Reducing emissions through speed optimization in supply vessel operations

Ellen Karoline Norlund a,b,⇑, Irina Gribkovskaia a

a Molde University College, Pb. 2110, N-6402 Molde, Norway
b Statoil ASA, Pb. 7200, N-5020 Bergen, Norway

ARTICLE INFO

Keywords:
Emissions
Supply vessel operations
Ship routing and scheduling
Speed optimization

ABSTRACT

This paper examines how optimizing sailing speeds can reduce supply vessels emissions in the upstream supply chain to offshore installations. We introduce several speed optimization strategies to be used in construction of periodic vessel schedules. The strategies consider vessel waiting times before the start of service at installations and at supply base. Tests carried out on real instances from Statoil’s activities on the Norwegian continental shelf indicate that a 25% emissions and fuel cost reductions can be achieved without fleet size increase.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

In cargo transportation to offshore installations, supply vessels are the main source of air pollution. Here we study reductions in emissions in supply vessel operations by optimizing speed while planning vessel schedules and fleet size for a supply base. Supply vessels are one of the major cost drivers in offshore upstream logistics. Reducing emissions can be achieved through low speed sailing, which results in increased fleet size and higher costs. In this paper we develop speed optimization strategies which reduce fuel consumption and emissions without increasing the number of vessels. We test and compare the strategies on instances from the oil and gas company Statoil, which supplies approximately 50 fixed installations and exploration rigs on the Norwegian continental shelf (NCS) from seven bases on the coast of Norway.

2. Background

The oil and gas industry relies on transport of cargo to and from offshore installations. Careful planning of supply vessel operations is therefore essential. In this planning emissions reductions should be considered. The emissions from supply vessels depend on fuel consumed during sailing and loading/unloading activities.

Supply vessels use diesel or liquefied natural gas (LNG) and produce emissions of CO₂, methane (CH₄), NOₓ, SOₓ, and particles (PM₂.₅). The greenhouse gas CO₂ is determined by the carbon content in the fuel, and is therefore linearly dependent on fuel consumption. LNG has a lower carbon content than diesel and therefore yields reduced CO₂ emissions, but the unburned methane emissions may reduce this gain. NOₓ emissions are dependent on the engine conditions under which the fuel is burned. SOₓ emissions originate from sulphur in the fuel oxidized in the combustion process, and emissions of particles are related to the type of fuel. Emissions of pollutant i can be approximated by:
\[ E_i = \sum_m (FC_m \cdot EF_{i,m}). \]  

where \( E_i \) is the emission of pollutant \( i \) in kilograms or tons, \( FC_m \) is the mass of fuel type \( m \), and \( EF_{i,m} \) is the emission factor for pollutant \( i \) and fuel type \( m \) (European Environment Agency, 2009).

Eq. (1) is used to estimate the annual emissions for a supply vessel in Statoil’s activities on the NCS. Table 1 shows emissions for a supply vessel using either marine gas oil (MGO) or LNG. The majority of the supply vessels fuelled by MGO have implemented the selective catalytic reduction (SCR) system reducing NO\(_x\) emissions by approximately 83% (Nielsen and Stenersen, 2010). The LNG vessels have dual fuel engines, and diesel is used to ignite the gas. The fuel consumption is calculated as an average for vessels used more than 8000 h in supply activities in 2010. The emission factors are taken from Sandmo (2011) for CO\(_2\) and PM\(_{2.5}\), from Buhaug et al. (2009) for NO\(_x\) and SO\(_x\) emissions, and from Nielsen and Stenersen (2010) for CH\(_4\) and NO\(_x\) for LNG fuelled vessels.

There are two ways of improving the environmental performance of a vessel: design and operations. Offshore supply vessels operating on the NCS are worldwide leaders regarding green ship design improvements. For example the world’s first LNG supply vessels have been in use since 2003. Speed reduction is an operational measure that can reduce the amount of fuel consumed. We examine ways to reduce fuel consumption in supply vessel operations through speed optimization. To this end, we use for fuel consumption per time unit when sailing:

\[ FC(v) = (FC_0 / v_0^3) v^3, \]

where \( FC(v) \) is fuel consumption at sailing speed \( v \), and \( FC_0 \) is fuel consumption at design speed \( v_0 \). In addition, a supply vessel consumes fuel when waiting and performing loading/unloading activities at installations and at the supply base.

For many oil and gas fields on the NCS, the supply base location is licensed, and thus each base mainly services its own set of installations, and supply vessel planning is performed independently for each. The scheduling of vessels from a supply base is done on a weekly basis. Each installation requires one or more visits per week, and the departures to the installations from the supply base should be fairly spread throughout the week. The number of installations served from the supply base and the activity level at the installations vary over time, and hence does the need for supply vessels. Supply vessel planning consists of simultaneously determining the fleet composition and the schedules for the vessels, combined in a weekly schedule. A vessel schedule consists of consecutive vessel voyages. A voyage is defined by the start day and the sequence of installations to visit.

The normal duration of a voyage on the NCS is limited to 2 or 3 days since installations prefer short lead times and frequent deliveries. Vessel departures from the base are planned at a fixed time of the day. To start a voyage on a certain day, the vessel has to be at the base some hours before departure to perform unloading and loading operations. We define voyage time window as the time interval between the vessel departure from the base and the earliest start for unloading operations after return to the base. For example, a 2-day voyage starting at 04:00 pm on Monday has a time window of 40 h, which ends at 8:00 am on Wednesday, assuming unloading/loading at the base takes 8 h. If the vessel returns to the base before the end of the voyage time window, it will incur a waiting time at the base. Installations with drilling operations are usually open for supply services 24 h a day, while installations with only production activities at the NCS are closed at night. Opening hours at installations yield multiple time windows in supply vessel planning; there will be waiting if a vessel arrives at an installation outside opening hours. Voyage waiting time is calculated as the sum of waiting times at installations and at the base.

In supply vessel planning installation cargo demands are assumed to be uniformly distributed among the visits. However, installation service time is dependent on actual demand and current weather conditions. The main factor affecting the duration of offshore loading and unloading operations is wave height. Due to safety regulations, these operations are not allowed if wave height is above five meters, and the vessel must then wait. In bad weather, sailing and service times offshore therefore increase. To avoid delays in scheduled vessel activities, various strategies are used to build robustness in the schedules. One way to handle uncertainty is to include a required slack to the voyage by extending its pre-calculated duration. Particularly, Statoil uses this approach and adds a 4-h slack to the voyage end during winter months.

We incorporate speed optimization into the supply vessel-planning algorithm of Halvorsen-Weare et al. (2012) used by Statoil (Fig. 1). It consists of two stages. In the voyage generation stage, all shortest feasible voyages are generated for each vessel and set of installations. These voyages are then used as an input to an optimization model that determines the vessels and voyages to use in the weekly schedule by assigning voyages to start days. The model is a voyage-based set covering model with side constraints, ensuring that all installations receive their required number of visits, and that the duration of all voyages sailed by a vessel does not exceed the number of days in the planning horizon. There are also base capacity constraints, constraints ensuring that voyages of the same vessel do not overlap, and constraints ensuring that the departures
from the base and the visits to an installation are spread in time. The objective is to minimize the sum of vessel charter costs, the major component, and voyage fuel costs, related to fuel consumption.

Weekly schedules, generated with constant speed, may include waiting for opening hours at installations, waiting for loading and unloading at supply base, and idle time between voyages. Our research is based on the idea that schedule waiting time, consisting of waiting time within voyages and waiting time between voyages, can be used to reduce speed and thus fuel consumption. We develop speed optimization strategies which use the schedule waiting time in different ways. One way is to define a constant optimized voyage speed when generating the weekly schedule with the optimization model by making use of inter-voyage waiting time. Another way is to compute the optimized speed on each voyage leg during the generation of voyages by using the intra-voyage waiting time. To generate voyages with our speed strategies, we optimize the speed on shortest feasible voyages initially constructed with a constant speed. Fuel consumption reduction on speed optimized voyages is achieved through a decrease in sailing speed and in waiting time. To achieve a weekly schedule with low emissions and minimal number of vessels, we use a cost minimization objective including charter and fuel costs in the optimization model. Emissions are minimized by minimizing fuel costs since fuel costs are linearly related to fuel consumption.

3. Speed strategies

Four speed optimization strategies are developed to construct voyages with optimized speed. The initial step of all strategies is the generation of feasible shortest voyages with design speed \( v_d \) for each vessel. It is assumed that each vessel has a speed interval between a minimum vessel speed, \( v_{\text{min}} \) and a maximum speed, \( v_{\text{max}} \). In the D strategy the optimized speed is found by iterative speed decrease for all voyages. Other algorithms optimize speed on voyage legs by using the voyage waiting time. The W strategy reduces speed on legs with waiting time. In the S strategy speed is sequentially optimized, while in the R strategy it is done recursively.

The D strategy is developed for a homogenous fleet since supply vessels are relatively uniform. However, it can be modified for a heterogeneous fleet. It is assumed that the same speed \( v \) is used on all legs of all voyages. Under this strategy, the weekly schedules are iteratively constructed by decreasing speed \( v \) from \( v_d \) by a constant value \( D_v \). The iteration process stops when a schedule with an increase in the fleet size is generated or when the minimum vessel speed \( v_{\text{min}} \) is reached. Finally, the weekly schedule with the speed yielding the lowest fuel consumption is chosen. The pseudo-code for the D strategy is select \( v \) yielding schedule with the lowest fuel consumption:

```plaintext
for \( v \) from \( v_d \) to \( v_{\text{min}} \)
    generate schedule with \( v \)
    if the fleet size increases then
        stop to iterate
    end if
    set \( v = v - \Delta v \)
end for
```

The effects of iterative speed decrease on fuel consumption, fleet size and number of voyages in weekly schedule are illustrated in Fig. 2. The top element shows an example where speed reduction leads to an increased fleet size, and bottom illustrates that fuel consumption does not always decrease with reduced speed since the number of voyages may increase. This explains why the algorithm may require the generation of all schedules with a discrete speed value in the interval \([v_{\text{min}}, v_d]\).

In the W strategy we sequentially look at each leg of an initial voyage, and reduce speed only on legs with waiting time for opening hours at the installations, or with waiting time for the end of the voyage time window. The speed reduction may change the arrival times, but does not change the start time of service at the installations and the voyage time window.
Assuming, \( v_{ij} \) is the optimized speed on leg \((i,j)\), \( d_{ij} \) is the leg’s sailing distance, \( t_{ij}^w \) is the leg’s waiting time, and \( t_{ij} \) is the sum of the leg’s sailing and waiting time, the pseudo-code for the W strategy is:

\[
\begin{align*}
\text{for each leg (i,j) do} & \\
\quad \text{if } t_{ij}^w > 0 & \text{ then} \\
\quad \quad \text{set } v_{ij} = d_{ij}/t_{ij} \\
\quad \quad \text{if } v_{ij} < v_{\min} & \text{ then} \\
\quad \quad \quad \text{set } v_{ij} = v_{\min} \\
\quad \text{end if} & \\
\quad \text{end if} & \\
\text{end for}
\end{align*}
\]

The S strategy optimizes speed sequentially on each voyage leg, rescheduling arrival times at the installations and the arrival back to the supply base. The minimum voyage speed is first calculated by dividing the voyage sailing distance by the sum of the voyage sailing and waiting times. Then, on each voyage leg, the installation arrival time is calculated from the previous installation departure time and the minimum voyage speed. If the vessel arrives at an installation before it opens, we decrease the speed. If the arrival is too late to complete the service within the opening hours, we increase the speed to reach the time window. If the modified speed is not within the vessel speed interval, it is set to the minimum vessel speed. Further, the new arrival time with the optimized speed is calculated. If the arrival to an installation is outside the time window, waiting will occur. In some cases, the vessel may have to wait for the next time window, thus increasing the voyage time window by 24 h.

For the pseudo-code for the S strategy we denote the initial voyage sailing distance as \( d \) and the sum of voyage sailing and waiting times as \( t \). The minimum voyage speed is denoted by \( v_{\min} \), and \( v_{ij} \) is the optimized speed on voyage leg \((i,j)\). The parameter \( t_i^l \) is the service duration for loading and unloading at installation \( i \). The daily time window \([e_i, l_i]\) for installation \( i \) is defined by the opening time \( e_i \) for service, and the time \( l_i \) when the service closes. We denote by \( t_i^e \) and \( t_i^d \) the time elapsed from the midnight the day the voyage starts to the arrival to and to the departure from installation \( i \). The arrival time at installation \( i \) is calculated as \( t_i^e \mod 24 \). For the first leg, \( t_1^d \) is the start of the voyage time window. For the last leg, \( t_{n+1}^d = 0 \), and \( e_n \) and \( l_n \) are set to the end of the voyage time window:
Set \( v^r = d/t \)

if \( v^r < v_{\text{min}} \) then
  set \( v^r = v_{\text{min}} \)
end if

for each leg \((i,j)\) do
  set \( t_{ja} = t_{ja} + d_{ij}/v_{ij} \)
  if \( t_{ja} \mod 24 < e_j \) then
    set \( v_{ij} = d_{ij}/(e_j - t_{ja}^i) \)
  else if \( t_{ja} \mod 24 > l_j - t_{j}^i \) then
    set \( v_{ij} = d_{ij}/(l_j - t_{j}^i - t_{ja}^i) \)
  end if
  if \( v_{ij} < v_{\text{min}} \) or \( v_{ij} > v_{\text{max}} \) then
    set \( v_{ij} = v_{\text{min}} \)
  end if
  set \( t_{ja} = t_{ja} + d_{ij}/v_{ij} \)
  if \( t_{ja} \mod 24 < e_j \) then
    set \( t_{ja} = t_{ja} + e_j - t_{ja} \mod 24 + t_{ja} \)
  else if \( t_{ja} \mod 24 > l_j - t_{j} \) then
    set \( t_{ja} = t_{ja} + e_j + 24 - t_{ja} \mod 24 + t_{ja} \)
  else
    set \( t_{ja} = t_{ja} + t_{ja} \)
  end if
end for

For some instances, the S strategy may generate voyages with long waiting times, yielding a schedule with high fuel consumption. To avoid this, both the initial and the speed optimized feasible voyage with extended duration in days are used as inputs to the optimization model.

The R strategy uses the idea of the recursive smoothing algorithm of Norstad et al. (2010), defining optimal speed on a route given as a path with a single time window and zero service time at each node (Hvattum et al., 2013). It is not possible to apply the algorithm directly on a supply vessel voyage since the installations have multiple time windows, the return time to the supply base is not given a priori, and service times at installations need to be considered. To optimize speed recursively as in Norstad et al. (2010), we set time windows for installations closed at night to the day of visit in the initial voyage. The time window for the other installations and the voyage itself are set to the initial voyage time window, which guarantees that the R strategy will always generate a feasible voyage.

In the R strategy the minimum voyage speed \( v^r \) is calculated as the voyage sailing distance divided by the sum of voyage sailing and waiting times, as in the S strategy. Then, in the first iteration, the minimum voyage speed is used on all legs of the voyage, and the installation \( k \) with the largest time window violation is identified. The arrival time at this installation is adjusted to the nearest feasible value, and the voyage is then split into two partial voyages. For each partial voyage the same procedure is repeated until there are no time window violations.

For the partial voyage from \( a \) to \( b \), we denote by \( d_{ab} \) the sailing distance, and by \( t_{ab} \) the partial voyage time window duration minus the sum of service times. Further, \( \delta \) is the largest time window violation along the partial voyage. The remaining notation is similar as in the S strategy, beside that the start and the end of the time window for each installation and the voyage are measured in hours from the midnight the day the voyage starts. Before the algorithm is called, \( a \) and \( b \) are set equal to the start and the end of the initial voyage time window, and are then updated after each split for both partial voyages:

set \( \delta = 0 \)
set \( k = a \)
set \( t^r = d_{ab}/t_{ab} \)
if \( t^r < v_{\text{min}} \) then
  set \( t^r = v_{\text{min}} \)
end if

for each leg \((i,j)\) of partial voyage from \( a \) to \( b \) do
  set \( t_{ij} = t^r \)
(continued on next page)
The R strategy attempts to optimize speed by fully using the voyage waiting time. However, if the minimum partial voyage speed \( v \) has a lower value than \( v_{\text{min}} \) there will be waiting at the end of the partial voyage.

We illustrate how to optimize speed on legs with the W, S and R strategies on the 2-day voyage shown in Fig. 3. Here, node 0 is the supply base and the other nodes are the installations. Distances are shown in nautical miles on the arcs, and service times in hours are shown at the nodes. For installations closed at night, the opening hours are shown in square brackets next to the nodes. The installations are visited in a sequence yielding the shortest voyage duration calculated with a design speed \( v_d \) of 12 knots. The minimum speed \( v_{\text{min}} \) and maximum speed \( v_{\text{max}} \) are set to 10 knots and 14 knots.

Fig. 4 provides speeds and times for operations on each voyage leg obtained under the W, S and R strategies. The double vertical lines define the voyage time window from 4:00 pm the start day to 8:00 am, 40 h later. The brackets define the opening hours for installations closed at night. The medium grey slots represent the sailing times with corresponding speeds. The light grey slots show the service times at the corresponding installations, and the waiting times are shown in dark grey. The first line illustrates the initial voyage generated with the design speed. This voyage has waiting time at installation two and at supply base.

The second line illustrates the voyage generated with the W strategy. Here, the speed is reduced from 12 to the minimum allowed speed of 10 knots on legs with waiting time.

The speeds optimized with the S strategy are shown in line S. As the calculated minimum voyage speed of 9.7 knots is less than the minimum allowed, it is set to 10 knots. With this speed the vessel will arrive at installation three too late to complete service within the opening hours. To serve installation three before it closes, the speed on the corresponding leg must be increased to 12.5 knots.
The speeds computed with the recursive algorithm are shown in line R. Since the minimum voyage speed of 10 knots results in a time window violation for installation three, the initial voyage is split into two partial voyages: from the base to installation three, and from the installation to the base. After setting arrival time to the installation three to 5:00 pm, the calculated minimum voyage speed of 10.7 knots is used on all legs of the first partial voyage. As the start of the other partial voyage is at 7:00 pm, the calculated voyage speed of 8.3 knots is adjusted to the minimum allowed speed of 10 knots, resulting in waiting time at the base.

4. Experiments

The four speed optimization strategies were tested on several instances from a real supply vessel-planning problem faced by Statoil.¹

4.1. Test instances and test description

We have created 16 test instances derived from real supply vessel operations data from one of Statoil’s supply bases on the NCS. The small instances involve five installations and nine visits per week, the medium instances have seven installations and 15 visits, and the major instances include 10 installations and 25 visits. We have generated instances of each size with voyage duration limited to 2 and 3 days, with and without 4-h voyage slack requirements, and four installations closed at night. To test how fuel consumption reduction depends on the number of installations closed at night, we have generated five similar major instances that differ in this number. The instances are described by four numbers. The first is the number of installations. The second number is the maximum duration in days for voyages departing from the supply base from Monday to Thursday. Voyages starting other days can last for 2 or 3 days in all instances. The third number is the required voyage slack in hours. The last is the number of installations closed at night. For example, in instance 10–2–4–4 there are ten installations, only 2-day voyages are allowed to depart from the base from Monday to Thursday, the required voyage slack is 4 h, and four installations are closed at night. In all test instances, three identical vessels are available with a design speed of 12 knots. Vessel capacity, fuel consumption and cost data are provided from one of the supply vessels used by Statoil.

To compare the speed optimization strategies, we first construct a weekly schedule for each instance with design speed. This weekly schedule is called a basic solution. Then, for the weekly schedule generated with each strategy, the fuel consumption is compared with that of the basic solution. The fuel consumption for each voyage is calculated as the sum of fuel consumption when sailing, waiting and loading/unloading at the installations. The fuel consumption per hour of waiting and loading/unloading at the supply base is set to zero to compare fuel consumption when vessels are offshore. Speed optimization, however, decreases waiting time spent in port, and hence emissions there as a side effect.

In our tests, the minimum and maximum allowed speeds are set to 10 and 14 knots. In the D strategy we decrease speed by one knot in each iteration step.

4.2. Results

Table 2 shows for each instance the percentage reduction in fuel consumption for all strategies, together with the percentage of waiting time and the number of vessels in basic solution. The waiting time and the number of vessels in the basic solution vary from instance to instance due to the underlying structure of the vessel planning problem, including installations’ and slack requirements. The largest fuel consumption reduction for each instance is shown in bold. The reduction potential is limited by the minimum allowed speed of 10 knots.

¹ We implemented the speed optimization algorithms using Visual Studio 2008 in the voyage generator written in C++. The optimization model written in Xpress-IVE 1.19.01 with Xpress Mosel 2.4.0 is solved with Xpress-Optimizer 19.00.00. All tests were run on a 2.19 GHz Intel Core 2 Duo PC with 1.99 GB RAM.
The fuel consumption reduction achieved with the D strategy varies from zero to 25%. The performance of this strategy is dependent on the waiting time in basic solution, illustrated by the best results on instances with a large amount of waiting time. However, even for such instances the reduction may be small due to little inter-voyage waiting time (as for 10–3–0–6). Moreover, for instances with little waiting time (as 10–2–4–4) a reduction may not be possible because an equal decrease of speed on all voyages may result in a fleet size increase. The application of this strategy may be time consuming since it requires several runs of the optimization model. The W strategy yields on average the smallest reduction in fuel consumption, from eight to 18%, since this strategy only optimizes speed on legs with waiting time. However, the W strategy is the easiest to implement and yields good results on instances with a large number of installations closed at night. The S strategy performs better than the W strategy since it optimizes speed on all legs. At the same time, its performance is dependent on the waiting time in the basic solution as for the D strategy. In cases of major instances with little waiting time, the S strategy outperforms the D strategy because it uses both intra- and inter-voyage waiting times. The overall best performance is achieved with the R strategy, yielding a 17 to 25% reduction in fuel consumption. This strategy yields the best utilization of intra-voyage waiting time, and produces good results even for instances with little waiting time.

We have also applied the W, S and R strategies a posteriori to compare them with an a priori approach. While in the a priori approach the strategies are applied for generation of all voyages that are input to the optimization model, in the a posteriori approach speed is optimized only on voyages from the weekly schedule in basic solution. The percentage reductions in fuel consumption obtained with both approaches are shown in Table 3 for small and medium size instances. The W strategy yields similar results with both approaches since the difference in fuel consumption for a speed optimized and a non-speed optimized voyage is not substantial. The S strategy performed worse than the W strategy because it uses both intra- and inter-voyage waiting times. The overall best performance is achieved with the R strategy, yielding a 17 to 25% reduction in fuel consumption. This strategy yields the best utilization of intra-voyage waiting time, and produces good results even for instances with little waiting time.

We have also applied the W, S and R strategies a posteriori to compare them with an a priori approach. While in the a priori approach the strategies are applied for generation of all voyages that are input to the optimization model, in the a posteriori approach speed is optimized only on voyages from the weekly schedule in basic solution. The percentage reductions in fuel consumption obtained with both approaches are shown in Table 3 for small and medium size instances. The W strategy yields similar results with both approaches since the difference in fuel consumption for a speed optimized and a non-speed optimized voyage is not substantial. The S strategy performed worse than the W strategy because it uses both intra- and inter-voyage waiting times. The overall best performance is achieved with the R strategy, yielding a 17 to 25% reduction in fuel consumption. This strategy yields the best utilization of intra-voyage waiting time, and produces good results even for instances with little waiting time.

5. Conclusions

We have studied how to reduce emissions from supply vessel operations by incorporating speed optimization in the supply vessel planning. The speed optimization strategies use inter- and intra-voyage waiting time during voyage generation.
and construction of a weekly schedule. The strategy of reducing speed on legs with waiting time is easy to implement, but it yields the smallest reduction in fuel consumption. The strategy of speed decrease for all voyages and the sequential speed optimization strategy applied a priori yield good results for instances with long waiting time in the schedules. The recursive speed optimization strategy provides the overall best results applied both a priori and a posteriori. We expect the schedules generated with optimized speed to be robust since they can adapt to stochasticity in weather and demand requirements through speed adjustments.

The tests of speed strategies conducted on the Statoil instances show savings of up to 25% in fuel consumption without fleet size increase. In current operations, Statoil applies a posteriori the strategy of speed reduction on legs with waiting time. With an average of 10% in fuel consumption reduction, as we have obtained in our tests, this strategy results in an annual reduction of 900 tons of CO₂ emissions for a single vessel.

References


