Extending the MIP toolbox to crack the Liner Shipping Fleet Repositioning Problem

Robin Pearce, Alexis Tyler and Michael Forbes

School of Mathematics and Physics
University of Queensland, Australia
robin.pearce@uqconnect.edu.au

July 28-29, 2016

What's the Problem?

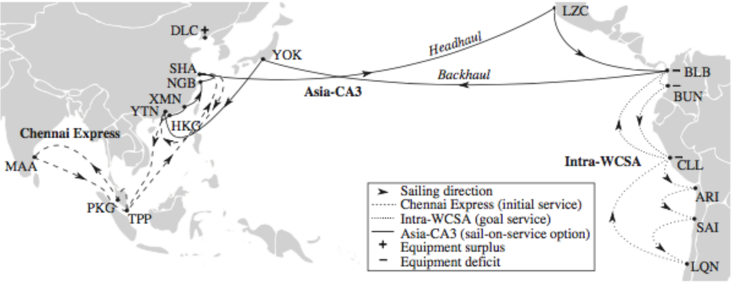


Figure: Source: Tierney et. al. (2015)

Definitions

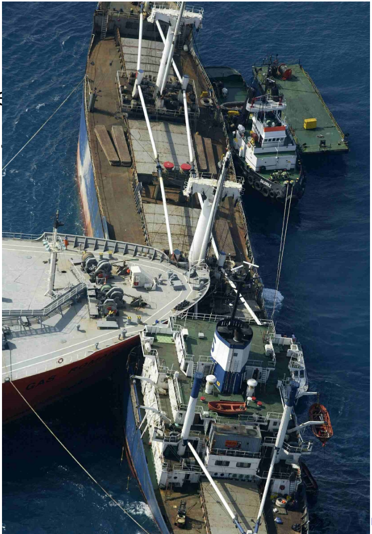
- Visit: A time and a place where demand can be loaded or unloaded
- Demand: A commodity which can be moved from one port to one of a set of ports for profit.
- Demand triplets: (o, d, q) where o is the origin node, d is a set of potential destination nodes and q is the cargo type.
- Graph sink: a node connected to all final destinations.

The Network

- Each node represents a place AND a time
- Each demand has a specific origin node
- The network is acyclic
- Each node can be visited by at most one ship

The Network

- Each node repre
- Each demand h
- The network is
- Each node can



Revised Formulation

Sets:

V	The set of nodes, including the graph sink τ
V'	The set of nodes, not including the graph sink τ
A	The set of arcs
A'	The set of arcs not connected to the graph sink τ
S	The set of ships
Q	The set of demand types (dry, refrigerated)
M	The set of demand triples (o, d, q)

Revised Formulation

Parameters:

u_s^q	The capacity of ship s for cargo type q
v_s	Starting visit of ship s
$r^{(o,d,q)}$	Revenue per unit of demand triplet (o,d,q)
c_i^{Mv}	Cost of loading and unloading containers at node i
c_{sij}^{Sail}	Cost of ship s sailing on arc (i,j)
c_{si}^{Port}	Port fee for ship s at node i
$a^{(o,d,q)}$	Available demand for triplet (o,d,q)
$In(i), Out(i)$	Set of nodes with an arc connecting to node i

Revised Formulation

Variables:

$y_{ij}^{s} \in \{0, 1\}$ 1 if vessel s is sailing on arc $(i, j) \in A$, 0 otherwise

$x_{ij}^{s,(o,d,q)} \in \mathbb{R}_0^+$ Amount of flow of demand triplet $(o, d, q) \in M$ on arc $(i, j) \in A'$ and ship $s \in S$.

Objective

$$\max \quad \Bigg\{ \sum_{(o,d,q)\in M} \sum_{s\in S} \sum_{j\in d} \sum_{i\in In(j)} (r^{(o,d,q)} - c_o^{Mv} - c_j^{Mv}) x_{ij}^{s,(o,d,q)}$$

$$- \sum_{s\in S} \sum_{(i,j)\in A'} c_{sij}^{\text{Sail}} y_{ij}^s - \sum_{j\in V'} \sum_{i\in In(j)} \sum_{s\in S} c_{sj}^{\text{Port}} y_{ij}^s \Bigg\}$$

(Revenue from servicing demand minus loading and unloading costs) -
(Cost of sailing on arcs) - (Cost of visiting ports)

Constraints

Only one ship visits each node

$$\sum_{s\in S}\sum_{i\in In(j)} y_{ij}^s \leq 1, \quad \forall j \in V'$$

Conservation of flow

$$\sum_{j\in Out(i)} y_{ij}^s = 1, \quad \forall s \in S, i = v_s$$

$$\sum_{i\in In(\tau)}\sum_{s\in S} y_{i\tau}^s = |S|$$

$$\sum_{i\in In(j)} y_{ij}^s - \sum_{i\in Out(j)} y_{ji}^s = 0, \quad \forall j \in V' \setminus \bigcup_{s\in S} v_s, s \in S$$

Constraints

Capacity constraints for each arc and cargo type

$$\sum_{(o,d,rf)\in M} x_{ij}^{s,(o,d,rf)} \leq u_s^{rf} y_{ij}^s, \qquad \forall (i,j) \in A', s \in S$$

$$\sum_{(o,d,q)\in M} x_{ij}^{s,(o,d,q)} \leq u_s^{dc} y_{ij}^s, \qquad \forall (i,j) \in A', s \in S$$

Can only move what is available, and only if the ship leaves the origin port

$$\sum_{i\in Out(o)} x_{oi}^{s,(o,d,q)} \leq a^{(o,d,q)} \sum_{i\in Out(o)} y_{oi}^s, \qquad \forall (o,d,q) \in M$$

Constraints

Flow conservation of Demand

$$\sum_{i \in In(j)} x_{ij}^{s,(o,d,q)} - \sum_{k \in Out(j)} x_{jk}^{s,(o,d,q)} = 0, \quad \forall (o,d,q) \in M, \quad j \in V' \backslash (o \cup d), s \in S;$$

Tighter capacity constraint

$$x_{ij}^{s,(o,d,q)} \leq y_{ij}^{s} \min(a^{(o,d,q)}, u_s^q), \qquad \forall (i,j) \in A', s \in S \quad (o,d,q) \in M$$

Paths Through the Network

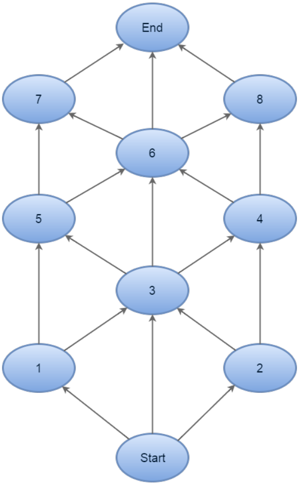


Figure: A simplified physical network

Paths Through the Network

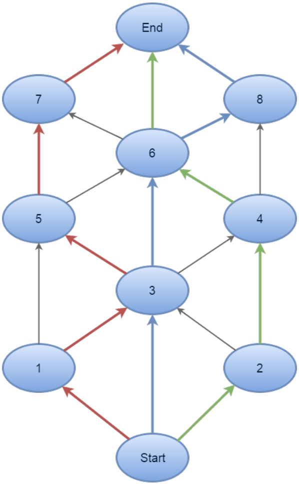


Figure: Example paths through the network

Column Generation

Parameters:

P	The set of paths through the network
$C_{sp} \in \mathcal{R}^+$	The profit of vessel s sailing on path p
$\delta_{isp} \in \{0, 1\}$	1 if vessel s sailing on path p goes through node $i \in V'$

Variables:

$Z_{sp} \in \{0, 1\}$ 1 if vessel s sails on path $p \in P, 0$ otherwise

Column Generation

Objective:

$$\max \sum_{p \in P} \sum_{s \in S} C_{sp} Z_{sp}$$

Constraints:

Each ship travels on one path

$$\sum_{p \in P} Z_{sp} = 1 \quad \forall s \in S;$$

One ship per node

$$\sum_{p \in P} \sum_{s \in S} \delta_{isp} Z_{sp} \leq 1 \quad \forall i \in V'$$

Result

Table: Comparison of solution times for different implementations

Instance ID	Reduced MIP time (secs)	Revised MIP time (secs)	Column Generation time (secs)
repos29p	2.34	2.85	3.52
repos30p	4.59	3.09	5.83
repos31p	9.82	6.95	12.19
repos32p	6.96	3.89	6.42
repos39p	Time	539.08	374.75
repos40p	Time	541.28	370.84
repos41p	Time	88.84	113.24
repos42p	Time	Time	1271.69
repos43p	Time (Time)	Time (Time)	Time (9632.16)
repos44p	Time (Time)	Time (Time)	Time (8924.85)

Nice Aspects of the Network

- Acyclic

Nice Aspects of the Network

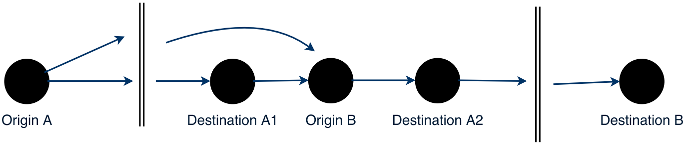
- Acyclic
- No two ships can visit the same node

Nice Aspects of the Network

- Acyclic
- No two ships can visit the same node
- No two ships can carry the same demand triple

Nice Aspects of the Network

- Acyclic
- No two ships can visit the same node
- No two ships can carry the same demand triple
- If any two demands are ever carried together at any time, then they will always be carried together.



Lazy reformulation of the sub-problems

Variables:

$x_s^{(o,d,q)} \in \mathbb{R}_0^+$	Amount of flow of demand triplet $(o,d,q) \in M$ on ship $s \in S$.
$y_{ij}^s \in \{0,1\}$	1 if vessel s is sailing on arc $(i,j) \in A$, 0 otherwise

New Sets:

M_i^{Orig}	The set of demands that start at node i
$\bar{M}_s$	The set of all demands which can be moved by ship s
$\bar{V}_{sq}^{\text{Orig}}$	The set of all nodes from which ship s can pick up demand of type q from $\bar{M}_s$
$A^{(o,d,q)}$	The set of arcs on which demand (o,d,q) can possibly travel

Lazy Reformulation of the sub-problems

For each $s \in S$:
Cannot pick up more than the capacity of the ship at any port, and only if we visit that port

$$\sum_{(k,d,q) \in M_k^{\text{Orig}}} x_s^{(k,d,q)} \leq \sum_{\substack{j \in Out(k) \\ (k,j) \in A'}} u_s^{dc} y_{kj}^s, \quad \forall k \in \bigcup_{q \in Q} \bar{V}_{sq}^{\text{Orig}}$$

$$\sum_{(k,d,rf) \in M_k^{\text{Orig}}} x_s^{(k,d,rf)} \leq \sum_{\substack{j \in Out(k) \\ (k,j) \in A'}} u_s^{rf} y_{kj}^s, \quad \forall k \in \bar{V}_{s,rf}^{\text{Orig}}$$

Lazy Reformulation of the sub-problems

For each $s \in S$:
Cannot carry any more of one demand than the capacity of the ship or the availability of the demand, and only if the ship passes through the origin and at least one of the destinations.

$$x_s^{(o,d,q)} \leq \min\left(a^{(o,d,q)}, u_s^q\right) \sum_{\substack{j \in Out(o) \\ (o,j) \in A^{(o,d,q)}}} y_{oj}^s \quad \forall (o,d,q) \in M$$

$$x_s^{(o,d,q)} \leq \min\left(a^{(o,d,q)}, u_s^q\right) \sum_{j \in d} \sum_{\substack{i \in In(j) \\ (i,j) \in A^{(o,d,q)}}} y_{ij}^s \quad \forall (o,d,q) \in M$$

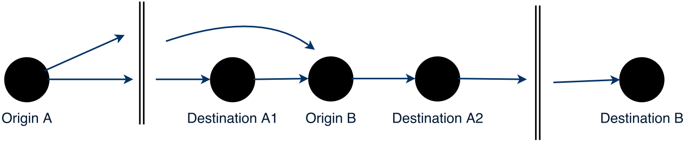
Lazy Reformulation of the sub-problems

For all nodes a ship visits, if the sum of the demands at node i exceed the capacity of the ship, add one of:

$$\sum_{\substack{(o,d,q)\in M \\ j\in Out(i) \\ (i,j)\in A^{(o,d,q)}}} x_s^{(o,d,q)} \leq u_s^{dc} \qquad \sum_{\substack{(o,d,rf)\in M \\ j\in Out(i) \\ (i,j)\in A^{(o,d,rf)}}} x_s^{(o,d,rf)} \leq u_s^{rf}$$

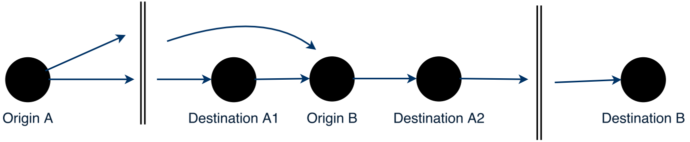
(The sum of all demands that could possibly travel through node i must obey the relevant capacity of the ship)

Splitting Demands



A demand triple, (o, d, q), must be split if the following criteria are met:

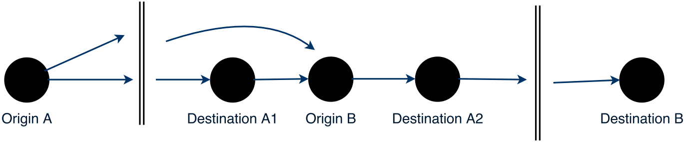
Splitting Demands



A demand triple, (o, d, q), must be split if the following criteria are met:

- Two destinations $d_1, d_2 \in d$ such that d_2 is reachable from d_1,

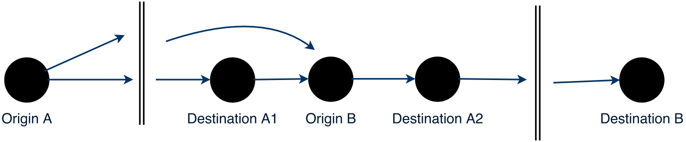
Splitting Demands



A demand triple, (o, d, q), must be split if the following criteria are met:

- Two destinations $d_1, d_2 \in d$ such that d_2 is reachable from d_1,
- There exists another demand triple (o^*, d^*, q^*) such that o^* is reachable from d_1, and d_2 is reachable from o^*,

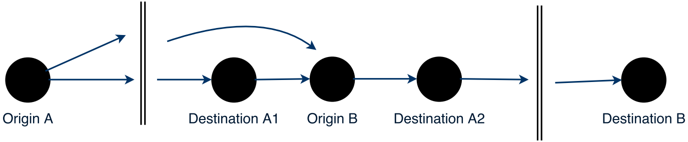
Splitting Demands



A demand triple, (o, d, q), must be split if the following criteria are met:

- Two destinations $d_1, d_2 \in d$ such that d_2 is reachable from d_1,
- There exists another demand triple (o^*, d^*, q^*) such that o^* is reachable from d_1, and d_2 is reachable from o^*,
- There exists a destination $d_3 \in d^*$ such that d_3 is reachable from d_2, and

Splitting Demands



A demand triple, (o, d, q), must be split if the following criteria are met:

- Two destinations $d_1, d_2 \in d$ such that d_2 is reachable from d_1,
- There exists another demand triple (o^*, d^*, q^*) such that o^* is reachable from d_1, and d_2 is reachable from o^*,
- There exists a destination $d_3 \in d^*$ such that d_3 is reachable from d_2, and
- There exists a path between o and o^* that does not pass through d_1.

Splitting Demands

A demand triple, (o, d, q), must be split if the following criteria are met:

- Two destinations $d_1, d_2 \in d$ such that d_2 is reachable from d_1,
- There exists another demand triple (o^*, d^*, q^*) such that o^* is reachable from d_1, and d_2 is reachable from o^*,
- There exists a destination $d_3 \in d^*$ such that d_3 is reachable from d_2, and
- There exists a path between o and o^* that does not pass through d_1.

We then split the demand triple (o, d, q) into separate variables $x_s^{(o,d_i,q)}$ $\forall d_i \in d$, and add a constraint to enforce the availability of the demand:

$$\sum_{d_i \in d} x_s^{(o,d_i,q)} \leq a^{(o,d,q)}$$

Result

Instance ID	Revised MIP time (secs)	Column Generation time (secs)	Lazy Constraints time (secs)
repos29p	2.85	3.52	1.70
repos30p	3.09	5.83	2.78
repos31p	6.95	12.19	6.88
repos32p	3.89	6.42	3.19
⋮	⋮	⋮	⋮
repos39p	539.08	374.75	35.12
repos40p	541.28	370.84	29.06
repos41p	88.84	113.24	17.00
repos42p	Time	1271.69	57.41
repos43p	Time	Time	223.21
repos44p	Time	Time	222.63

Table: Comparison of solution times for different implementations

Model Size Comparison

Table: Comparison of model sizes for instance repos44p

Method	Rows	Columns	Non-zeros
Reduced MIP	171102	269796	1858378
Column Generation sub-problems ($\times 11$)	136391	76612	426025
Col. Gen. and Lazy Constraints sub-problems ($\times 11$)	1627	6715	33475

Number of Added Constraints

	# Ships		# Constraints	
Instance	Dry	Reef	Dry	Reef
repos10p-repos14p	1	0	4	0
repos15p-repos16p	0	3	0	7
repos36p	0	1	0	1
repos39p	0	1	0	2
repos42p	3	10	8	21
repos43p	2	10	4	22
repos44p	2	10	5	27

Table: Number of ships affected by lazy constraints, and number of constraints added for each instance.

Papers

Tierney, Áskelsdóttir, Jensen and Pisinger (2015) Solving the liner shipping fleet repositioning problem with cargo flows, *Transportation Science* **49**(3), pp. 652-674

Pearce, Tyler and Forbes (2016) Column Generation and Lazy Constraints for solving the Liner Ship Fleet Repositioning Problem with cargo flows *(In Publication)*, pre-print available at http://arxiv.org/abs/1603.02384