Innovative Applications of O.R.

Optimal fleet composition and periodic routing of offshore supply vessels

Elin E. Halvorsen-Wearea,b,⁎, Kjetil Fagerholt a, Lars Magne Nonåsb, Bjørn Egil Asbjørnslett c

aDepartment of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Alfred Getz veg 3, NO-7491 Trondheim, Norway
bNorwegian Marine Technology Research Institute (MARINTEK), PO Box 4125 Valentinlyst, NO-7450 Trondheim, Norway
cDepartment of Marine Technology, Norwegian University of Science and Technology, O. Nielsens vei 10, NO-7491 Trondheim, Norway

ARTICLE INFO

Article history:
Received 1 July 2010
Accepted 10 June 2012
Available online 17 June 2012

Keywords:
Transportation
Fleet composition
Ship routing and scheduling
Supply operations
Periodic routing

ABSTRACT

The supply vessel planning problem is a maritime transportation problem faced by amongst others the energy company Statoil. A set of offshore installations requires supplies from an onshore supply depot on a regular basis, a service performed by a fleet of offshore supply vessels. The problem consists of determining the optimal fleet composition of offshore supply vessels and their corresponding weekly voyages and schedules. We present a voyage-based solution method for the supply vessel planning problem. A computational study shows how the solution method can be used to solve real-life problems. Statoil has implemented a planning tool based on the voyage-based solution method and reports significant savings.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Many oil and gas producers operate offshore installations that need regular supplies of commodities from land. Specialized offshore supply vessels are used to bring supplies from onshore supply depots out to offshore installations. These supply vessels are a costly resource in the offshore supply chain. Daily time charter rates for one supply vessel can amount to tens of thousands of USDs. To achieve a cost-efficient supply service, good planning of supply vessels is required.

Norway is a major oil and gas producing country with offshore installations in the Norwegian Sea and North Sea. Statoil is the leading operator on the Norwegian continental shelf and controls several onshore supply depots along the Norwegian coastline where supply vessels load cargoes to and discharge cargoes from the offshore installations. Each supply depot comprises a separate planning problem due to government regulated licenses that constrain which supply depots are to supply which offshore installations. Statoil acquires supply vessels on time charter for the supply service and determines a weekly sailing schedule. This schedule is typically valid for a few months depending on when large changes in demand are observed or expected. This could be, for example, when new exploration rigs are to be serviced from a supply depot, or the main operation for an installation changes from drilling to production, or from drilling and production to only production. The supply vessel planning problem thus consists of determining the optimal fleet of supply vessels and their corresponding weekly routes and schedules from an onshore supply depot. The supply vessel planning problem is a fleet composition and periodic routing problem, and the routes and schedule that are determined will be repeated on a weekly basis.

To attain solutions for the supply vessel planning problem, Statoil has previously been using manual planning methods. On request from Statoil we started a project aiming to develop a tool that could be used to solve the supply vessel planning problem. In this paper, we present the new model and solution method that were developed for this purpose. The solution method has been tested on problem instances based on a real supply vessel planning problem faced by Statoil.

This paper is organized as follows: The supply vessel planning problem is described in Section 2. In Section 3 some relevant literature is discussed. The solution approach is described in Section 4, while Section 5 presents the computational study along with some practical considerations. Section 6 concludes the paper.

2. Problem description and modeling assumptions

The supply vessel planning problem consists of identifying the optimal fleet composition of supply vessels that are to service a given number of offshore installations from one common onshore depot while at the same time determining the weekly routes and schedules for these vessels. A route in this setting is a combination
of one or more voyages, starting and ending at the supply depot, which a vessel sails during a week. During a voyage, the vessels may visit one or more offshore installations. The problem is illustrated in Fig. 1 for an instance with eight offshore installations.

The objective for the supply vessel planning problem is to minimize the costs while at the same time maintaining a reliable supply service. The costs that are to be minimized are primarily the time charter costs for the supply vessels, then the sailing costs of the voyages. A number of constraints and assumptions need to be assessed. In the following, the constraints are shown in italics.

The onshore supply depot and the offshore installations may have opening hours during which they can unload and load supply vessels. The supply depot is normally open for service during regular Norwegian working hours (0800–1600). The installations are normally either closed for loading and unloading operations at night (1900–0700) or open for such service around the clock. The turnaround time for the vessels at the supply depot can be estimated to about eight hours, which coincides with the opening hours. A supply vessel will therefore need to be at the supply depot before 0800 to start on a new voyage the same day. It is assumed that all voyages start from the supply depot at 1600.

Limited capacity at the supply depot sets a bound on the number of vessels that may be prepared for a new voyage on a given day. This number may vary during the week due to personnel availability, typically with less capacity at weekends.

A supply vessel has a given deck and bulk capacity, service speed and time charter rate. The deck capacities for the vessels vary from about 600 to 1100 meter$^2$, and bulk capacities from about 4000 to 8000 tons. Historical data show that the deck capacity is the binding capacity resource for the supply vessels, thus all demands from installations are given in m$^2$ deck capacity. There are also backloads that need to be transported from the offshore installations to the onshore supply depot, but the backload volume will in almost every case be less than the demand. It is therefore assumed that there will be enough capacity to cover the backload.

The installations’ demands are estimated as weekly demands that need to be delivered. The installations also need to be visited a given number of times during the week, so that the demand for each visit is the weekly demand divided by the number of visits. The demand is not necessarily evenly spread throughout the week, so the demand for each installation visit is upscaled by a load factor (set to 20%) to allow for some variations. The total demand for all installations visited on a voyage cannot exceed the capacity of the vessel sailing the voyage. All installations have a given service time for unloading and loading operations of the supply vessels. The departures from the supply depot to a given installation should be evenly spread throughout the week. It is more important to spread the departures to an installation than the actual visits, as the demand from an installation is reported continuously. This means, for example, that if an installation requires three service loads a week and the supply vessels visiting the installation leave the supply depot on three consecutive days, a demand may be called in after the third vessel has left and it will be almost five days until the next departure. In such cases it may be necessary to reroute other supply vessels or send out a helicopter to meet the demand. This will in most cases be very costly, and can to a large extent be avoided if the departures are evenly spread.

Fig. 2 illustrates voyages with evenly spread departures in a solution to a small instance with two supply vessels and seven offshore installations. Installations 1, 5, and 7 require one weekly visit each, while installations 2, 3, 4, and 6 require two weekly visits each. The figure shows the four voyages sailed by the two vessels, and departure days for the voyages. The departures to the installations that require two visits a week are spread so that there are at least three days between each departure.

The duration of a voyage is given as an integer number of days and will vary depending on the number of installations visited on the voyage, their service times, the sailing distances between them, and the service speed of the supply vessel sailing the voyage. Too short voyages with too few visits are not desired as they will not exploit the supply vessels’ capacities properly, and too long voyages should also be avoided as they involve more uncertainty regarding sailing time. Therefore minimum and maximum durations of voyages are introduced. For the same reasons, there will be a minimum and maximum number of visits for each voyage.

3. Relevant literature

The supply vessel planning problem is a combined fleet composition and periodic routing problem that has similarities to the well-studied vehicle routing problem (VRP). For example, there is one common onshore supply depot where the vessels’ voyages start and end that can be viewed as the depot in a VRP. However, the supply vessel planning problem is more complex than a traditional VRP. Where the planning horizon for a VRP is typically one day (time period), we study a horizon of one week (multiple time periods). In addition, there are other complicating constraints that are not considered in the traditional VRP, such as opening hours at the installations and the onshore supply depot, installations that require more than one visit during the planning horizon and
vessels that may sail more than one voyage. Also, this is a periodic routing problem where part of the objective is to create a weekly sailing schedule that repeats itself.

Previous studies have addressed some of the aspects of the problem discussed in Section 2. Some of these will now be presented.

The fleet composition problem concerns the strategic decision of determining an optimal fleet of vehicles. The models very often include routing decisions since it is necessary to consider the underlying structure of the operational planning problem, as discussed by Christiansen et al. (2007). This class of problems is known as fleet size and mix vehicle routing problems (FSMVRP). An early reference to the FSMVRP is Golden et al. (1984). The authors present a mathematical formulation for the problem and suggest several heuristic solution procedures. A fleet composition problem from the maritime industry is presented by Fagerholt (1999). Hoff et al. (2010) present a literature survey covering fleet composition and routing problems in both road-based and maritime transportation. The survey paper discusses industrial aspects of the problem and presents basic mathematical models for the problem found in the literature. In total, the authors reviewed about 120 scientific papers that address the combination of fleet composition and routing.

For VRPs with multiple use of vehicles each vehicle may be assigned to more than one trip during the planning horizon. Ronen (1992) studies the problem of assigning a set of trips to a heterogeneous fleet of trucks, and proposes two heuristic methods to solve larger sized problems. Other references are Taillard et al. (1996) who propose a tabu search heuristic for the problem and Brandão and Mercer (1997) who propose a tabu search heuristic to a more complicated version of the VRP with multiple use of vehicles. In Brandão and Mercer (1998) the authors present a simplified version of their tabu search algorithm to compare it with the one from Taillard et al. (1996). An exact algorithm for the VRP with time windows and multiple use of vehicles can be found in Azi et al. (2010) where the authors introduce a branch-and-price approach.

The periodic vehicle routing problem (PVRP) is a generalization of the VRP where vehicle routes are constructed for a period of time that can be more than one day (see Francis et al., 2008). The problem then consists of deciding when to visit the customers during the time period (and there could be more than one visit to each customer), assigning the visits to a vehicle, and then optimizing the routes for each vehicle. The supply vessel planning problem contains one more obstacle compared with the PVRP: A route (voyage) may last more than one day, and it is therefore necessary to ensure that a vessel starting on a voyage at the end of one time period is not assigned to a new voyage in next period before it has returned to the supply depot.

There are several case studies and algorithms for the PVRP. An early study by Beltrami and Bodin (1974) proposed a heuristic solution method for a PVRP for municipal waste collection. A heuristic algorithm for the PVRP based on an algorithm designed for the VRP (see Fisher and Jaikumar, 1981) can be found in Tan and Beasley (1984). Some later references are the papers by Baptista et al. (2002) that presents a two stage heuristic solution method for a real PVRP for the collection of recycling paper containers, and Hemmelmayr et al. (2009) that propose a heuristic algorithm based on variable neighborhood search. Mourgaya and Vanderbeck (2007) present a two-phase approach for the tactical problem of deciding dates to visit customers and assigning them to vehicles, leaving the sequencing decisions to be decided on an operational level.

Literature on the PVRP where one single route may last more than one day is scarce. Two relevant papers are Fagerholt (1999) and Fagerholt and Lindstad (2000), where the aspect of multiple use of vehicles is also present.

There are a few papers that consider routing and logistics problems in the offshore supply business. Aas et al. (2009) discuss the role of supply vessels in the offshore logistics chain, as these are a costly resource. The focus of the paper is on analyzing the design of the supply vessels to see how their design can be improved to better support the operations. Fagerholt and Lindstad (2000) study supply operations in the Norwegian Sea, and present an integer programming model based on a priori generation of voyages. This model can only be used to solve a simplified version of the supply vessel planning problem as complicating aspects such as spread of departures, service capacity constraints for the onshore supply depot, and maximum and minimum duration of voyages are omitted. These are all important real-life considerations.

The routing problem arising in the service of offshore installations is studied by Gribkovskaia et al. (2008). The authors consider a single vehicle pickup and delivery problem with capacitated customers, also taking into account the backload from the installations and the limited storage capacity on the installations. The objective is finding a least-cost voyage that starts and ends at the supply depot and visits each installation exactly once.

4. Solution method

In this section we present the voyage-based solution method developed to solve the supply vessel planning problem as it is described in Section 2. An arc-flow model formulation for the problem that gives a more detailed mathematical description of the problem has also been developed and can be found in Halvorsen-Weare (2012). A schematic overview of the voyage-based solution method is given in Section 4.1. Then the voyage generation procedure and the voyage-based model formulation is presented in Sections 4.2 and 4.3, respectively.

4.1. Solution method overview

The voyage-based solution method consists of two phases:

Phase one: Generate all candidate voyages the vessels may sail

Phase two: Solve the voyage-based model

Fig. 3 provides a schematic overview of the solution method procedure.

As illustrated in Fig. 3 some input data, such as sailing distances between installations and the installations’ demands, are only used when generating candidate voyages. Other input, such as requirements for the spread of departures and capacity for loading and unloading supply vessels at the supply depot, are used only when solving the voyage-based model.

Output from the voyage generation procedure is all candidate voyages a vessel can sail. The voyage data contain all relevant information needed in the voyage-based model like sailing costs, what installations that are visited, visiting sequence and time for visits, and the duration of the voyage (in days).

The results after solving the voyage-based model include the optimal fleet of supply vessels and which voyages they should sail on which days.

4.2. Voyage generation

To generate all candidate voyages, we start by defining all possible subsets of offshore installations that may be visited by a given supply vessel. The size of the subsets will be limited by the minimum and maximum number of installations to visit on a voyage and the maximum capacity of the supply vessels.
Then, for each of the subsets we need to solve a traveling salesman problem (TSP) with multiple time windows. There are multiple time windows when one or more of the offshore installations in the subset have opening hours for unloading and loading supply vessels since these are open for such service every day between 0700 and 1900 and a voyage can last for several days. For subsets containing no offshore installations with opening hours we only need to solve a standard TSP. The distances between each offshore installation and the installations and the supply depot are given. Then the sailing speed for the supply vessel decides the corresponding sailing times. The sailing speed may vary during a voyage based on weather conditions and delays according to schedule, but each vessel has a given service speed that is used when planning voyages. This service speed is used when calculating sailing times.

The cost of each voyage is calculated as fuel consumption multiplied by the cost of fuel. We operate with two different fuel consumption rates: one while sailing and one while laying still at an offshore installation. The TSPs are solved by finding the voyage with shortest duration, which will in most cases also be the lowest cost voyage.

Due to the limited sizes of the TSPs we need to solve for the supply vessel planning problem faced by Statoil, we solve the TSPs by a full enumeration procedure: All possible voyages, that is, all possible visiting sequences for the installations in the subset, are compared and the one with the shortest duration is chosen. This will normally also be the one with the shortest travel distance. In the cases where the voyage with shortest duration is not equivalent to the one with shortest travel distance, and this only has a marginally longer duration, the voyage with shortest travel distance is chosen as this then will be the lowest cost voyage.

The duration of possible voyages is calculated as follows: Starting from the onshore supply depot the sailing time to the next offshore installation is calculated and added to the total duration based on the distance and sailing speed of the vessel. Then, if the offshore installation is closed for service upon arrival, the waiting time until opening time is added to the total duration. When, or if, the installation is open for service, the service time is added to the total duration. If the unloading and loading operations cannot be completed before the offshore installation closes for service, a 12 hours waiting time is added to the total duration. Then the sailing time to the second offshore installation is calculated, and this time together with the waiting time and service time at this installation is added to the total duration. This procedure continues until the next visit is to the onshore supply depot.

A pseudocode for the voyage generating procedure is given in Table 1.

The output from the voyage generation procedure is the set of all candidate voyages that the vessels may sail, denoted $R$. This is then input to the voyage-based model presented in next section.

### 4.3. Voyage-based model formulation

The voyage-based model solves the supply vessel planning problem by choosing the most cost-effective supply vessels and picking the best of the pregenerated voyages that in combination fulfill the constraints. We start by introducing the notation before the general model, called the base case model, is presented. Additional notation and constraints will then be introduced.

<table>
<thead>
<tr>
<th>Table 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>The voyage generation procedure.</td>
</tr>
<tr>
<td><strong>Voyage generation procedure</strong></td>
</tr>
<tr>
<td>Create sets of vessels $\text{VesselSet}$ with equal sailing speed</td>
</tr>
<tr>
<td>For all $\text{VesselSet}$</td>
</tr>
<tr>
<td>Find vessel in $\text{VesselSet}$ with largest loading capacity, $\text{vesselMax}$</td>
</tr>
<tr>
<td>Enumerate all sets of installations $\text{InstallationSet}$ that fulfill minimum and maximum requirements</td>
</tr>
<tr>
<td>on number of installations in a voyage and that does not exceed the capacity of $\text{vesselMax}$</td>
</tr>
<tr>
<td>For all $\text{InstallationSet}$</td>
</tr>
<tr>
<td>Find a voyage by solving a traveling salesman problem with time windows starting and ending at the supply depot where all installations in $\text{InstallationSet}$ are visited exactly once</td>
</tr>
<tr>
<td>If voyage does not violate minimum and maximum duration requirements</td>
</tr>
<tr>
<td>Add voyage to $\text{VoyageSet}$ for $\text{vesselMax}$ $(\text{VoyageSet vesselMax})$</td>
</tr>
<tr>
<td>End If</td>
</tr>
<tr>
<td>End For all $\text{InstallationSet}$</td>
</tr>
<tr>
<td>For all vessels in $\text{VesselSet}$ not $\text{vesselMax}$</td>
</tr>
<tr>
<td>For all voyages in $\text{VoyageSet}$ $\text{vesselMax}$</td>
</tr>
<tr>
<td>If voyage does not violate capacity of vessel</td>
</tr>
<tr>
<td>Add voyage to $\text{VoyageSet}$ $\text{vessel}$</td>
</tr>
<tr>
<td>End If</td>
</tr>
<tr>
<td>End If</td>
</tr>
<tr>
<td>End For all voyages</td>
</tr>
<tr>
<td>End For all vessels in $\text{VesselSet}$</td>
</tr>
<tr>
<td>End For all $\text{VesselSet}$</td>
</tr>
<tr>
<td>Return all $\text{VoyageSet}$s</td>
</tr>
</tbody>
</table>
Let $V$ be a set containing all supply vessels available for time charter, and let $N$ be the set of all offshore installations. Then set $R_v$ contains all pregenerated voyages vessel $v$ may sail. Further let $T$ be the set of days in the planning horizon (a week), and $L$ be a set containing all possible voyage durations in days (two or three). Then subset $R_v \subseteq R_v$ contains all candidate voyages of duration $l$ that vessel $v$ may sail.

$C_p^v$ represents the weekly time charter cost for vessel $v$, and $C_s^v$ all sailing and service costs for voyage $r$ sailed by vessel $v$. Further $D_r$ is the duration of voyage $r$ sailed by vessel $v$ in days (rounded up to the nearest integer), $F_r$ the number of days vessel $v$ may be used during the week, $S_r$ the number of weekly visits required by installation $i$, $B_l$ the maximum number of supply vessels that may be serviced at the supply depot on day $t$ and the binary parameter $A_{vr}$ is 1 if vessel $v$ visits installation $i$ on voyage $r$, and 0 otherwise.

Finally, the decision variables are as follows:

$$
\delta_v = \begin{cases} 
1, & \text{if vessel } v \text{ is used} \\
0, & \text{otherwise}
\end{cases}$$

$$
x_{vr} = \begin{cases} 
1, & \text{if vessel } v \text{ sails voyage } r \text{ starting on day } t \\
0, & \text{otherwise}
\end{cases}
$$

The base case voyage-based model formulation then becomes:

$$
\min \sum_{v \in V} C_p^v \delta_v + \sum_{\substack{v \in V \cap R_v \cap T \cap L \cap T \cap \mathbb{Z}^+}} \sum_{r \in R_v \cap T} C_s^r x_{vr},
$$

subject to

$$
\sum_{r \in R_v \cap T} \sum_{v \in V} A_{vr} x_{vr} \geq S_i, \quad i \in N,
$$

$$
\sum_{r \in R_v \cap T} D_r x_{vr} - F_r \delta_v \leq 0, \quad v \in V,
$$

$$
\sum_{r \in R_v \cap T} x_{vr} \leq B_t, \quad t \in T,
$$

$$
\delta_v \in \{0, 1\}, \quad v \in V, \quad r \in R_v, \quad t \in T.
$$

The objective function (1) minimizes the sum of the time charter costs and the sailing costs. As the time charter costs are much higher than the sailing costs, the primary objective is to find the most cost-effective fleet composition. Constraints (2) ensure that all installations get their required number of visits during the planning horizon, and constraints (3) ensure that the total duration of all voyages sailed by a vessel does not exceed the maximum number of days the vessel may be in service during the planning horizon. These constraints also ensure that if a vessel $v$ is used, the binary variable $\delta_v$ must equal 1. Further, constraints (4) ensure that there are no more supply vessels at the supply depot on day $t$ than there is capacity to service. Constraints (5) ensure that a vessel does not start on a new voyage before it has returned to the supply depot after the last voyage. Finally, constraints (6) and (7) set the binary requirements for the $\delta_v$ and $x_{vr}$ variables, respectively.

### 4.3.1. Spread of departures

The following constraints may be used to ensure evenly spread departures. The constraints are determined by how many visits a given installation requires during the planning horizon of one week. The sets $N_2, N_3, N_4$, and $N_5$ contain the installations that require 2, 3, 4, and 5 visits a week, respectively.

$$
\sum_{i \in N_2, t \in T} A_{vi} x_{vt} \leq 1, \quad i \in N_2, t \in T,
$$

$$
\sum_{i \in N_3, t \in T} A_{vi} x_{vt} \leq 1, \quad i \in N_3, t \in T,
$$

$$
\sum_{i \in N_4, t \in T} A_{vi} x_{vt} \leq 2, \quad i \in N_4, t \in T,
$$

$$
\sum_{i \in N_5, t \in T} A_{vi} x_{vt} \leq 3, \quad i \in N_5, t \in T.
$$

Constraints (8) ensure that there will be maximum one voyage starting on day $t$ that is to visit a given installation $i$. These constraints are sufficient for installations that require six or seven visits a week. Constraints (9) ensure that for installations requiring two visits a week, there will be at most one departure during any three day period. Then constraints (10) ensure that there will be at least one departure during any three day period for installations requiring three visits a week. Further, constraints (11) ensure that installations requiring four visits a week will have at least two departures during any four day period. Finally, for installations requiring five visits a week, constraints (12) ensure that there will be at least one departure during any two day period.

Constraints (8)–(12) are one way of formulating constraints that ensure spread departures from the supply depot, and there are other formulation options. Some will be equivalent to these constraints, while others will give a smaller or larger solution space. For example, constraints (9) could be formulated differently by ensuring that there will be minimum one visit during any four day period. This would result in the same solution space. Constraints (10) could be replaced by constraints ensuring that there
will be maximum one visit during any two day period. This would provide a stronger formulation as constraints (10) allow for departures on two consecutive days, reducing the solution space. Constraints (8)–(12) have been chosen as these were the ones preferred by Statoil.

Fig. 4 shows some examples of departure patterns for installations requiring two, three, four and five visits a week that are feasible with regard to constraints (8)–(12).

5. Computational study and practical considerations

The voyage-based solution method presented in Section 4 has been tested on problem instances based on a real supply vessel planning problem faced by Statoil. In Section 5.1 these problem instances are described. This is followed by the numerical results in Section 5.2. Section 5.3 considers some practical considerations concerning the supply vessel planning problem, while Section 5.4 shows how the voyage-based solution method can be used for what-if scenario analysis.

5.1. Problem instances

Twenty-two problem instances have been generated based on real data provided by Statoil. The problem instances are made based on 11 installations that are permanently serviced from Mongstad supply depot, and also three floating rigs that at a point in time were serviced from this supply depot. This supply depot represents the largest supply vessel planning problem that Statoil needs to solve.

The problem instances have been numbered after how many installations there are and how many of them that have opening hours for unloading and loading operations. For example, problem instance 3-0 has three installations where none of them have opening hours, while problem instance 14-3 has 14 installations where three of them have such opening hours. The opening hours mean that the supply vessels need to be serviced between 0700 and 1900.

The number of installations in the problem instances varies from three to fourteen. The number of visits the installations requires every week varies from one to six and the total number of visits for each problem instance varies from 16 to 59. The weekly demands for each installation vary from 250 to 960 meter$^2$. These demands are then divided by the number of visits an installation requires during a week to get the demand for each visit. These again vary from 50 to 334 meter$^2$. The service times at the installations vary from 2.25 to 7 hours.

For all problem instances five supply vessels are available that may be chartered. The vessels’ loading capacities vary from 900 to 1090 meter$^2$. The time charter rates for all vessels are above USD 100,000 per week. The rates vary, depending on the capacity of the vessel, the vessel with least capacity having the lowest time charter cost. The sailing speed for all supply vessels is 12 knots.

For all problem instances the duration of each voyage has to be either two or three days. Minimum number of visits on a voyage is limited by the minimum duration of a voyage, and maximum number of visits is eight installations. The capacity at the supply depot is three supply vessels a day, Monday to Saturday, and zero on Sunday as the supply depot is closed on this day.

All results referred to in this paper were obtained on a 2.16 Giga Hertz Intel Core 2 Duo PC with 2 Giga Byte RAM. The voyage-based model formulation was implemented in Xpress-IVE 1.19.00 with Xpress-Mosel 2.4.0 and solved by Xpress-Optimizer 19.00.00. Maximum CPU time when solving the Xpress-MP models was set to 10,000 seconds. The voyage generator was written in C++ using Visual Studio 2005.

5.2. Numerical results

Table 2 shows the results from solving the problem instances using the voyage-based solution method. The CPU times are reported for the voyage generation procedure, for solving the voyage-based model, and the total CPU times. Opt. gap refers to the optimality gap reported from Xpress-MP (gap between objective value and best lower bound), # voy. generated refers to the number of voyages generated by the voyage generation procedure, and # vess. and # voy. refer to the total number of vessels and voyages used in the solution, respectively.

The results show that the voyage-based solution method gives optimal solutions within the CPU time limit for all but the largest problem instances. We observe that when optimal solutions are not obtained, the optimality gap is less than 1% for all but one instance. For the instance with the higher optimality gap, the gap is still small enough to conclude that it is not possible to reduce the size of the fleet. This means that the optimal fleet composition is found, which is the main objective for the supply vessel planning problem.

The problem instances developed represent a good span of the real supply vessel planning problem Statoil needs to solve, and we obtain good results for all of them with the voyage-based solution method. This indicates that our method is well suited for solving the supply vessel planning problem faced by Statoil.

5.3. Other practical aspects of the problem

Several other aspects, which have not been discussed so far in this paper, may appear when solving real-life instances of the supply vessel planning problem. In the following, such practical aspects of the supply vessel planning problem that appeared while working on the problem together with planners from Statoil are discussed. Problem instance 11-3 will be used for illustrative purposes.

5.3.1. Case specific constraints

For some instances of the supply vessel planning problem, there may be case specific considerations that can be modeled as constraints in the voyage-based model. Here we present two such considerations that appeared while analyzing Statoil’s planning problem.

5.3.1.1. Departure to certain installations on given days

For the supply vessel problem at the Mongstad supply depot, there are certain installations that require visits by vessels departing from the supply depot on given days.

These departure specific considerations may be formulated as constraints. For example, the following constraint will ensure that installation 10 is visited by a vessel departing from the supply depot on Wednesdays:

$$\sum_{v \in V_{10}} \sum_{r \in R} A_{vr} x_{vr,10} \geq 1.$$  \hspace{1cm} (13)

Constraints like (13) will reduce symmetry, and will in most cases make the voyage-based model easier to solve to optimality.

5.3.1.2. Duration requirement varies throughout the week

For all problem instances described in Section 5.1, the duration of the voyages was minimum two and maximum three days. In addition, no voyages could start on Sundays. In real-life, voyages with a duration of only two days are preferred, and voyages with a duration of three days are only allowed to start on Fridays and Saturdays.

To enforce this, the following constraints may be added:
ensure that vessel departures to a given installation are spread to prove optimum within the maximum CPU time limit. When only A and B were added, it decreased even more. However, when only that even minor adjustments would require the use of a fourth supply vessel. This indicates the optimal solution for problem instance 11-3 was quite tight compared with instance 11-3.

Table 3 shows the results when the constraints discussed above are added to the voyage-based model individually and in combination for the original problem instance 11-3. A and B refer to the constraints described in Sections 5.3.1.1 and 5.3.1.2, respectively. The CPU times for solving the voyage-based model are reported in the table. It was only necessary to adjust the voyage-based model for the original problem instance 11-3. A and B refer to the constraints. The CPU time for the voyage generator, so there was no need to regenerate voyages. The CPU time for the voyage-based constraints described in Sections 5.3.1.1 and 5.3.1.2, respectively.

The spread of departure constraints introduced in Section 4.3 for the practical situations, be undesirable as the two vessels will not be able to service the installation simultaneously.

Constraints like (14) are likely to make the voyage-based model easier to solve to optimality as they reduce the number of variables by forcing many of those that represent voyages with a duration of three days to equal zero.

Table 3 shows the results when the constraints discussed above are added to the voyage-based model individually and in combination for the original problem instance 11-3. A and B refer to the constraints described in Sections 5.3.1.1 and 5.3.1.2, respectively. The CPU times for solving the voyage-based model are reported in the table. It was only necessary to adjust the voyage-based model, so there was no need to regenerate voyages. The CPU time for the voyage generator for the original problem instance 11-3 was 73.4 seconds as reported in Table 2. The gap from 11-3 column refers to the percentage difference in the cost of the solution compared with instance 11-3.

We observe from Table 3 that the added constraints result in a solution requiring the use of four supply vessels compared with three vessels for the original problem instance 11-3. This indicates that the optimal solution for problem instance 11-3 was quite tight and that even minor adjustments would require the use of a fourth supply vessel.

For the CPU times we observe that when constraint set B was added, the CPU time decreased considerably, and when both A and B were added, it decreased even more. However, when only constraint set A was added, the Xpress-Optimizer did not manage to prove optimum within the maximum CPU time limit.

5.3.2 Collision constraints

The spread of departure constraints introduced in Section 4.3 ensure that vessel departures to a given installation are spread throughout the week. But these constraints do not prevent more than one vessel being at the same installation at the same time. For example, an installation may be visited late on a voyage starting on Tuesday, and early on a voyage starting on Wednesday, and this again may result in the two vessels sailing the voyages reaching the installation at about the same time. This will again, in most practical situations, be undesirable as the two vessels will not be able to service the installation simultaneously.

Table 3 Case specific constraints.

<table>
<thead>
<tr>
<th>Problem instance</th>
<th>CPU voy. model (seconds)</th>
<th>Opt. gap (%)</th>
<th>Gap from 11-3 (%)</th>
<th># Vess.</th>
<th># Voy.</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-3 A</td>
<td>10.00</td>
<td>10.31</td>
<td>21.94</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>11-3 B</td>
<td>72.5</td>
<td>0.00</td>
<td>22.90</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>11-3 AB</td>
<td>7.1</td>
<td>0.00</td>
<td>23.08</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>

\[
\sum_{v \in V} \sum_{r \in R} x_{vrt} = 0, \quad t = \{1, 2, 3, 4, 7\}. \tag{14}
\]

Constraints like (14) are likely to make the voyage-based model easier to solve to optimality as they reduce the number of variables by forcing many of those that represent voyages with a duration of three days to equal zero.

Table 3 shows the results when the constraints discussed above are added to the voyage-based model individually and in combination for the original problem instance 11-3. A and B refer to the constraints described in Sections 5.3.1.1 and 5.3.1.2, respectively. The CPU times for solving the voyage-based model are reported in the table. It was only necessary to adjust the voyage-based model, so there was no need to regenerate voyages. The CPU time for the voyage generator for the original problem instance 11-3 was 73.4 seconds as reported in Table 2. The gap from 11-3 column refers to the percentage difference in the cost of the solution compared with instance 11-3.

We observe from Table 3 that the added constraints result in a solution requiring the use of four supply vessels compared with three vessels for the original problem instance 11-3. This indicates that the optimal solution for problem instance 11-3 was quite tight and that even minor adjustments would require the use of a fourth supply vessel.

For the CPU times we observe that when constraint set B was added, the CPU time decreased considerably, and when both A and B were added, it decreased even more. However, when only constraint set A was added, the Xpress-Optimizer did not manage to prove optimum within the maximum CPU time limit.

5.3.2 Collision constraints

The spread of departure constraints introduced in Section 4.3 ensure that vessel departures to a given installation are spread throughout the week. But these constraints do not prevent more than one vessel being at the same installation at the same time. For example, an installation may be visited late on a voyage starting on Tuesday, and early on a voyage starting on Wednesday, and this again may result in the two vessels sailing the voyages reaching the installation at about the same time. This will again, in most practical situations, be undesirable as the two vessels will not be able to service the installation simultaneously.

To prevent such events, constraints can be added that take arrival and departure times at the installations on a voyage as input and prevent two vessels from being at an installation at the same time.

Let $G_{str}$ and $H_{str}$ be the arrival time at and departure time from installation $i$ by vessel $v$ on voyage $r$, respectively. The arrival and departure times are given as hours after departure from the supply depot, and are calculated by the voyage generator. Further, let $U$ be the turnaround time at the supply depot and let $P$ represent a user-specified parameter representing the minimum number of hours required between planned departures and arrivals at an installation. The following constraints can then be added:

\[
\sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt} \leq (48 - U) \left( \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+1\mod|T|) - \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+1\mod|T|) \right) + \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+1\mod|T|) (24 + G_{str} - P),
\]

\[
i \in N, t \in T. \tag{15}
\]

\[
\sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt} \leq (72 - U) \left( \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+2\mod|T|) - \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+2\mod|T|) \right) + \sum_{v \in V} \sum_{r \in R} a_{str} x_{vrt}/(t+2\mod|T|) (48 + G_{str} - P),
\]

\[
i \in N, t \in T. \tag{16}
\]

Constraints (15) ensure that if there is a departure to installation $i$ on day $t+1$, the arrival time for this vessel at the installation has to be minimum $P$ hours after the departure by a vessel visiting the same installation departing from the supply depot on day $t$. Similarly, constraints (16) ensure that a vessel departing on day $t+2$ arrives at the installation a minimum of $P$ hours after the
departure by a vessel departing on day \( t \). These constraints are sufficient if the maximum duration of a voyage is three days.

Since only the voyage with the optimal visiting sequence for a given set of installations is generated and not all feasible voyages, adding these constraints may destroy the optimality of the solution and transfer the voyage-based solution method from an exact method to a heuristic. Therefore a double voyage module has been added to the voyage generator. This gives the option of generating two voyages for a given set of installations instead of just one. The second voyage is created by letting the last installation visited on the first voyage be the first one visited on the second voyage if this installation does not have opening hours; if it has, let the second last visited be the first one and so on. The rest of the visiting sequence for the second voyage is decided by finding the best way of visiting the remaining installations. In most cases this will result in two generated voyages where the second one is close to the reverse of the first one.

Table 4 shows the results from solving problem instance 11-3 with constraint sets A and B without the double voyage module. The user specified parameter \( P \) is set to six hours. The costs of the solutions found are compared with the cost of the solution found when solving problem instance 11-3 with constraint sets A and B (see Table 3). From the results we observe that we get a slightly better solution when using the double voyage module at the cost of an increase in CPU time. The cost of this solution is only marginally higher than the one achieved with constraint sets A and B without the collision constraints (0.01%).

### 5.3.3. Robust schedules

Uncontrollable events, such as rough weather, may cause delays to planned voyages, and schedules. For example, if the significant wave heights at sea are above certain critical limits, the supply vessels cannot perform any operations at the installations and thus need to wait until the weather conditions improve. In addition, rough weather will reduce the sailing speed for the vessels. In the sea outside the Norwegian coast, the probability of such events is relatively high in the winter season.

The probability of rough weather conditions creates a need for more robust schedules, where a delay to a voyage will not necessarily affect other voyages performed by a supply vessel. One way of dealing with this is to require a given slack for each voyage, where slack is defined as hours available after finishing a voyage before starting to prepare for next voyage at the supply depot. For instance, if it is a three day voyage and the turnaround time at the supply depot is eight hours, total duration of the voyage may not exceed 64 hours. If the voyage then lasts for 58 hours, the voyage has six hours slack.

Table 5 shows the results from solving problem instance 11-3 with constraint sets A and B, collision constraints and 2 or 4 hours slack requirements for each voyage. All problem instances are solved with both single voyage generation and double voyage generation. The results are compared with that of problem instance 11-3 included constraint sets A and B, collision constraints and the double voyage module (see Table 4), referred to in Table 5 as 11-3°.

We observe from Table 5 that the sailed voyage with the least slack has slack just above the minimum requirement for all problem instances (column Min slack). Average slack refers to the average slack over all sailed voyages. The column named Total slack shows the total idle time for the supply vessels. This includes the slack of the sailed voyages and the days where the supply vessels are not used. We observe that the total slack is higher when there are no robustness requirements. This is because the vessels sail fewer voyages with less slack. Consequently, there are then more days when the vessels stay idle.

Instead of requiring a minimum number of hours of slack for each voyage, it is also possible to penalize voyages with less slack than a given number of hours so that these voyages will be selected only if found very profitable compared with other voyages with more slack.

### 5.4. What-if scenario analyses

The voyage-based solution method may be used to perform what-if scenario analyses. The model has been used by Statoil for this purpose, and many of the practical considerations discussed above are examples of such analyses.

What-if analyses can be useful especially in cases where it may be possible to use one less supply vessel if some adjustments are made. We notice, for example, that for most of the practical constraints introduced, it was necessary to use four vessels compared with three vessels for the original problem instance 11-3 (see Table 2). This indicates that it may be possible to reduce the number of supply vessels to three also when practical constraints are added.

As an example, we will use problem instance 11-3 with constraint sets A and B, collision constraints and the double voyage module and see if it may be possible to reduce the number of supply vessels through the following scenario analyses:

(a) What if all installations are open for service 24 hours?
(b) What if the number of visits to installations 1, 8, and 9 are reduced from six to five?
(c) What if installation 4 is visited from a different supply depot?

<table>
<thead>
<tr>
<th>Problem constraints.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single instance</td>
</tr>
<tr>
<td>Single</td>
</tr>
<tr>
<td>Double</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Robust schedules.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem instance</td>
</tr>
<tr>
<td>11-3°</td>
</tr>
<tr>
<td>2 Hours single</td>
</tr>
<tr>
<td>2 Hours double</td>
</tr>
<tr>
<td>4 Hours single</td>
</tr>
<tr>
<td>4 Hours double</td>
</tr>
</tbody>
</table>
Scenario (a) will increase the flexibility of the voyages by not forcing the three installations with opening hours to be serviced between 0700 and 1900. The scenario under (b) does not reduce the installations’ weekly demand; consequently each visit’s demand will increase. It will reduce the required sailing distance for the vessels as the number of weekly installation visits is reduced by three. Finally, (c) refers to the case where the location of an installation is close to a different supply depot, and it may be more beneficial to let it be serviced from that supply depot (even though license regulations restrict from doing so for the time being).

Table 6 shows the results from solving the scenarios described above. The cost of each solution is compared with the cost of the solution to problem instance 11-3 with added constraint sets and double voyage module as in Table 4. We observe that for scenarios (a) and (b) four supply vessels are still needed, and only small cost reductions are obtained, while for scenario (c) we manage to reduce the number of vessels from four to three. We also get a solution with lower costs than the original problem instance 11-3. This is not surprising since we only have 10 installations in this scenario.

Such analyses as the ones shown here may be very useful if for example installation 4 may be visited from a different supply depot without needing to increase the number of supply vessels there. In such cases, a cost reduction of more than 100,000 USD a week may be obtained.

The main objective for the voyage-based solution method is to determine the optimal fleet composition for the supply vessel planning problem. To obtain this objective, the sailed voyages and schedules also need to be considered and the resulting sailing schedule should represent a schedule that realistically can be put into life. However, the actual sailed schedule will in most cases differ from the planned one as the operational problem involves many uncontrollable factors that may result in rescheduling. When we incorporate practical aspects and do what-if scenarios like the ones discussed in this section, we exploit the problem nature and give planners valuable information regarding the optimal fleet composition’s capabilities.

6. Concluding remarks

In this paper we have considered a fleet composition and periodic routing problem that appears in the offshore supply vessel service. This is a real-life planning problem and the work originates from a project performed together with Statoil, the leading operator on the Norwegian continental shelf.

We have presented a voyage-based solution approach for the problem where all candidate voyages that the supply vessels may sail are generated a priori. Then the fleet composition and what voyages to sail when by each vessel are decided by solving a voyage-based model. The computational study shows that the voyage-based solution method can be used to solve real-sized problem instances.

A decision support tool based on the voyage-based model has shown to be very valuable for the planners in Statoil. So far, the tool has played an essential role in the process of reducing the number of supply vessels (from five to four) while maintaining an efficient and reliable supply service at one onshore supply depot. According to Statoil, the annual cost savings obtained by the tool at this supply depot are estimated to be roughly USD 3 million, reflecting the cost savings of reducing the supply vessel fleet by one vessel while still maintaining the number of weekly visits.

We observe that the voyage-based solution method could not find or prove optimal solutions to the largest problem instances within the predefined CPU time limit. This indicates that the solution method reaches its limits, and may not be used to solve larger problem instances. As the method is sufficient to solve Statoil’s supply vessel planning problems, it was not necessary to exploit other solution algorithms. Column generation schemes or heuristic methods may be implemented in order to solve even larger problem instances.

This paper only briefly touches on the subject of finding robust schedules although robustness is an important factor in maritime transportation. The robustness approach presented in this paper is a simplified way of getting solutions that are somewhat more resistant to delays due to unforeseen events. Future research should focus on how to create robust schedules where small delays will have little or no consequence for other planned voyages.

Acknowledgements

This work was supported by the projects MARRISK, MARFLIX, and DESIMAL partly funded by the Research Council of Norway. The authors would like to thank Endre Vik, Ellen Karoline Norlund, Tore Raa and Frode Helgesen at Statoil for good project cooperation. Thanks go also to four anonymous referees whose comments have helped improve the presentation of this paper.

References


Table 6
What-if scenario analyses.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>CPU voy. gen. (seconds)</th>
<th>CPU voy. model (seconds)</th>
<th>CPU total (seconds)</th>
<th>Gap from 11-3 (%)</th>
<th># Vess.</th>
<th># Voy.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>84.0</td>
<td>39.5</td>
<td>123.5</td>
<td>–0.23</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>(b)</td>
<td>65.6</td>
<td>80.4</td>
<td>146.0</td>
<td>–0.56</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>(c)</td>
<td>27.4</td>
<td>41.5</td>
<td>68.9</td>
<td>–36.00</td>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>


