Extended abstract for
"Approximate column generation for some ship scheduling problems"

Geir Brønmo & Bjørn Nygreen
Section of Managerial Economics and Operations Research
Norwegian University of Science and Technology
7491 Trondheim
Norway

Jens Lysgaard
Department of Business Studies
Aarhus School of Business
Fuglesangs Allé 4, 8210 Aarhus V
Denmark

Abstract

We present a Dantzig-Wolfe procedure for the ship scheduling problem with flexible cargo sizes. This problem is similar to the well-known pickup and delivery problem with time windows, but the cargo sizes are defined to be within intervals instead of having fixed values. We show that the introduction of flexible cargo sizes to the column generation framework is not straightforward, and we handle the flexible cargo sizes heuristically when solving the subproblems. This leads to convergence issues in the branch-and-price search tree, and an optimal solution cannot be guaranteed. Hence we have introduced a method that generates an upper bound on the optimal maximization objective. We have compared our method with an a priori column generation approach, and our computational experiments on real world cases show that the Dantzig-Wolfe approach is faster than the a priori generation of columns, and we are able to deal with larger or more loosely constrained instances. By using the techniques introduced here, a more extensive set of real world cases can be solved either to optimality or within a small deviation from optimality.

Keywords: Transportation, integer programming, dynamic programming.

1 INTRODUCTION

Ship scheduling is generally concerned with determining sequences of ports to be visited by ships, taking temporal aspects into account. Many specific types of problems fall into the category of ship scheduling. Due to their inherent complexities, an effective solution approach must invariably be tailored to the particular problem at hand. This involves a decision on an appropriate mathematical model for which we hope to find an optimal or near-optimal solution.

At the level of abstraction corresponding to a mathematical model, ship scheduling may be viewed as being rather similar to vehicle routing and scheduling, representing transportation by sea and road, respectively. The considerable effort in the literature that has been put in attacking vehicle routing and scheduling problems – see e.g. Toth & Vigo (2002) – suggests that a large number of mathematical models and associated solution procedures would be available for being adopted to ship scheduling. However, from a practical perspective, ship scheduling contains certain problem variations which are very rarely encountered in road transportation. Christiansen et al. (2004) recently conducted a thorough survey of ship scheduling research.

The complication that we pay particular attention to here is that the decision about which cargos to carry on a particular ship also involves a decision about the size of each cargo. This type of problem was introduced by Brønmo et al. (2006), in which the problem was solved as a Set Partitioning Problem (SPP). The approach in their paper is based on a priori generation of all feasible columns, which is reasonable in cases of small or tightly restricted instances. They also showed that problems with flexible cargo sizes usually have solutions with better economy. A priori column generation has been the most common optimization method used within ship scheduling research; see for instance Christiansen and Fagerholt (2002), Sherali et al. (1999) and Bausch et al. (1998). On larger or loosely restricted instances where the a priori column generation fails, a dynamic column generation scheme might work better.

Here we consider the possibilities for dynamic column generation in ship scheduling with flexible cargo sizes. Our motivation for addressing this issue is that the a priori generation of all feasible columns is intractable when attacking some of the actual problems that we have encountered in practice. We show that the introduction of flexible cargoes to the column generation framework is not straightforward, and therefore we handle the flexible cargo sizes heuristically when solving the subproblem. We have conducted computational experiments on a variety of practical instances in order to compare the performance of our column generation approach to that of the a priori generation of columns in Brønmo et al. (2006).

The rest of this extended abstract is organized as follows: A problem description and mathematical model is presented in Section 2, while the column generation approach is described in Section 3. In Section 4 a computational study is presented and finally, concluding remarks are given in Section 5.

2 PROBLEM DESCRIPTION

The studied short-term ship scheduling problem for the tramp market is closely related to the multi-vehicle pickup and delivery problem with time windows. In the pickup and delivery problem a set of vehicles must satisfy a set of transportation requests having an origin and a destination with time windows and a fixed quantity. In the problem studied here, each transportation request or cargo is given a profit rate, and not all transportation requests must be fulfilled. Some of the cargoes are contracted, and must be carried, while the rest of the cargoes are negotiated on the spot marked. Before an agreement is made with the cargo owner, the spot cargoes can be treated as optional.

Each transportation request has a flexible cargo size, i.e. the cargo size is given by an interval. The objective is to maximize the profit. Brønmo et al. (2006) give a mathematical model of this problem. In this extended abstract we only repeat the part of their model that explicitly deals with the flexible cargos.

2.1 Mathematical model

There are N cargoes in the problem. The pickup nodes are numbered form 1 to N and the delivery nodes are numbered from $N+1$ to $2N$. The index sets used in this formulation are: $\mathcal{V}$ for ships, $\mathcal{A}_v$ for arcs that ship v can sail and $\mathcal{N}_{Pv}$ for the pickup nodes that can be served by ship v. The data are: R_i = the profit for one unit of cargo i, T_{Qiv} = the time for load or unload one unit of cargo i to or from ship v, T_{Sijv} = time to travel from node i to node j with vessel v, Q_{MNi} and Q_{MXi} are the minimal and maximal size of cargo i.

Let $x_{ijv}, v \in \mathcal{V}, (i,j) \in \mathcal{A}_v$ be the binary flow variable representing the decision whether or not to sail directly from port *i* to port *j* with ship *v*. Further, let $t_{iv}, v \in \mathcal{V}, i \in \mathcal{N}_v$ be the continuous time variable denoting the time for start of service in port *i* for ship *v*. $l_{iv}, v \in \mathcal{V}, i \in \mathcal{N}_v$ is a continuous variable representing the total load onboard ship *v* at the departure from port *i*, while $q_{iv}, v \in \mathcal{V}, i \in \mathcal{N}_{Pv}$ is a continuous variable denoting the quantity of cargo *i* loaded on ship *v*.

The flexible cargo size part of our short-term tramp ship scheduling problem can be formulated as follows:

$$\max \left[\sum_{v \in \mathcal{V}} \sum_{i \in \mathcal{N}_{Pv}} R_i q_{iv} \right] \text{ - Sailing costs,} \tag{2.1}$$

$$x_{ijv}(t_{iv} + T_{Qiv} q_{iv} + T_{Sijv} - t_{jv}) \leq 0, \quad \forall v \in \mathcal{V}, (i,j) \in \mathcal{A}_v, \tag{2.2}$$

$$t_{iv} + T_{Qi} q_{iv} + T_{Si,N+i,v} - t_{N+i,v} \leq 0, \quad \forall v \in \mathcal{V}, i \in \mathcal{N}_{Pv}, \tag{2.3}$$

$$x_{ijv}(l_{iv} + q_{jv} - l_{jv}) = 0, \quad \forall v \in \mathcal{V}, (i,j) \in \mathcal{A}_v | j \in \mathcal{N}_{Pv}, \tag{2.4}$$

$$x_{i,N+j,v}(l_{iv} - q_{jv} - l_{N+j,v}) = 0, \quad \forall v \in \mathcal{V}, (i,N+j) \in \mathcal{A}_v | j \in \mathcal{N}_{Pv}, \tag{2.5}$$

$$\sum_{j \in \mathcal{N}_v} Q_{MNi} x_{ijv} \leq q_{iv} \leq \sum_{j \in \mathcal{N}_v} Q_{MXi} x_{ijv}, \quad \forall v \in \mathcal{V}, i \in \mathcal{N}_{Pv}, \tag{2.6}$$

Equation (2.1) defines the profit maximizing objective. Constraints (2.2) and (2.3) make sure that the correct arrival times are calculated due to the variable amount of the cargoes. Constraints (2.2) have to be adjusted slightly for the *N+i* nodes, since variable q_{iv} is defined for pickup nodes (cargoes), only. Constraints (2.4) and (2.5) make sure that the load quantities are calculated so that the load quantity intervals (2.6) can be controlled.

3 COLUMN GENERATION

We use a column generation approach to solve the model mentioned in Section 2. When we use Dantzig-Wolfe decomposition on the total model given in Brønmo et al. (2006), we get a subproblem for each ship and a masterproblem to ensure that each cargo at most is lifted once. After introduction of explicit slack variables for cargoes not taken, we get a set covering master problem, as in Desrosiers et al. (1995).

The subproblem is solved by solving one problem for each ship by use of the current master duals. Each of these problems is solved by dynamic programming as a resource constraint shortest path problem.

To find feasible columns, we can put different effort into the solution of the shortest path problems:

1. Use a fixed size for each cargo
2. Use several discrete sizes for each cargo
3. Use continuous size for each cargo

We have only tried the two first because the third would result in a too large state space in the dynamic programming. The reason for this is that optimal size for each cargo can only be calculated for parts of routes between ports where the ship is empty. With the possibility of many cargo combinations in such part routes, the needed state space (label combinations) in the DP might be too large.

For a fixed route, the optimal cargo sizes can be found by solving an LP with bounded arrival time and cargo size variables and ship capacity constraints.

3.1 Solving the subproblem

We have used a slightly enhanced version of the shortest path problem for pickup and delivery problems with time windows given by Dumas et al. (1991), for solving our shortest path problems.

When solving each subproblem we first find *p* (= 60) shortest paths in stead of just one. Then we solve the cargo size LP as discussed in Brønmo et al. (2006) for each of the *p* paths. All the quantity optimized paths with positive (promising) reduced costs are transferred to the masterproblem as new columns.

One result of this approach is that master duals are calculated with cargo sizes different from those used in the "pure" shortest path problem. This means that we might not be able to reproduce the basic master columns when we solve the subproblem. This also means that there might be a column that is promising after size optimization, even if it is not among the *p* shortest paths before size optimisation.

For this reason, we can not guarantee that our approach converges to the optimal solution.

3.2 Solving the masterproblem

The linear relaxation of the masterproblem is solved by a library call to XpressMP. The branch and price part of the master problem is solved by a pure depth first tree search with continued column generation in each tree node.

In a given node in the search tree with a fractional solution, we use the following procedure to select the type of branching to use, and what entity to branch on:

IF there are cycles in the solution

Choose time window branching.

Select one of the cycles. The cycle is characterized by at least two arrivals at the loading node of cargo *i*. Calculate the average of the earliest and latest arrivals at the loading node in the given sequence and choose it as the branching time.

ELSE

Choose cargo-ship branching.

Calculate the *solution weight* of each cargo-ship combination. The solution weight of a cargo-ship combination is defined as the sum of the values of the basis variables that contain this combination. Choose the cargo-ship combination for which the solution weight is closest to 0.5.

END

3.3 Upper bounds

Since the result from our column generation approach is not necessarily optimal, we calculate an upper bound to the problem. We introduce an *upper bound discretization* setup where the income of each of the fixed cargo sizes in the discretization set is increased, while the resource consumption is the same as in the original problem. The income is set so that any original problem solution corresponds to an upper bound discretization solution with equal or higher profit. Now, we find the optimal solution to the upper bound discretization setup by using the column generation scheme without quantity optimization, and we have computed an upper bound to the original problem.

4 COMPUTATIONAL TESTS

We have tested our solution approach on ten test cases based on real data from two different tramp shipping companies. The motivation was to evaluate the approach in terms of response time and solution quality.

Five cases are collected from a shipping company, which operates in northern Europe and transports dry bulk commodities such as rock, iron ore and cement. 4 of these cases are based on scheduling problems for a fleet of four ships. The time windows are large for both loading and unloading of the cargoes. In the last of these cases the fleet is increased to six ships and the time windows are made somewhat narrower.

In addition we have used 5 other cases taken from a chemical commodity shipping company, which operates between Europe and the Caribbean. In these cases the number of ships varies from 3 to 7. The ships come from the same fleet. In 3 of the cases some ships are engaged in fulfilling other obligations, while in 2 of the cases the whole fleet is available. In most of the cases the time windows are typically narrow for the loading and wide for the unloading of the cargoes

In the computational study we have compared our method with the a priori column generation approach of Brønmo et al. (2006) for six of their real data cases. The results show that column generation is more efficient than a priori column generation also in the case of flexible cargo sizes. Our column generation approach found the optimal solution to all the six cases. In order to show that a more extensive set of practical instances can be solved, we have also used four cases that were solved by our method but could not be solved by the a priori column generation algorithm. In these four cases the column generation solution was 0.7% – 3.0% from the computed upper bound.

5 CONCLUDING REMARKS

We have presented a column generation procedure for the tramp ship scheduling problem with flexible cargo sizes. A modified version of the dynamic programming method of Dumas et al. (1991) is used to find the *p* best solutions of the subproblem with discretized load quantities. For all the *p* best subproblem solutions we optimize load quantities by using the LP model and solution algorithm introduced by Brønmo et al. (2006). If any of the columns have a positive reduced cost, it is transferred to the restricted master problem. The column generation scheme stops when no more positive reduce cost columns can be found. Finally we use a branch-and-price approach to find an integer solution.

Our results indicate that in the case of relatively small or tightly constrained instances, the a priori column generation approach should be chosen since optimality is guaranteed. In larger or more loosely constrained cases the column generation approach gives a very good compromise between solution time and quality.

ACKNOWLEDGEMENTS

This work was carried out with financial support from the TOP project on Transport Optimization through the Research Council of Norway. The authors are grateful to Aarhus School of Business for hosting Geir Brønmo during this work. Geir Brønmo's stay at Aarhus School of Business has been supported by a mobility scholarship from the Nordic Research Board (NordForsk).

REFERENCES

Bausch DO, Brown G G, Ronen D. Scheduling short-term marine transport of bulk products. Maritime Policy and Management 1998; 25(4); 335-348.

Brønmo G, Christiansen M, Nygreen B. Ship Scheduling with Flexible Cargo Sizes. Journal of the Operational Research Society 2006; (forthcoming).

Christiansen M, Fagerholt K. Robust ship scheduling with multiple time windows. Naval Research Logistics 2002; 49(6); 611-625.

Christiansen M, Fagerholt K, Ronen D. Ship Routing and Scheduling: Status and Perspectives. Transportation Science 2004; 39(1); 1-18.

Desrosiers JY, Dumas Y, Solomon MM, Suomis F 1995. Time Constrained Routing and Scheduling. In: Ball MO, Magnanti TL, Monna CL and Nemhauser GL (Eds), Handbooks in Operations Research and Management Science: Network Routing. North Holland: Amsterdam; 1995. p. 35-139.

Dumas Y, Desrosiers J, Suomis F. The pickup and delivery problem with time windows. European Journal of Operational Research 1991;54;7-22.

Sherali HD, Al-Yakoob SM, Hassan MM. Fleet management models and algorithms for an oil tanker routing and scheduling problem. IIE Transactions 1999; 31; 395-406.

Toth P, Vigo D (eds). Monographs on Discrete Mathematics and Applications: The Vehicle Routing Problem S.I.A.M.: Philadelphia, PA; 2002.