A Stochastic Programming Model for Container Dispatching of on Sea

Bin Wang

ABSTRACT

The paper adopted chance-constrained programming to build the model of optimization of dispatching containers on sea. In the model, the objective function is to maximize the profit of dispatching container on every lines. And the constraints to the model include satisfying the need of containers, the limit to transport ability and the number of empty container supported. In the model, the requirement of empty containers is a random variable. We turn the chance-constrained model into a integer programming. Then the linear model is solved by Lingo9.0. The aim of the paper is to provide a reasonable project of choosing shipping container route, so the profit of a shipping company can be maximized.1

KEYWORDS

line; container; container dispatching; chance-constrained programming; integer programming.

INTRODUCTION

In the late 1960s, Regular ship transportation are divided into grocery and container ones. Because of numbers of superiority of the container transportation, it plays key role in the world trade.

1 Bin Wang, College of international Vocational Education in Shanghai Polytechnic University, 2360 Jinhai Road, Shanghai, 201209, China.
Many scholars did much research in the subject. Xin S [1] used integer programming to study empty container repositioning on sea-bound, and he did simulation to analyze the influence of cost and income on Strategy of repositioning. Hengjiang L. and Xin S [2] built the model of empty container repositioning based on Petri net and did simulation by the use of EXSPECT. Daozhi Z. and Jian H. [3] constructed model of empty container repositioning of sea-carriage and land-carriage with aim of cost minimized. Zhenye W., Tiansheng S. and Ke Z. [4] discussed the optimization of container on sea. In the model, the empty containers and heavy containers are combined into a unified system, but the empty containers and heavy ones were not identified.

MATHEMATICAL MODEL

Parameters

$I$ : set of ports. $S_k$ : set of ports that provide empty containers ($S_k \subseteq I$). $D_k$ : set of ports that require empty containers ($D_k \subseteq I$). $(S_z, D_z)$ : set of ports pair that provide and require heavy containers. $S_z$ shows the ports that supply heavy containers. ($S_z \subseteq I$), $D_z$ shows the ports that need heavy containers. ($D_z \subseteq I$). For $\forall (i, j) \in (S_z, D_z)$, it shows that heavy containers shall be loaded from port $i$ and transported to port $j$. In port $j$, the heavy containers shall be unloaded. $H$ : the set of lines. $C^L_{ij}$ : loading cost of one container from port $i$ to port $j$ by line $h_i$ ($i \in (S_k \cup S_z), j \in (D_k \cup D_z), h \in H$). $C^U_{ij}$ : unloading cost of one container from port $i$ to port $j$ by $h_i$ ($i \in (D_k \cup D_z), j \in (S_z \cup S_k), h \in H$). $d_{qh}$ : profit of transporting one heavy container from port $i$ to port $j$ by $h_i$ ($i \in (S_k \cup S_z), j \in (S_z \cup S_k), h \in H$). $C^z_{ij}$ : cost caused by shortage of one heavy container between ports pair $(i, j)$ ($i \in (S_z, D_z), j \in (S_z, D_z)$). $C^k_{ij}$ : cost of renting an empty container in port $j$ ($j \in D_k$). $D^h_{i,j}$ : numbers of heavy containers that are required between ports pair $(i, j)$ ($i \in (S_z, D_z), j \in (S_z, D_z)$). $D^e_{i,j}$ : numbers of empty containers needed in port $j$. The number is a random variable. $S^i_k$ : limit to the supply ability for port $i$. $L^h$ : limit to the number of containers by line $h_i$. $L^k$ is decided by the number of regular ships and the category of the ships. ($h \in H$). $M^h$ : limit to the weight of containers by line $h_i$. $M^k$ is decided by the number of regular ships and the category of the ships. ($h_i \in H$). $l$ : volume of a container. $m_1$ : weight of a heavy container. $m_2$ : weight of a empty container.
Variables

\( x_{ij}^{zh} \): numbers of heavy containers that will be loaded from port \( i \), shipped by line \( h \) and be transported to and unloaded in port \( j \). \( (i, j) \in (S_z, D_z) \). \( S_z \) is the set of origination of heavy container, \( D_z \) is the destination of heavy container. \( H \) is the set of lines.

\( y_{ij} \): shortage numbers of heavy containers of \( (i, j) \) \( (i, j) \in (S_z, D_z) \)

\( x_{ij}^{kh} \): numbers of empty containers that will be loaded from port \( i \), shipped by line \( h \) and be transported to and unloaded in port \( j \). \( (i \in S_k, j \in D_k, h \in H) \). \( S_k \) is the origination of empty containers and \( D_k \) is the destination of empty containers.

\( x_j^k \): numbers of empty containers which are rented by port \( j \). \( (j \in D_k) \)

Objective Function

Max\( \sum_{(i, j) \in (S_z, D_z)} \sum_{h \in H} d_{ij}^{zh} x_{ij}^{zh} - \sum_{(i, j) \in (S_z, D_z)} \sum_{h \in H} \left( C_{ij}^L x_{ij}^{zh} + C_{ij}^h y_{ij}^{zh} + C_{ij}^U x_{ij}^{zh} + \sum_{(i, j) \in (S_z, D_z)} \sum_{k \in D_k} C_{ij}^k x_j^k \right) \) \( - \sum_{i \in S_k} \sum_{j \in D_k} \sum_{h \in H} \left( C_{ij}^L x_{ij}^{kh} + C_{ij}^h x_{ij}^{kh} + C_{ij}^U x_j^k \right) + \sum_{j \in D_k} C_j^k x_j^k \right) \) \( \cdots \) (1)

Min\( \sum_{k \in D_k} x_j^k \) \( \cdots \) (2)

Turn multi objective function into a linear programming.

- Max\( \sum_{k \in D_k} x_j^k \) \( \cdots \) (3)

Max\( \lambda_1 (1) + \lambda_2 (2) \) \( \cdots \) (4)

Constraints

Pr\{ \sum_{i \in S_k} \sum_{j \in D_k} \sum_{h \in H} x_{ij}^{kh} + x_j^k \leq D_j^k \} = \alpha \) \( \cdots \) (5)

\( \sum_{(i, j) \in (S_z, D_z)} \sum_{h \in H} x_{ij}^{zh} + x_{ij}^z = D_{(i, j)}^z \) \( \cdots \) (6)

\( \forall i \in S_k, \sum_{j \in D_k} x_{ij}^{kh} \leq S_i^k \) \( \cdots \) (7)
\[ \forall h_i \in H, \sum_{n \in S_k} \sum_{j \in D_k} l_{y}^{kh} + \sum_{(i,j) \in (S_k, D_k)} l_{y}^{zh} \leq L^h_i \] (8)

\[ \forall h_i \in H, \sum_{(i,j) \in (S_k, D_k)} m_i x_{ij}^{zh} + \sum_{n \in S_k} \sum_{j \in D_k} m_{ij} x_{ij}^{zh} \leq M^h \] (9)

\[ x_{ij}^{zh}, x_{(i,j)}, x_{ij}^{kh}, x_j^k \text{ are integers} \] (10)

**NUMERICAL EXAMPLES**

![Routing Network](image)

**REFERENCES**


