Research Article

On the Origin-Destination Demands Linear Programming Model for Network Revenue Management with Customer Choice

Feng Liu, Qizong Wu and Ying Qu

1Department of Information Management, the Central Institute for Correctional Police, Baoding 071000, China
2Department of Management and Economics, Beijing Institute of Technology, Beijing, 100081, China
3College of Economic and Management, Hebei University of Science and Technology, Shijiazhuang, 050018, China

Abstract: In this study, we research the problem of network revenue management with customer choice based on the Origin-Destination (O-D) demands. By dividing customers into different segments according to O-D pairs, we consider a network capacity control problem where each customer chooses the open product within the segment he belongs to. Starting with a Markov Decision Process (MDP) formulation, we approximate the value function with an affine function of the state vector. The affine function approximation results in a new Linear Program (LP) which yields tighter bounds than the Choice-based Deterministic Linear Program (CDLP). We give a column generation procedure for solving the LP within a desired optimality tolerance and present numerical results which show the policy perform from our solution approach can outperform that from the CDLP.

Keywords: Choice behavior, dynamic programming, linear programming, network revenue management

INTRODUCTION

Traditionally, revenue management systems have been built upon the independent demand model assumption. This assumption views demands for products as completely independent of the capacity controls being applied by the seller. But among both practitioners and researchers, there is growing interest in modeling customer choice behavior in revenue management problems, which stems partly from the dissatisfaction with the limitations of the independent demand model.

Under the independent demand model assumption, Adelman (2007) studied an Approximate Dynamic Programming (ADP) approach for computing dynamic bid-prices. The idea is to formulate the underlying dynamic program as a LP by making an affine functional approximation to the value function.

In most of capacity control models of network revenue management, uncertain demands are considered for each product (each product for a specific fare class). However, the exploration of models based on stochastic demands between O-D pairs will probably become increasingly important as opportunities for code-sharing within strategic partnerships increases the breadth of choice in customers itinerary selections. Motivated by this consideration and the work of Adelman (2007) and Liu et al. (2011) provided an independent demand model which is developed with O-D demands. The model can be used to compute dynamic bid-prices and provides stronger bounds and better policy performance than the Deterministic Linear Program (DLP) approximation based on the O-D demands. This study will focus on researching the network capacity control problem with customer choice based on the O-D demands and developing a column generation algorithm to solve the problem within a desired optimality tolerance. Our numerical study shows the policy perform from our solution approach can outperform that from the CDLP.

LITERATURE REVIEW

There have been a lot of independent demand models for solving the network revenue management problem. For a detailed discussion of these models, (Talluri and Van Ryzin, 2004b). Due to the deficiency of independent demand models, many researchers have studied the problems with rich customer choice behavior.

Several researches have been done on choice behavior for single-leg revenue management problems. Belobaba (1987a, b) propose the buy-up heuristics to modify the expected marginal seat revenue (EMSR) heuristics. Belobaba and Weatherford (1996), Brumelle et al. (1990) and Zhao and Zheng (2001) also consider...
...which serve various O-D pairs in the network. Typically, there are multiple routes that can serve a given O-D pair. The firm sells k products (Each product is defined by a route and fare class combination). The set of products is denoted by set \( j \in J = \{1, \ldots, k\} \). Let \( J_n \subseteq J \) be the set of products which belong to O-D pair \( n \), then \( J = \cup_{n \in N} J_n \). Furthermore, we have \( J_n \cap J_m = \emptyset \) for \( n \neq m \). The fare for product \( j \) is \( f_j \).

Define the incidence matrix \( A = [a_{ij}] \), where \( a_{ij} = 1 \) if product \( j \) uses leg \( i \) and \( a_{ij} = 0 \) otherwise; The \( j \)th column of \( A \), denoted \( A^j \), is the incidence vector for product \( j \). We let \( A^n \) denote the set of legs used by product \( j \).

Time is discrete, there are \( T \) periods and the index \( t \) represents an arbitrary time (with the time indices running forward, so \( t = T \) is the time of service).

Within each time period \( t \), at most one customer arrives. The probability of having an arrival in each time period is denoted by \( \lambda \) and no customer arrives with probability \( 1 - \lambda \). Assuming that an arriving customer first chooses which O-D pair he belongs to and then chooses the product within the given segment. From the firm’s perspective, each arriving customer belongs to segment \( n \) with probability \( p_n \), with probability \( \sum_{n \in N} p_n = 1 \). Hence, the arriving stream of segment-\( n \) customers is a Poisson process with rate \( \lambda n = \lambda p_n \) and the total arrival rate \( \lambda \) verifies \( \lambda = \sum_{n \in N} \lambda_n \).

When a customer arrives, the firm must decide what products to offer. Let \( S \subseteq J \) he the set of the total available products which are offered by the firm. Given the set \( S \), let \( P_{nj}(S) \) denote the probability that a segment-\( n \)-customer chooses the product \( j \in J_n \cap S \). To determine the purchase probability \( P_{nj}(S) \), define a preference vector \( v_j \geq 0 \), which indicates the customer “preference weight” for each product contained in \( J_n \) and the no-purchase preference value \( v_{nj} \). Then:

\[
P_{nj}(S) = \frac{v_j}{\sum_{j, j \in S} v_j + v_{nj}}
\]

If \( j \notin J_n \cap S \) or \( j \notin J_n \), then \( v_{nj} = 0 \) (and hence \( P_{nj}(S) = 0 \)). Let \( P_j(S) \) be the probability that the product \( j \in S \) is chosen by an arriving customer. Noting that the seller ex ante cannot distinguish which segment each arriving customer belongs to, then:

\[
P_j(S) = \sum_{n \in S} p_n P_{nj}(S).
\]

Let \( P_n(S) \) denote the no-purchase probability and by total probability \( \sum_{j \in J} P_{nj}(S) + P_n(S) = 1 \), i.e., \( P_j(S) = 1 - \sum_{n \in S} P_n(S) \sum_{j \in J} P_{nj}(S) \).
The state of the network is described by a vector \( x = (x_1, \ldots, x_m) \) of remaining leg capacities, the initial state is denoted by vector vector \( c = (c_1, \ldots, c_m) \). Vector \( x \) satisfies:

\[
x \in X_t = \{ \{ c \} \text{ if } t = 1,
\{ \{ x \in \mathbb{Z}^n_t \mid x_t \in \{ 0, 1, \ldots, c_i \} \forall i \text{ if } t = 2, \ldots, T \}
\]

If a single unit of product \( j \in S \) is sold, the state of the network changes to \( x-A_j \), ignoring cancellations and no-shows.

Let \( v_t(x) \) be the maximum total expected revenue over periods \( t, \ldots, T \) starting at state \( x \) at the beginning of period \( t \). Then \( v_t(x) \) must satisfy the Bellman equations:

\[
v_t(x) = \max_{j \in J} \left\{ \sum_{n \in N} \lambda_n \sum_{p \in P_n} p_n(s) \left[ f_j + v_{i+1}(x-A_j) \right] + \lambda x - v_{i+1}(x) \right\}
\]

with the boundary condition \( v_{T+1}(x) = 0 \ \forall x \). In the above, the second equation follows from the fact that \( \lambda_n = \lambda p_n \) and the set:

\[ J(x) = \{ j \in J : x \geq \lambda A \} \]

is the set of products that can be offered when the state is \( x \).

The value function at initial state \( C \) can be computed by the linear program:

\[
(P0) \min_c \ v_t(c)
\]

s.t. \( v_t(x) \geq \sum_{n \in N} \lambda_n \sum_{p \in P_n} p_n(s) \left[ f_j - v_{i+1}(x) - v_{i+1}(x-A_j) \right] \]

subject to: \( v_{i+1}(x), \ \forall t, x \in X_t, S \subseteq \{ J(x) \} \)

CDLP Formulation: In general, (1) and (P0) are intractable because of the high-dimensional state space. To circumvent this complexity, the standard approach to revenue management is to approximate the dynamic programming with a LP.

Let \( R_j(S) \) denote the revenue from one arriving customer who belongs to segment \( n \) when the set \( S \) is offered. Then:

\[
R_j(S) = \sum_{p \in P_n} f_j p_n(S), \ \forall n \in N, S \subseteq J.
\]

Given offer set \( S \), let \( Q_n(S) \) denote the resource consumption rate on leg \( i \in \{ 1, \ldots, m \} \) which can be used by products which belong to segment \( n \), when a customer arrives. Then the vector \( Q_n(S) = (Q_n1(S), \ldots, Q_nm(S)) \) is the vector of resource consumption rate of segment \( n \). Furthermore, if let \( P_n(S) = (P_n1(S), \ldots, P_nk(S)) \) be the vector of purchase probabilities of segment \( n \), then:

\[
Q_d(S) = AP_d(S)
\]

Since the demands are deterministic and the purchase probabilities are time homogeneous, only the total time each set \( S \) is offered matters. Let \( t(S) \) be the total time the set \( S \) is offered, then we have the following LP:

\[
(LP) Z_L = \max \sum_{j \in J} \left( \sum_{x \in \mathbb{N}} \lambda_x R_x(S) \right) \gamma(S)
\]

s.t. \( \sum_{j \in J} \left( \sum_{x \in \mathbb{N}} \lambda_x Q_x(S) \right) \gamma(S) \leq c \)

\[
\sum_{j \in J} t(S) = T
\]

\[
t(S) \geq 0, \ \forall S \subseteq J
\]

If \( S = \emptyset \), the decision variable \( t(\emptyset) \) means the total time that no products are offered. We allow the variables \( t(S) \) to be continuous. (LP) is similar to the CDLP model proposed in Gallego et al. (2004), but we consider demands at O-D pair level. The dual of (LP) is:

\[
\min_{\pi, \mu} \pi^T c + T \mu
\]

subject to:

\[
\pi^T \lambda \gamma(S) + \mu \geq \sum_{x \in \mathbb{N}} \lambda_x R_x(S), \ \forall S \subseteq J,
\]

\[
\pi \geq 0,
\]

where \( \pi \) is the vector of dual prices on (4) and \( \mu \) is the dual price on (5), respectively.

(LP) can be solved by column generation techniques efficiently (Liu and Van Ryzin, 2008).

**FUNCTIONAL APPROXIMATION**

As mentioned, (P0) is intractable because of the enormous size of the state space. The only practical approach is to try to approximate the decision problem. In this section, first, we use a set of affine functions to approximate \( v_t(.) \) and then give the resulting primal-
Formulation: Consider the affine functional approximation:

\[ v_i(x) = \theta_i + \sum_{j} \pi_{ij} x_j, \]

where, \( \theta_i \) is a constant offset and \( \pi_{ij} \) estimates the marginal value of a seat on leg \( i \) in period \( t \). We assume \( \theta_{T+1} = 0 \) and \( \pi_{T+1} = 0, \forall i \). Plugging (7) into (P0) yields that:

\[
(P1)\min_{x} \sum_{i=1}^{m} c_i \theta_i + \sum_{i} \pi_{ij} x_j
\]

s.t. \( \theta_i - \theta_i \sum_{j} \pi_{ij} x_j = \sum_{j} \lambda_j \sum_{j} \sum_{S} \pi_j(S) a_{ij} \)

\[ \geq \sum_{j} \lambda_j \sum_{j} \sum_{S} \pi_j(S) f_{ij}, \forall t, x \in X_i, \forall S \subseteq J(x). \]

The dual of (P1) is:

\[
(D1)\max_{\gamma} \sum_{S} \gamma_{t,S} = \gamma_0 \geq 0
\]

s.t.

- \( c_i \) if \( t = 1 \),
- \( \sum_{x \in X_i, S \subseteq J(x)} \left( x - \sum_{s \in S} \sum_{j} \sum_{j} \pi_j(S) a_{ij} \right) \forall t, \forall S \subseteq J(x), \gamma_{t+1} \geq 0 \),

\[
\sum_{S} \gamma_{t,s} = \left\{ \begin{array} {c} 1 \quad \text{if} \ t = 1, \\
0 \quad \text{otherwise} \end{array} \right\} \gamma_0 \geq 0
\]

The constraints (10) means:

\[
\sum_{x \in X_i, \forall S \subseteq J(x)} \gamma_{t,x,S} = 1, \forall t.
\]

Therefore the decision variables \( \gamma_{t,x,S} \) can be interpreted as approximated state-action probabilities; i.e., \( \gamma_{t,x,S} \) is the probability that the state is \( x \) and the sets \( S \) is offered at time \( t \). The constraints (9) is a flow balance constraint.

An optimal solution to (LP) specifies the total time each set should be offered, but the sequence in which the sets are offered is ambiguous. Let \( (x^*, \theta^*) \) be the optimal solution for (P1). Using the approximation:

\[ v_i(x) - v_i(x - A^*) = \sum_{i} a_{ij} \pi_{ij}^*, \]

then we can select an offer set dynamically in period \( t \) and state \( x \) by solving:

\[
\max_{S \subseteq J(x)} \sum_{j} \lambda_j \sum_{j} \sum_{j} \pi_j(S) a_{ij} \left( f_{ij} - \sum_{i} a_{ij} \pi_{ij}^* \right)
\]

Relationship to (LP): To derive (LP) from (D1), define:

\[ t(S) = \sum_{i,t \in x} \gamma_{i,t,S}, \forall S \subseteq J. \]

The objective function (3) follows immediately from (2) and (8). Now fix \( i \) and sum (9) over \( t \) to obtain:

\[
\sum_{t=2, \ldots, T, x \in X_i, S \subseteq J(x)} \left( x - \sum_{s \in S} \sum_{j} \sum_{j} \pi_j(S) a_{ij} \right) t(S) = \gamma_{t,x,S}.
\]

Canceling terms and rearranging yields:

\[ c_i = \sum_{t=2, \ldots, T, x \in X_i, S \subseteq J(x)} \lambda_j \sum_{j} \sum_{j} \pi_j(S) a_{ij} \gamma_{t,x,S} + \sum_{x \in X_i, S \subseteq J(x)} \gamma_{t,x,S}. \]

If \( \gamma_{t,x,S} > 0 \), we have \( x \geq a_{ij}, \forall i, j \in S \), so \( x \geq \sum_{x \in X_i, S \subseteq J(x)} \sum_{j} \sum_{j} \pi_j(S) a_{ij}, \forall i \). Therefore, (13) implies:

\[ c_i \geq \sum_{t=2, \ldots, T, x \in X_i, S \subseteq J(x)} \lambda_j \sum_{j} \sum_{j} \pi_j(S) a_{ij} \gamma_{t,x,S} = \sum_{S \subseteq J(x)} \sum_{j} \sum_{j} \pi_j(S) a_{ij} t(S), \forall i, \]

which yields (4). In addition, summing (10) over \( t \), we derive \( \sum_{t=2, \ldots, T} t(S) = T \).

The arguments above show that \( Z_{LP} \geq Z_{DP} \). As mentioned, any feasible solution to (P0) gives an upper bound to the optimal value from the Eq. (1). We summarize the results in the following proposition.

Proposition 1: Any feasible solution to (D1) yields a feasible solution to (LP) having the same objective value. Hence \( Z_{LP} \geq Z_{DP} \geq v_1(c) \).

Liu and Van Ryzin (2008) show that the bound \( Z_{LP} \) is asymptotically optimal, i.e., converges to \( v_1(c) \), as demands, capacity and time horizon scale linearly, that is, \( Z_{LP} \) is also asymptotically optimal.

COLUMN GENERATION ALGORITHM

The program (D1) has a large number of variables but relatively few constraints, so we can solve it via
column generation. Denote the reduced profit of $\gamma_{t,x,S}$ by:

$$
\omega_{t,x,S} = \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S) f_j - \sum_{n \in N} \pi_n x_n - \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S)a_{i,j} - \theta_i + \theta_{tl}.
$$

Proposition 2: A feasible solution to (D1) is:

$$
\hat{c}_{t,x,S} = \begin{cases} 
1 & \text{if } x = c, S = \emptyset, \forall t, x \in X, S \subseteq J(x) \\
0 & \text{otherwise,}
\end{cases}
$$

Proof: For all $t$ and $i$, the left-hand side of (9) is:

$$
\sum_{x \in J(x)} x \hat{c}_{t,x,S} = x S c c t \hat{c}_{t,x,\emptyset} = c_i.
$$

Likewise, for all $t > 1$, the right-hand side of (9) is:

$$
\sum_{x \in J(x)} \left( x - \lambda_n \sum_{j \in S} P_j(S)a_{i,j} \right) \hat{c}_{t-1,x,S} = c_i \hat{c}_{t-1,x,\emptyset} = c_i.
$$

Given an initial feasible solution to (D1) supplied by Proposition 2, denoting the resulting prices by $\theta, \pi$, now solve:

$$
\max_{t,n,x,S} \omega_{t,x,S} = \max_{t,n,x,S} \left( \lambda_n \sum_{j \in S} P_j(S) f_j - \sum_{n \in N} \pi_n x_n - \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S)a_{i,j} - \theta_i + \theta_{tl} \right)
$$

If the optimal function value is nonpositive, then we have attained optimality; otherwise, we add the column to the existing set of columns for (D1). For fixed $t > 1$, this is equivalent to solving the following optimization problem:

$$
\max_{t,n,x,S} \omega_{t,x,S} = \max_{t,n,x,S} \left( \lambda_n \sum_{j \in S} P_j(S) f_j - \sum_{n \in N} \pi_n x_n - \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S)a_{i,j} - \theta_i + \theta_{tl} \right)
$$

Under the multinomial logit (MNL) model, a choice set $S$ can be represented by an availability vector. We let the binary vector $u \in \{0,1\}^S$ be the characteristic vector of set $S$. It indicates which products are offered at any period, $u_j = 1$ if $j \in S$ and $u_j = 0$ otherwise. So we can then express (S0) in terms of the binary variables $u_j$:

$$
\max_{t,n} \sum_{j \in S} u_j v_j \left( f_j - \sum_{i \in J} a_{i,j} \pi_{i,j} \right) - \sum_{i \in J} \left( \pi_{i,j} - \pi_{i+1,j} \right) x_i - \theta_i + \theta_{tl}
$$

s.t. $a_{i,j} \leq x_j, \forall i, j \in S, x_j \in \{0,...,c_j\}, \forall j$.

In fact, (S1) is a Mixed Integer Non-Linear Programming (MINLP) problem. Solving such a problem is most challenging, since there is no a direct method capable of doing it efficiently. In the following, we transform (S1) into a Mixed Integer Linear Programming (MILP) problem. The advantage of this transformation is that any mixed integer programming (MIP) software package can be used to solve (S1):

Let:

$$
\alpha_n = \frac{1}{\sum_{j \in J} u_j v_j + v_{n0}}, \forall n \in N,
$$

$$
z_{n,j} = \alpha_n u_j, \forall n \in N, j \in J.*
$$

For all $n \in N, j \in J.*$, variable $z_{n,j}$ can be represented by the following linear inequalities: (1) $z_{n,j} \leq K - K u_j$; (2) $z_{n,j} \leq \alpha_n$; (3) $z_{n,j} \leq K u_j$; (4) $z_{n,j} \geq 0$, where $K$ is a large number (i.e., greater than $\alpha_n$). Furthermore, by the definition of $\alpha_n$ and $z_{n,j}$, we have:

$$
\sum_{j \in J} v_{n,j} z_{n,j} + v_{n0} \alpha_n = 1, \forall n \in N,
$$

$$
\alpha_n \geq 0, \forall n \in N.
$$

Then (S1) can be rewritten as:

$$
\max_{t,n,x,S} \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S) f_j - \sum_{n \in N} \pi_n x_n - \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S)a_{i,j} - \theta_i + \theta_{tl}
$$

s.t. $a_{i,j} \leq x_j, \forall i, j \in S, x_j \in \{0,...,c_j\}, \forall i.$

$$
\max_{t,n,x,S} \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S) f_j - \sum_{n \in N} \pi_n x_n - \sum_{n \in N} \lambda_n \sum_{j \in S} P_j(S)a_{i,j} - \theta_i + \theta_{tl}
$$

s.t. $a_{i,j} \leq x_j, \forall i, j \in S, x_j \in \{0,...,c_j\}, \forall i.$

664
Theorem 1: (S1) is equivalent to (S2); i.e., both optimization problems have the same optimal objective value and an optimal solution to one can be obtained from an optimal solution of the other. So we only need to solve (S2) to find the maximum reduced profit for each t > 1.

Proof: Since (S2) is obtained from (S1) through change of variables, it can be shown that an optimal solution to (S1) is a solution to (S2) and both optimization problems have the same optimal objective value at the solution.

Suppose \((\hat{x}, \hat{u}, \hat{z}, \hat{a})\) is an optimal solution to (S2). Then \((\hat{x}, \hat{u})\) clearly satisfies the constraints in (S1). Furthermore, \(\forall n \in N, j \in J_a:\)

\[
\sum_{j \in J_a} \frac{\hat{u}_j}{v_{nj} + \alpha_n} = \hat{u}, \quad \sum_{j \in J_a} \frac{\hat{z}_j}{v_{nj} + \alpha_n} = \hat{z},
\]

where the last equation follows from (14). It then follows that the two optimization problems have the same objective value at the given solution. Combining the results in all cases yields Theorem 1. Where the last equality follows from (11). This relation is true for all feasible solutions \(\gamma\). Particularly, for an optimal solution \(\gamma^*\) to (D1), there is objective value \(Z(\gamma^*) = Z_{D1}\). Furthermore, from strong duality applied to the restricted problem, we have:

\[
\sum_i \pi_{i,t} \xi_i - \hat{\theta}_t = Z(\hat{\gamma}) = Z_{\hat{\gamma}}.
\]

As a result, we obtain:

\[
Z_{D1} \leq Z_{\hat{\gamma}} + \sum_{t=1}^r \hat{\omega}_t^*.
\]

For (D1), Proposition 3 gives an upper bound on the optimality gap between an optimal solution and a given feasible solution. To ensure that the objective value of the current solution \(\hat{\gamma}\) based on columns \(\xi\) is within \(\Omega\) of an optimal solution, i.e., \(Z_{D1}/Z_{\hat{\gamma}} \leq 1 + \Omega\), it suffices to ensure that:

\[
\sum_{t=1}^r \hat{\omega}_t^* \leq \Omega.
\]
Algorithm Column generation

Set $\xi = \{ (i, c, \omega) \mid t \}$, solve the restricted
problem (D1(\xi)), and set $a_i^t = \infty$ for all $i$.
while $\sum_{i} a_i^t > 0$, do
  for all $t \in \{1, \ldots, T\}$
    compute $a_i^t = \max_{j \in \mathcal{J}} a_{i,j}^t$
    select an $(x, S) = \arg\max_{j \in \mathcal{J}} a_{i,j}^t$
    update $\xi \leftarrow \xi \cup \{ (i, x, S) \}$.
  solve (D1(\xi)).

Fig. 1: Column generation algorithm for solving (P1) to
within an optimality tolerance of $\Omega$

Fig. 2: Hypothetical airline network with five legs, six O-D
pairs and ten routes

We employ this as a stopping criterion for the
algorithm. The full algorithm is described in Fig. 1.
Let $(\pi^*, \theta^*)$ be the optimal solution for (P1). For the
MNL choice model, (12) reduces to:

$$\max_{u_i \in \{0, 1\} \mid x \in \mathcal{A}'} \sum_{i \in \mathcal{N}} \lambda_i \sum_{j \in \mathcal{J}} u_j \gamma_j \left( f_j - \sum_{i \in \mathcal{N}} a_{i,j}^t \gamma_i \right)$$

(15)

A control policy in period $t$ and state $x$ can
be computed by solving (15). The constraint $u_i \in \{0, 1\} \mid x \in \mathcal{A}'$ in (15) incorporates the constraint on capacity. The
maximization in (15) can be solved efficiently using
simple ranking procedure (Liu and Van Ryzin, 2008).

NUMERICAL RESULTS

Figure 2 illustrates a hypothetical airline network
which consists of five legs, six O-D pairs and ten
routes. Furthermore, two fare classes (Business and
Leisure) are offered for each route. Business fares are
drawn from the Poisson distribution with mean 200 and
Leisure fares are drawn from the Poisson distribution
with mean 100. For simplicity, we considered
stationary demands with the probability 0.2 for having
no customer arrival in a period. We generated problem
instances with $T \in \{20, 50, 100, 200, 500\}$. For each
instance, we set the initial capacity, $c$, to be the same
for each leg.

We tested the following methods:

- **ADP**: Solve (D1) – (P1) once. Given a set of
dynamic bid prices, use the policy given by (12).
- **LP**: This method implements the static (LP)
solution. As mentioned, the optimal solution to
(LP) gives the total time to offer each set, but the
sequence in which the sets are offered is ambiguous.
We assumed that sets were offered according to the order that the solutions to (6) were
generated.

The numerical experiments also studied the upper
bound given by (D1) as compared with (LP). We solved
(D1) with an optimality tolerance of $\Omega = 5\%$
and simulated each instance 100 times for each policy,
using the same sequence of customer demands across
different policies. The results are shown in Table 1.

<table>
<thead>
<tr>
<th>$T$</th>
<th>Capacity per leg</th>
<th>(D1) Bound</th>
<th>(LP) Bound</th>
<th>ADP Mean (S.E.)</th>
<th>LP Mean (S.E.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>3</td>
<td>1636.90</td>
<td>2199.20</td>
<td>1456.40 (135.21)</td>
<td>1300.90 (122.24)</td>
</tr>
<tr>
<td>50</td>
<td>7</td>
<td>3753.50</td>
<td>4542.40</td>
<td>3468.40 (302.75)</td>
<td>2554.80 (220.36)</td>
</tr>
<tr>
<td>100</td>
<td>15</td>
<td>9761.60</td>
<td>11961.00</td>
<td>9132.80 (608.42)</td>
<td>8624.30 (523.50)</td>
</tr>
<tr>
<td>200</td>
<td>29</td>
<td>16214.00</td>
<td>23906.00</td>
<td>13480.00 (985.34)</td>
<td>12004.00 (853.72)</td>
</tr>
<tr>
<td>500</td>
<td>76</td>
<td>41352.00</td>
<td>57634.00</td>
<td>33208.00 (1362.20)</td>
<td>31342.00 (1330.40)</td>
</tr>
</tbody>
</table>

CONCLUSION

Currently, data from past sales typically provide
the basis for forecasting future demands. This data is
itinerary based and customer choices regarding selected
itineraries can be difficult to discern. In our model, the
demands are for O-D pairs rather than specific
itineraries within the network. We also explicitly
recognize that a given O-D pairs can be served by
multiple itineraries. As a result, current itinerary-based
demand forecasting techniques can be used-with the
added step of aggregating demands over the various
itineraries that service an O-D pair.

In this study, we consider a network capacity
control problem where customers choose the open
product according to their O-D pair. Starting with a
Markov Decision Process (MDP) formulation, we make
an affine functional approximation to the optimal
dynamic programming value function. Then, we derive
the program (D1) which yields tighter bounds than the
CDLP based on the O-D demands. We give a column
generation procedure for solving (D1) within a desired
optimality tolerance. The numerical results also show
our conclusion and the policy perform from our
solution approach can outperform that from the CDLP.
REFERENCES