A Greedy Construction Heuristic for the Liner Shipping Network Design Problem

Løfstedt, Berit

Publication date: 2010

Document Version
Early version, also known as pre-print

Citation (APA):
A greedy construction heuristic for the Liner Service Network Design Problem

Berit Løfstedt

Operations Research
Department of Management Engineering
Technical University of Denmark

May 5, 2010
The Liner Service Network Design Problem (LS-NDP)

Methods based on integer and linear programming relaxations

LS-NDP as a multilayered Multiple Quadratic Knapsack Problem

The greedy construction heuristic

Critique of model and method

Future work
Outline

1. The Liner Service Network Design Problem (LS-NDP)
2. Methods based on integer and linear programming relaxations
3. LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
4. The greedy construction heuristic
5. Critique of model and method
6. Future work
1 The Liner Service Network Design Problem (LS-NDP)
2 Methods based on integer and linear programming relaxations
3 LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
4 The greedy construction heuristic
5 Critique of model and method
6 Future work
Outline

1. The Liner Service Network Design Problem (LS-NDP)
2. Methods based on integer and linear programming relaxations
3. LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
4. The greedy construction heuristic
5. Critique of model and method
6. Future work
Outline

1. The Liner Service Network Design Problem (LS-NDP)
2. Methods based on integer and linear programming relaxations
3. LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
4. The greedy construction heuristic
5. Critique of model and method
6. Future work
The Liner Service Network Design Problem (LS-NDP)
Methods based on integer and linear programming relaxations
LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
The greedy construction heuristic
Critique of model and method
Future work
1. The Liner Service Network Design Problem (LS-NDP)
2. Methods based on integer and linear programming relaxations
3. LS-NDP as a multilayered Multiple Quadratic Knapsack Problem
4. The greedy construction heuristic
5. Critique of model and method
6. Future work
Problem definition

The Liner shipping network design problem

Given a complete graph $G'$ between a set of ports $P$, a fleet divided into vessel classes $A$ and a set of commodities $K$ determine a minimum cost network $G = (V, E)$ consisting of disjoint non-simple cyclic vessel routes to transport the most profitable subset of the commodities.
Characteristics of a service

Cyclic
Non-simple
Inbound vs. outbound direction

Figure: Example of a single service
Characteristics of a network

Figure: Network design

- Transhipment of cargo at transhipment hubs and main ports
- Capacity classes: feeder, panamax, super panamax
- Fixed schedule - mainly based on weekly port visits
Selection of previous work

Focus:
- Multiple routings (i.e. network design)
- Multiple hubs

Relevant literature:
- \#models = \#articles
- Main difference: transhipment

Figure: Transhipment of cargo
### Previous work

<table>
<thead>
<tr>
<th>Article</th>
<th>Method</th>
<th>Optimal</th>
<th>Transhipment</th>
<th>vessels/ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td>Lagrange, Benders</td>
<td>No</td>
<td>No</td>
<td>3v, 20p</td>
</tr>
<tr>
<td>[2]</td>
<td>Branch-&amp;-Cut</td>
<td>Yes</td>
<td>Yes, handling cost per container</td>
<td>6v, 20p</td>
</tr>
<tr>
<td>[3]</td>
<td>greedy, column generation, Benders</td>
<td>No</td>
<td>Yes, no cost</td>
<td>50v, 10p</td>
</tr>
<tr>
<td>[4]</td>
<td>tabu search, LP solver</td>
<td>No</td>
<td>Yes, individual cost per container</td>
<td>100v, 120p</td>
</tr>
</tbody>
</table>

**Table:** Overview of main articles with multiple route construction

- [1]: Rana & Vickson 1991
- [2]: Reinhardt & Kallehauge 2007
- [3]: Agarwal & Ergun 2008
- [4]: Alvarez 2009
Challenges

Scaling to a global liner shipping network
200+ ports, 200+ vessels

Scalability Issues:

Symmetry:
- Cyclic Routing
- Vessel Specs

Large scale multicommodity flow problem
Motivation

Good solutions to the liner shipping network design problem

- Competitive network
- Low cost network
- Inclusion of dynamic non-linear bunker cost calculation
- No optimality guarantee

Figure: Fictitious example of non-linear bunker curve
- Create a good model including bunker cost
- Build a local search framework (ALNS)
- Create a good model including bunker cost
- Build a local search framework (ALNS)
- Combining sets of:
  1. Construction Heuristics
  2. Destruction Heuristics
Create a good model including bunker cost
Build a local search framework (ALNS)
Combining sets of:
  1. Construction Heuristics
  2. Destruction Heuristics
Topic of this talk:
Create a good model including bunker cost
Build a local search framework (ALNS)
Combining sets of:
  1. Construction Heuristics
  2. Destruction Heuristics

Topic of this talk:
First building block:
  1. Greedy construction heuristic
Create a good model including bunker cost
Build a local search framework (ALNS)
Combining sets of:
1. Construction Heuristics
2. Destruction Heuristics

Topic of this talk:
First building block:
1. Greedy construction heuristic
Create a good model including bunker cost
Build a local search framework (ALNS)
Combining sets of:
1. Construction Heuristics
2. Destruction Heuristics

Topic of this talk:
First building block:
1. Greedy construction heuristic
2. Based on a simplified LS-NDP model with simplified cost structures
Model simplifications

Rephrase the problem:
Model simplifications

Rephrase the problem:

1. A set of routes
Model simplifications

Rephrase the problem:

1. A set of routes
2. Place port calls on routes
Model simplifications

Rephrase the problem:
1. A set of routes
2. Place port calls on routes

Multiple Quadratic Knapsack Problem (MQKP)
Routes=Knapsacks
Port calls=items
Model simplifications

Rephrase the problem:
1. A set of routes
2. Place port calls on routes

Avoid evaluating a large scale multicommodity flow problem

Multiple Quadratic Knapsack Problem (MQKP)
Routes=Knapsacks
Port calls=items
Model simplifications

Rephrase the problem:
1. A set of routes
2. Place port calls on routes

Avoid evaluating a large scale multicommodity flow problem

Multiple Quadratic Knapsack Problem (MQKP)
Routes=Knapsacks
Port calls=items

Profit function, $f$:
\[ f(distance, demand, transhipment) \]
## Layer characteristics

<table>
<thead>
<tr>
<th>Layer</th>
<th>Port types</th>
<th>Distances</th>
<th>Direct</th>
<th>Transport to Hub</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feeder</td>
<td>Spokes</td>
<td>Short secondary primary</td>
<td>1-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Main ports</td>
<td>Hubs</td>
<td>Medium primary secondary</td>
<td>3-8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Super Main ports</td>
<td>Panamax Hubs</td>
<td>Long secondary primary</td>
<td>6-12</td>
<td>Panamax</td>
<td></td>
</tr>
</tbody>
</table>

Table: Layer classification

Berit Løfstedt (DTU Management)  LS-NDP  May 5, 2010  12/22
Layer characteristics

<table>
<thead>
<tr>
<th>Layer</th>
<th>Port types</th>
<th>Distances</th>
<th>Direct</th>
<th>Transport to Hub</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feeder</td>
<td>Spokes</td>
<td>Short</td>
<td>secondary</td>
<td>primary</td>
<td>1-3</td>
</tr>
<tr>
<td></td>
<td>Main ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hubs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Layer characteristics

<table>
<thead>
<tr>
<th>Layer</th>
<th>Port types</th>
<th>Distances</th>
<th>Direct</th>
<th>Transport to Hub</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feeder</td>
<td>Spokes</td>
<td>Short</td>
<td>secondary</td>
<td>primary</td>
<td>1-3</td>
</tr>
<tr>
<td></td>
<td>Main ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hubs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Panamax</td>
<td>Main ports</td>
<td>Medium</td>
<td>primary</td>
<td>secondary</td>
<td>3-8</td>
</tr>
<tr>
<td></td>
<td>Hubs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Layer characteristics

<table>
<thead>
<tr>
<th>Layer</th>
<th>Port types</th>
<th>Distances</th>
<th>Direct</th>
<th>Transport to Hub</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feeder</td>
<td>Spokes</td>
<td>Short</td>
<td>secondary</td>
<td>primary</td>
<td>1-3</td>
</tr>
<tr>
<td></td>
<td>Main ports Hubs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Panamax</td>
<td>Main ports Hubs</td>
<td>Medium</td>
<td>primary</td>
<td>secondary</td>
<td>3-8</td>
</tr>
<tr>
<td>Super panamax</td>
<td>Main ports Hubs</td>
<td>Long</td>
<td>secondary</td>
<td>primary</td>
<td>6-12</td>
</tr>
</tbody>
</table>

**Table:** Layer classification
Multilayered algorithm

Figure: Multi layered knapsack interpretation of the LS-NDP

- Three layers: feeder, panamax and super panamax
- Port items: Scheduled port visits
- Each layer may have multiple visits to a port
Solve an MQKP for each layer

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>287</td>
<td>306</td>
</tr>
<tr>
<td>1</td>
<td>-25</td>
<td>42</td>
<td>742</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>513</td>
<td>0</td>
</tr>
</tbody>
</table>

Table: Profit matrix

- $V_{layer}$: items (scheduled port calls with the capacity class of this layer)
- $R_{layer}$: knapsacks (Services)
- Services are assigned a standard number of vessels
- Number of vessels = Duration in weeks
Specialised MQKP - Mathematical model

\[
\text{maximize}(\text{MQKP}) = \sum_{r \in \mathcal{R}} \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} p_{ij} x_i^f x_j^f + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{V}} p_j x_j^f
\]

subject to:

\[
\sum_{r \in \mathcal{R}} x_i^f = 1 \quad \forall i \in \mathcal{V}
\]

(Mutually exclusive)

\[
x_i^f x_j^f \geq y_{ij}^f \quad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R}
\]

(Activate edge variable)

\[
\sum_{j \in \mathcal{V}} y_{ij}^f - \sum_{j \in \mathcal{V}} y_{ji}^f = 0 \quad \forall i \in \mathcal{V}, r \in \mathcal{R}
\]

(Cyclic)

\[
\sum_{j \in \mathcal{V}} y_{ij}^f \leq 1 \quad \forall i \in \mathcal{V}, r \in \mathcal{R}
\]

(Simple)

\[
u_i^f - u_j^f + y_{ij}^f \sum_{i \in \mathcal{V}} x_i^f \leq \sum_{i \in \mathcal{V}} x_i^f - 1 \quad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R}
\]

(Connected)

\[
\sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} y_{ij}^f (t_{ij} + t_i) \leq \sigma(C_a) \quad \forall r \in \mathcal{R}_a, a \in \mathcal{A}
\]

(Duration)

\[
x_i^f \in \{0, 1\} \quad \forall i \in \mathcal{V}, r \in \mathcal{R}
\]

\[
y_{ij}^f \in \{0, 1\} \quad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R}
\]

\[
u_i^f \in \mathbb{Z}^+ \quad \forall i \in \mathcal{V}, r \in \mathcal{R}
\]

Quadratic objective function - heuristic solution method
Greedy parallel insertion

The football teaming principle
The knapsacks take turn at choosing the most profitable item among the remaining items

- Principle: parallel insertion
- Motivation: Distribution of difficult items
Algorithm

GreedyConstruction (instance)
1  layers ← FleetToLayers(instance)
2  ScheduleToItems(instance, layers)
3  profitIncrease ← TRUE
4  for each layer ∈ layers
5     do MAKEKnapsacks()
6       while (V_layer ≠ ∅ ∪ profitIncrease )
7          do profitIncrease ← FALSE
8           for each r ∈ R_{layer}
9              best ← NULL
10             bestValue ← 0
11           for each i ∈ V_{layer}
12               deltaValue ← \sum_{j \in r} p_{ij}
13               if (deltaValue > bestValue)
14               then
15                   bestValue ← deltaValue
16                   best ← i
17               if (bestValue > 0)
18               then
19                 profitIncrease ← TRUE
20                 UPDATEDemandMatrices(knapsack, best)
21                 r ← best
22               V_{layer} ← V_{layer} \backslash best
Results

- Solve an instance of 234 ports and roughly 14000 demands in 33 seconds
- Evaluated by Network specialists at Maersk Line
  1. The routings are overall realistic
  2. Emphasis on direct transportation
  3. Transhipment facilities are weak
  4. Good basis for a local search

Conclusion:
Good construction heuristic as initial solution for further local search
Critique of the approach

- Not based on the true objective i.e. the MCF problem
- Little interaction between layers
- Only tested on a single instance of the Maerskline network
- No transhipment cost, bunker cost or vessel deployment cost
- **Note:** Integration in ALNS will provide evaluation of true cost
Future work for MQKP heuristic

- Interaction between layers
- More realistic goal function
  1. Solve uncapacitated MCF
  2. Evaluate the transit times and the potential throughput
- Test on real life data (Benchmark suite in progress)
- Compare results to the network cost of the initial schedule
Future work for ALNS framework

- Fast delta evaluation of multi commodity flow problem
- Destruction/ construction heuristics
- Benchmark suite for Liner shipping
[K. Rana and R.G. Vickson]
“Routing container Ships Using Lagrangian Relaxation and Decomposition”
*Transportation Science* 25, 201-214 (1991)

[L. B. Reinhardt and B. Kallehauge]
“Network Design for Container Shipping Using Cutting Planes”
Conference on Maritime & Intermodal Logistics, Singapore, December 2007

[R. Agarwal and O. Ergun]
"Ship Scheduling and Network Design for Cargo routing in Liner Shipping”

[J.F. Alvarez]
“ Joint Routing and Deployment of a Fleet of Container Vessels”

[S. Røpke and D. Pisinger],
“A general heuristic for vehicle routing problems”,

[I. Akio and K. Shintani and S.Papadimitriou]
"Multi-port vs. Hub-and-Spoke port calls by containerships”
*Transportation Research Part E* 42 , 740-757 (2009)

[D. Pisinger, S. Ropke]
"Large Neighborhood Search”

*Book Chapter, http://www.diku.dk/hjemmesider/ansatte/sropke/Papers/Lns.pdf*