Datalogisk Institut
KØBENHAVNS UNIVERSITET

DIKU

Liner Shipping Revenue Management with Respositioning of Empty Containers

Berit Løfstedt
David Pisinger
Simon Spoorendonk

Technical Report no. 08-15
ISSN: 0107-8283

Dept. of Computer Science
University of Copenhagen • Universitetsparken 1
DK-2100 Copenhagen • Denmark

Liner Shipping Revenue Management with Repositioning of Empty Containers

Berit Løfstedt · David Pisinger · Simon Spoorendonk
berit@diku.dk pisinger@diku.dk spooren@diku.dk
DIKU Department of Computer Science, University of Copenhagen
Universitetsparken 1, DK-2100 Copenhagen Ø, Denmark

November 12, 2008

Abstract

Repositioning of empty containers pose a significant cost in the shipping industry due to the large difference in export and import between some parts of the world, e.g., North America and Asia. Dejax and Crainic [9] estimate, that movement of empty containers comprise up to 40% of all container movements. This paper presents a revenue management model for a liner shipping company where the repositioning of empty containers is taken into account. The aim is to maximize the profit of transported cargo in a network, subject to the cost and availability of empty containers. The model is an augmented multi-commodity flow problem with additional inter-balancing constraints to control repositioning of empty containers. An arc flow formulation is Dantzig-Wolfe decomposed to a path flow formulation, where the LP relaxation is solved with a delayed column generation algorithm. A feasible IP solution is hereafter found by rounding down the LP solution and adjusting flow balance constraints with leased containers. Computational results are reported for eight instances based on real-life shipping networks. Solving the LP relaxed path flow model with a column generation algorithm outperforms solving the LP relaxed arc flow model with the *CPLEX* barrier solver even for very small instances. The proposed algorithm is able to solve instances with 234 ports, and 293 vessels for 9 time periods in 34 minutes. The integer solutions found by rounding are computed in less than 5 seconds and the gap is within 0.01% of the LP upper bound, which is assumed to be below the level of uncertainty of the input data. The solved instances are quite large compared to computational results in the reviewed literature on models for empty container repositioning.

1 Introduction

This paper presents a revenue management model for strategic planning within a *liner shipping* company. A revenue management model is a strategic tool, that given a schedule and a fleet over time decides which orders are profitable to transport with the planed capacity. A mathematical model is presented for maximizing the profit of cargo transportation while considering the possible cost of repositioning empty containers. The model is denoted revenue management with repositioning of empty containers (RMREC). Empty containers tend to accumulate at import intensive regions due to a significant imbalance in world trade. Therefore, repositioning empty containers to export intensive regions impose a large cost on liner shippers. RMREC incorporates the potential repositioning cost such that the profit of an order takes into account the derived demand for empty containers. As opposed to most models RMREC permits load rejection, since we believe, that an unprofitable order may be rejected due to capacity constraints in the liner shipping network.

A *liner shipping* company is a shipping operator with a public itinerary and schedule visiting certain ports at a given service frequency. The objective is to maximize profit for freighting optional cargo between ports. A liner shipping company differs from *industrial shipping*, where the objective is to minimize the transportation cost of delivering all cargo, and from *tramp shipping*, where the objective is to maximize the profit of optional cargo while delivering obligated cargo. Current practice with regards to empty containers is to have conservative stock policies and empty deadweight on vessels to ensure sufficient availability of empty containers. With RMREC, we hope to reduce deadweight on board vessels and to minimize stock of empty containers. Furthermore, the model may be used to investigate alternative leasing policies for liner shippers.

The strategic booking decision of a liner shipper considering empty container repositioning can be described as a specialized multi-commodity flow problem with inter-balancing constraints to control the flow of empty containers. A commodity in logistic terms is a pair (O, D), where O is the origin and D is the destination of a container demand. The set of commodities is denoted K. The network is represented by a graph $G = (N, A)$, where the node set N represents the ports and the arc set A represent the scheduled itineraries. The capacity associated with each arc is determined by the assignment of vessels to the schedule. The objective is to find a set of feasible paths in the network such that the profit of routing cargo between the (O, D) port pairs is maximized.

The classical formulation of the standard multi-commodity flow problem is the arc flow formulation with $|K||A|$ variables and $|A| + |K||N|$ constraints due to flow conservation at every node. Although the number of variables is polynomially bounded, it will be huge for a global shipping network. In addition, a large constraint set results in poor performance for the simplex method. Dantzig-Wolfe decomposition can be applied to generate a path flow formulation with only $|A|+|K|$ constraints. However, the number of variables in the path flow formulation may exponential. To circumvent this problem we use delayed column generation, as it can be proven that at most $|K| + |A|$ paths carry positive flow [1]. The pricing problem is a shortest path problem, where the cost of a path represents the reduced cost of a path variable.

RMREC is an augmented multi-commodity flow problem where the extra constraints stem from the inter-balancing constraints which ensure repositioning or leasing of empty containers at nodes with a positive net flow. Due to the structure of the dual problem, the arc costs of the pricing problem for the path flow model are positive which results in a polynomially solvable pricing problem when considering column generation. As containers cannot be split, RMREC is an integer multi-commodity flow problem which is $\mathcal{NP}$-hard. A nice property of the path flow formulation is, that flow conservations constraints are implicitly satisfied on a path. Hence, a feasible integer solution can be obtained by rounding down all fractional variables and supplying empty containers through a leasing variable at nodes with violated inter-balancing constraints.

Solving both standard and augmented integer multi-commodity flow problems is a well-studied area within airline management, e.g., crew scheduling [10] and fleet assignment [11, 3] where a branch-and-price algorithm on the path flow formulation is used to find an integer solution. The integer variables of these known problems are mostly binary. However, the integer variables of the arc flow model of RMREC are large integer numbers because demands are expressed in containers as the total demand between any two (O, D) port pair. Because the number of containers transported in a global shipping network is huge, rounding down the fractional part of demands may be considered insignificant. This is confirmed by experimental results in this paper. Therefore, we consider the LP relaxation of path model of RMREC and obtain a heuristic integer solution by rounding down the LP solution of the path flow model.

The contribution of this paper is to present an augmented multi-commodity flow formulation of the container transportation problem considering repositioning of empty containers (RMREC). A basic arc flow model of RMREC is decomposed into a path flow model. The LP relaxation of RMREC is solved with a delayed column generation algorithm. Computational results are reported for eight instances based on real life shipping networks. The results show that the delayed column generation algorithm for the path flow model clearly outperforms solving the arc flow model with the *CPLEX* barrier solver. Instances with up to 234 ports and 293 vessels for 9 time periods were solved in less than 34 minutes with the column generation algorithm. The largest instance solved for 12 time periods contains 151 ports and 222 vessels and was solved in less than 75 minutes. It is shown, that high quality integer solutions within 0.01% from the LP upper bound of the path flow formulation can be found by a simple rounding heuristic.

The following section describes related work. Section 3 describes the network representation used throughout this paper. Section 4 presents the arc flow model, and Section 5 presents the decomposed path flow model and the pricing problem used in the delayed column generation algorithm. Section 6 reports our computational results on eight generated test instances. Section 7 provides some concluding remarks and future work of RMREC.

2 Literature Overview

According to Ronen et al. [16] papers on optimization based decision support systems within shipping are scarce. There is an increasing interest in operations research within the area of shipping, but most papers concern scheduling and routing of vessels. Furthermore, most papers are concerned with industrial and tramp shipping. Within the area of liner shipping only a few references are found and they concern deployment of vessels [16]. Christiansen et al. [5] describe models for designing shipping networks for a traditional liner operation as well as for a hub-and-spoke liner network. According to the paper, a booking is accepted if there is space available on a vessel. This may lead to non-optimal decisions since the space may be used more profitably by demands in subsequent ports on the route. However, the issue of empty container availability is not regarded as a component of cargo profitability, although the connection to profit is evident. The paper encourages research into the area of revenue management and booking models, but very little work has been published on this subject as mentioned by, e.g., Ronen et al. [16], Christiansen et al. [5].

The airline industry holds several similarities with the booking models and flow constraints encountered in the maritime industry. Especially with regard to the relation of the underlying network structure. RMREC was originally inspired by Bartodziej and Derigs [2] who presented a revenue management model for Cargo Airlines. The model is a special multi-commodity flow problem which is solved by column generation.

Hane et al. [11] solve an airline fleet assignment problem as a multi-commodity problem, where air-crafts need repositioning and where aggregation of the graph is considered. Desaulniers et al. [10] solve a crew scheduling problem using multi-commodity flows, where the problem of repositioning crew is mentioned, but not thoroughly treated. Bélanger et al. [3] solve a fleet assignment problem with time windows as a multi-commodity flow problem.

Empty container repositioning can be regarded as empty flow in a network. Empty flows have been studied within all areas of the transportation sector because they represent a significant cost. In a survey on empty flows, Dejax and Crainic [9] estimate that up to 40% of all movements for rail cars and containers are empty. The need for models considering empty and loaded movements simultaneously is emphasized. Crainic et al. [7] present a multi mode multi-commodity location-distribution problem with inter-depot balancing requirements. The model is primarily a location problem deciding the number and locations of inland depots for empty vehicles. However, it also determines the empty flows between depots according to the inter-balancing constraints. The model has been tested on data from a European company, which operated 23 major European ports at the time. A large reduction in the number of inland depots is reported, which along with management of the empty flows represent a 47% annual saving for the company. An international liner shipping company of today will span the globe and service hundreds of ports. Furthermore, container vessels have increased from 5,9% to 9% of the world fleets total deadweight capacity [16] from 1995 to 2001.

Crainic et al. [8] present a dynamic and stochastic model for the empty container allocation problem. The context is an international shipping company with focus on the land operations, i.e., movements between customers and depots. The paper gives a very thorough description of the container trade with regards to the space and time of events along with the complex issue of asymmetric substitution between container-types. A single- and a multi-commodity model with containers as commodities are presented using a time-space network in a rolling horizon manner.

Shen and Khoong [17] present a decision support system for empty container distribution planning for a shipping company at port level. The paper describes a network optimization model and a heuristic to solve the problem but actual implementation and results are not reported.

Cheung and Chen [4] have developed a two stage stochastic network model for the dynamic empty container allocation problem. The model minimizes the total cost of repositioning or leasing containers at deficit ports and is highly related to Crainic et al. [8].

Li et al. [13] present the empty container problem as a non standard inventory model at a port to reduce holding of redundant empty containers. The paper also explores finite and infinite horizon methods.

The above papers all assume no load rejection as competition in the trade is fierce. Hence simultaneous optimization of loaded movements is irrelevant. Shintani et al. [18] are the first to discuss the possibility of rejecting unprofitable cargo within the context of designing container ship networks with regard to empty container repositioning. The model is a knapsack formulation choosing the ports to call, with an underlying model choosing the optimal port calling sequence under the assumption that all demand is satisfied in ports called. Repositioning of empty containers is only allowed in case of excess capacity making the cost of the repositioning negligible. The incurred cost is the penalty cost of storing or leasing empty containers. Using a genetic algorithm, results are reported for a model with 20 ports in Asia.

None of the above papers solve problems of a size corresponding to present shipping networks. Cheung and Chen [4] perform experiments for 3 randomly generated networks, where the largest instance has 10 ports, 6 voyages/vessels and 42 time periods. Shintani et al. [18] solve test instances for 5-8 ports out of 20 potential ports. The number of voyages/vessels is not declared. The number of time periods is 52.

3 Network Representation

RMREC may be modeled as a multi-commodity flow problem with inter-balancing constraints having as objective to maximize the profit of the demanded flow in a capacitated network.

The network consists of a set of unique ports P connected by the services offered by the liner shipping company. All services are cyclic. Since a service may take months to rotate, the network must be modeled over time. Let T be the set of time periods.

A *time-space* network is created as a graph $G = (N, A)$, where $N = \{p^t \mid p \in P, t \in T\}$ is the set of nodes. Let $A = A_G \bigcup A_R$ and let $A_G = \{(p^t, p^{t+1}) \mid p^t, p^{t+1} \in N\}$ be the set of ground arcs representing the stock at a port between two subsequent time periods. Let $A_R = \{(p^t, q^{t'}) \mid p^t, q^{t'} \in N,\ t \leq t',\ p \neq q\}$ be the set of travel arcs representing a voyage on a vessel between two ports $p, q \in P$ departing at time $t \in T$ and arriving at time $t' \in T$. The capacity of an arc $a \in A_R$ is given by the capacity of the vessels on the service at that specific time. An illustration of the *time-space* network may be seen in Figure 1 where time spans the x-axis and space, i.e., the geographical location of ports spans the y-axis.

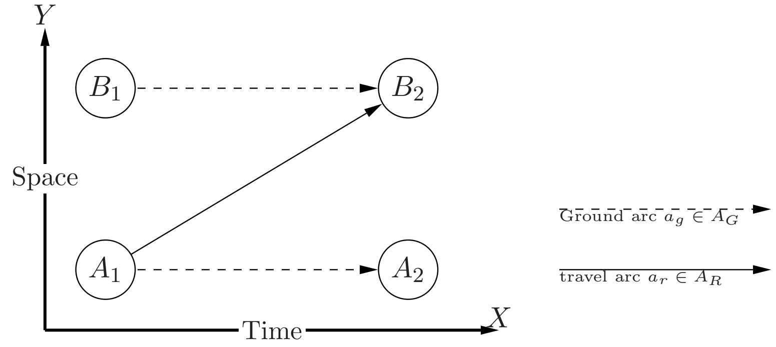


Figure 1: Example of time-space network - dotted arcs are ground arcs.

Each origin O_k and destination D_K of commodity k is given by a port $p \in P$ and time $t \in T$. For instance the tuple $(O_k, D_k, d_k, s_k) = (\texttt{BUA}_{08}, \texttt{BRV}_{09}, 2000, 220)$, means that the origin O_k is Buenos Aires (BUA) in month 08, the destination D_k is Bremerhaven (BRV) in month 09, the demand is 2000 TEU (twenty foot container type) and the sales price s_k is 220. The origin and destination time may be thought of as a *time window* for the delivery of the commodity.

In RMREC no consideration is taken for substitution of container types and it is assumed that all demands are accounted for with the smallest container type, enabling us to scale all larger container types to the smallest container type.

The granularity of time may be defined in two ways resulting in networks with different properties. T may represent the schedule directly if all events for a set of voyages is defined by a point in time, i.e., the voyage of a vessel gives rise to a totally ordered set of time units. This is illustrated in Figure 2. The resulting graph is directed, acyclic and can be topologically sorted. Hence, finding a shortest path can be done in $O(N + A)$ time.

The network will be very sparse, but may increase the number of path variables for a commodity, if the time windows of the commodities are not tight. Each vessel represents an itinerary and has its own nodes and arcs in the network. However, several vessels offer the same service with regard to visited ports on a voyage. This means, that there may be several daily departures from a port heading to the same destination. Figure 2 illustrates the issue: a commodity from port $A1$ to port $C4$ can be serviced by four possible paths with an identical

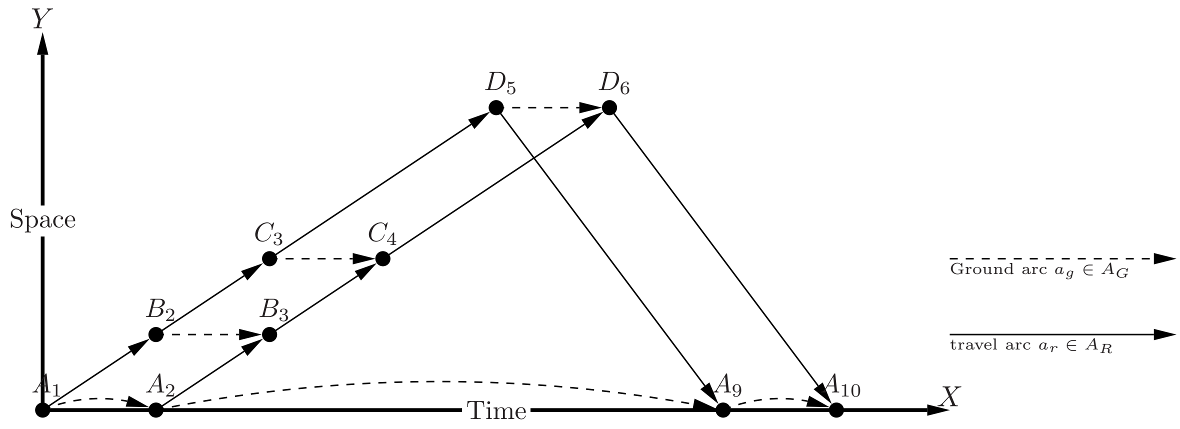


Figure 2: Time-space network with the schedule as time units. Service 1: $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$, vessel $v_{1,1}$ departing A at time $t = 1$ vessel $v_{1,2}$ departing A at time $t = 2$.

geographical port sequence ($A \rightarrow B \rightarrow C$). This can be avoided by tight time windows, but finding a feasible itinerary for a commodity may then become a problem. RMREC is intended for long-term planning preferably spanning 6 months or more. If a very time detailed network is applied on a problem with hundreds of ports with several daily departures, the network will be extremely large, making it impractical to solve. Hence, it seems reasonable to search for a more compact representation.

The shipping network may also be described by representing each port in a given time interval and aggregating all arrival and departure events to/from this port within the specified period. This definition of time gives a less detailed, but more compact graph, where the paths of the individual vessels are aggregated into a path per rotation for the given time period. This will lead to fewer variables as well as fewer constraints in the path flow formulation.

Compared to the graph representing the detailed time-space network, the compact graph is not acyclic. In the compact graph a service can rotate within a single time period (see Figure 3), and the services may contain cycles within them (see Figure 4). If a service takes more than one time period to rotate, then an itinerary is a directed path from port h_t to port h_{t+i}, where i represents the number of time periods it takes the vessel to rotate (see Figure 5). However, if the time unit is set to one month, we believe most services will rotate in a single time period. RMREC is intended for long term planning and a detailed schedule is not needed at this point. Hence, the compact aggregated time definition is preferred.

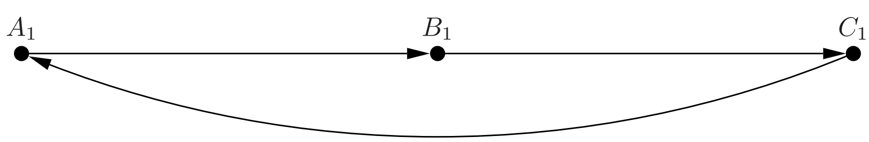


Figure 3: Service $A \rightarrow B \rightarrow C \rightarrow A$ rotates within one time period.

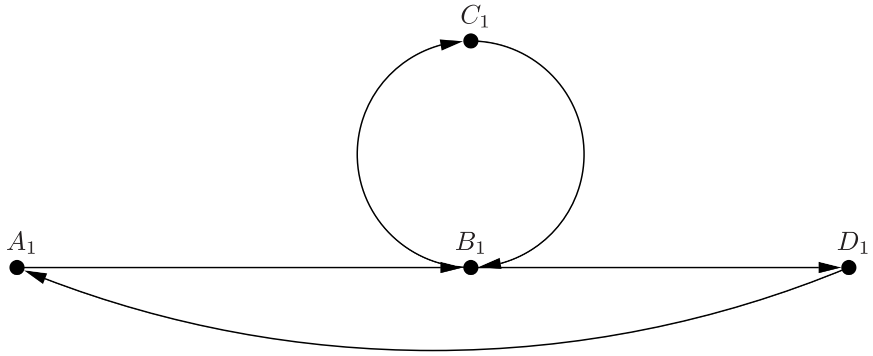


Figure 4: Service $A \to B \to C \to B \to D \to A$ with an internal cycle.

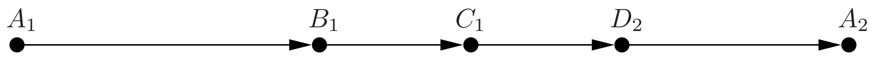


Figure 5: Service $A \to B \to C \to D \to A$ rotates in two time periods.

4 Arc Flow Formulation

Although the integer version of RMREC is $\mathcal{NP}$-hard, the LP relaxation may be solved in polynomial time. The challenge lies in the expected size of a liner shipping network and the number of periods in the planning horizon, which result in very large LPs.

In the arc flow formulation we have a set of commodities K, defined on a graph with nodes N and arcs A. The unit cost of arc (i,j) for commodity k is denoted c_{ij}^k. The non-negative integer variable x_{ij}^k is the flow on arc (i,j) of commodity k. The capacity of arc (i,j) is u_{ij} and d_k is the demand for commodity k. Finally, O_k is the origin of commodity k and D_k is the destination. A commodity in the network is defined as the tuple (O_k, D_k, d_k, s_k) which represents a demand of d_k from node $O_k = p^t$ to node $D_k = q^{t'}$ with a sales price per unit of s_k.

The standard multi-commodity flow model does not consider the supply of empty containers. Inter-balancing constraints are applied to every node to account for availability of empty containers. The constraints require, that the amount of containers arriving at a port must be at least the amount of containers leaving the port for all commodities. Therefore, we get a demand for empty containers depending on the actual allocation of loaded commodities. The inter-balancing constraints also introduce a new set of variables representing leased containers at a node. The cost of leasing is modeled in the objective. Let c_l^i be the cost of leasing a container at port i, while l_i is the leasing variable at port i. If the demand for empty containers are seen as commodities, a set of empty commodities with no revenue and a derived demand is needed. In the arc flow formulation a set of empty commodities must be defined consisting of every possible (O,D) pair with no upper bound on the flow and with no sales price. The set would be huge and the constraints redundant, as they do not impose bounds on the flow. The only purpose of the constraints would be to define origin and destination of empty flows. Since flow conservation is redundant and defining origin and destination of empty flows is already done by the inter-balancing constraints, an empty super commodity without flow conservation constraints may be defined in the arc flow model. The empty super commodity is defined for all arcs in the network, allowing empty flows to start and end anywhere needed in the network. The empty super commodity has no flow conservation constraints and appear in the objective with a cost and in the bundled capacity and inter-balancing constraints. For convenience the commodity set is split into the loaded commodities and the empty super commodity: Let K_F be the set of commodities with a cargo, a sales price and a demand. Let k_e for all arcs in A be the empty super commodity with no cargo, no sales price and a demand implicit derived from K_F. Finally, let $K = K_F \cup K_e$. Load rejection and capacity constraints mean that demand may not be met for all demand pairs. The net flow of commodity k at the origin node reveals the quantity of demand transported in the network and hence the revenue of commodity k. The cost of transport, leasing and the empty super commodity must be subtracted. The arc flow model of RMREC with a profit maximizing objective, an empty super commodity, leasing variables, load rejection and inter-balancing constraints is stated as:

$$\max \sum_{k \in K_F} \sum_{j \in N} s^k (x^k_{O_k j} - x^k_{j O_k}) - \sum_{k \in K} \sum_{(ij) \in A} c^k_{ij} x^k_{ij} - \sum_{i \in N} c^i_l l^i \tag{1}$$

$$\begin{aligned}
\text{s.t.} \sum_{j \in N} x^k_{ij} - \sum_{j \in N} x^k_{ji} &\leq d_k & i = O_K \quad & k \in K_F & (2)\\
\sum_{j \in N} x^k_{ji} - \sum_{j \in N} x^k_{ij} &\leq d_k & i = D_k \quad & k \in K_F & (3)\\
\sum_{j \in N} x^k_{ij} - \sum_{j \in N} x^k_{ji} &= 0 & i \in N \setminus \{O_k, D_K\} \quad & k \in K_F & (4)\\
\sum_{k \in K} x^k_{ij} &\leq u_{ij} & & (i,j) \in A & (5)\\
\sum_{k \in K} \sum_{j \in N} x^k_{ij} - \sum_{k \in K} \sum_{j \in N} x^k_{ji} - l^i &\leq 0 & & i \in N & (6)\\
x^k_{ij} \in \mathbb{Z}_+ & & & k \in K, (i,j) \in A & (7)\\
l_i \in \mathbb{Z}_+ & & & i \in N & (8)
\end{aligned}$$

The objective (1) is to maximize the profit of the demanded flow of all commodities in K on the arcs. Constraints (2)-(4) are the flow conservation constraints which ensure flow of a demand from origin to destination. Furthermore, the flow is bounded by the demanded quantity d_k. Constraints (5) are the bundle constraints ensuring that the flow of all commodities do not exceed the capacity of the arcs. Constraints (6) are the inter-balancing constraints which gives a derived demand for the empty super commodity and leased containers. Constraints (7)-(8) ensure non-negative and integral flow and leasing variables. The formulation is polynomial in the input size as the number of variables is $O(|K||A| + |N|)$ and the number of constraints is $O(|N||K| + |A| + |N|)$. Although the problem size is polynomially bounded, models of large networks have a vast number of variables and a substantial number of constraints, which deteriorates the performance of the simplex algorithm when solving the LP relaxation [19].

It should be noted that when a container is leased, it remains in the network for the remainder of the period. This corresponds to long-term leasing for the first period and short term leasing in the last periods. The cost of a leasing variable should depend on the amount of time periods remaining in T at the node where it is leased. Off-leasing, that is terminating the lease of a container, can be modeled by defining an off-leasing variable and making cost dependent on the net leasing between in- and off-leasing variables at the nodes. This changes the objective function (1) to:

$$\max \sum_{k \in K_F} \sum_{j \in N} s^k (x^k_{O_k j} - x^k_{j O_k}) - \sum_{k \in K} \sum_{(i,j) \in A} c^k_{ij} x^k_{ij} - \sum_{i \in N} c^i_l (l^i_{in} - l^i_{off})$$

and the inter-balancing constraints (6):

$$\sum_{k \in K} \sum_{j \in N} x^k_{ij} - \sum_{k \in K} \sum_{j \in N} x^k_{ji} - l^i_{in} + l^i_{off} \leq 0 \qquad i \in N$$

The above model assumes off-leasing can occur at any port. Off-leasing can be restricted to certain ports by defining off-leasing variables accordingly. When the container is leased the remainder of the optimization period is paid for. When it is off leased the remainder of the optimization period at the off-leasing point is refunded.

Various services are offered by leasing companies, that own half the maritime container fleet worldwide, see [12]. Leasing services vary from one-trip and round-trip leases to short-, medium- and long-term leasing ranging from one month to 42 months, see [20]. For the current RMRECit is chosen to model leasing on a monthly basis, i.e., one time period, which may range from a single month to the entire time period optimized upon. However, the leasing mode may be altered, if different leasing services are explored.

5 Path Flow Formulation

In the following we introduce a path flow model for RMREC. Solving the LP relaxed path flow model compared to the arc flow model has several advantages:

- Although there is a polynomial bound on the number of variables in the arc flow formulation, it is a large polynomial factor. Even though the number of paths in the network *may* be exponential, it depends on how dense the network is. If the network is sparse, like in most liner shipping networks, there will be few path variables for each commodity $k \in K$.
- The network size of an international liner shipping company means that column generation is our only hope to solve the problem in reasonable time. This is true, even for the LP relaxation.
- An LP-solution to the path flow formulation can be transformed to a feasible IP solution to RMREC by rounding, as flow conservation is respected implicitly in the path variables (see figures 6 and 7). This makes it possible to translate the solution directly into itineraries for the demand pairs.

The RMREC given as the arc flow model (1)-(8) has block-angular structure with $|K_F|$ subproblems given by the flow conservation constraints for each full commodity. The commodity subproblems are tied together by the bundle constraints (10), i.e., the arc capacity constraints, and the inter-balancing constraints (12) regarding the supply of empty containers. Using Dantzig-Wolfe decomposition we get a master problem considering paths for all

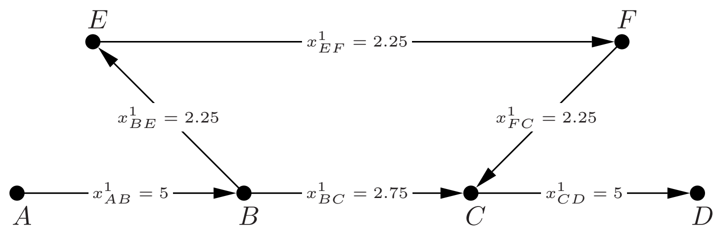


Figure 6: A fractional solution to the arc flow model: Containers may be split at every node.

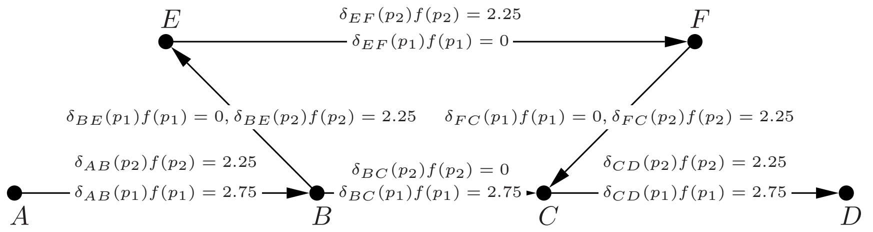


Figure 7: Fractional solution to the path flow model: Containers may only be split at the origin.

commodities, and a subproblem defining the possible paths for each commodity $k \in K$. Due to *the flow decomposition theorem* (Chapter 3.5 in [1]), which states that every non-negative arc flow can be represented as a non-negative path and cycle flow, this can be formulated such that master variables define the flow of a commodity on a path, and the subproblem find valid paths for that commodity.

Let $\overline{p}$ be a path connecting O_k and D_k and P_k be the set of all paths belonging to commodity k. The flow on path $\overline{p}$ is denoted by the variable $f(\overline{p})$. The binary indicator $\delta_{ij}(\overline{p})$ is one if and only if arc (i,j) is on the path $\overline{p}$. Finally, $c^k_{\overline{p}} = \sum_{(i,j)\in A} \delta_{ij}(\overline{p}) c^k_{ij}$ is the cost of path $\overline{p}$ for commodity k. The master problem is:

$$\max \quad \sum_{k\in K_F} \sum_{\overline{p}\in P_k} (s^k - c^k_{\overline{p}}) f(\overline{p}) - \sum_{(i,j)\in A} c^{K_E}_{ij} x^{K_E}_{ij} - \sum_{i\in N} c^i_l l^i \tag{9}$$

$$\text{subject to} \quad \sum_{k\in K_F} \sum_{\overline{p}\in P_k} \delta_{ij}(\overline{p}) f(\overline{p}) + x^{K_E}_{ij} \leq u_{ij} \qquad (i,j) \in A \tag{10}$$

$$\sum_{\overline{p}\in P_k} f(\overline{p}) \leq d_k \qquad k \in K_F \tag{11}$$

$$\sum_{k\in K_F} \sum_{\overline{p}\in P_k} \sum_{j\in N} (\delta_{ij}(\overline{p}) - \delta_{ji}(\overline{p})) f(\overline{p}) + x^{K_E}_{ij} - x^{K_E}_{ji} - l^i \leq 0 \qquad i \in N \tag{12}$$

$$f(\overline{p}) \in \mathbb{Z}_+ \qquad \overline{p} \in P_k k \in K_F \tag{13}$$

$$x^{K_E}_{ij} \in \mathbb{Z}_+ \qquad (i,j) \in A \tag{14}$$

$$l^i \in \mathbb{Z}_+ \qquad i \in N \tag{15}$$

Where the x^k_{ij} variables are replaced by $x^k_{ij} = \sum_{\overline{p}\in P_k} \delta_{ij}(\overline{p}) f(\overline{p})$ according to the flow decomposition theorem for all $k \in K_F$. The subproblems for commodities $k \in K_F$ are given by the polytopes:

$$\mathcal{P}_k = \left\{ \begin{array}{llr} \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} & = 1 & i = O_K \\ \sum_{j \in N} x_{ji} - \sum_{j \in N} x_{ij} & = 1 & i = D_k \\ \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} & = 0 & i \in N \setminus \{O_k, D_K\} \\ x_{ij} \geq 0 & & (i,j) \in A \end{array} \right\}$$

The convexity constraints for the individual subproblems (11) bound the flow between the (O_k, D_k) pair from above (a maximal flow of d_k is possible). The extreme points of $\mathcal{P}_k$ model a path between the (O_k, D_k) pair.

The exponential number of variables is handled by considering only a small subset in a restricted master problem. Paths/columns are then generated on the fly using delayed column generation. The dual variables corresponding to the three constraint sets are:

- w_{ij} for each $(i,j) \in A$ corresponding to the bundle constraints (10)
- σ^k corresponding to the convexity constraints (11) for each commodity $k \in K$
- α^i corresponding to the inter-balancing constraints (12) for each port $i \in N$.

The reduced cost $\hat{c}$ of a path $\overline{p} \in P_k$ for the variable $f(\overline{p})$ in subproblem $k \in K_F$ is given as:

$$\hat{c}_{\overline{p}} = s^k - \sum_{(i,j) \in A} \delta_{ij}(\overline{p}) \left(c^k_{ij} - w_{ij} \right) - \sigma^k - \alpha^{O_k} + \alpha^{D_k}$$

Note, that the dual values α^i cancel each other out for intermediate ports on a path in constraints (12), i.e., only the supply port O_k and the demand port D_k of a path are affected by these constraints.

As the subproblems are only dependent on the arc variables x_{ij} the constant terms given by commodity k may be treated isolated implying:

$$\sum_{(i,j) \in A} \delta_{ij}(\overline{p})(c^k_{ij} + w_{ij}) < s^k - \sigma^k - \alpha^{O_k} + \alpha^{D_k}$$

Minimizing $\sum_{(i,j) \in A}(c_{ij} + w_{ij})x_{ij}$ when finding an extreme point of $\mathcal{P}_k$ will return the path variable with the best reduced cost for the subproblem belonging to k. The subproblem corresponds to an ordinary shortest path problem with positive arc costs as $c_{ij}, w_{ij} \geq 0$:

$$\min \sum_{(ij) \in A} (c_{ij} + w_{ij}) x_{ij} \tag{16}$$

$$\text{s.t.} \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} = 1 \qquad i = O_K \tag{17}$$

$$\sum_{j \in N} x_{ji} - \sum_{j \in N} x_{ij} = 1 \qquad i = D_k \tag{18}$$

$$\sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} = 0 \qquad i \in N \setminus \{O_k, D_K\} \tag{19}$$

$$x_{ij} \in \{0, 1\} \qquad (i,j) \in A \tag{20}$$

6 Computational Results

The experimental results are performed on data based on real life shipping networks. The test instances are created from a snapshot of the Containership Databank [6] from 2005. The set P of ports (and hence the set N of nodes) and the set of arcs A with capacities are created from services found at [6]. Cost and demand functions are generated randomly but such that both profitable and unprofitable products are present, demands are asymmetric in the sense that an area such as Asia should have more export than import (and vice versa for, e.g., Europe and North America), the total demand must exceed the capacity of the network for some areas, and the profit of some products must be able to support the price of leasing containers. Liner shipping operators are chosen so that the instances vary in size from 34 ships to 316 ships. Test instances are named according to the number of ships in the fleet, see Table 1. Note, that instance 293 is larger than instance 316 in terms of the number of ports and unique rotation legs.

Test instance	Ports	Unique rotation legs	Average out degree	Fleet capacity in TEU
34	44	101	2.295	21035
62	60	104	1.733	111004
98	58	122	2.103	348356
136	96	198	2.063	383179
159	117	253	2.162	422796
222	151	326	2.156	633719
293	234	565	2.415	846447
316	185	455	2.459	992479

Table 1: Test instances - instance name denotes the fleet size, e.g., 34 has a fleet size of 34 ships

All tests were performed on an *Intel(R) Xeon(R) CPU 2.66 GHz processor* with 8 GB RAM. We have used the LP solvers from *ILOG's CPLEX 10.2* and the open source *CLP* solver from *COIN-OR*. All tests were performed with the *CPLEX* Barrier solver, *CPLEX* dual simplex solver and the *CLP* dual simplex solver. The best results for the arc flow model were obtained with *CPLEX* barrier solver and the best results for the path flow model were obtained with *CLP* dual simplex. Computational results are stated for the best results for each model for 1, 3, 6, 9, and 12 time periods. The models are solved to LP-optimality. For larger test instances the arc flow model cannot be generated with the available memory. In all tests where it has been possible to generate the arc flow model, the objective value for the two models are identical. This confirms the correctness of the path flow model and the implementation of the column generation algorithm.

In the following we compare performance of the arc flow and the path flow model for test instances with 1 and 3 time periods, where the arc flow model can be generated for most instances. Next we present results for the solution times for large instances using the path flow model in conjunction with delayed column generation. For the path flow model all test instances up to 9 time periods can be solved within an hour. For 12 time periods all tests that may be generated with the available memory are solved in less than 75 minutes. Lastly, we present the integer solutions for a simple rounding heuristic applied to the LP solutions of the path flow model. The IP solutions presented are within a very reasonable distance of the LP upper bound and the gap is sufficiently small to discard the need for a branch-and-price algorithm as well as more sophisticated heuristics.

6.1 Arc flow and path flow compared

The result tables and graphs abbreviate the arc flow model to `A` and the path flow model to `P`. The size of the respective models is stated as $m \times n$. `MEM` indicates that memory was not sufficient for the process to complete. The size of the LP for the arc flow model is calculated for comparison with the size of the master problem of the path flow model. Column `time` denotes the CPU time in seconds to solve the respective model, while `iter` denotes the number of iteration for the column generation algorithm. The arc flow model is solved in one LP iteration.

Table 2 shows the relative performance of the arc flow model and the path flow model for 1 and 3 time periods respectively. The path flow formulation in conjunction with delayed column generation outperforms the arc flow model by a wide margin even for small instances. The size of the LPs for the column generation algorithm is surprisingly small and the column generation algorithm is at least two orders of magnitude faster than the arc flow model for one time period and three orders of magnitude faster for three time periods.

	arc flow model			path flow model			
Test no.	$m \times n$	objective	time	$m \times n$	objective	time	iter
34-01	21860 × 9605	$2.59 \cdot 10^{07}$	4.76	360 × 378	$2.59 \cdot 10^{07}$	0.02	4
62-01	35732 × 20684	$1.21 \cdot 10^{09}$	23.30	506 × 542	$1.21 \cdot 10^{09}$	0.03	4
98-01	60448 × 28832	$5.37 \cdot 10^{09}$	46.40	674 × 889	$5.37 \cdot 10^{09}$	0.06	4
136-01	166020 × 80646	$4.32 \cdot 10^{09}$	180.00	1131 × 1589	$4.32 \cdot 10^{09}$	0.20	6
159-01	264249 × 122401	$4.27 \cdot 10^{09}$	407.00	1413 × 2155	$4.27 \cdot 10^{09}$	0.31	8
222-01	416779 × 193304	$7.92 \cdot 10^{09}$	952.00	1754 × 2285	$7.92 \cdot 10^{09}$	0.34	5
293-01	1237583 × 510835	$1.1 \cdot 10^{10}$	1670.00	2987 × 4008	$1.1 \cdot 10^{10}$	1.03	8
316-01	878790 × 357690	$1.22 \cdot 10^{10}$	1140.00	2570 × 3352	$1.22 \cdot 10^{10}$	0.70	6
34-03	252718 × 85663	$7.9 \cdot 10^{07}$	580	1168 × 2962	$7.9 \cdot 10^{07}$	0.40	5
62-03	501300 × 209232	$4.3 \cdot 10^{09}$	1730	1771 × 2159	$4.3 \cdot 10^{09}$	0.63	10
98-03	844638 × 305330	$1.98 \cdot 10^{10}$	5420	2407 × 3638	$1.98 \cdot 10^{10}$	1.73	15
136-03	2245890 × 823602	$1.55 \cdot 10^{10}$	5690	3930 × 5863	$1.55 \cdot 10^{10}$	2.98	11
159-03	3518550 × 1244586	$2.04 \cdot 10^{10}$	10600	4886 × 8894	$2.04 \cdot 10^{10}$	6.34	18
222-03	5860293 × 2075114	$2.62 \cdot 10^{10}$	57600	6310 × 10039	$2.62 \cdot 10^{10}$	14.10	16
293-03	16257902 × 5260738	-	MEM	10382 × 16620	$3.65 \cdot 10^{10}$	38.20	14
316-03	11965115 × 3829015	-	MEM	9185 × 14097	$5.06 \cdot 10^{10}$	33.50	24

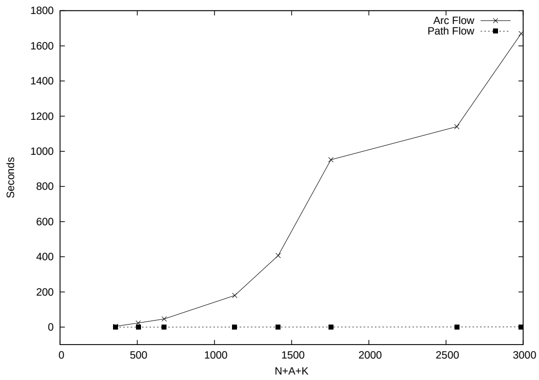
Table 2: Test instances for 1 and 3 time periods

Figures 8(a)–8(b) plot the solution time of the algorithms as a function of the size of the LP model. The unit $|N|+|A|+|K|$ is chosen at the $x-$axis because it is the decisive factors in the size of the LP constraint set for both models. Figure 8(a) shows a fast growth in solution time for the arc flow model (full lines). Using a logarithmic scale in Figure 8(b) it is seen that the growth is exponential. The solution times of the path flow model (dotted lines) grow more moderately. Figures 8(c)–8(d) correspond to figures 8(a)–8(b) when considering three time periods. Again, Figure 8(d) shows an exponential growth of the arc flow model (the two largest instances could not be generated due to memory limitations).

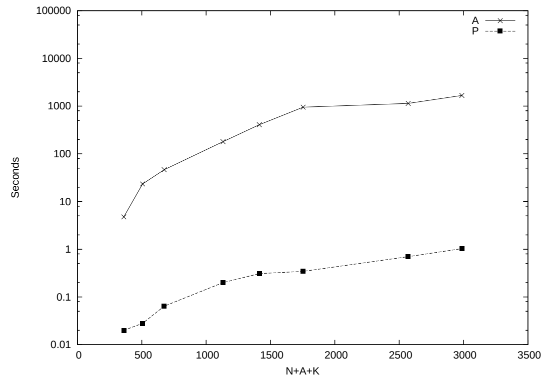
6.2 Solution of large instances

We now consider the solution times for 6, 9 and 12 time periods. The arc flow model cannot be generated for the largest instances and hence is not discussed further in this section.

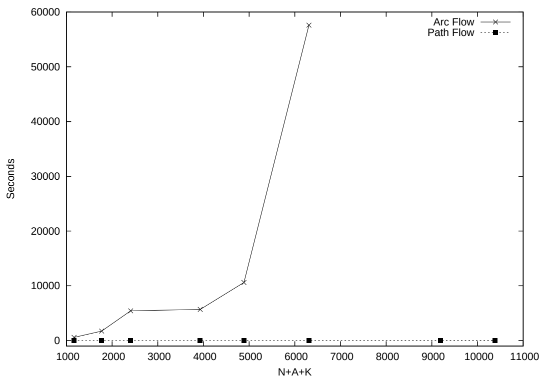
Table 3 shows that test instances with 6 time periods solved with the path flow model complete within 10 minutes. The master problems are small compared to the arc flow model and the number of iterations is reasonable. The sparsity of the networks probably results in few path variables for a commodity, which leads to relatively fast convergence of the delayed



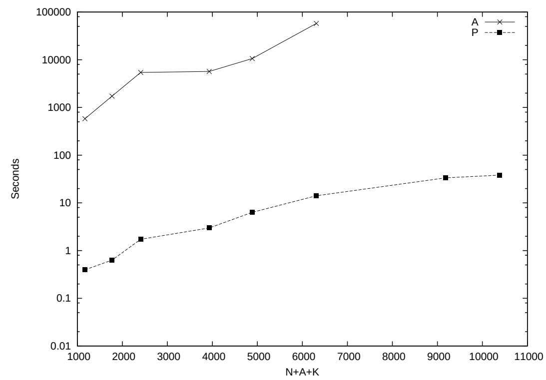
(a) Performance for test instances 1 time period



(b) Performance for test instances 1 time period - logarithmic scale



(c) Performance for test instances 3 time periods

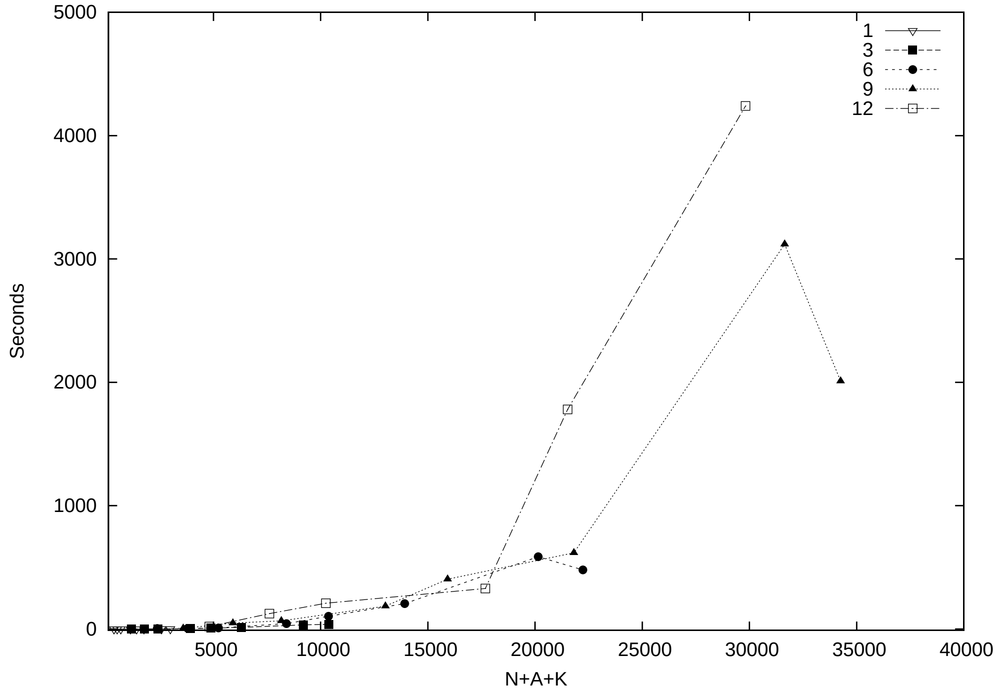


(d) Performance for test instances 3 time periods - logarithmic scale

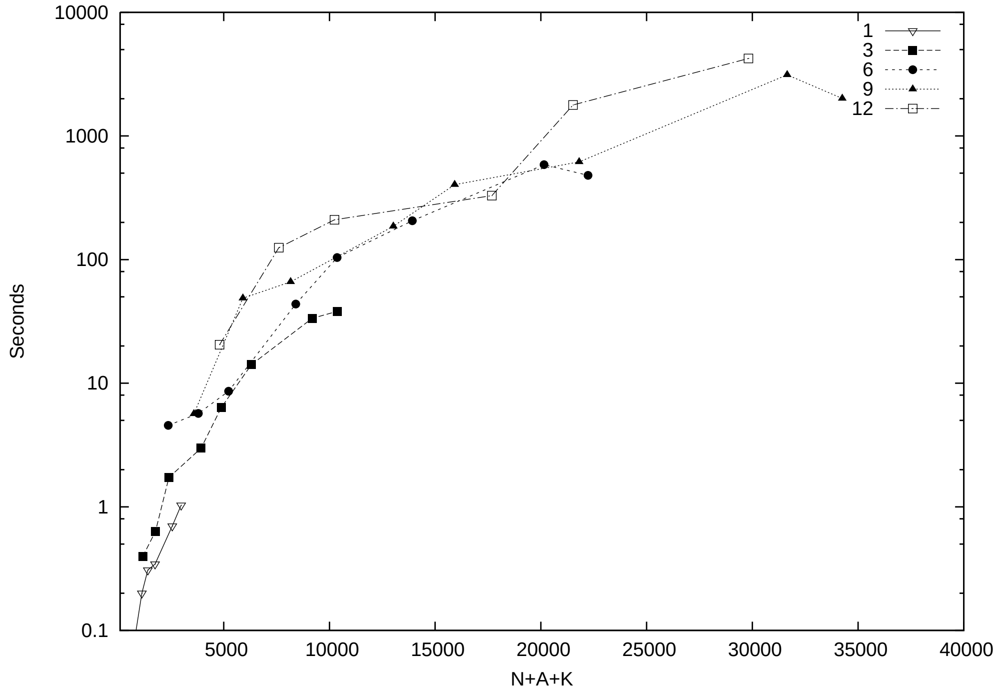
Figure 8: Relative performance of the arc flow and path flow model, 1 and 3 time periods

column generation algorithm. Note, that test 316 is slower than test 293 in spite of a smaller LP. This might be specific for test 316 but may also be due to degeneracy, ϵ rounding or many cache misses. Test instances with 9 time periods may all be completed in less than one hour. All but the two largest tests, 293 and 316 complete within 12 minutes. The number of columns in the two largest test instances is significant. Again we see, that test 316 needs more time and iterations to complete than test 293 although the LP is smaller. The increased solution time seems to be problem specific and it is interesting to note that the network of test 316 is denser than that of test 293. This supports the theory that the sparsity of shipping networks is a key to success for the path flow model and the column generation algorithm. For 9 time periods we see, that the number of iterations varies from 9 to 89. However, execution times still grow steadily and the size of the LPs is reasonable considering the size of the networks. For 12 time periods the LPs have reached a critical size and tests 293 and 316 do not have sufficient memory to complete (the * indicates the size of the LP at the last iteration before the process was aborted). The number of iterations for the remaining test instances is reasonable and the solution times are still within 75 minutes which is equitable for large models representing a shipping network for a whole year.

Figure 9(a)-9(b) show the solution time of the path flow model for 1, 3, 6, 9, and 12 time periods as a function of the size of the LP. It is seen, that the growth in solution times



(a) Performance of model P for 3, 6, 9, and 12 time periods



(b) Performance of model P for 3, 6, 9, and 12 time periods - logarithmic scale

Figure 9: Relative performance of the path flow model - 1,3,6,9 and 12 time periods

	arc flow model			path flow model			
Test no.	$m \times n$	objective	time	$m \times n$	objective	time	iter
34-06	1066630 × 341650	$2.49 \cdot 10^{08}$	3410	2380 × 11070	$2.49 \cdot 10^{08}$	4.56	7
62-06	2325144 × 906684	$9.18 \cdot 10^{09}$	7300	3799 × 5305	$9.18 \cdot 10^{09}$	5.70	19
98-06	3951400 × 1346390	$5.03 \cdot 10^{10}$	35100	5235 × 7989	$5.03 \cdot 10^{10}$	8.61	12
136-06	3552132 × 10282128	-	MEM	8407 × 15251	$3.9 \cdot 10^{10}$	43.70	17
159-06	5310627 × 15903588	-	MEM	10366 × 24020	$4.09 \cdot 10^{10}$	104.00	24
222-06	9339332 × 27929628	-	MEM	13918 × 25635	$7.29 \cdot 10^{10}$	206.00	23
293-06	22762597 × 74147688	-	MEM	22231 × 41419	$7.71 \cdot 10^{10}$	480.00	23
316-06	17078785 × 56225975	-	MEM	20147 × 36229	$1.18 \cdot 10^{11}$	586.00	29
34-09	767917 × 2441692	$4.98 \cdot 10^{08}$	12500	3592 × 11205	$4.98 \cdot 10^{08}$	5.7	9
62-09	2134956 × 5595156	-	MEM	5906 × 12790	$1.78 \cdot 10^{10}$	48.7	89
98-09	3176366 × 9500606	-	MEM	8165 × 14201	$7.11 \cdot 10^{10}$	66.1	21
136-09	8302998 × 24496164	-	MEM	13020 × 24212	$7.17 \cdot 10^{10}$	186.0	41
159-09	12274875 × 37445355	-	MEM	15919 × 51822	$7.36 \cdot 10^{10}$	404.0	28
222-09	22172150 × 67565663	-	MEM	21812 × 42444	$1.26 \cdot 10^{11}$	618.0	17
293-09	52866028 × 175158501	-	MEM	34252 × 69529	$1.37 \cdot 10^{11}$	2010.0	27
316-09	17078785 × 56222321	-	MEM	31643 × 63309	$2.18 \cdot 10^{11}$	3120.0	45
34-12	1364464 × 4377904	-	MEM	4804 × 20557	$3.69 \cdot 10^{08}$	20	5
62-12	3587508 × 9502560	-	MEM	7607 × 36750	$1.4 \cdot 10^{10}$	125	27
98-12	5183822 × 15650086	-	MEM	10242 × 39919	$8.62 \cdot 10^{10}$	210	18
136-12	15092328 × 44953488	-	MEM	17681 × 36455	$1.24 \cdot 10^{11}$	329	30
159-12	22176291 × 68270220	-	MEM	21518 × 87218	$9.42 \cdot 10^{10}$	1780	48
222-12	40655981 × 125020921	-	MEM	29818 × 67841	$1.72 \cdot 10^{11}$	4240	32
293-12	95531887 × 319187701	-	MEM	46302 × 114088*	-	MEM	25*
316-12	74721600 × 252239000	-	MEM	43369 × 90759*	-	MEM	40*

Table 3: Test instances for 6, 9 and 12 time periods. * indicates the size of the LP and the last iteration of the process when aborting due to insufficient memory

is relatively steady for 3 time periods. Figure 9(a) shows an exponential tendency for the graphs of 9 and 12 time periods, where the LPs have reached a critical size. The trend is even more explicit in figure 9(b) where the graphs are plotted on a logarithmic scale. The trend is particular for the larger test instances, which have denser networks and hence, more path variables per commodity. The exponential tendency is very clear in the graph of 12 time periods although the two largest tests did not complete.

The delayed column generation method shows good convergence for the generated test instances, and methods to reduce the master problem constraint set has not been required. We are able to solve instances of large shipping networks spanning 9 months in less than one hour. For 12 months the two largest instances cannot be generated with the available memory, but the remaining tests which have a significant size are solved within 75 minutes. The convergence is suspected to be correlated to the sparsity of the networks. The results of the tests show that the column generation technique is very effective for solving RMREC for sparse networks. The path flow model and column generation algorithm outperforms solving the arc flow model by a wide margin. It appears that the number of path variables for real life liners is very modest and therefore the number of variables in the restricted master problem is relatively small. Services, capacities and ports are based on the real world, the cost structure is randomly generated. This indicates that RMREC will perform well on real life problems, but the real partition of surplus/deficit zones for empty containers and the commodity set of a real life instance might be harder to solve than the generated instances presented in this paper.

6.3 Quality of integer solutions and speed of rounding heuristic

In this section we present the integer solutions obtained by a simple rounding heuristic of the LP solution provided by the path flow model. We show that the integer solutions have a very small gap to the LP upper bound. An integer solution to RMREC is obtained by rounding down all fractional variables. At nodes with violated inter-balancing constraints we supply empty containers through the leasing variable to maintain feasibility. Table 4 shows the integer solutions obtained by rounding. For each test instance we report the fraction of fractional basis variables (`Frac/basis`), the percentage of fractional basis variables (`Frac %`), the fractional moves as a percentage of total moves (`Rounded %`), the gap (`obj gap`) and gap percentage (`gap %`) between the LP and IP solution and the CPU time in seconds (`time`).

Test	Frac/basis	Frac %	Rounded %	obj gap	gap %	time
34-01	0/183	0.0	0.0	0	0.0	0.01
62-01	0/214	0.0	0.0	0	0.0	0.01
98-01	0/331	0.0	0.0	0	0.0	0.01
136-01	10/538	1.9	$2.7 \cdot 10^{-4}$	13800	$3.2 \cdot 10^{-6}$	0.01
159-01	0/464	0.0	0.0	0	0.0	0.02
222-01	0/836	0.0	0.0	0	0.0	0.02
293-01	23/1430	1.6	$2.2 \cdot 10^{-4}$	2940	$2.7 \cdot 10^{-6}$	0.06
316-01	28/1210	2.3	$2.5 \cdot 10^{-4}$	34100	$2.8 \cdot 10^{-6}$	0.05
34-03	0/588	0.0	0.0	0	0.0	0.02
62-03	46/723	6.4	$1.4 \cdot 10^{-3}$	81900	$1.9 \cdot 10^{-6}$	0.03
98-03	116/1020	11.4	$1.06 \cdot 10^{-3}$	271000	$1.37 \cdot 10^{-5}$	0.03
136-03	0/1680	0.0	0.0	0	0.0	0.08
159-03	16/1500	1.1	$1.32 \cdot 10^{-4}$	62000	$3.05 \cdot 10^{-6}$	0.15
222-03	324/2600	12.5	$1.81 \cdot 10^{-3}$	767000	$2.93 \cdot 10^{-5}$	0.20
293-03	375/4450	8.4	$1.41 \cdot 10^{-3}$	902000	$2.47 \cdot 10^{-5}$	0.48
316-03	161/3620	4.4	$5.03 \cdot 10^{-4}$	518000	$1.02 \cdot 10^{-5}$	0.36
34-06	0/1060	0.0	0.0	0	0.0	0.04
62-06	102/1330	7.7	$1.7 \cdot 10^{-3}$	457000	$4.98 \cdot 10^{-5}$	0.10
98-06	290/1960	14.8	$1.37 \cdot 10^{-3}$	1340000	$2.66 \cdot 10^{-5}$	0.14
136-06	453/3290	13.8	$2.17 \cdot 10^{-3}$	1610000	$4.13 \cdot 10^{-5}$	0.36
159-06	401/3130	12.8	$1.88 \cdot 10^{-3}$	1620000	$3.96 \cdot 10^{-5}$	0.56
222-06	969/5060	19.1	$2.40 \cdot 10^{-3}$	3850000	$5.28 \cdot 10^{-5}$	0.86
293-06	1114/9460	11.8	$2.14 \cdot 10^{-3}$	3650000	$4.74 \cdot 10^{-5}$	2.01
316-06	1281/7660	16.7	$2.04 \cdot 10^{-3}$	5220000	$4.44 \cdot 10^{-5}$	1.55
34-09	0/1560	0.0	0.0	0	0.0	0.10
62-09	259/1990	13.0	$3.22 \cdot 10^{-3}$	1810000	$1.02 \cdot 10^{-4}$	0.21
98-09	673/3080	21.8	$2.18 \cdot 10^{-3}$	3770000	$5.30 \cdot 10^{-5}$	0.29
136-09	744/4800	15.5	$2.44 \cdot 10^{-3}$	4260000	$5.94 \cdot 10^{-5}$	0.82
159-09	637/4910	13.0	$1.89 \cdot 10^{-3}$	2960000	$4.03 \cdot 10^{-5}$	1.27
222-09	1568/7610	20.6	$3.05 \cdot 10^{-3}$	8070000	$6.41 \cdot 10^{-5}$	1.99
293-09	1838/13900	13.2	$2.39 \cdot 10^{-3}$	7540000	$5.49 \cdot 10^{-5}$	4.93
316-09	2357/11500	20.5	$2.61 \cdot 10^{-3}$	12900000	$5.92 \cdot 10^{-5}$	3.70
34-12	0/2030	0.0	0.0	0	0.0	0.15
62-12	50/2470	2.0	$4.53 \cdot 10^{-4}$	366000	$2.61 \cdot 10^{-5}$	0.38
98-12	532/3300	16.1	$1.25 \cdot 10^{-3}$	2870000	$3.32 \cdot 10^{-5}$	0.49
136-12	630/6280	10.0	$1.63 \cdot 10^{-3}$	5470000	$4.41 \cdot 10^{-5}$	1.46
159-12	984/6410	15.4	$2.18 \cdot 10^{-3}$	4870000	$5.17 \cdot 10^{-5}$	2.30
222-12	2163/10600	20.3	$3.05 \cdot 10^{-3}$	13500000	$7.89 \cdot 10^{-5}$	3.59
293-12	-	-	-	-	-	-
316-12	-	-	-	-	-	-

Table 4: Rounded integer solutions - 1-12 time periods

Table 4 shows that 10 out of 38 ($\approx 26\%$) of the LP solutions are already integer. Test instance 34 is integer throughout. The remaining integer LP solutions are found in time periods one and three. 20 out of 38 ($\approx 53\%$) LP solutions have less than 10% fractional basis variables. The highest percentage among the remaining 18 LP solutions is 21.8%. The most fractional solutions seems to be test instances with 9 time periods. Despite having more than 20% fractional basis variables the amount of flow rounded is never more than 0.3% of the total flow and the gap percentage in terms of the objective value is never higher than 0.01%. This confirms that the rounded integer solution is a good solution in terms of the gap to the LP upper bound. This is probably due to generally large flows on path variables making the rounding insignificant. Execution times are mostly less than one second but on larger instances execution times rise to at most 5 seconds. The optimal integer solution might be slightly better, but given a gap percentage less than 10^{-4} the computation time needed for a branch-and-price algorithm does not seem justified since the gap is smaller than the data uncertainty.

7 Concluding Remarks

We have presented a mathematical model for the container revenue management problem considering empty repositioning and solved it to near-optimality using delayed column generation and rounding. To the best of our knowledge a revenue management model considering empty repositioning has not been presented in the literature before. The mathematical model is surprisingly simple. The inter-balancing constraints, which ensure repositioning of empty containers, results in an augmented multi-commodity flow problem. It appears that these constraints do not complicate the model to an extent where the solution time is significantly affected. Furthermore, test results show that the inter-balancing constraints ensure transportation of low profitable products before repositioning empty containers to deficit ports, see Løfstedt [14] for a detailed discussion. This demonstrates the importance of considering empty container repositioning in a booking model for liner shipping. Also, the size of the instances created and solved in this paper are significantly larger than previously reported in the reviewed literature.

Solving the LP relaxed path flow model with delayed column generation turned out to be very successful compared to solving the arc flow model with the *CPLEX* barrier solver. The column generation algorithm is at least two orders of magnitude faster for one time period and three orders of magnitude faster for three time periods. The column generation algorithm is able to solve all instances for 6 time periods in 586 seconds. For 9 time periods test instance 316 containing 1665 nodes (185 ports in 9 periods), 5575 arcs and 24403 commodities is solved in 3120 seconds. For 12 periods the two largest test instances cannot be solved within the space limit. The largest instance completed contains 1812 nodes (151 ports in 12 periods), 5573 arcs and 22433 commodities and is solved in 4240 seconds. A rounding heuristic is applied to the LP solutions of the path flow model with great success. The heuristic finds a solution in less than 5 seconds. All integer solutions have a gap to the LP upper bound of at most 0.01% which is well below the data uncertainty.

An area of future work is to incorporate booking and empty repositioning into routing/scheduling decisions of the vessel fleet as the overall cost of running a liner shipping company is the fixed cost of committing to a schedule. Also, it would e very interesting to incorporate substitution of containers into the path flow model of RMREC. Some modifications may result in hard pricing problems (complexity and computationally wise), but we believe that even a complex pricing problem may be solved in reasonable time as the graph may be split into subgraphs according to time periods, and because paths are generally very short. Furthermore, pricing problems may be solved in parallel to decrease solution times. Reinhardt

[15] solve a multi-objective shortest path problem for liner shipping with non-additive costs. These techniques could be relevant for RMREC because it is likely that several criteria needs to be taken into account when defining the attractiveness of a path and various strategic goals may have a non-additive cost structure.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows - Theory, Algorithms and Applications*. Prentice Hall, 1993. ISBN 0-13-617549-X.

[2] P. Bartodziej and U. Derigs. *Experimental and Efficient Algorithms*. Springer Berlin/Heidelberg, 2004. ISBN 978-3-540-22067-1.

[3] N. Bélanger, G. Desaulniers, F. Soumis, and J. Desrosiers. Periodic airline fleet assignment with time windows, spacing constraints, and time dependent revenues. *European Journal of Operational Research*, 175(3):1754–1766, 2006. doi: 10.1016/j.ejor.2004.04.051.

[4] R.K. Cheung and C.-Y. Chen. A two-stage stochastic network model and solution methods for the dynamic empty container allocation problem. *Transportation Science*, 32(2): 142–162, 1998. doi: 10.1287/trsc.32.2.142.

[5] M. Christiansen, D. Ronen, B. Nygreen, and K. Fagerholt. Maritime transportation. *Handbooks in Operations Research and Management Sciences*, 14:189–284, 2007.

[6] Containership Databank. Containership databank. URL `http://www.mdst.co.uk`.

[7] T.G. Crainic, P.J. Dejax, and L. Delorme. Models for multimode multicommodity location problems with interdepot balancing requirements. *Annals of Operational Research*, 18(1):277–302, 1989. doi: 10.1007/BF02097809.

[8] T.G. Crainic, M. Gendreau, and P.J. Dejax. Dynamic and stochastic models for the allocation of empty containers. *Operations Research*, 41(1):102–126, 1993. doi: 10.1287/opre.41.1.102.

[9] P.J. Dejax and T.G. Crainic. Survey paper - a review of empty flows and fleet management models in freight transportation. *Transportation Science*, 21(4):227–247, 1987. doi: 10.1287/trsc.21.4.227.

[10] G. Desaulniers, J. Desroisers, Y. Dumas, S. Marc, B. Rioux, M. M. Solomon, and F. Soumis. Crew pairing at Air France. *European Journal of Operations Research*, 97:245 – 259, 1997. doi: 10.1016/S0377-2217(96)00195-6.

[11] C.A. Hane, C. Barnhart, E.L. Johnson, R.E. Marsten, G.L. Nemhauser, and G. Sigismondi. The fleet assignment problem solving a large scale integer program. *Mathematical Programming*, 70:211–232, 1995. doi: 10.1007/BF01585938.

[12] IICL. Institute of international container lessors. URL `http://www.iicl.org`.

[13] J. Li, K. Liu, S.C.H. Leung, and K.K Lai. Empty container management in a port with long-run average criterion. *Mathematical and Computer Modelling*, 40:85–100, 2004. doi: 10.1016/j.mcm.2003.12.005.

[14] B. Løfstedt. Crmer - container revenue management with empty repositioning. Master's thesis, Department of Computer Science, Copenhagen University, 2007.

[15] L.B. Reinhardt. Multi objective shortest path for cargo transportation. Master's thesis, Department of Computer Science, Copenhagen University, 2005.

[16] D. Ronen, K. Fagerholt, and M. Christiansen. Ship routing and scheduling: Status and perspectives. *Transportation Science*, 38(1):1–18, 2004. doi: 10.1287/trsc.1030.0036.

[17] W.S. Shen and C.M. Khoong. A DSS for empty container distribution planning. *Decision Support Systems*, 15(1):75–82, 1995. doi: 10.1016/0167-9236(94)00037-S.

[18] K. Shintani, A. Imai, E. Nishimura, and S. Papadimitriou. The container shipping network design problem with empty container repositioning. *Transportation Research Part E: Logistics and Transportation Review*, 43(1):39–59, 2007. doi: 10.1016/j.tre.2005.05.003.

[19] H.A. Taha. *Operations Research*. Prentice Hall, 7 edition, 2003. ISBN 0-13-048808-9.

[20] TAL. Tal international container corporation. URL `http://www.talinternational.com`.