APPLICATION OF THE MINIMUM COST FLOW PROBLEM IN CONTAINER SHIPPING

Srečko Krile

Sveučilište u Dubrovniku (University of Dubrovnik)
Odjel računarstva (Department of Computing)
Čira Cara 4, 20000 Dubrovnik, Hrvatska
Tel: 385 20 445-739, Fax: 385 20 435-590,
e-mail: srecko.krile@unidu.hr

Abstract: In this paper the efficient algorithm for optimal cargo transport of N types of containers with limited ship capacity, minimizing the transport costs, is being developed. It can be applied for transport planning on a voyage route with multiple loading ports and multiple ports of discharge. The problem is solved with network optimization approach that can be seen as the Minimum Cost Multi-Commodity Flow Problem (MCMCF). The implemented algorithm is able to find appropriate load planning sequence and to ensure minimal loading, discharging and transshipment costs, but with fulfillment of cargo demands in a number of destination ports on the voyage route. Application of this efficient tool may significantly reduce transport costs and ensure maximal possible profit. It may improve the operation process in maritime transport technology.

Keywords: Transportation problem, Minimum Cost Multi-Commodity Flow Problem, Optimal transshipment cost, Transport of container ships

1. INTRODUCTION

One of the four different problem types concerning of cargo transportation is to find the sequence of cargo distribution between the sources and the destination, minimizing the transportation cost. Our problem has extension to transportation of different types of cargo by one means from multiple sources to multiple destinations. Such problem is well known in transportation of different types of containers by ship on voyage route with multiple loading ports and multiple ports of discharge (unload). Amount of containers is in firm correlation because the total capacity of ship is limited (TEU or GT). Taking into account cargo demands (in ports of discharge) for each type of cargo, and various loading ports with sufficient number of containers, we need optimal transportation plan and loading sequence to minimize shipping and loading expenses, transshipment cost and cost of ship’s stay in port (connected with duration of loading process). The problem of optimal transportation for two and more different commodities (cargo types) from multipleseveral ports of departure (sources) to multiple destinations with fulfillment of given demands in discharging ports (sinks) is very hard to solve. In scientific literature such problem is well known as the Minimum Cost Multiple Commodities Flow Problem (MCMCF); see [1]. Such problem exists in many fields and it is still the subject of many scientific papers. For example, the paper [4] deals about such problem between only one port of loading and only one port of discharge on the route.
Some interesting papers in that field are [2] and [3], but many other authors are investigating similar problems. In special circumstances of ship transportation we solved MCMCF by using network optimization approach. The mathematical model is formulated in section 2. The algorithm implementation is explained in section 3.

2. MATHEMATICAL MODEL

Different kinds of good (container types) are differentiated with \( i \) for \( i = 1, 2, \ldots, N \). The ship with defined cargo capacity is shipping from first port to last, marked with \( M \). The objective is to find a loading and transshipment strategy that minimizes the total cost incurred over the whole voyage route consisting of \( M \) ports. We need the loading plan for various container types in each port to serve \( N \) demand types in ports of destinations (port of discharge). The loading strategy consists of load/discharge plan for each port for each cargo type. The starting port on the route can be only for loading and last port on the route can be only for discharging; other ports on the route may be for both.

The transportation problem can be represented by a flow diagram of oriented acyclic network. The problem can be solved with network optimization technique as shortest path problem. On figures 1. and 2. there are examples of the Multi-Commodity Multi-Source Multiple Destination Network, and it is obvious that problem can be treated as the Minimum Cost Multi-Commodity Flow Problem (MCMCF). With such approach we enable optimisation, using significant less effort in programming than with other programming techniques. Figure 1. gives a network flow representation of MCMCF for \( N=3 \). Common node “O” is the source of containers for each cargo type with possible limitations. Some source ports can have limitation on charging capacity, but most of them are hub ports with capacity exceeding the ship’s earning capacity (than capacity of ship). The \( i \)-th row of nodes represents the capacity state of \( i \)-th type of container after loading in port \( m \). The links between nodes represent the amount of cargo transported between ports (in TEU).

![Figure 1. A network flow representation of the problem for \( N=3 \).](image-url)
In the mathematical model the following notation is used:

- $i, j$ and $k$ = indices for cargo type. The $N$ facilities are not ranked, just present different types of containers from 1, 2, ..., $N$.
- $m$ = indices the port of loading (charging) or discharging. The number of port of calls on the voyage including departure port $M$ ($m = 1, ..., M$).
- $u, v$ = indices for ports in sub-problem, $0 \leq u, ... , v \leq M$.
- $r_{im}$ = demands for loading of cargo $i$-th type in port $m$. For convenience, the $r_{im}$ is assumed to be integer. All traffic demands must be satisfied after discharging in last port on the route.
- $I_{im}$ = the amount of cargo of capacity type $i$ at the arrival in port $m$ (or, equivalently, at the departure in port $m-1$). We assumed that initially there is no idle capacity or capacity shortages before first port of loading, $I_{i1} = 0$.
- $k_{Ii}$ = the lowest step of possible capacity charging and discharging for capacity type $i$. In numerical examples it can be set e.g. $k_{Ii} = 10$ TEU.
- $x_{im}$ = quantity of $i$-th type of containers (cargo) being loaded on board in port $m$ (TEU)
- $z_m$ = the total loading amount for all types of containers in port $m$, i.e., $z_m = \sum_{i=1}^{N} x_{im}$
- $Q$ = ship’s deadweights in tons
- $W$ = ship’s transport capacity (TEU); $W \geq \sum_{i=1}^{N} I_{im}$ for any $m$; $m = 1, ..., M$
- $a_i$ = weight per unit of the $i$-th type of container,
- $Q \geq \sum_{i=1}^{N} I_{im} a_i$ for any $m$; $m = 1, ..., M$
- $g_{ir}$ = average number of TEUs of $i$-th type of container that can be loaded on board or discharged from board on daily basis,

The total cost over time includes:

a) Transshipment cost on distance between ports $m$ and $m+1$: $c_m = C_m \frac{d_m}{s}$

\begin{equation}
(2.1)
\end{equation}

$C_m$ = cost of container ship during voyage (per day); $d_m$ = distance (in nautical miles); $s$ = speed of container ship (in knots).

b) Loading and discharging cost in port $m$:

\begin{equation}
h_m = H_m \sum_{i=1}^{N} \left( \frac{x_{im} + r_{im}}{g_{im}} \right)
\end{equation}

\begin{equation}
(2.2)
\end{equation}

$H_m$ = cost of container ship stay in port $m$ (per day), and:

c) Penalty cost in form of loading set-up cost $p_a(z_m)$. This cost can be linear function with negative value to stimulate large loadings in the port.

We assume that all costs functions are linear or concave and non-decreasing. Some of them are reflecting economies of scale, so we have nonlinear optimization problem.

Total duration of voyage and ships stay in port during loading can be expressed as:

\begin{equation}
T = \sum_{m=1}^{M-1} \left( \frac{d_m}{s} + \sum_{j=1}^{N} \frac{x_{jm} + r_{jm}}{g_{jm}} \right)
\end{equation}

\begin{equation}
(2.3)
\end{equation}

The MCMCF problem can be formulated as follows:

\begin{equation}
\min \sum_{m=1}^{M-1} c_m(d_m) + \sum_{i=1}^{N} h_m(x_m) + p_m(z_m)
\end{equation}

\begin{equation}
(2.4)
\end{equation}
so that we have:
\[ I_{m+1} = I_m + x_m + -r_m \] (2.5)
\[ I_{i0} = I_{i,M+1} = 0 \] (2.6)
for \( m = 1, 2, \ldots, M; i = 1, 2, \ldots, N; \)

3. ALGORITHM IMPLEMENTATION

3.1. Definition of the capacity point

Generalizing the concept of the capacity state after loading in port \( m \), in which capacity of each cargo type is known in defined bounds, we define as a capacity point. In (3.1) \( \alpha \) denotes the vector of capacities \( I_{im} \) for all cargo types after loading in port \( m \), and we call it the capacity point.

\[ \alpha_m = (I_{1m}, I_{2m}, \ldots, I_{Nm}) \] (3.1)
\[ \alpha_1 = \alpha_{M+1} = (0, 0, \ldots, 0) \] (3.2)

Each column on flow diagram from fig. 2. represents a capacity point, consisting of \( N \) capacity values. (3.2) imply that idle no cargo for any type of capacity exists at the beginning and at the end of voyage route.

Let \( C_m \) be the number of possible capacity point values in port \( m \). If we consider that before any loading in starting port, same as after discharging in the last port: \( C_1 = C_{M+1} = 1 \), the total number of capacity points is:

\[ C_p = \sum_{m=1}^{M+1} C_m \] (3.3)

The total number of possible connections between capacity points is:

\[ N_d = \sum_{i=1}^{M} C_i \left( \sum_{j=1}^{M+1} C_j \right) \] (3.4)

Connections between two successive capacity points in two different ports represent minimum costs of transportation \( d_{uv}(\alpha_u, \alpha_{v+1}) \). Suppose that all costs are known, so the optimal solution for MCMCF can be found by searching for the optimal sequence of capacity points and their associated connections. As shown in figure 3. such problem can be formulated as a shortest path problem for an acyclic network in which the nodes represents all possible values of capacity points. In that case Dijkstra’s algorithm or any similar algorithm can be applied.

It is very important to reduce that number of capacity points and that can be done through imposing any of appropriate capacity bounds or by introduction of adding constraints.

![Figure 2. A network flow representation of a sub-problem for N=3](image-url)
3.2. Solving of a sub-problem

Associated value between two capacity points, that represents minimum transportation cost $d_{uv}(\alpha_u, \alpha_{v+1})$ we denoted as sub-problem. In MCMCF we have to find many cost values $d_{uv}(\alpha_u, \alpha_{v+1})$ that emanates two capacity points, from each node $(u, \alpha_u)$ to node $(v+1, \alpha_{v+1})$ for $v > u$. Most of the computational effort is spent on computing the sub-problem values. Any of them if it cannot be a part of the optimal sequence is set to infinity.

Then we can define $d_{uv}(\alpha_u, \alpha_{v+1})$ as follows:

$$
\min \left\{ \sum_{m=0}^{v} \left( \sum_{i=1}^{N} c_{im}(x_m) + h_{im}(I_{i,m}) + p_{im}(z_m) \right) \right\}
$$

where

$$I_{i,m+1} = I_{i,m} + X_i - R_i(u,v)$$

$$X_i = \sum_{m=0}^{v} x_m$$

$$R_i(u,v) = \sum_{m=0}^{v} r_{im}$$

for $i = 1, 2, 3, ..., N$

To compute the sub-problem value $d_{uv}$ it is convenient to describe this problem as a multi-facility multi-source and multi-destination network, that is represented by a flow diagram as we can see on fig. 1. and 2. At each node $(i,m)$ there is an external demand increment $r_{im}$. The nodes are connected by links, where each flow represents the amount of container transportation.

The MCMCF algorithm has to find ports of loading and unloading for all three capacity types over whole interval in sub-problem. We assume that all changes in one sub-problem are made in the same port that is much appropriate for real application. So, the algorithm is looking for unique port $m$ between two ports $u \leq m \leq v$ to minimize the equation (3.2.1). Such limitation reduces the computation complexity significantly without influence on optimal loading sequence and real application.

3.3. Single Location Problem

Approach described in paragraph above requires solving repeatedly a certain single location problem. Let SLS$_{ij}(m, X_i, \ldots, X_j)$ be a Single Location Sub-Problem associated with
port \( m \) for capacity type \( i, i+1, \ldots, j \) and corresponding values of capacity change intention \( X_i, X_{i+1}, \ldots, X_j \). For this problem the loading solution has to satisfy some basic properties:

\[
x_{it} \geq 0 \quad (3.3.1)
\]

It can be shown that a feasible flow in the network given in fig. 2. corresponds to an extreme point solution of the sub-problem only if it is not the part of any cycle (loop) with positive flows, in which all flows satisfy given bounds. One may observe that the absence of cycles with positive flows implies that each node has at most one incoming flow from the source node. That means that optimal solution of \( d_{uv} \) has at most one loading for each facility. Using a network flow approach, properties of extreme point solution are identified. These properties are used to develop an efficient search for the link costs \( d_{uv} \). Absence of such cycles with positive flows implies that extreme point solutions for sub-problem satisfy the following property:

\[
I_{im} \cdot x_{im} \leq 0, \quad (3.3.2)
\]

for: \( i, k, j = 1, 2, 3 \); \( m = 1, \ldots, M \)

Property (3.3.2) implies that the capacity of any capacity type is increased at a given port only if it doesn’t make cycles with positive flows. That approach is restrictive and it has a sense only if we allow loading of each container type in single loading port on the route. Any acceptable SLS13 loading solution for any sub-problem have to satisfy properties (3.3.2), so many loading solutions that are not part of optimal sequence could be eliminated from further computing. If none of the optimal solutions satisfies all properties we set \( d_{uv} = \infty \).

### 4. CONCLUSIONS

An efficient algorithm for optimal cargo transport of \( N \) types of containers with limited ship capacity is being developed. It can be applied for transport planning on the voyage route with multiple loading ports and multiple ports of discharge. This heuristic algorithm is capable to obtain the best possible result (near-optimal loading sequence) that is proved through many numerical examples. The numbers of capacity points are the measure of the complexity of the problem. That number, satisfying basic and additional properties of optimal flow, can picture the efficiency of this algorithm, which is proportionally reflected on computation time. The computation effort required to solve one sub-problem is \( O(N^2M) \). The number of all possible \( d_{uv} \) values depends of the total number of capacity points and requires the effort of \( O(M^3N^4R^{2(N-1)}) \). If there are no limitations on capacity state values the complexity of such heuristic approach is pretty large and increases exponentially with \( N \).

### LITERATURE


