Winter 12-5-2004

E-Booking Control Problem in Sea Cargo Logistics

Chengxuan Cao

James S. K. Ang

Hengqing Ye

E-Booking Control Problem in Sea Cargo Logistics

Chengxuan Cao[1], James S. K. Ang[2], Hengqing Ye[2]

[1] The Logistics Institute - Asia Pacific, National University of Singapore, Singapore 119260
[2] School of Business, National University of Singapore, Singapore 117592

ABSTRACT

In this paper, we consider the e-booking control problem which includes optimal pricing and capacity allocation. In pricing model, customers may refuse bookings based on their willingness to pay the quoted price. Since customer behavior and characteristics are highly varying and not known in advance, we develop a stochastic pricing model which tracks customer behavior as well as the arrival process to maximize profit. For the multi-period capacity allocation of e-booking control problem, we present a two-stage stochastic mixed integer programming model and a heuristic algorithm. The solution to the model is found by maximizing the expected profit over the possible control decisions under the uncertainty of shipping capacity. Finally, we give numerical experiments demonstrate the efficiency of the algorithm.

***Keywords*:** e-booking control, container shipping industry, mathematical programming

1. INTRODUCTION

Alliances and partnerships are prevailing characteristics of the container shipping industry. A brutally competitive environment, low freight rates, and the need for carriers to increase their revenue have led to joint carrier operations. Carriers have to band together in operating alliances with the objectives of improving capacity utilization and more service flexibility. Unfortunately, this consolidation has not resulted in upward pressure on shipping rates, nor has it increased profitability for the carriers involved. The high cost of capital equipment and low rates of return that characterize container shipping compel carriers seek out every opportunity to improve their own with respect to operational efficiency and performance (see [3], [5]).

Effective booking control is a key to improved revenues and reduced container shipping costs and it has become more important when the container shipping industry has widely adopted information technologies. E-booking service supports e-business strategy for container shipping industry, which aims to make business processes easier and more cost efficient. As a part of sea cargo e-commerce strategy, web-based e-booking system enables the customer to electronically book shipments and to channel other shipment requirements to the carrier, such as cargo collection, delivery requests, and information required for customs clearance. In general, e-bookings may be made up to 30 days in advance. The customers can specify the number and type of containers needed, and service time windows at the origin and destination locations. Minimally, the time window information must include the earliest time containers may be loaded at the origin, and the latest time containers should be delivered to the destination. The objective of e-booking service is to help customers enjoy the competitive advantage of the Internet and provide an efficient, flexible and fast way of doing business. Customer Service staff can also help customers with more complicated bookings that require special expertise. The whole industry-shippers, forwarders, carriers and consignees-benefit from minimizing or eliminating duplicate data entry. It makes the shipment process more transparent and improves the accountability of service provide. This means that the monthly freight schedule booking process can be done by pressing one button. A new e-business strategy will improve customer service and efficiency for both carriers and their customers. This move to a more efficient and effective way of making cargo booking is an important part of operational level service route planning in container shipping industry.

To obtain service, customers first check the price quote for a given origin-destination service in e-booking website, and then subsequently make bookings under the quote. With the exception of certain ancillary charges, the carrier charges the customer a fixed price for transportation, and pays the transportation service providers directly out of this fee. Therefore, it is in the carrier's interest to minimize the transportation costs for most shipments. However, there must be a level of “reasonableness” in transit times, and some customers may be willing to pay a premium for faster service.

In this paper, we consider the e-booking control problem in the container shipping industry, which includes pricing and capacity allocation. The pricing and revenue management problem is formulated as nonlinear programming in which some parameters obtained from stochastic marked point process. The capacity allocation problem is formulated as two-stage stochastic mixed integer programming model which accounts for total profit in multi-period and constraints of limited shipping capacity. Efficient algorithms for the models and numerical experiments are proposed at last. The rest of the paper is organized as follows. The problem formulation and algorithm are described in Section 2. Empirical results are given in Section 3. We conclude the paper in Section 4.

2. PROBLEN FORMULATION

2.1. Pricing and Revenue Management

Consider an e-booking system for an ocean liner operator. Cargo bookings arrive at the system according to a compound Poisson process $\{X(t)\}_{t\geq 0}$. Its jump times T_1, T_2, T_3, … form a Poisson process with rate λ >0, and the jump magnitudes Y_1, Y_2, Y_3, …are *i.i.d.* random variables with Pareto distribution. The Pareto principle states that a large income stems from a very small number of customers. It is reasonable to assume that the fright volume of a booking follows Pareto distribution (see [1]). The Pareto probability density function with shape parameter a >1 and scale parameter b >0 is defined as

$$f(x) = \frac{ab^a}{x^{a+1}}, \text{ for } x \geq b \tag{1}$$

Figure 1 illustrates the Pareto probability density function for different values of shape a and scale b=40.

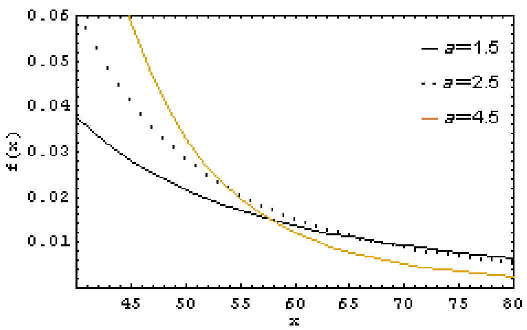

Figure 1. Pareto probability density function

When the quoted price is p, an arrival customer may accept or reject the price. Let ξ be the binary random variable, i.e., ξ =1 if the arrival customer accepts the quoted price; ξ =0, otherwise. We assume that the random variable ξ is independent of booking arrival process and the probability that an arriving customer accepts the quoted price p is a decreasing and differentiable function of p. The probability is chosen as

$$g(p) = 1 - (\frac{p}{\theta})^{\delta}, \text{ for } p \leq \theta \tag{2}$$

where δ and θ are constants. By varying the parameter δ, we can make the willingness as elastic as desired. The lower the value of parameter δ, the more willing are customers to book under the quoted price. Figure 2 shows the function $g(p)$ for different values of parameter δ and θ=10. The total profit in time t is:

$$Z(t) = \sum_{i=1}^{\infty} Y_i 1_{0 \leq T_i \leq t} \, p 1_{\xi=1} \tag{3}$$

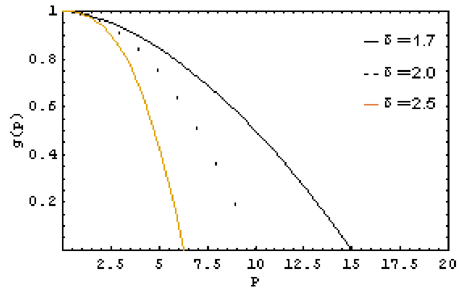

Figure 2. Probability of acceptance under quoted price p

Let total available capacity is C. Then we formulate the pricing and revenue management problem for single cargo type and single period as following nonlinear programming (M1):

$$\text{Maximize } EZ(t) = E\sum_{i=1}^{\infty} Y_i 1_{0 \leq T_i \leq t} \, p 1_{\xi=1} \tag{4}$$

subject to

$$E\sum_{i=1}^{\infty} Y_i 1_{0 \leq T_i \leq t} 1_{\xi=1} \leq C \tag{5}$$

Obviously, stochastic process $\{Z(t)\}_{t\geq 0}$ is a marked point process and the expectation of $Z(t)$ is (for details of marked point process, see [2]):

$$EZ(t) = EY_1 \cdot \lambda tp \cdot P\{\xi = 1\}$$
$$= \frac{ab\lambda tp}{a-1}[1 - (\frac{p}{\theta})^{\delta}] \tag{6}$$

Similarly,

$$E\sum_{i=1}^{\infty} Y_i 1_{0 \leq T_i \leq t} 1_{\xi=1} = \frac{ab\lambda t}{a-1}[1 - (\frac{p}{\theta})^{\delta}] \tag{7}$$

In the following theorem, we give the optimal of above nonlinear programming.

Theorem 2.1. The optimal solution of the nonlinear programming (M1) is

$$p^* = \max(\hat{p}, \overline{p}) \tag{8}$$

where

$$\hat{p} = \frac{\theta}{(\delta+1)^{1/\delta}}, \text{ and } \overline{p} = \theta[1 - \frac{c(a-1)}{ab\lambda t}]^{1/\delta}.$$

Proof. Let

$$\eta_1(p) = \frac{ab\lambda tp}{a-1}[1 - (\frac{p}{\theta})^{\delta}], \quad \eta_2(p) = \frac{ab\lambda t}{a-1}[1 - (\frac{p}{\theta})^{\delta}],$$

then we easily verify that the function $\eta_1(p)$ is concave in (0, +∞) and it takes maximum at point:

$$\hat{p} = \frac{\theta}{(\delta+1)^{1/\delta}} \tag{9}$$

On the other hand, the function $\eta_2(p)$ is decreasing in (0, θ) and it takes zero at point:

$$\overline{p} = \theta[1 - \frac{c(a-1)}{ab\lambda t}]^{1/\delta} \tag{10}$$

Therefore, the optimal solution of non-linear programming (M1) is $p^* = \max(\hat{p}, \overline{p})$.

2.2. Capacity Allocation

We assume that the capacity allocation decisions (for the entire planning horizon) have to be made, with only some knowledge of future scenarios of parameters. The overall objective is to determine a capacity allocation plan, such that the sum of each expected profit is maximized. To incorporate the uncertainty in the parameters, we assume that these parameters can be realized as one of S scenarios. The probability of scenario s will be denoted by p^s.

Before formally stating the problem, we introduce some notation:

Index Sets

$\widetilde{T}$: set of time periods $\{1, 2, \cdots, t, \cdots, T\}$.

$\widetilde{K}_t$: set of all cargoes received in period t, i.e. $\widetilde{K}_t = \{1, 2, \cdots, k, \cdots, K_t\}$.

$\widetilde{S}$: set of all scenarios $\{1, 2, \cdots, s, \cdots, S\}$.

Deterministic Parameters

r_{tkd} : per volume profit of cargo k which is received in period t and delivered in period d. It can be interpreted as the per volume net profit of cargo k, i.e., per volume profit of cargo k minus its per volume inventory cost.

τ_k : due date of cargo k. Each cargo has its due date requested by shipper in its booking status.

$v_{tk\tau_k}$: the volume of cargo k received in period t ready for delivery before its due date τ_k.

$w_{tk\tau_k}$: weight of cargo k received in period t ready for delivery before its due date τ_k.

Random Data

V_t^s : total available volume capacity in period t under scenario s.

W_t^s : maximum allowable weight capacity in period t under scenario s.

q_{d1}^s : overage cost per unit overage of volume capacity in period d under scenario s.

q_{d2}^s : shortage cost per unit short of volume capacity in period d under scenario s.

q_{d3}^s : overage cost per unit overage of weight capacity in period d under scenario s.

q_{d4}^s : shortage cost per unit short of weight capacity in period d under scenario s.

Decision Variables

x_{tkd} : binary variable, i.e., $x_{tkdj_k} = 1$ if cargo k is received in period t and is ready for delivery in period d before its due date τ_k, 0, otherwise.

y_{d1} : amount of overage of available volume capacity in period d.

y_{d2} : amount of shortage of available volume capacity in period d.

y_{d3} : amount of overage of allowable weight capacity in period d.

y_{d4} : amount of shortage of allowable weight capacity in period d.

The multi-period capacity allocation problem can then be formulated as following two-stage stochastic mixed integer programming model (M2):

$$Maiximize\ z = \sum_{t=1}^{T} \sum_{k \in \widetilde{K}_t} \sum_{d=t}^{\tau_k} v_{tk\tau_k} r_{tkd} x_{tkd} - \sum_{s=1}^{S} p^s Q^s(x) \quad (11)$$

subject to

$$\sum_{d=t}^{\tau_k} x_{tkd} \le 1 \quad \forall\, t \in \widetilde{T}, \quad k \in \widetilde{K}_t \quad (12)$$

$$x_{tkd} \in \{0, 1\}, \quad (13)$$

For $\forall\, t \in \widetilde{T}, \quad k \in \widetilde{K}_t, \quad d \in \{t, t+1, \cdots, \tau_k\}$,

where for all s,

$$Q^s(x) = Minimize \sum_{d=1}^{T} \sum_{i=1}^{4} q_{di}^s y_{di} \quad (14)$$

subject to

$$y_{d1} - y_{d2} = V_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \ge d}} v_{tk\tau_k} x_{tkd} \quad \forall\, d \in \widetilde{T} \quad (15)$$

$$y_{d3} - y_{d4} = W_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \ge d}} w_{tk\tau_k} x_{tkd} \quad \forall\, d \in \widetilde{T} \quad (16)$$

where all y_{di} are non-negative for $d \in \widetilde{T}$, i=1,2,3,4.

Because all random variables in (M2) are discretely distributed, and their joint distribution has a finite number of realizations, (M2) can be rewritten as the following large-scale mixed integer programming model (M3):

$$Maximize\ z = \sum_{t=1}^{T} \sum_{k \in \widetilde{K}_t} \sum_{d=t}^{\tau_k} v_{tk\tau_k} r_{tkd} x_{tkd} - \sum_{s=1}^{S} p^s \sum_{d=1}^{T} \sum_{i=1}^{4} q_{di}^s y_{di}^s \quad (17)$$

subject to

$$\sum_{d=t}^{\tau_k} x_{tkd} \le 1 \quad \forall\, t \in \widetilde{T}, \quad k \in \widetilde{K}_t, \quad (18)$$

$$y_{d1}^s - y_{d2}^s = V_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} v_{tk\tau_k} x_{tkd} \,, \qquad (19)$$

for $\forall\, d \in \widetilde{T}, \quad j \in \widetilde{J}, \quad s \in \widetilde{S}$

$$y_{d3}^s - y_{d4}^s = W_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} w_{tk\tau_k} x_{tkd} \,, \qquad (20)$$

for $\forall\, d \in \widetilde{T}, \quad s \in \widetilde{S}$

where $x_{tkd} \in \{0, 1\}$, for $\forall\, t \in \widetilde{T}, \quad k \in \widetilde{K}_t$, $d \in \{t, t+1, \cdots, \tau_k\}$ and all y_{di}^s are non-negative for $s \in \widetilde{S}$, $d \in \widetilde{T}$, $i = 1, 2, 3, 4$.

A common attitude in solving NP-hard combinatorial optimization problems (see [4]) is to not insist on optimality but dedicate research efforts to designing fast and high quality approximation methods. A greedy algorithm is chosen for solve the problem (M3). Its robust and implicit enumerative character ensures to achieve the optimal solution or a near optimal solution. In cases like this we can sacrifice the guarantee of optimality that is provided by it in favor of getting a reasonable answer quickly.

Let $E_d = \min\{E_d^s : s \in \widetilde{S}\}$, $V_d = \min\{V_d^s : s \in \widetilde{S}\}$, $W_d = \min\{W_d^s : s \in \widetilde{S}\}$, $\bar{v}_{tk\tau_k}^d = v_{tk\tau_k} / V_d$, $\overline{w}_{tk\tau_k}^d = w_{tk\tau_k} / W_d$. Then, we get a binary integer programming as follows (M4):

$$Maximize\ z = \sum_{t=1}^{T} \sum_{k \in \widetilde{K}_t} \sum_{d=t}^{\tau_k} v_{tk\tau_k} r_{tkd} x_{tkd} \qquad (21)$$

subject to

$$\sum_{d=t}^{\tau_k} x_{tkd} \leq 1 \quad \forall\, t \in \widetilde{T}, \quad k \in \widetilde{K}_t \qquad (22)$$

$$\sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} \bar{v}_{tk\tau_k}^d x_{tkd} \leq 1 \quad \forall\, d \in \widetilde{T} \qquad (23)$$

$$\sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} \overline{w}_{tk\tau_k}^d x_{tkd} \leq 1 \quad \forall\, d \in \widetilde{T} \qquad (24)$$

where $x_{tkd} \in \{0, 1\}$, $d \in \{t, t+1, \cdots, \tau_k\}$, $\forall\, t \in \widetilde{T}$, $k \in \widetilde{K}_t$,

Toyoda's heuristic for the Multi-Dimensional Knapsack Problem (MDKP) (see [6]) starts with no items (or all *x*'s being zero), and adds one item at a time iteratively as long as the solution is feasible. Following his approach, we propose a heuristic algorithm EBHA that provides a near-optimal solution to (M2) by means of concepts such as penalty vector and effective gradient introduced in [6].

The heuristic algorithm for the mixed integer programming (M3) is presented as follows.

Algorithm EBHA:

Step 1: Initialization.

Step 1.1: Let $z \leftarrow 0$, $\hat{z} \leftarrow 0$, $x_{tkd} \leftarrow 0$, $\hat{x}_{tkd} \leftarrow 0$.

Step 1.2: Let $K_U \leftarrow \phi$, where K_U is the set of accepted items.

Step 1.3: Assign all items to $K_D = K - K_U$, where K_D is the set of items not in K_U and $K = \{k : k \in \widetilde{K}_t, t \in \widetilde{T}\}$.

Step 1.4: Let $A_U^d \leftarrow (0, 0)$, where A_U^d is the total quantity vector of accepted items in period d.

Step 2: Let $K_C \leftarrow \{k : k \in K_D, \exists d \leq \tau_k\ s.t.$ $(\bar{v}_{tk\tau_k}^d, \overline{w}_{tk\tau_k}^d) \leq (1, 1) - A_U^{dj_k}\}$, where K_C is the set of candidate items.

Step 3: Check K_C. If K_C is empty, goto **Step 7**. Otherwise, proceed to the next step.

Step 4: Let $\overline{K}_C = \{(k, d) : k \in K_D,$ $(\bar{v}_{tk\tau_k}^d, \overline{w}_{tk\tau_k}^d) \leq (1, 1) - A_U^d\}$.

Step 4.1: If A_U^d is a zero vector, then we set

$$G_{kd} \leftarrow \frac{\sqrt{2}\, v_{tk\tau_k} r_{tkd}}{\bar{v}_{tk\tau_k}^d + \overline{w}_{tk\tau_k}^d}, \text{ for } (k, d) \in \overline{K}_C$$

Step 4.2: Otherwise, let $K_U^d = \{k' : k' \in K_U, x_{tk'd} = 1\}$, for $(k, d) \in \overline{K}_C$, we set

$$G_{kd} \leftarrow \frac{v_{tk\tau_k} r_{tkd} \sqrt{(\sum_{k' \in K_U^d} \bar{v}_{tk'\tau_k}^d)^2 + (\sum_{k' \in K_U^d} \overline{w}_{tk'\tau_k}^d)^2}}{\bar{v}_{tk\tau_k}^d \sum_{k' \in K_U^d} \bar{v}_{tk'\tau_k}^d + \overline{w}_{tk\tau_k}^d \sum_{k' \in K_U^d} \overline{w}_{tk'\tau_k}^d},$$

Step 5: Find that item k whose effective gradient is the largest in a period, i.e.,

$$G_{kd} = \max\{G_{k'd'} : (k', d') \in \overline{K}_C\}.$$

Step 6: Accept k. Let $K_U \leftarrow K_U + \{k\}$,

$$A_U^d \leftarrow A_U^d + (\bar{v}_{tk\tau_k}^d, \overline{w}_{tk\tau_k}^d), \quad z \leftarrow z + v_{tk\tau_k} r_{tkd},$$

$K_D \leftarrow K_D - \{k\}$, $x_{tkd} \leftarrow 1$. Then, goto **Step 2**.

Step 7: Let $z \leftarrow \hat{z}$, $x_{tkd} \leftarrow \hat{x}_{tkd}$,

$$y_{d1}^s \leftarrow V_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} v_{tk\tau_k} x_{tkd}, \forall\, d \in \widetilde{T}, s \in \widetilde{S},$$

$$y_{d3}^s \leftarrow W_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} w_{tk\tau_k} x_{tkd}, \forall\, d \in \widetilde{T}, s \in \widetilde{S},$$

$$z \leftarrow z - \sum_{s=1}^{S} p^s \sum_{d=1}^{T} (q_{d1}^s y_{d1}^s + q_{d3}^s y_{d3}^s),$$

$$\hat{z} \leftarrow z, \ \hat{x}_{tkd} \leftarrow x_{tkd},$$
$$\text{for} \quad t \in \widetilde{T}, \quad k \in \widetilde{K}_t, \quad d \in \{t, t+1, \cdots, \tau_k\}.$$

Step 8: If $K_D = \phi$, the procedure terminates. Otherwise, let $z \leftarrow 0$, $y_{di}^s \leftarrow 0$, for $s \in \widetilde{S}$, $d \in \widetilde{T}$, $j \in \widetilde{J}$, $i = 1,2,3,4$, and proceed to the next step.

Step 9: we set

$$G_{kd} \leftarrow \frac{\sqrt{0.125}\, v_{tk\tau_k} r_{tkd}}{\overline{v}_{tk\tau_k}^d + \overline{w}_{tk\tau_k}^d}, \text{ for } k \in K_D.$$

Step 10: Let $G_{kd} \leftarrow \max\{G_{k'd'} : k' \in K_D,$
$d' \in \{t, t+1, \cdots, \tau_k\}\}$.

Step 11: Let $K_D \leftarrow K_D - \{k\}$, $x_{tkd} \leftarrow 1$.

Step 12: For $d \in \widetilde{T}, s \in \widetilde{S}$, if $V_d^s \geq \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} v_{tk\tau_k} x_{tkd}$,

then $y_{d1}^s \leftarrow V_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} v_{tk\tau_k} x_{tkd}$,

otherwise, $y_{d2}^s \leftarrow \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} v_{tk\tau_k} x_{tkd} - V_d^s$;

if $W_d^s \geq \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} w_{tk\tau_k} x_{tkd}$, then

$$y_{d3}^s \leftarrow W_d^s - \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} w_{tk\tau_k} x_{tkd},$$

otherwise, $y_{d4}^s \leftarrow \sum_{t=1}^{d} \sum_{\substack{k \in \widetilde{K}_t \\ \tau_k \geq d}} w_{tk\tau_k} x_{tkd} - W_d^s$;

$$z \leftarrow \sum_{t=1}^{T} \sum_{k \in \widetilde{K}_t} \sum_{d=t}^{\tau_k} v_{tk\tau_k} r_{tkd} x_{tkd} - \sum_{s=1}^{S} p^s \sum_{d=1}^{T} \sum_{i=1}^{4} q_{di}^s y_{di}^s$$

Step 13: if $\hat{z} < z$, then $\hat{z} \leftarrow z$, $\hat{x}_{tkd} \leftarrow x_{tkd}$.
Otherwise, $x_{tkd} \leftarrow 0$.

Step 14: If $K_D = \phi$, the procedure terminates.
Otherwise, let $z \leftarrow 0$, $y_{di}^s \leftarrow 0$, for $s \in \widetilde{S}$, $d \in \widetilde{T}$, $j \in \widetilde{J}$, $i = 1,2,3,4$, and goto **Step 10**.

3. EMPIRICAL RESULTS

In this section, we implement the heuristic algorithm EBHA and compare its solution to optimal solution or LP (relaxation) optimal solution (as the upper bound for optimal solution). The algorithm has been coded in C++ and run under Microsoft Windows Server 2003 Standard Edition using a Server (Intel(R) Xeon(TM) CPU 3.06GHz and 1.0GB of RAM). CPU times were obtained through the C++ function clock(). To conduct our experiments we used randomly generated instances.

For each set of parameters T and K, we generated 10 random small scale instances, for which optimal solutions can be obtained by CPLEX 8.0. We tested heuristic solutions and optimal solutions or LP optimal solutions for all 10 instances, and tabulated the average relative gap and average computation time. Let z_{LP} be the optimum of LP relaxation, z_H be the lower bound by heuristic and z_O be the optimum of the problem. In table 1, the relative gap g_O is defined as

$$(\mathrm{z_O} - \mathrm{z_H}) / \mathrm{z_O} \times 100\%.$$

In table 1 and table 2, the relative gap g_L is defined as

$$(\mathrm{z_{LP}} - \mathrm{z_H}) / \mathrm{z_{LP}} \times 100\%.$$

Table 1 shows the results obtained for a set of small test problems. Test problems 1 have 2 scenarios, 2 periods and 150 items (cargoes); test problem 2 has 3 scenarios, 3 periods, 4 destination ports and 71 items, and so on. For comparison, the optimal solution has been computed using CPLEX 8.0. As can be seen from table 1, the obtained results seem to be encouraging. The gap between the optimal solution and the heuristic solution is small and the computation time is very short. Table 2 shows the results obtained for a set of large scale problems.

From our preliminary computation experiment, we believe that heuristic algorithm would be a very good candidate for solving the problem in time critical or real-time applications such as capacity allocation in e-booking control problem where a near optimal solution is acceptable, and fast computation is more important than guaranteeing optimal value.

4. SUMMARY

Critical to the e-booking control problem is understanding sea fright marketing environment and optimization technique, its impact on the utilization of available capacity and service route planning for liner operator. Since customer behavior and characteristics are highly varying and not known in advance, we develop a stochastic pricing model which tracks customer behavior as well as the arrival process to maximize profit. On the other hand, we have formulated the capacity allocation problem as the two-stage stochastic mixed integer programming model, and presented effective heuristic algorithm which provide fast and near optimal solution. We also presented experimental results to evaluate the algorithm using a wide range of problem instances. The results strongly suggest that the heuristic algorithm is very effective for time critical tactical or operations level decisions, where a near optimal solution is acceptable and fast computation is more important than guaranteeing optimal value.

S	T	K	Number of variables	Number of constraints	Instances tested	Average relative gap (%)	
						g_O	g_L
2	2	126	268	134	10	0.13	0.22
2	3	121	387	133	10	0.21	0.46
3	4	101	452	125	10	0.78	1.93
4	8	43	472	107	10	1.34	2.41
3	5	91	515	121	10	1.11	1.76
3	7	63	525	105	10	1.26	2.22

Table 1. Results for small test problems

S	T	K	Number of variables	Number of constraints	Instances tested	Average relative gap g_L (%)	Average CPU time (sec)	
							Heuristic	*LP*
3	3	586	1794	604	10	0.73	0.48	393.79
5	6	286	1836	346	10	2.34	0.28	45.08
4	6	473	2934	521	10	1.22	0.86	146.02
3	7	464	3332	506	10	1.34	0.77	147.74
3	4	886	3592	910	10	0.86	1.47	674.78
4	8	764	6240	828	10	1.12	2.33	2084.95

Table 2. Results for large scale problems

ACKNOWLEDGEMENT

This research is supported in part by the Academic Research Fund and the Centre for E-Business of National University Singapore, and the Strategic Research Programme (SRP) sponsored by Agency for Science, Technology and Research (A*STAR) of Singapore.

REFERENCES

[1] Arnold, B.C., *Pareto Distributions*, Burtonsville, Maryland: International Cooperative Publishing House, 1983.

[2] Bremaud, P., *Point Processes and Queues*, Springer Verlag, 1981.

[3] Cheung, R.K., C.-Y. Chen, "A two-stage stochastic network model and solution methods for the dynamic empty container allocation problem", *Thansp. Sci.*, Vol.32, No.2, pp142-162, 1998.

[4] Garey, M.R., D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, freeman, 1979.

[5] Lai, K.K., K. Lam, W. K. Chan, "Ship container logistics and allocation", *Journal of the Operational Research Society*, Vol. 46, No. 6, 687-697, 1995.

[6] Toyoda, Y., "A simplified algorithm for obtaining approximate solution to zero-one programming problems", *Management Science*, Vol. 21, pp1417-1427, 1975.