An adaptive large neighborhood search heuristic for fleet deployment with voyage separation requirements

Rikard Bakkehaug, Jørgen G. Rakke, Kjetil Fagerholt
Department of Marine Technology, Engineering Science and Technology, Norwegian University of Science and Technology, Trondheim, Norway
Email: rikard.bakkehaug@ntnu.no

Gilbert Laporte
Department of Management Sciences, HEC Montréal, Montréal, Canada

1 Introduction

Within maritime shipping efficient use of vessels is recognized as one of the most important factor for company profit. Operating a ship is costly and extra sailings with the sole purpose of repositioning the vessel should be kept to a minimum as these do not generate income.

The shipping industry is a large industry, and it can be divided into different modes of operations (tramp, industrial and liner (Lawrence, 1972)). To deploy the vessels in a best possible way there are different challenges in the three modes. In this paper we present a ALNS heuristic for a real ship routing and scheduling problem faced by Saga Forest Carriers, a Norwegian shipping company transporting forest products and break bulk cargoes around the world. The company operates in between liner and tramp shipping. In liner shipping the shipping company publishes schedules, almost like a bus company. In tramp shipping the company operates more as a taxi. Saga Forest Carriers operates on several trade routes where they publish a schedule of when to sail. In addition they take on spot voyages to which they charter out vessels to perform. This planning problem is a ship routing and scheduling problem, and is within the category of tactical decisions (Stopford, 2009). For liner container shipping Powell and Perkins (1997), Gelareh and Pisinger (2011), Blander Reinhardt and Pisinger (2012), Wang and Meng (2012a), Meng and Wang (2012) and Wang and Meng (2012b), among others, have studied the fleet deployment and scheduling problems.
The ALNS has been implemented for several vehicle routing problems (VRP) with great success. Ropke and Pisinger (2006) introduces the ALNS and implement it for the VRP with time-windows (VRPTW). The heuristic is tested for over 350 benchmark instances and the computational results show improved solutions on many of these. Pisinger and Ropke (2007) gives a general description of the framework, and a general ALNS framework for the vehicle routing problem is presented. They test the heuristic for five different versions of the vehicle routing problem and show good results.

The capacitated arc-routing problem with stochastic demands is studied in Laporte et al. (2010) where they propose an ALNS heuristic which give good results. Another vehicle routing problem for which the ALNS has been used is the cumulative capacitated VRP (Mattos Ribeiro and Laporte, 2012). This type of VRP is good for modeling situations where the response time is important. The ALNS outperforms other available and published heuristics for this problem.

The ALNS implementations for the VRPs are not directly transferable to our problem. In this fleet deployment problem it is multiple vessels of different types, no central depot and voyage separation requirements on the trades. The voyage separation requirement means that the voyages on a trade have to be "fairly evenly spread", meaning that all sailings could not be performed at the same time, but should be spread within the planning horizon. This implies that the problem contains linking constraints between the routes. A problem which is closer related to the one studied in this paper than pure VRPs is ship routing and scheduling problem with split loads for tramp shipping. For this problem Korsvik et al. (2011) presents an ALNS heuristic. However, the problem does still not have the aforementioned characteristics of voyage separation requirements. For the latest literature survey within ship routing and scheduling we refer to Christiansen et al. (2004).

2 Problem description

Saga Forest Carrier serves several intercontinental trades between regions around the world. A region is for example "North America East Coast", "Oceania" or "Europe". A trade is two regions with a transportation demand between them, for example "Europe - Oceania". Figure 1 gives an example of such a network of regions. In the example there is three regions; "A", "B" and "C", and two trades; "A-B" and "C-A". In addition, between every node (a region) in the network there is a bi-directional arc (stippled) corresponding to the possibility of repositioning the vessel (ballast sailing).

On each of the trades there is a given number of voyages that is to be served during the planning period. The voyages on a trade has an estimated duration which includes the sailing time between all port calls on along the route and the time spent in the ports. The ships have different sailing speeds and thus the voyage duration is ship specific.
To serve the obliged voyages and the spot voyages the company operates a vessel fleet of heterogeneous vessels. The vessels differ in capacity, equipment, sailing speed and cost structure. As this is a tactical planning problem the vessel fleet is assumed fixed. If the vessel fleet turns out too small the shipping company can charter out entire voyages (they rent a vessel and perform a sailing on one trade). Each vessel has its own starting position at the beginning of the planning horizon.

For the customers of Saga Forest carriers it is important, to balance the inventory, that the voyages on a trade is "fairly evenly spread" in time. This means that the time between two consecutive voyages has to be greater than a minimum spread, which is trade specific. This tactical planning problem of Saga Forest Carrier is presented in Norstad et al. (2013), where it is solved both by an arc-flow model and a path-flow model. In the path-flow model the routes are generated a-priori.

3 Solution Method

An adaptive large neighborhood search (ALNS) is developed for the problem. The ALNS heuristic was introduced by Ropke and Pisinger (2006). It builds on the principle of large neighborhood search (LNS) (Shaw, 1997) where an initial solution is destroyed by some destroy operator and then repaired by a repair operator. In ALNS there are multiple destroy and repair operators. Which one to use in a given iteration is determined randomly based on the weight given to the operators according to their performance in earlier iterations. In our implementation of the ALNS we pair all destroy and repair operators that are feasible with each other, and then score the pair based on that pairs’ performance. In each iteration we then pick a pair of operators to use. As there are differences in the time consumption of the operators we normalize the scores of the pairs to account for this. Instead of doing this based on the theoretical complexity as some have done, the time consumption is recorded during runtime and once every \( k \)’th iteration we adjust the
scores. This not to favor very time consuming operators that give much better solution over operators that gives slightly better solution much faster.

We use the simulated annealing acceptance criterion when evaluating new solutions. If a new solution is better than the current solution, the new solution is accepted. But if the new solution is worse than the current solution the new solution is accepted with a probability $p$. $p$ is larger with higher "temperature" $T$. We use a decreasing temperature $T$ with the number of iterations, but with reheating at some points. This to control the diversification and intensification of the search.

The ALNS bases it search on an existing solution, so to make the first solution a construction heuristic is developed. The construction heuristic generate a feasible solution that not necessarily are any good.

4 Computational study and results

To evaluate the heuristic we solve the same test instances as Norstad et al. (2013) solves with an exact method. The instances are generated based on real data from a global Ro-Ro company, and range from small instances up to the real life problem size. Results will be presented and compared with the results of Norstad et al. (2013).

References


