capacities
n	Products
f_j	Fare for product j
$A=(a_{ij})$	Incidence matrix, $a_{ij}>0$ if and only if product j needs a_{ij} units (integer) of resource i
A^j	jth column of A
τ	Amount of discrete time periods (departure in period $\tau+1$)
λ	Arrival probability of a customer in any time period
$S\subseteq N$	Offer set S out of the finite set N of all products
$P_j(S)$	General purchase probability of an arrived customer for product j given offer set S
$X=\{0,\dots,,c_1\}\times\cdots\times\{0,\dots,c_m\}$	State space

to a purchase or non-purchase decision. For example, a customer might be interested in a flight from A to B, but considers several flights with close-by departure times, or several class options. The probability that the customer chooses product j given the set of offered fares S (conditioned to arrival of a customer) is denoted by $P_j(S)$. We keep the choice model general until discussing the column generation procedure where we assume that customers choose according to the multinomial logit choice model with distinct consideration sets.

Time is discrete and partitioned into τ time periods that are small enough such that there is at most one customer arrival with probability λ and no arrival with probability $1-\lambda$. The notation is summarized in Table 1.

4. Current Solution Approaches

Let $v_t(x)$ denote the expected revenue-to-go from time period t until the final period τ, given the vector $x\in X:=\{0,\dots,,c_1\}\times\cdots\times\{0,\dots,c_m\}$ of still available resources in the network. The well-known optimality equation for maximizing expected revenue is then given by

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

where $N(x)=\{j\in N: x\geq A^j\}$ is the set of all feasible products to offer, and $v_{\tau+1}(x)=0$ for all x is the boundary condition. The decision to be made within each time period is which set of products to offer before we can observe demand in the corresponding period. In the independent demand model, in contrast, decisions and demand are decoupled. The problem (1) can –in theory–

be solved via backward dynamic programming, but for each time period the value function must be computed for all states which renders this method intractable in presence of a huge state space.

4.1. Choice-Based Deterministic LP

In order to reduce the problem to a tractable size, Gallego et al. (2004) and van Ryzin and Liu (2008) propose a choice-based deterministic linear program (CDLP) where demand is treated as known and being equal to its expected value. The problem reduces then to an allocation problem where we need to decide for how many time periods a certain set of products S shall be offered, denoted by $h(S)$. 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

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

Then the choice-based deterministic linear program is given by

$$\begin{aligned}
(\textbf{CDLP}) \quad z_{\text{CDLP}} = \max_h & \sum_{S \subseteq N} \lambda R(S) h(S) \\
& \sum_{S \subseteq N} \lambda Q_i(S) h(S) \leq c_i, && \forall i, \\
& \sum_{S \subseteq N} h(S) = \tau, \\
& h(S) \geq 0, && \forall S \subseteq N.
\end{aligned}$$

It essentially parallels the well-known deterministic LP for the no-choice case, provides with z_{CDLP} an asymptotically tight upper bound on the optimal expected revenue under fluid scaling and is very fast, but has the disadvantage that it does not provide the order in which the optimal sets shall be used since every order yields the same expected revenue under these model assumptions. The dual variables of the capacity constraints can be used to construct bid prices, however, they suffer from the static nature of the model, namely, that there is no dependency on the time. To remedy this shortcoming, van Ryzin and Liu (2008) propose a dynamic programming decomposition which transforms the static bid prices in dynamic ones. We outline this approach in Section 7.2.

4.2. Alternative Deterministic LP

Kunnumkal and Topaloglu (2008) addressed the issue of static bid prices obtainable from the CDLP by proposing an alternative deterministic linear program (ADLP) which generates bid prices that do depend on how much time is left until the time of departure. This formulation also results in an asymptotically tight upper bound on the optimal expected revenue, but none of the bounds generated by CDLP and ADLP dominate each other in general. However, their numerical experiments indicate that the ADLP can provide tighter bounds than the CDLP.

$$
\begin{aligned}
(\mathbf{ADLP}) \qquad & z_{\text{ADLP}} = \max_h \sum_{t=1}^{\tau} \sum_{S\subseteq N} \lambda R(S) h_t(S) \\
& \sum_{k=1}^{t-1} \sum_{S\subseteq N} \lambda Q_i(S) h_k(S) + \sum_{S\subseteq N} \mathbf{1}_{\{j\in S\}} a_{ij} h_t(S) \le c_i, && \forall i,t,j\in N, \\
& \sum_{S\subseteq N} h_t(S) = 1, && \forall t, \\
& h_t(S) \ge 0, && \forall S\subseteq N, t.
\end{aligned}
$$

Note that the CDLP and ADLP have a similar structure: While $h(S)$ was in the CDLP the scalar that indicated how much time to allocate over the full time horizon for the offer set S, in the ADLP, the variable $h_t(S)$ indicates how much time to allocate within time period t to the offer set S. We can also interpret $h_t(S)$ as the frequency of offering S in period t.

5. Approximation Based on the Equivalent LP

The following linear programming formulation will serve as the starting point of our considerations. It is equivalent to the dynamic program (1) and, for that reason, we denote it by (**EQ**). The equivalence can be derived from fundamental results of value iteration, see for example Powell (2007).

$$
\begin{aligned}
(\mathbf{EQ}) \quad & \min_{v(\cdot)} v_1(c) \\
& v_t(x) \ge \lambda \sum_{j\in S} P_j(S) \left[f_j - (v_{t+1}(x) - v_{t+1}(x - A^j)) \right] + v_{t+1}(x), \qquad \forall t, x, S\subseteq N(x).
\end{aligned}
$$

The decision variables are $v_t(x)$, for all t, x, and therefore the problem is also intractable for a large state space.

The basic idea is now to approximate $v_t(\cdot)$ by a given set of κ basis functions $\phi_k(\cdot)$ in order to reduce the number of variables:

$$
v_t(x) \approx \sum_{k=1}^{\kappa} V_{t,k} \phi_k(x), \qquad \forall\, t, x.
$$

Our approach is based on Zhang and Adelman (2007), who consider the affine approximation

$$
v_t(x) \approx \theta_t + \sum_{i=1}^{m} V_{t,i} x_i, \qquad \forall\, t, x,
$$

with boundary conditions $\theta_{\tau+1} = 0$ and $V_{\tau+1,i} = 0$ for all flight legs i. In this approximation, $V_{t,i}$ estimates the marginal inventory value on flight i in period t without taking into account how many seats are still available on this flight leg. Their basis functions are given by

$$\phi_k(x) := \begin{cases} x_i, & k = i \in \{1,\dots,m\}, \\ 1, & k = m+1. \end{cases}$$

Plugging the resulting approximation into (**EQ**) and constructing its dual yields the linear program:

$$\textbf{(AFF)} \quad z_{\text{AFF}} = \max_Y \sum_{t,x,S\subseteq N(x)} \sum_{j\in S} \lambda P_j(S) f_j Y_{t,x,S}$$

$$\sum_{x,S} x_i Y_{t,x,S} = \begin{cases} c_i, & \text{for } t = 1, \\ \sum_{x,S}(x_i - \sum_{j\in S} \lambda P_j(S) a_{ij}) Y_{t-1,x,S}, & \forall\, t = 2,\dots,\tau, \end{cases} \qquad \forall\, i, t,$$

$$\sum_{x,S} Y_{t,x,S} = \begin{cases} 1, & \text{for } t = 1, \\ \sum_{x,S} Y_{t-1,x,S}, & \forall\, t = 2,\dots,\tau, \end{cases}$$

$$Y_{t,x,S} \geq 0, \qquad \forall t, x, S \subseteq N(x).$$

Their approach can be seen as a special case of our approximation, which we elaborate in the following: Our essential refinement is to split the inventory of every resource i into K_i inventory level ranges, and then to assign for each range k a variable $V_{t,i,k}$ which estimates the marginal resource value at any inventory level within this range at time period t. The number of inventory levels contained within range k is denoted by s^i_k, and can reach from unit size 1 to resource capacity c_i. Note, in particular, that it can vary between resources. For notational convenience, we introduce for each resource i a function

$$r(\cdot) : \{0, 1, \dots, c_i\} \to \mathbb{N},$$

for which $r(0) := 0$ and for $x_i > 0$ we set $r(x_i) := k$ if and only if inventory level x_i is contained in range k. We approximate the value function with

$$v_t(x) \approx \theta_t + \sum_{i=1}^{m} \left[\sum_{k=1}^{r(x_i)-1} s^i_k V_{t,i,k} + (x_i - \sum_{k=1}^{r(x_i)-1} s^i_k) V_{t,i,r(x_i)} \right]. \tag{2}$$

The non-linear approximation has the particular advantage that it can reflect concavity of the value function across inventory levels, thus sometimes we refer to our approach as the *concave approximation.* On the boundary we define $V_{\tau+1,i,k} = 0$ for all i, k, and $\theta_{\tau+1} = 0$. Figure 1 gives three examples of how we could aggregate inventory levels: The solid line represents aggregation of the entire inventory of a resource, so that we only have one marginal inventory value $V_{t,i}$ for any inventory level. If done for all resources, then the problem reduces to the affine approximation by Zhang and Adelman (2007). On the other extreme, we might disaggregate completely so that

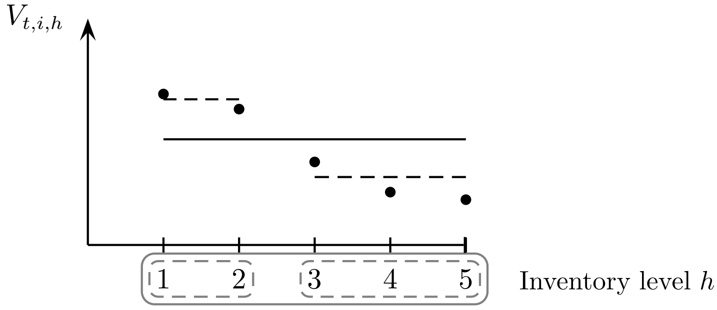

Figure 1 **Examples of inventory aggregations on a resource i with $c_i = 5$.**

we have a potentially different marginal value $V_{t,i,h}$ for each inventory level h, which would correspond to the dots in Figure 1, however, computationally it becomes quickly expensive to solve the associated problem for larger networks. Any other aggregation is possible, for example, we could split the inventory and obtain two dashed ranges. Furthermore, it is also possible to aggregate across time to further reduce the size of the linear program by exploiting that the marginal values of capacity typically stay nearly constant when there is much time left to departure. Likewise, inventory aggregations could be designed to change over time. In this paper, however, we stick to static inventory aggregation to increase readability.

Plugging the approximation (2) into (**EQ**) results in the following linear program, where we made use of the assumption $a_{ij} \leq 1$ in order to simplify notation, since it implies that $0 \leq r(x_i) - r(x_i - a_{ij}) \leq 1$.

$$
\begin{aligned}
(\mathbf{D})\ \min_{\theta,V} & \sum_{i=1}^{m}\sum_{k=1}^{K_i} s_k^i V_{1,i,k} + \theta_1 \\
& \theta_t - \theta_{t+1} + \sum_{i=1}^{m}\Bigg[\sum_{k=1}^{r(x_i)-1} s_k^i V_{t,i,k} + \Big(x_i - \sum_{k=1}^{r(x_i)-1} s_k^i\Big) V_{t,i,r(x_i)} - \sum_{k=1}^{r(x_i)-2} s_k^i V_{t+1,i,k} \\
& \quad + \Bigg(- s^i_{r(x_i)-1} + \lambda \sum_{j\in S} P_j(S) \mathbf{1}_{\{r(x_i - a_{ij}) < r(x_i)\}} \big[s^i_{r(x_i)-1} - x_i + a_{ij} \\
& \quad + \sum_{k=1}^{r(x_i - a_{ij})-1} s_k^i \big] \Bigg) V_{t+1,i,r(x_i)-1} + \Bigg(\Big(\sum_{k=1}^{r(x_i)-1} s_k^i - x_i \Big) + \lambda \sum_{j\in S} P_j(S) \big[x_i \\
& \quad - \sum_{k=1}^{r(x_i)-1} s_k^i - \Big(x_i - a_{ij} - \sum_{k=1}^{r(x_i-a_{ij})-1} s_k^i\Big) \mathbf{1}_{\{r(x_i - a_{ij}) = r(x_i)\}} \big] \Bigg) V_{t+1,i,r(x_i)} \Bigg] \\
& \geq \lambda \sum_{j\in S} P_j(S) f_j, \qquad \forall t, x, S \subseteq N(x).
\end{aligned}
$$

To increase readability, let us abbreviate the coefficients of $V_{t+1,i,r(x_i)-1}$ and $V_{t+1,i,r(x_i)}$ as stated in (**D**) by $\beta_{i,x,S}$ and $\gamma_{i,x,S}$, respectively. The dual of the above problem is given by:

$$
(\mathbf{P}) \quad z_{\text{conc}} = \max_{Y} \sum_{t,x,S} \lambda \sum_{j\in S} P_j(S) f_j Y_{t,x,S}
$$

$$\sum_{x,S}\Bigg(s^i_k\mathbf{1}_{\{k\le r(x_i)-1\}}+(x_i-\sum_{\tilde k=1}^{r(x_i)-1}s^i_{\tilde k})\mathbf{1}_{\{k=r(x_i)\}}\Bigg)Y_{t,x,S}=s^i_k, \qquad \text{for } t=1,\forall i,k,$$

$$\sum_{x,S}\Bigg(-s^i_k\mathbf{1}_{\{k\le r(x_i)-2\}}+\beta_{i,x,S}\mathbf{1}_{\{k=r(x_i)-1\}}+\gamma_{i,x,S}\mathbf{1}_{\{k=r(x_i)\}}\Bigg)Y_{t-1,x,S}+\sum_{x,S}\Bigg(s^i_k\mathbf{1}_{\{k\le r(x_i)-1\}}$$

$$+(x_i-\sum_{\tilde k=1}^{r(x_i)-1}s^i_{\tilde k})\mathbf{1}_{\{k=r(x_i)\}}\Bigg)Y_{t,x,S}=0, \qquad \text{for } t>1,\forall i,k,$$

$$\sum_{x,S}Y_{t,x,S}=1, \qquad \text{for } t=1,$$

$$\sum_{x,S}(Y_{t,x,S}-Y_{t-1,x,S})=0, \qquad \forall t>1,$$

$$Y_{t,x,S}\ge 0, \qquad \forall\, t,x,S\subseteq N(x).$$

We refer to (**P**) in the special case that $K_i=c_i$, $s^i_k=1$, $\forall\, k,i$ as CONC-c, and for $K_i=2$, $s^i_k=\lfloor c_i/2\rfloor$, $\forall\, k,i$ as CONC-2. To provide some intuition regarding constraints and variables of (**P**), note that the decision variables $Y_{t,x,S}$ can be interpreted as state-action probabilities since they are non-negative and satisfy $\sum_{x,S}Y_{t,x,S}=1$ for all t. With this in mind, the first set of constraints in (**P**) can be understood as the expected available capacity on resource i within inventory range k at the beginning of the booking horizon, which has to equal the size of the range s^i_k. The second set of constraints can similarly be seen as expected available capacity in range k on resource i at time t, which equals expected available capacity in that range at time $t-1$ minus expected consumption within period $t-1$.

Proposition 1*For an arbitrary inventory aggregation, any feasible solution to the corresponding linear program* (**P**) *yields a feasible solution to* (**AFF**) *having the same objective function value. We have the following upper bounds on the optimal expected revenue* $v_1(c)$*:*

$$z_{\text{CDLP}}\ge z_{\text{AFF}}\ge z_{\text{conc}}\ge v_1(c).$$

In particular, the objective in problem (**P**) *is asymptotically optimal.*

Proof. Suppose (Y) solves (**P**). In order to show the second inequality, we need to show that Y yields a feasible solution to (**AFF**) yielding the same objective function value. Apparent from feasibility to (**P**), we have

$$\sum_{x,S}Y_{t,x,S}=1, \qquad \text{for } t=1,$$

$$\sum_{x,S}Y_{t,x,S}=\sum_{x,S}Y_{t-1,x,S}, \qquad \forall\, t>1,$$

$$Y_{t,x,S}\ge 0, \qquad \forall\, t,x,S\subseteq N(x).$$

From the first set of constraints in (**P**), we obtain for a fixed resource i and $t=1$ by summation over all inventory level ranges $k \in \{1,\dots,K_i\}$:

$$\sum_{x,S}\Big(\sum_{k=1}^{r(x_i)-1} s_k^i + x_i - \sum_{k=1}^{r(x_i)-1} s_k^i\Big) Y_{t,x,S} = \sum_{k=1}^{K_i} s_k^i$$
$$\Leftrightarrow \sum_{x,S} Y_{t,x,S} x_i = c_i.$$

For $t>1$, fix a resource i and sum the second set of constraints in (**P**) over all ranges $k=1,\dots,K_i$:

$$\sum_{x,S} Y_{t,x,S} x_i + \sum_{x,S} Y_{t-1,x,S} \Bigg\{ -\sum_{k=1}^{r(x_i)-2} s_k^i + \mathbf{1}_{\{r(x_i)>1\}} \Bigg(-s^i_{r(x_i)-1} + \lambda \sum_{j\in S} P_j(S)[s^i_{r(x_i)-1} - x_i + a_{ij}$$
$$+ \sum_{k=1}^{r(x_i-a_{ij})-1} s_k^i]\mathbf{1}_{\{r(x_i-a_{ij})<r(x_i)\}}\Bigg) + \Bigg(\sum_{k=1}^{r(x_i)-1} s_k^i - x_i + \lambda \sum_{j\in S} P_j(S)[x_i - \sum_{k=1}^{r(x_i)-1} s_k^i$$
$$-(x_i - a_{ij} - \sum_{k=1}^{r(x_i-a_{ij})-1} s_k^i)\mathbf{1}_{\{r(x_i-a_{ij})=r(x_i)\}}]\Bigg)\Bigg\} = 0.$$

Consider the term in curly brackets:

- If $r(x_i)=0$, then $x_i=0$, and due to $S\subseteq N(x)$ we have $a_{ij}=0\,\forall j\in S$, resulting in $\{\cdots\}=-x_i$.
- If $r(x_i)=1$, then $\{\cdots\} = -x_i + \lambda\sum_j P_j(S)a_{ij}$ follows directly.
- If $r(x_i)\in\{2,\dots,K_i\}$: then we obtain the following: (without loss of generality, we assume $a_{ij}\le 1$ to simplify notation)

$$\{\cdots\} = -x_i + \lambda\sum_{j\in S} P_j(S)[(s^i_{r(x_i)-1} - x_i + a_{ij} + \sum_{k=1}^{r(x_i-a_{ij})-1} s_k^i)\mathbf{1}_{\{r(x_i-a_{ij})<r(x_i)\}}$$
$$+ x_i - \sum_{k=1}^{r(x_i)-1} s_k^i - (x_i - a_{ij} - \sum_{k=1}^{r(x_i-a_{ij})-1} s_k^i)\mathbf{1}_{\{r(x_i-a_{ij})=r(x_i)\}}]$$
$$= -x_i + \lambda\sum_{j\in S} P_j(S) a_{ij}.$$

Thus we obtain feasibility to (**AFF**), and therefore validity of the inequality $z_{\text{AFF}} \ge z_{\text{conc}}$. Alternatively, we can obtain this result from using the dual instead of the primal problem by starting from an optimal dual solution $(\theta^*, V^*_{t,i})$ to (**AFF**). Setting $\theta := \theta^*$ and $V_{t,i,k} := V^*_{t,i}$ yields a solution feasible to (**D**) with the same objective, yielding the desired result.

The last inequality $z_{\text{conc}} \ge v_1(c)$ follows from the fact that every feasible solution to (**EQ**) is an upper bound to the exact value function. This fact is a standard result in value iteration, see for example Theorem 3.4.1 in Powell (2007).

Zhang and Adelman (2007) showed that (**AFF**) has a tighter bound than the deterministic LP, and since van Ryzin and Liu (2008) proved that the deterministic LP is asymptotically optimal (that is, z_{CDLP} converges to $v_1(c)$ as demand, capacity and time horizon are linearly scaled up), both (**AFF**) and (**P**) are as well asymptotical optimal in that respect. □

Furthermore, we can also show that the affine approximation problem (**AFF**) results in a tighter upper bound on the optimal expected revenue than the ADLP, which likewise seems to be a new result.

Proposition 2*Any feasible solution to* **(AFF)** *yields a feasible solution to* (**ADLP**) *having the same objective value. We have the following bounds on the objective value $v_1(c)$:*

$$z_{\text{ADLP}} \geq z_{\text{AFF}} \geq v_1(c).$$

Proof. Let Y be a feasible solution to (**AFF**). We define

$$h_t(S) := \sum_x Y_{t,x,S}, \qquad \forall S \subseteq N, t,$$

and need to show that this is a feasible solution to (**ADLP**) with the same objective value.

First, we have directly from $Y \geq 0$ and the definition of $h_t(S)$ that $h_t(S) \geq 0$ for all S, t. Next, note that the second set of equality constraints in (**AFF**) actually reduces to the condition $\sum_{x,S \subseteq N(x)} Y_{t,x,S} = 1$ for all t. Using the definition of $h_t(S)$, we obtain

$$\sum_{S \subseteq N(x)} h_t(S) = \sum_{S \subseteq N} h_t(S) = 1,$$

for all t as required, where the first equality stems from $Y_{t,x,S} = 0$ if $S \nsubseteq N(x)$ because of Y's feasibility to (**AFF**). It remains to show that the first set of inequalities in (**ADLP**) holds, and that the objective value stays the same. As for the objective value, we defined earlier the total expected revenue from offering set S by $R(S) := \sum_{j \in S} P_j(S) f_j$. Substituting this into the objective in (**AFF**) and making use of the definition of $h_t(S)$ shows the equivalence of the objective. Finally, in order to show that the first set of inequalities in (**ADLP**) holds, we keep i fixed and sum the first set of equality constraints of (**AFF**) over time from 1 to some fixed $t \in \{1, \dots, \tau\}$:

$$\sum_{k=1}^{t} \sum_{x,S} x_i Y_{k,x,S} = c_i + \sum_{k=2}^{t} \sum_{x,S} x_i Y_{k-1,x,S} - \sum_{k=2}^{t} \sum_{x,S} \sum_{j \in S} \lambda P_j(S) a_{ij} Y_{k-1,x,S}.$$

Cancelling terms and rearranging yields

$$c_i = \sum_{x,S} x_i Y_{t,x,S} + \sum_{k=1}^{t-1} \sum_{S} \lambda Q_i(S) h_k(S),$$

where the total expected consumption on resource i is $Q_i(S) := \sum_{j \in S} P_j(S) a_{ij}$ as defined earlier. Due to the feasibility of Y to (**AFF**), $Y_{t,x,S} > 0$ only if $S \subseteq N(x) = \{j \in N : a_{ij} \leq x_i \, \forall i \in A^j\}$. Therefore we have

$$\sum_{x,S} x_i Y_{t,x,S} \geq \sum_{S} \mathbf{1}_{\{j \in S\}} a_{ij} h_t(S), \qquad \forall j \in N,$$

which concludes the proof. $\square$

From Proposition 1 and Proposition 2 it follows that our approach (**P**) provides also a tighter bound than the ADLP.

Corollary 1 *We have the following bounds on the optimal expected revenue $v_1(c)$:*

$$z_{\text{ADLP}} \geq z_{\text{conc}} \geq v_1(c).$$

6. Solution via Column Generation

The problem (**P**) has many variables and, for realistic network sizes, cannot be solved in moderate time unless techniques such as column generation are used to deal with problem size. This method builds upon the observation that for large problems most columns never enter the basis matrix and therefore do not need to be stored. Apparently, the main task is then to provide a way of how to find the next column to enter the basis without having to generate the whole coefficient matrix. We show in the following that if we use the *multinomial logit choice model with disjoint consideration sets* this so-called column generation subproblem reduces to solving a small linear mixed integer program. For the sake of improved readability, we confine ourselves to demonstrate the derivation of the subproblem for a special case only, namely $K_i = c_i$ for every resource i. For any other choice of aggregation, the derivation works in the same way. The considered case is the approximation proposed by Farias and Van Roy (2007) in the no-choice context which reflects the fact that the marginal value of capacity also depends on the quantity of remaining unused inventory. Our approximation for general aggregation (2) reduces in this case to

$$v_t(x) \approx \theta_t + \sum_{i=1}^{m} \sum_{k=1}^{x_i} V_{t,i,k},$$

with boundary conditions $V_{\tau+1,i,k} = 0$ for all i, k and $\theta_{\tau+1} = 0$. An initial feasible solution for the column generation procedure is given by

$$Y_{t,x,S} = \begin{cases} 1, & \text{if } x = c, S = \emptyset, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall\, t, x, S.$$

Next, we need to check for optimality and in case that it is not attained yet, we also need to find the next column that shall enter the basis. Given the dual values V at some iteration, this is achieved by finding the column with maximal reduced profit, the latter being given by

$$\max_{t,x,S \subseteq N(x)} \left(\lambda \sum_{j \in S} P_j(S) f_j - \sum_{i=1}^{m} \left[\sum_{k=1}^{x_i} V_{t,i,k} - \sum_{k=1}^{x_i - 1} V_{t+1,i,k} - (1 - \lambda \sum_{j \in S} P_j(S) a_{ij}) V_{t+1,i,x_i} \right] + \theta_{t+1} - \theta_t \right). \tag{3}$$

If the result is nonpositive then optimality has been reached, otherwise we add the corresponding column to the basis. Several variants of the column generation algorithm exist, for example, we could retain all columns that once entered the basis and thus obtain a system of growing size, or we could remove all columns that exit the basis, or use some other rule in between. The most important question, however, is whether the maximal reduced profit can be found quickly and inexpensively. The maximization in (3) could potentially be expensive to solve, so let us focus on this subproblem. Rearrangement of terms yields:

$$\max_{t,x,S\subseteq N(x)} \lambda \sum_{j\in S} P_j(S)(f_j - \sum_{i=1}^{m} a_{ij} V_{t+1,i,x_i}) - \sum_{i=1}^{m}\sum_{k=1}^{x_i}(V_{t,i,k} - V_{t+1,i,k}) + \theta_{t+1} - \theta_t. \quad (4)$$

Difficulties stem from the probability term $P_j(S)$ since it makes the problem nonlinear, and in particular the requirement $S \subseteq N(x)$ forces S to be dependent on x which makes the two variables non-separable. We consider choice probabilities $P_j(S)$ derived from the multinomial logit choice model with disjoint consideration sets. For this model, we divide customers into L segments, where customers within a given segment $l \in \{1,\dots,L\} =: \tilde{L}$ are considered to be homogenous in that they all have the same consideration set $C_l \subset N$ 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 base for example on itinerary and departure time (early morning, midday etc). We assume that the consideration sets are disjoint, that is $C_{l_1} \cap C_{l_2} = \emptyset$ for any segments $l_1 \neq l_2 \in \tilde{L}$. The probability that a customer in segment l purchases product j when we offer the fare set S is given by $P_{lj}(S) = v_{lj}/(\sum_{j\in C_l\cap S} v_{lj} + v_{l0})$ for $S \subseteq N$, where v_{l0} is the preference for not buying anything. These preference values could, for example, be derived from the reservation price of the segment for a particular product, and set equal to the maximum of this reservation price minus the actual price and zero. 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. 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:

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

The purchase probability for product j given the arrival of a customer is then defined by

$$P_j(S) = p_l P_{lj}(u_l(S)),$$

where $p_l = \lambda_l/\lambda$ and $u_l(S)$ is a vector with $u_{lj} = 1$ if $j \in S \cap C_l$ and $u_{lj} = 0$ otherwise. Plugging this choice probability into the column generation subproblem (4) and keeping the time t fixed results in a nonlinear maximization problem over the variables x and u. We perform a change of variables as done by Zhang and Adelman (2007) for a similar problem: Define $z_{lj} = u_{lj}/(\sum_{k\in C_l} v_{lk}u_{lk} + v_{l0})$ for all products $j \in C_l$ and all segments $l \in \tilde{L}$, furthermore, define $\alpha_l = 1/(\sum_{k\in C_l} v_{lk}u_{lk} + v_{l0})$ for all segments $l \in \tilde{L}$. The change of variables from u to z and α and a replacement of a nonlinear constraint with a linear one results in the mixed integer program stated below; we refer to Zhang and Adelman (2007) for details. We call this auxiliary problem (**AUX**) for reference.

$$(\mathbf{AUX}) \quad \max_{x,z,\alpha} \sum_{l\in\tilde{L}} \sum_{j\in C_l} \lambda_l v_{lj} \left[f_j - \sum_i a_{ij} V_{t+1,i,x_i} \right] z_{lj} - \sum_i \sum_{k=1}^{x_i} (V_{t,i,k} - V_{t+1,i,k}) + \theta_{t+1} - \theta_t$$

$$\begin{aligned}
&\sum_{j\in C_l} v_{lj} z_{lj} + v_{l0}\alpha_l = 1, && \forall\, l \in \tilde{L}, \\
&x_i \geq a_{ij} v_{l0} z_{lj}, && \forall\, i, j \in C_l, l \in \tilde{L}, \\
&x_i \in \{0,\ldots,c_i\}, && \forall\, i, \\
&z_{lj} \in \{0, \alpha_l\}, && \forall\, j \in C_l, l \in \tilde{L}, \\
&\alpha_l \geq 0, && \forall\, l \in \tilde{L}.
\end{aligned} \tag{5}$$

The parameters V_{t+1,i,x_i} depend on x, so some more auxiliary binary variables are needed to reformulate the problem as a linear mixed integer program:

Proposition 3 *Suppose the preference for non-purchase is positive for all segments, that means $v_{l0} > 0$ for all $l \in \tilde{L} := \{1,\ldots,L\}$, $a_{ij} \in \{0,1\}$ and let M be an arbitrary scalar greater than or equal to 1. We only need to solve the following linear mixed integer program to find the solution for problem (4) for each $t \geq 1$:*

$$\begin{aligned}
\max_{x,y,z,\alpha} \quad & \sum_l \sum_{j\in C_l} \sum_i (-\lambda_l v_{lj} a_{ij}) \left[V_{t+1,i,1} y_{lj}^{1,i} + \sum_{k=2}^{c_i} (V_{t+1,i,k} - V_{t+1,i,k-1}) y_{lj}^{ki} \right] + \\
& + \sum_l \sum_{j\in C_l} (\lambda_l v_{lj} f_j) z_{lj} + \sum_i \sum_{k=1}^{c_i} (V_{t+1,i,k} - V_{t,i,k}) x^{ki} + \theta_{t+1} - \theta_t
\end{aligned}$$

$$\begin{aligned}
&\sum_{j\in C_l} v_{lj} z_{lj} + v_{l0}\alpha_l = 1, && \forall\, l, \\
&\sum_{k=1}^{c_i} x^{ki} \geq a_{ij} v_{l0} z_{lj}, && \forall\, i, l, j \in C_l, && (6) \\
&x^{k-1,i} \geq x^{ki}, && \forall\, i, k \in \{2,\ldots,c_i\}, && (7) \\
&y_{lj}^{ki} \leq x^{ki}, && \forall\, l, j \in C_l, k, i, && (8) \\
&y_{lj}^{ki} \leq z_{lj}, && \forall\, l, j \in C_l, k, i, && (9) \\
&y_{lj}^{ki} \geq z_{lj} - M(1 - x^{ki}) && \forall\, l, j \in C_l, k, i, && (10) \\
&x^{ki} \in \{0,1\}, && \forall\, i, k \in \{1,\ldots,c_i\},
\end{aligned}$$

$$y_{lj}^{ki} \geq 0, \qquad \forall\, l, j \in C_l, k, i, \tag{11}$$

$$z_{lj} \in \{0, \alpha_l\}, \qquad \forall\, l, j \in C_l,$$

$$\alpha_l \geq 0, \qquad \forall\, l.$$

Proof. We start from problem (**AUX**) and introduce new binary variables $x^{ki} \in \{0,1\}$ for all $k \in \{1,\dots,n\}$ and all resources i such that $x_i = \sum_k x^{ki}$. With these new variables we can rewrite

$$\sum_i \sum_{k=1}^{x_i} (V_{t+1,i,k} - V_{t,i,k}) = \sum_i \sum_{k=1}^{c_i} (V_{t+1,i,k} - V_{t,i,k}) x^{ki}.$$

By imposing constraints (7) we ensure that x^{ki} is monotone decreasing in k for fixed i and hence that we have a one-to-one correspondence between a vector $[x^{1,i},\dots,x^{c_i,i}]$ and x_i. Furthermore,

$$V_{t+1,i,x_i} = \sum_{k=1}^{c_i-1} V_{t+1,i,k}(x^{ki} - x^{k+1,i}) + V_{t+1,i,c_i} x^{c_i,i}. \tag{12}$$

The constraints (5), which ensure that only allowable offer sets are used, become under the new variable (x^{ki}) the constraints (6). Note that by allowable offer sets we mean offer sets $S \subseteq N(x)$, that is we have sufficient capacity to accommodate at least one request for any product $j \in S$. We carry out the change of variables from x to (x^{ki}), which leaves us with an indefinite quadratic program featuring the nonlinear terms $x^{ki} z_{lj}$ (originating from substituting (12) for V_{t+1,i,x_i}) in the objective. Hence we further introduce new variables $y_{lj}^{ki} = x^{ki} z_{lj} \in \{0, z_{lj}\}$. Imposing the constraints (8–11) guarantees that $y_{lj}^{ki} = z_{lj}$ if $x^{ki} = 1$ and $y_{lj}^{ki} = 0$ otherwise. Since $z_{lj} \leq 1$ by definition, every $M \geq 1$ can be used in (10). □

7. Policies

In this section, we address the question of how the solution to (**D**) can actually be used to obtain a control policy that tells us which set of fares S to offer at any given time t and state x of the network. Again, in order to improve readability we discuss the policies for the entirely disaggregated case, that is $K_i = c_i$ for all resources i. For any other aggregation of inventory, similar policies can be derived by the same argumentation.

7.1. Bid Prices Directly from (D)

A standard approach of finding such a policy is to use the optimal solution V^* of (**D**) as a means to approximate the opportunity cost of the resources. In a given time period, we would only offer products j that have more revenue f_j associated with them than the sum of the marginal inventory values of the resources $i \in A^j$ that this product uses, evaluated at the respective resource inventory level. From those that have higher revenue, not necessarily all will be offered since the customer's choice probabilities $P_j(S)$ depend on the offer set. We need to solve a maximization problem for each segment l in each time period t and state x to obtain the optimal offer set S_l, and Bellman equation (1) indicates that it has the form

$$\arg\max_{S_l \subseteq C_l \cap N(x)} \sum_{j \in C_l} P_j(S_l) \left[f_j - \sum_{i=1}^{m} \sum_{k=x_i-a_{ij}+1}^{x_i} V^*_{t+1,i,k} \right]. \tag{13}$$

Note that we approximated the opportunity cost in the following way, using the optimal solution V^* to (**D**):

$$v_{t+1}(x) - v_{t+1}(x - A^j) \approx \sum_i \sum_{k=x_i-a_{ij}+1}^{x_i} V^*_{t+1,i,k}.$$

Furthermore, keep in mind that the consideration sets C_l are assumed to be disjoint, and thus segment-wise maximization is feasible. For the MNL choice model with preference vector v_l for segment l, rewriting the above maximization problem (13) in terms of a binary availability vector u_l yields a system of the form

$$\max_{u_{lj} \in \{0, \mathbf{1}_{\{x \geq A^j\}}\} \forall j \in C_l} \frac{\sum_{j \in C_l} v_{lj} u_{lj} w_j}{\sum_{j \in C_l} v_{lj} u_{lj} + v_{l0}} = \max_{u_l \in \{0,1\}^{|C_l|}} \frac{\sum_{j \in C_l} v_{lj} u_{lj} \mathbf{1}_{\{x \geq A^j\}} w_j}{\sum_{j \in C_l} v_{lj} u_{lj} \mathbf{1}_{\{x \geq A^j\}} + v_{l0}}, \qquad \forall x, t, \tag{14}$$

where w_j is the "worth" of product j, that means its revenue minus its approximated opportunity cost

$$w_j := f_j - \sum_i \sum_{k=x_i-a_{ij}+1}^{x_i} V^*_{t+1,i,k}.$$

The maximization (14) can be solved in the following way:

Proposition 4 *Consider the optimization problem (14). Rank the values w_j in a decreasing order; that is, $w_{[1]} \geq \cdots \geq w_{[i]} \geq \cdots \geq w_{[|C_l|]}$. Then there is a critical value h^*, $1 \leq h^* \leq |C_l|$, such that the optimal solution to the above problem is given by*

$$u^*_{lj} = \begin{cases} 1 & \text{if } w_j \geq w_{[h^*]} \text{ and } x \geq A^j, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. Defining $\tilde{v}_{lj} := v_{lj} \mathbf{1}_{\{x \geq A^j\}} \forall j \in C_l$ in the maximization (14), the ranking procedure follows by applying Proposition 6 in van Ryzin and Liu (2008). The optimal policy $\tilde{u}^*_l$ is such that

$$\tilde{u}^*_{lj} = \begin{cases} 1 & \text{if } w_j \geq w_{[h^*]}, \\ 0 & \text{otherwise.} \end{cases}$$

It is trivial that this policy leads to the same objective value like using the policy u^*_l as defined above for the original preference vector v_l. □

We denote this policy by D-CONC-c, where the "c" refers to the property that for each resource i inventory is split in as many ranges as we have capacity c_i. In Section 8 we present numerical results for this policy, as well as for a so-called D-CONC-2 policy. The latter relies on splitting the inventory of each resource i into two equal ranges and assuming that the marginal value of capacity $V_{t,i,a}$ and $V_{t,i,b}$ are constant across each range, respectively (see Figure 1). The policy D-CONC-2 is defined in the same way as D-CONC-c, except that the opportunity cost approximation is now $v_{t+1}(x) - v_{t+1}(x - A^j) \approx \sum_i \nu_i$, where ν_i is defined by

$$\nu_i = \begin{cases} a_{ij} V_{t+1,i,a}, & x_i \leq \lfloor c_i/2 \rfloor, \\ a_{ij} V_{t+1,i,b}, & x_i > \lfloor c_i/2 \rfloor, \end{cases} \text{ for all } i.$$

In the same manner, the solutions of (**CDLP**), (**AFF**) and (**ADLP**) can be used to construct policies based on the dual values of the corresponding capacity constraints. We call the resulting policies D-CDLP, D-AFF and D-ADLP, respectively.

7.2. Dynamic Programming Decomposition using CDLP

A popular method of solving network revenue management problems is to decompose them into a set of resource-level problems, that is for every resource i in the network we have one single leg problem with associated value function $v_t^i(x_i)$. One possible approach is to use the choice-based deterministic linear program which was introduced in Section 4.1: Given a resource i, we approximate the network value function by

$$v_t(x) \approx v_t^i(x_i) + \sum_{k \neq i} \pi_k^* x_k,$$

where π^* is the static vector of optimal bid prices obtained from (**CDLP**), that means the dual variables to the capacity constraints in (**CDLP**) at the optimal solution. We plug this approximation into the dynamic programming formulation (1) and obtain a one-dimensional problem with displacement-adjusted revenues $f_j - \sum_{k \neq i} \pi_k^* a_{kj}$ which can be quickly solving by backwards induction. Having done that for all resources i, the network value function is then approximated by

$$v_t(x) \approx \sum_i v_t^i(x_i).$$

Again plugging this approximation into the Bellman equation (1) yields a maximization like in (14) but with $w_j := f_j - \sum_i (v_t^i(x_i) - v_t^i(x_i - a_{ij}))$. We call this policy DP-CDLP and refer to van Ryzin and Liu (2008) for a more detailed discussion of this approach. Through this procedure we transform the static bid prices π into dynamic ones. However, their quality is based on the relatively crude opportunity cost approximation in terms of π; the bid prices resulting from our approach are already dynamic and approximate the opportunity cost better. For any aggregation of inventory levels, the resulting bid prices can potentially also be improved by DP decomposition where we would need to solve a one-dimensional dynamic program for every resource. The corresponding Bellman equations share the same structure with the column generation sub-problem (3) so that we can solve them via mixed integer linear programs. Intuitively, the decomposition approach should exhibit better policy performance than DP decomposition based on the CDLP since the input of the decomposition procedure is more accurate. However, the finer the approximation the less improvement the decomposition seems to be able to achieve while even for the fully aggregated affine approach the results of Zhang and Adelman (2007) show little improvement of the decomposition approach based on (**AFF**) over D-AFF. For the concave approximation, we firstly would need to solve (**P**) quite accurately (which is expensive due to tailing off behavior of column generation), and secondly solve the m dynamic programs. Our experiments indicated negligible improvements over D-CONC-c at high expense. Hence we confine ourselves to directly using the bid prices from (**CONC**) to exhibit their improved quality, and show in the next section that even so this approach can outperform the CDLP with DP decomposition. Note that DP decomposition would be useful if we choose to aggregate over both time and inventory (resulting in a quite small number of constraints in (**P**))since in this case the linear program can be solved to optimality.

8. Numerical Results

In this section, we present the results of numerical experiments that shed light on the quality of the upper bounds and performance of policies obtained for our approach, compared with the above mentioned alternative approaches. We consider two versions of our model: First, splitting the inventory of each resource in two equally sized inventory level ranges, and second, considering all possible c_i inventory levels separately. The rationale is that we intend to demonstrate the obtainable gains by splitting up the inventory while balancing the computational effort required to solve (**P**). Our numerical examples provide a framework of what improvements can be expected for approximations in between the demonstrated ones. All computations were carried out with MATLAB using CPLEX on a 3 GHz PC.

Problem Instances

We test our approach on two small hub & spoke networks, called HS-a and HS-b. These examples correspond to test cases that, among others, were used in Zhang and Adelman (2007).

The first network HS-a is depicted in Figure 2 and represents a network with one hub and two non-hub nodes. There are four flight legs, and we assume that the segments are such that they

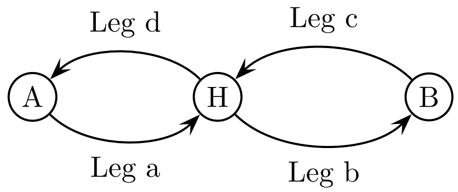

Figure 2 **Hub & Spoke network example HS-a.**

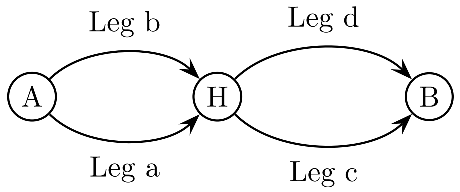

Figure 3 **Hub & Spoke network example HS-b.**

consider all products with the same origin-destination (O-D) combination. In this case, we have six segments which correspond to the six possible O-D combinations. For each itinerary there are two products, a high fare class and a low fare class. Preference values were generated from the Poisson distribution with mean 80 for high fares, with mean 200 for low fares, and mean 10 for the no-purchase preference and are given in Table 2, which also provides an overview of products, divided into the disjoint consideration sets C_l for each segment l. Fares are likewise indicated; they were drawn from the Poisson distribution with mean for high and low fares on local flight 30 and 10, respectively, and for high and low fares on multi-leg itineraries 300 and 100, respectively. We use three arrival rates $\lambda = \sum_l \lambda_l$ to vary the load factor of the considered network instances to having a low, medium and high load factor, given in Table 4. The empirical load factor is of course given by summing consumed capacity over the booking horizon for each sample path, averaging these numbers over all samples and dividing it by the total network capacity. Clearly, the load factor depends on the simulations, in particular with respect to the policies that were used. Since we intend to compare different policies under the same circumstances, we characterize the latter with the so-called *capacity tightness* instead of the empirical load factor. Capacity tightness is defined here as the total expected resource consumption of offering a specific set S^*, divided by the network capacity. In formulae,

$$\text{Capacity Tightness } \rho = \frac{\lambda \sum_{t=1}^{\tau} \sum_{j \in S^*} \sum_{i=1}^{m} a_{ij} P_j(S^*)}{\sum_{i=1}^{m} c_i},$$

where S^* is the revenue maximizing set given no capacity constraints,

$$S^* \in \arg\max_{S \subseteq N} \sum_{j \in S} P_j(S) f_j.$$

The second hub and spoke network example HS-b is depicted in Figure 3, and segments are described in Table 3 and Table 5. It consists of two parallel flights from location A to H, and further two parallel flights from H to B. On each itinerary we again have two fare classes high and low, and we assumed that segments correspond to O-D combinations. The fares and preference values are again taken from the Poisson distribution with mean as in the network HS-a. For both networks

Table 2 Products, Segments and Preference Values for HS-a

	C_1		C_2		C_3		C_4		C_5		C_6	
Prod. j	1	2	3	4	5	6	7	8	9	10	11	12
Segment	A → H		A → H → B		H → B		B → H		B → H → A		H → A	
Fare	30	12	294	97	39	10	26	10	289	121	25	10
Legs	a	a	a,b	a,b	b	b	c	c	c,d	c,d	d	d
Pref. v_l	72	198	76	203	89	200	74	228	87	209	87	214

Product definitions for network HS-a. "Legs" indicates the resources which the respective product utilizes. No-purchase preference $v_{l0} = [6, 14, 7, 6, 9, 7]$.

Table 3 Products, Segments and Preference Values for HS-b

	C_1				C_2								C_3			
Prod. j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Segment	A → H				A → H → B								H → B			
Fare	28	16	31	11	325	91	279	117	308	91	316	118	27	10	26	5
Legs	a	a	b	b	a,c	a,c	a,d	a,d	b,c	b,c	b,d	b,d	c	c	d	d
Pref. v_l	70	205	99	216	81	214	80	218	94	213	74	197	84	217	85	200

"Legs" indicates the resources which the respective product utilizes. No-purchase preference $v_{l0} = [3, 6, 14]$.

Table 4 HS-a: Arrival rates.

Seg. l	Low λ	Med. λ	High λ
1	0.0997	0.1189	0.1534
2	0.0605	0.0722	0.0932
3	0.0962	0.1147	0.1479
4	0.1033	0.1232	0.1589
5	0.0890	0.1062	0.1370
6	0.0712	0.0849	0.1096
Σ	0.52	0.62	0.8

HS-a: Arrival rates λ_l for each segment l, for the three considered cases of $\lambda \in \{0.52, 0.62, 0.8\}$.

Table 5 HS-b: Arrival rates.

Seg. l	Low λ	Med. λ	High λ
1	0.1327	0.1598	0.2051
2	0.1886	0.2271	0.2914
3	0.1187	0.1430	0.1835
Σ	0.44	0.53	0.68

HS-b: Arrival rates λ_l for each segment l, for the three considered cases of $\lambda \in \{0.44, 0.53, 0.68\}$.

capacities are scaled up starting from $\hat{c} := [2, 4, 4, 2]$. We confine ourselves to very small network capacities since solution of the full-blown approach D-CONC-c becomes quickly computationally expensive, yet it is of interest because it provides an excellent benchmark that can be used for testing other policies such as D-CONC-2 and indicates the range of possible improvement due to inventory dependence. For practical implementations however, the aggregated approach should be used.

Upper Bound Quality

Upper bounds are useful as benchmarks in simulation studies and also potentially in designing new policies: For example, the approach of Siddappa et al. (2007) uses upper and lower bounds on the value function to construct policies. As stated in Proposition 1, the optimal objective value to (**P**) constitutes an upper bound on the optimal expected revenue and, in particular, the bound is at least as good as the one provided by the affine approximation approach. The natural question arises whether the new bound might turn out to be identical to the latter, or, if there is improvement, how much more accuracy was gained. We address this issue by comparing the upper bounds of the different solution approaches, all of which being applied to the problem instances as described above. All approaches were implemented using column generation. As stopping criterion for the column generation procedure we used the following "x% tolerance criterion": Stop generating columns if the sum over all time periods of the maximum reduced cost of each time period, say we denote it by $\mathcal{S}$, is within x% of objective value of the restricted master problem plus $\mathcal{S}$. The CDLP is solved to optimality, for CONC-c we used the 1% and for all other approaches the 0.5% stopping criterion.

We solve the problem HS-a and HS-b with CDLP, ADLP and AFF, and compare their corresponding upper bounds with our two-ranges approach CONC-2 and the individual seat-level approach CONC-c. Figures 4 and 6 highlight the percentage improvement of CONC-c relative to the other approaches over several problem instances. The underlying data is reported in Tables 9 and 10. The highest gains in accuracy are observed for medium load factors, which is intuitive since very low load factors imply simply offering the unconstrained revenue maximizing set, and for very high load factors one would simply offer the highest fares. For network HS-b, however, the improvement converges quickly to only 1% over any of the other approaches. In all cases we can observe the following ordering of the arising bounds:

$$z_{\text{CDLP}} \geq z_{\text{ADLP}} \geq z_{\text{AFF}} \geq z_{\text{CONC-2}} \geq z_{\text{CONC-c}}.$$

Small deviations from this ordering can occur due to stopping the column generation procedure according to the above mentioned tolerance criterion. This demonstrates that the bounds obtainable from the concave approach are indeed improvements. As we increase the time horizon and the leg capacity, the improvements are reduced but still are at least 5% compared to AFF in network HS-a. This decreasing difference can be explained with the asymptotic behavior of all approaches, that means, they all approach the optimal expected revenue as time and capacity are scaled up.

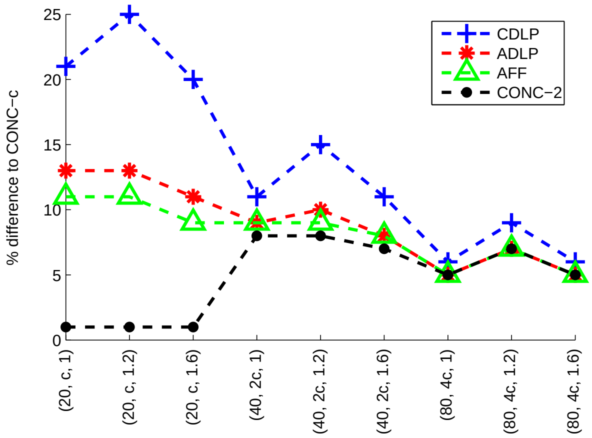

Figure 4 HS-a: Bound improvement.

Note. Problem instances are described by (time horizon, capacity vector, load factor).

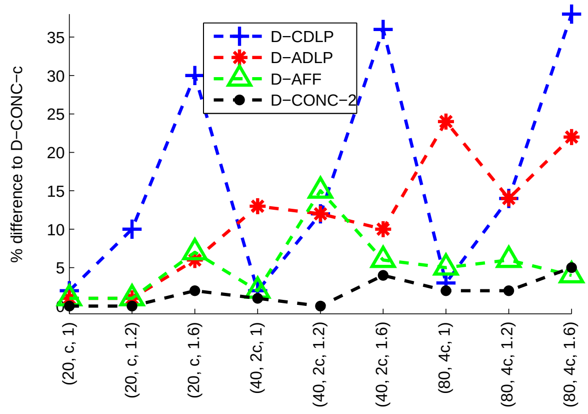

Figure 5 HS-a: Policy improvement.

Note. Problem instances are described by (time horizon, capacity vector, load factor).

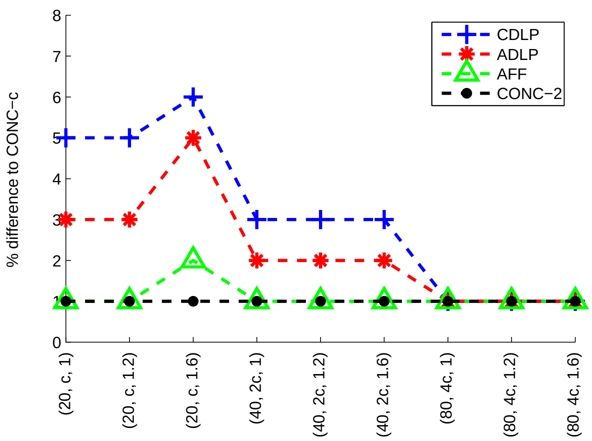

Figure 6 HS-b: Bound improvement .

Note. Problem instances are described by (time horizon, capacity vector, load factor).

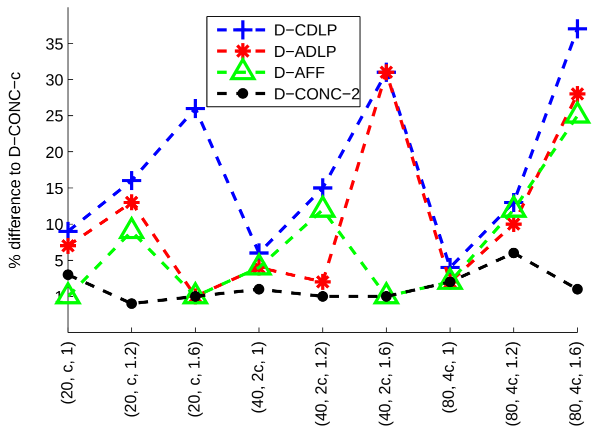

Figure 7 HS-b: Policy improvement.

Note. Problem instances are described by (time horizon, capacity vector, load factor).

Policy Performance

We claim that our proposed approach yields better bid prices through a better approximation of the value function, and support this by numerical results that were the outcome of using the opportunity cost information obtained from the various LP approaches directly to construct bid prices as described in Section 7. This provides insight into the improvement of the bid prices and the resulting potential in improving policies; in particular, dynamic programming policies are based on the quality of this opportunity cost information. Of course, in particular the solution of the CDLP would usually be used with DP decomposition as proposed by van Ryzin and Liu (2008) so as to obtain dynamic bid prices. We also compare this policy –we call it here DP-CDLP– with our direct bid price policy based on CONC-c in Table 6 and Table 7 for networks HS-a and HS-b, respectively, and emphasize that our approach can also be used with DP decomposition for any aggregation scheme, and the results can be expected to be better because the scheme can rely on improved opportunity cost information contained in the dual variables V. We tested the policies on the network examples HS-a and HS-b. Depending on the LP that was used to obtain the opportunity cost estimates, we denote the resulting policies by D-CDLP, D-ADLP, D-AFF, D-CONC-2 and D-CONC-c, where the latter two represent the policies based on an inventory split in two and c_i parts, respectively. In Figures 5 and 7 we summarized the outcome of our simulation study comparing D-CONC-c with the alternative approaches, the underlying data can be found in Tables 11 and 12. The relative errors of the simulations are at most 0.8% with 99% confidence. We find that the static bid prices do perform quite poorly as expected, and that ADLP and AFF show improved results because they incorporate time dependent bid prices. All approaches are outperformed by D-CONC-c and even D-CONC-2, the latter with the exception of one problem instance in HS-b. In particular, D-CONC-2 works already very well despite only using two marginal values per resource per time step, which indicates that some aggregation of inventory levels to enhance the computational performance will not necessarily severely deteriorate policy performance relative to D-CONC-c. For large networks, some aggregation will be needed depending on available computing power, as we elaborate in the following.

Computational Performance

The linear program (**P**) has $(\tau + \tau \sum_i K_i)$ constraints. Computational workload for solving a linear program grows proportionally to the number of constraints to the power of three (Bradley et al. (1977), p. 364), thus considering every inventory level separately on all resources –which corresponds to $K_i = c_i$ for all i– will be expensive. As an example for the grow of computational workload, we observed that solving CONC-2 takes about 5 times as long as solving AFF, see Table 8. Let us investigate how the marginal value of capacity actually varies across the inventory: In our numerical experiments, the difference between marginal values of capacity is small if the remaining time to departure is large relative to the capacity. To exemplify this observation, consider the contour plot in Figure 8. For the first 50 time periods, the marginal capacity values are almost constant over large inventory level ranges. Only in the last 30 time periods, the decline becomes more pronounced as it can be seen from the contour lines moving together. This can be intuitively explained by noting that these marginal values depend on the probability that we can sell all the seats up to the corresponding inventory level, and if the number of remaining time periods is large relative to the capacity, the probabilities should not differ very much. Therefore, first solving a large problem with a high level of aggregation and later resolving with refined approximation should be advantageous. Initially, an aggregation might be chosen with inventory ranges of equal length and large enough such that the model is still tractable. Having obtained a solution, we can guide the aggregation in the resolving process by examining the relative differences between resulting marginal values V_{t,i,k_l} and V_{t,i,k_r} of adjacent pairs of inventory ranges (k_l, k_r). If the difference between these values is greater than some specified threshold ϵ, then we could halve both ranges k_l and k_r so that in the next resolving process the change in the slope of the value function can be better represented in the approximation. On the other hand, if $|V_{t,i,k_l} - V_{t,i,k_r}| < \epsilon$, we would conclude that the value function is close to being linear in this area and we do not refine the approximation. In fact, we might even want to merge such two ranges into one to save computational effort. Also, we can exploit the flexibility of our model to approximate different flights with different levels of aggregation; those with low load factor will not need a fine approximation, and can thus be aggregated to enhance computational performance. Such legs could be identified by finding flights i that have $\pi_i = 0$ in the optimal CDLP dual solution.

9. Conclusion and Future Research

In the context of quantity-based network revenue management, we presented a linear programming approach to approximate dynamic programming with concave approximation of the value function with the specific feature that it incorporates both customer choice behavior as well as bid prices that depend on time and resource inventory level. As a result of the improved approximation, we obtain a better estimate of the opportunity cost, which is reflected in provably tighter upper bounds for any inventory aggregation and improved policy performance as observed in simulation studies. A policy based on the bid prices obtained directly from an approximate solution of our linear program using column generation outperforms alternative approaches including the choice-based linear program with dynamic programming decomposition as proposed by van Ryzin and Liu (2008). The solution of the linear program can be expensive, hence we propose to trade off accuracy with computational workload by aggregating inventory levels at the beginning of the booking horizon and later re-solving with refined inventory level resolution.

More research is needed on the question of how to aggregate inventory without losing too much accuracy. A promising way might be to resolve the linear program several times over the booking horizon with a process that guides the structure of inventory aggregations towards refinement where the value function exhibits non-linearity and aggregation where it is close to being linear. Such a process could be implemented by examining the difference in marginal values between every pair of adjacent inventory ranges and refining the ranges if this difference is large, or merging them into a single range if not. In addition, aggregation of time steps is also possible and can be used to reduce computational effort by exploiting that typically the value function at the beginning of the booking horizon is close to being linear.

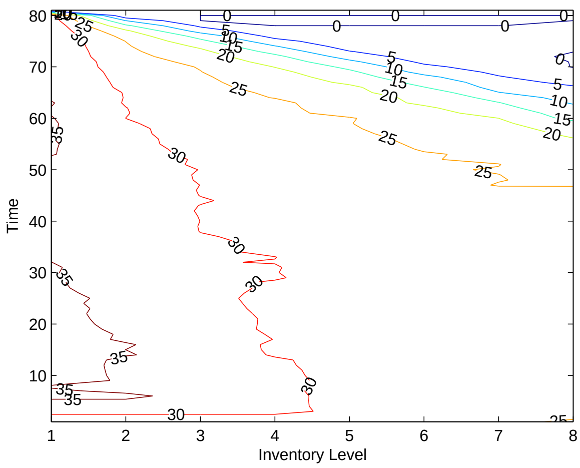

Figure 8 Contour plot of marginal value of capacity for leg d in network HS-b.

Note. Departures at time period $\tau = 81$, capacity of this resource is 8, capacity tightness $\rho = 1.2$.

Table 6 Simulation results for CDLP with dynamic programming decomposition on HS-a.

	τ	c	DP-CDLP	RE	D-CONC-c	RE	$\frac{\text{CONC-c}}{\text{DP-CDLP}}$
Low LF ($\rho = 1$)	20	$\hat{c}$	674	0.7	738	0.6	1.09
	40	$2\hat{c}$	1577	0.7	1627	0.8	1.03
	80	$4\hat{c}$	3422	0.6	3450	0.6	1.01
Med LF ($\rho = 1.2$)	20	$\hat{c}$	750	0.7	820	0.6	1.09
	40	$2\hat{c}$	1763	0.6	1816	0.7	1.03
	80	$4\hat{c}$	3859	0.5	3907	0.5	1.01
High LF ($\rho = 1.6$)	20	$\hat{c}$	895	0.7	940	0.7	1.05
	40	$2\hat{c}$	2004	0.5	2090	0.5	1.04
	80	$4\hat{c}$	4285	0.4	4389	0.4	1.02

CONC-c was implemented as direct bid price policy. RE is the percentage relative error of the sample mean with 99% confidence. The constant vector $\hat{c}$ is defined as $\hat{c} := [2, 4, 4, 2]$..

Table 7 Simulation results for CDLP with dynamic programming decomposition on HS-b.

	τ	c	DP-CDLP	RE	D-CONC-c	RE	$\frac{\text{CONC-c}}{\text{DP-CDLP}}$
Low LF ($\rho = 1$)	20	$\hat{c}$	990	0.6	1144	0.7	1.16
	40	$2\hat{c}$	2240	0.6	2378	0.7	1.06
	80	$4\hat{c}$	4771	0.4	4866	0.5	1.02
Med LF ($\rho = 1.2$)	20	$\hat{c}$	1212	0.5	1327	0.6	1.09
	40	$2\hat{c}$	2677	0.6	2783	0.7	1.04
	80	$4\hat{c}$	5639	0.5	5648	0.5	1.00
High LF ($\rho = 1.6$)	20	$\hat{c}$	1399	0.4	1588	0.5	1.14
	40	$2\hat{c}$	3131	0.4	3321	0.5	1.06
	80	$4\hat{c}$	6666	0.3	6878	0.4	1.03

CONC-c was implemented as direct bid price policy. RE is the percentage relative error of the sample mean with 99% confidence. The constant vector $\hat{c}$ is defined as $\hat{c} := [2, 4, 4, 2]$..

Table 8 CPU run time in seconds

	τ	c	AFF (s)	CONC-2 (s)	$\frac{\text{CONC-2}}{\text{AFF}}$
Low LF ($\rho = 1$)	20	$\hat{c}$	4.3	27	6.3
	40	$2\hat{c}$	8.7	41	4.7
	80	$4\hat{c}$	15.6	73	4.7
Med LF ($\rho = 1.2$)	20	$\hat{c}$	4.5	24	5.3
	40	$2\hat{c}$	9	49	5.4
	80	$4\hat{c}$	18.8	84	4.5
High LF ($\rho = 1.6$)	20	$\hat{c}$	5.1	18	3.5
	40	$2\hat{c}$	9.7	41	4.2
	80	$4\hat{c}$	22.8	115	5.0

CPU run times to solve problem HS-a with **AFF** and **CONC** for the concave approach with two marginal values per resource per time step. The constant vector $\hat{c}$ is defined as $\hat{c} := [2, 4, 4, 2]$.

Table 9 Upper Bounds for Network HS-a

	τ	c	z_{CDLP}	z_{ADLP}	z_{AFF}	$z_{\text{CONC-2}}$	$z_{\text{CONC-c}}$
Low LF ($\rho = 1$)	20	$\hat{c}$	925	866	851	775	766
	40	$2\hat{c}$	1850	1808	1803	1788	1661
	80	$4\hat{c}$	3701	3658	3655	3653	3488
Med LF ($\rho = 1.2$)	20	$\hat{c}$	1077	978	962	877	864
	40	$2\hat{c}$	2154	2065	2050	2026	1878
	80	$4\hat{c}$	4307	4234	4219	4214	3953
High LF ($\rho = 1.6$)	20	$\hat{c}$	1200	1102	1086	1008	997
	40	$2\hat{c}$	2400	2333	2321	2299	2153
	80	$4\hat{c}$	4800	4743	4742	4738	4529

Upper bounds on optimal expected revenue from HS-a. The constant vector $\hat{c}$ is defined as $\hat{c} := [2, 4, 4, 2]$.

Table 10 Upper Bounds for Network HS-b

	τ	c	z_{CDLP}	z_{ADLP}	z_{AFF}	$z_{\text{CONC-2}}$	$z_{\text{CONC-c}}$
Low LF ($\rho = 1$)	20	$\hat{c}$	1293	1273	1250	1243	1235
	40	$2\hat{c}$	2587	2565	2548	2548	2523
	80	$4\hat{c}$	5173	5147	5137	5140	5109
Med LF ($\rho = 1.2$)	20	$\hat{c}$	1495	1472	1447	1440	1430
	40	$2\hat{c}$	2990	2964	2943	2939	2917
	80	$4\hat{c}$	5980	5950	5930	5930	5897
High LF ($\rho = 1.6$)	20	$\hat{c}$	1817	1795	1746	1736	1715
	40	$2\hat{c}$	3633	3609	3580	3573	3537
	80	$4\hat{c}$	7266	7240	7220	7220	7170

Upper bounds on optimal expected revenue from HS-b. The constant vector $\hat{c}$ is defined as $\hat{c} := [2, 4, 4, 2]$.

Table 11 Simulation results for direct bid price policies on network instances HS-a.

	τ	c	D-CDLP	RE	D-ADLP	RE	D-AFF	RE	D-CONC-2	RE	D-CONC-c	RE
Low LF ($\rho=1$)	20	$\hat{c}$	727	0.7	732	0.7	732	0.7	738	0.6	738	0.6
	40	$2\hat{c}$	1594	0.8	1445	0.8	1591	0.8	1607	0.8	1627	0.8
	80	$4\hat{c}$	3364	0.7	2772	0.6	3297	0.6	3398	0.6	3450	0.6
Med LF ($\rho=1.2$)	20	$\hat{c}$	743	0.6	816	0.6	815	0.6	820	0.6	820	0.6
	40	$2\hat{c}$	1621	0.6	1622	0.7	1575	0.8	1814	0.7	1816	0.7
	80	$4\hat{c}$	3414	0.5	3421	0.5	3693	0.6	3830	0.6	3907	0.5
High LF ($\rho=1.6$)	20	$\hat{c}$	721	0.6	886	0.8	882	0.8	917	0.8	940	0.7
	40	$2\hat{c}$	1534	0.6	1893	0.6	1981	0.6	2001	0.6	2090	0.5
	80	$4\hat{c}$	3185	0.4	3598	0.4	4213	0.5	4186	0.5	4389	0.4

RE is the percentage relative error of the sample mean with 99% confidence. The constant vector $\hat{c}$ is defined as $\hat{c} := [2,4,4,2]$.

Table 12 Simulation results for direct bid price policies on network instances HS-b.

	τ	c	D-CDLP	RE	D-ADLP	RE	D-AFF	RE	D-CONC-2	RE	D-CONC-c	RE
Low LF ($\rho=1$)	20	$\hat{c}$	1053	0.7	1072	0.7	1146	0.8	1112	0.7	1144	0.7
	40	$2\hat{c}$	2242	0.7	2280	0.7	2294	0.7	2346	0.7	2378	0.7
	80	$4\hat{c}$	4693	0.5	4753	0.5	4760	0.5	4764	0.5	4866	0.5
Med LF ($\rho=1.2$)	20	$\hat{c}$	1148	0.6	1175	0.6	1222	0.6	1338	0.7	1327	0.6
	40	$2\hat{c}$	2419	0.5	2742	0.8	2495	0.5	2783	0.8	2783	0.7
	80	$4\hat{c}$	5003	0.4	5149	0.5	5033	0.4	5316	0.5	5648	0.5
High LF ($\rho=1.6$)	20	$\hat{c}$	1260	0.5	1585	0.5	1585	0.5	1586	0.5	1588	0.5
	40	$2\hat{c}$	2531	0.5	2542	0.5	3313	0.5	3315	0.5	3321	0.5
	80	$4\hat{c}$	5021	0.4	5361	0.3	5491	0.3	6832	0.4	6878	0.4

RE is the percentage relative error of the sample mean with 99% confidence. The constant vector $\hat{c}$ is defined as $\hat{c} := [2,4,4,2]$.

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.

Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA.

Boyd, E. A., R. Kallesen. 2004. The science of revenue management when passengers purchase the lowest available fare. *Journal of Revenue and Pricing Management* **3** 2.

Bradley, S.P., A.C. Hax, T.L. Magnanti. 1977. *Applied mathematical programming*. Addison-Wesley.

Bront, J. J. M., I. Méndez-Díaz, G. Vulcano. 2007. A column generation algorithm for choice-based network revenue management. URL `http://pages.stern.nyu.edu/\~{}gvulcano/ColGenChoiceRM-Rev2.pdf`. Forthcoming in *Operations Research*.

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.

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.

de Farias, D. P., B. Van Roy. 2004. On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research* **29** 462–478.

de Farias, D.P., B. Van Roy. 2003. The linear programming approach to approximate dynamic programming. *Operations Research* **51** 850–865.

Farias, V. F., B. Van Roy. 2007. An approximate dynamic programming approach to network revenue management. URL `http://web.mit.edu/\~{}vivekf/www/papers/ADP-rm-07-03.pdf`. Working paper.

Gallego, G., G. Iyengar, R. Phillips, A. Dubey. 2004. Managing flexible products on a network. Tech. Rep. TR-2004-01, Department of Industrial Engineering and Operations Research, Columbia University.

Kunnumkal, S., H. Topaloglu. 2008. A refined deterministic linear program for the network revenue mangement problem with customer choice behavior. *Naval Research Logistics* **55** 563–580.

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

Powell, W. 2007. *Approximate Dynamic Programming*. John Wiley & Sons, Hoboken, NJ.

Siddappa, S., D. Günther, J. Rosenberger, V. Chen. 2007. Refined experimental design and regression splines method for network revenue management. *Journal of Revenue and Pricing Management* **6** 188–199.

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

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. 2007. Using Lagrangian relaxation to compute capacity-dependent bid prices in network revenue management. URL http://people.orie.cornell.edu/~huseyin/publications/revenue_man.pdf. Forthcoming in *Operations Research*.

van Ryzin, G. J. 2005. Models of demand. *Journal of Revenue and Pricing Management* **4** 204–210.

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

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. 2007. An approximate dynamic programming approach to network revenue management with customer choice. URL http://faculty.chicagogsb.edu/daniel.adelman/research/network.pdf. Working paper, University of Chicago Graduate School of Business.

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