Robust Ship Scheduling with Multiple Time Windows

Marielle Christiansen,[1] Kjetil Fagerholt[2]

[1] *Section of Operations Research, Norwegian University of Science and Technology, Trondheim, Norway*

[2] *MARINTEK and Department of Marine Systems Design, Norwegian University of Science and Technology, Trondheim, Norway*

Received 19 September 2000; revised 5 February 2002; accepted 7 February 2002

Abstract: We present a ship scheduling problem concerned with the pickup and delivery of bulk cargoes within given time windows. As the ports are closed for service at night and during weekends, the wide time windows can be regarded as multiple time windows. Another issue is that the loading/discharging times of cargoes may take several days. This means that a ship will stay idle much of the time in port, and the total time at port will depend on the ship's arrival time. Ship scheduling is associated with uncertainty due to bad weather at sea and unpredictable service times in ports. Our objective is to make robust schedules that are less likely to result in ships staying idle in ports during the weekend, and impose penalty costs for arrivals at risky times (i.e., close to weekends). A set partitioning approach is proposed to solve the problem. The columns correspond to feasible ship schedules that are found *a priori.* They are generated taking the uncertainty and multiple time windows into account. The computational results show that we can increase the robustness of the schedules at the sacrifice of increased transportation costs.
 Naval Research Logistics 49: 611–625, 2002; Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/nav.10033

1. INTRODUCTION

Ocean shipping is already the major transportation mode for international trade, and the trend goes towards an increase in the use of ships. Often, the shipping industry is divided into deep sea and short sea shipping, where deep sea is shipping between continents and short sea shipping takes place within a continent. The shipping industry has a monopoly regarding transportation between continents for large volumes. This activity will probably increase in the future with the continuous growth in the world population, combined with product specialization and depletion of local resources. With an increase in deep sea activities we need feeder systems for short sea shipping, so that the latter is expected to increase as well. In addition, we will probably see an increase in the area of short sea shipping, due to heavily congested road networks and air corridors.

In this growth period, the shipping industry will meet new challenges. Efficient operation of the whole supply chain from the supplier to the end customer will be emphasized both in time

Correspondence to: M. Christiansen.

and costs. A key challenge in this respect is the construction of efficient routes and schedules for the ships in a particular fleet. This is also important due to ever-increasing competition between shipping companies, where the profit margins are squeezed to a minimum. A ship involves a major capital investment, and its daily costs can amount to thousands of dollars. Therefore, significant savings can be expected if routing and scheduling are optimal.

Transportation routing and scheduling have been widely discussed in the literature; see the bibliography by Laporte and Osman [25] containing 500 references on routing problems. However, relatively few contributions exist for ship routing and scheduling. Ronen [28] gives several reasons for the low level of attention to ship routing and scheduling. He states that one essential explanation is the long tradition of conservative thinking in the ocean shipping industry, making it unreceptive to new ideas such as the use of optimization models.

Fortunately, much of the research within vehicle routing can be adjusted to solving ship routing problems. However, the aim of this paper is to highlight two important differences between vehicle and ship planning problems, and show how these differences can be taken into account when searching for efficient routes and schedules. This work is related to a real planning problem within short sea shipping.

Many ports around the world have restricted operating hours. These ports are typically closed for service at night and during weekends. The *service,* either loading or discharging, of a ship may take up to several days. This means that the ship will stay idle much of the time in port. This balance between service time and the port's operating hours is considerably different from a common vehicle routing situation, but is normal at least in some sectors in the shipping industry.

For ship schedule planners, it is of great importance to minimize the idle time in port especially over the weekends. However, in practice this is often a difficult task. Ship scheduling is associated with a high degree of uncertainty, mainly due to bad weather at sea and unpredictable service times in ports. Assume a schedule where a ship is estimated to finish service at a port two hours before closing time on a Friday. This may be considered a nonrobust or "risky" schedule, as there is a relatively high probability of waiting in port until Monday morning. This may again create delays for the subsequent port calls in the planned schedule, perhaps with customer claims as a result. Therefore, an experienced planner would try to find alternative schedules that are more robust with respect to delays due to the restricted operating hours.

This paper considers a real ship scheduling problem involving the design of a set of robust schedules for a fleet of ships servicing dry bulk cargoes during the planning period. Each cargo consists of a pickup at a given port of loading and delivery at the corresponding port of discharge. Most of these ports have restricted operating hours. Both loading and discharging of the cargo must be started within specified cargo time windows. These features give a problem similar to the multi-vehicle Pickup and Delivery Problem with Time Windows (m-PDPTW), which is treated in [9] and [11]. Since the ports have restricted operating hours, wide time windows can actually be thought of as multiple time windows. As the ship may stay idle part of the time in port, the total service time in ports is regarded as a function of the arrival time of the ship. It is possible to impose a penalty cost for risky arrival times within the time windows to handle the robustness aspect of the ship schedules. The penalty cost increases as the planned end of service gets closer to the end of the operating hours for the particular port. Depending on the size of the penalty cost, the optimal solution will prefer port arrivals at times that are considered to be robust.

This problem is solved by an approach based on a set partitioning formulation. First, all feasible schedules are generated *a priori* for each ship in the fleet together with their operating and penalty costs. Then the scheduling problem is formulated and solved as a set partitioning problem where the columns correspond to the schedules generated a priori.

In Section 2, we compare the ship scheduling problem presented to related scheduling problems in the literature. Section 3 discusses the ship planning problem with focus on robust schedules and multiple time windows, and shows how these issues can be considered in a mathematical framework. In Section 4, we describe the solution approach. The mentioned foci are incorporated in the schedule generation part of the algorithm. Therefore, this part of the approach is focused. Computational results from a real ship scheduling problem are given in Section 5. Finally, some concluding remarks follow.

2. COMPARISONS TO OTHER RELATED SCHEDULING PROBLEMS

In the literature, there are various attempts to consider stochastic conditions within deterministic routing models. One such routing problem is the Vehicle Routing Problem with Soft Time Windows (VRPSTW); see, for instance, Balakrishnan [1], Koskosidis, Powell, and Solomon [23], and Taillard et al. [31]. In order to make the model of the vehicle routing problem with time windows less sensitive, the VRPSTW allows solutions with arrival times outside the time windows at an appropriate penalty cost. This penalty cost may reflect the cost of lost sales, goodwill, etc. due to the customer inconvenience of not meeting the time window. Therefore, the hard time window widths are extended to soft time window widths. This means that the feasible region of the problem increases. Depending on the size of the penalty costs, these costs will drive the solution towards the hard time window intervals.

We can also find a few contributions within ship scheduling where penalty costs are used in connection with time windows. In Fagerholt [15], the hard time windows are extended to soft ones like that of the VRPSTW. There, the penalty costs occur outside the hard time windows. Christiansen [6] describes a column generation approach for a combined ship routing and inventory management problem. The transported product is consumed in some port factories and produced in others. At all factories there exist hard inventory limits for the transported product. Christiansen and Nygreen [7] introduce another pair of soft inventory limits within the hard inventory limits to the same real planning problem to reduce the possibility of violating the inventory limits at the port factories. This means that those soft inventory limits can be violated at a penalty, but it is not possible to exceed the stock capacity or be under the lower inventory limit. They show that the soft inventory constraints can be transformed into soft time windows.

The VRPSTW has many similarities to our ship planning problem. However, we are not introducing soft time windows in addition to the hard time windows to take the uncertainty aspects into account. Instead, the costs and the service times are dependent on the arrival time within the hard time windows. The literature has examples of routing models where some time for the start of service is preferred. Ioachim et al. [22] propose an optimal dynamic programming algorithm for the shortest path problem with time windows and linear node costs. This problem arises as a subproblem when using decomposition approaches for solving for instance various aircraft scheduling problems. Sexton and Bodin [29] and Sexton and Choi [30] present polynomial algorithms for determining optimal schedules in the presence of convex time-dependent cost functions. Sexton and Bodin [29] only consider a latest arrival time rather than a time window, and the time-dependent cost function is linear. Based on a mathematical programming formulation of their problem, they use a Benders decomposition procedure. The result is a heuristic routing and scheduling algorithm. In Sexton and Choi [30], they do not consider any hard restrictions on the arrival time, but they impose a linear cost for services outside the desired time window. For fixed vehicle routes, Dumas, Soumis, and Desrosiers [12] present an algorithm where the time windows are hard, but convex time-dependent cost functions are imposed within the time windows.

The reviewed contributions, as well as the one studied here, approach the stochastic conditions by deterministic models. However, we can also find that uncertainty within routing problems has been taken into account by use of stochastic models (see the survey paper by Gendreau, Laporte, and Séguin [21]).

To our knowledge, there are no contributions where the balance between service time and operating hours for a port is focused on in the literature of routing and scheduling. However, in many short sea shipping operations the planners consider the minimization of the port idle time to be a highly preferential issue. This may also be an important issue in the scheduling of supply vessels (see Fagerholt and Lindstad [18]). In this case, a number of offshore installations are serviced from an onshore depot by a fleet of supply vessels. Some of these offshore installations are closed for service at night. It is therefore important to design schedules that have minimum probability of waiting at an installation during the night. In [18], the robustness aspects are treated by *a posteriori* manual adjustment of the model solutions.

To solve the problem, we have used a set partitioning approach. This approach has been widely used to optimally solve various ship scheduling problems with time windows, see for instance Bausch, Brown, and Ronen [2], Brown, Goodman, and Wood [3, 4], Darby-Dowman et al. [8], Fagerholt and Christiansen [16], and Fisher and Rosenwein [19]. However, none of these consider the robustness aspect of the schedules and the particular operating time structure in the ports.

3. PROBLEM DESCRIPTION

The studied ship scheduling problem corresponds to the pickup and delivery of bulk cargoes at minimum cost. Each cargo consists of a given quantity to be picked up at a given loading port and delivered to a corresponding port of discharge. Time windows are imposed for both the pickup and delivery of the cargoes. For this purpose, the company operates a heterogeneous fleet of ships with different cost structures, load capacities, and specific ship characteristics. Due to seasonal variations in demand, the fleet is not designed to handle all cargoes throughout the year. Some of the cargoes may therefore be serviced by spot carriers at a given market cost.

The cargo quantities and ship capacities are such that the ships may carry several cargoes simultaneously. This means that a new loading port can still be visited with some cargoes on board. This makes the problem more complex than many of the ship scheduling problems reported in the literature.

In the mathematical description of the problem each cargo is represented by the index i. Associated with the pickup port of cargo i, there is a node i, and with the corresponding delivery port a node $n + i$, where n is the number of cargoes to be serviced during the planning period. Note that different nodes may correspond to the same physical port. Let $N_P = \{1, \ldots, n\}$ be the set of pickup nodes and $N_D = \{n + 1, \ldots, 2n\}$ be the set of delivery nodes, and define $N = N_P \cup N_D$. Further, let V be the set of ships in the fleet indexed by v.

3.1. The Cost Structure

Time charter (TC)-costs for covering the capital costs exist for all ships. In the short term, it is of no interest to plan a change of the fleet size. So the TC-costs have no influence on the planning of optimal routes and schedules and do not need to be considered in the optimization phase.

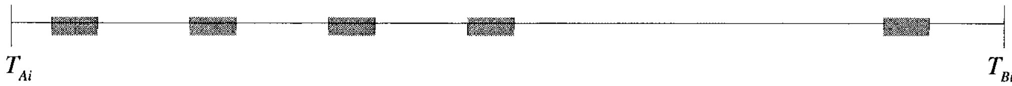

Figure 1. Time window construction for the pickup node for cargo i.

The operating costs consist of sailing costs (fuel and lube) and port service costs. The port service costs depend mainly on the size of the ship arriving at the port, the cargo quantity to be loaded or discharged, and the actual port visited. It should be noted that the costs of a given port visit are not dependent on the total time spent in port, but on the *effective* service time, i.e., the time the ship uses the loading/discharging facilities at the port.

In addition, we are concerned with the cost for the cargoes serviced by spot carriers.

3.2. The Time Window Structure

For this planning problem, the ports have restricted operating hours. Let us denote the time window for the pickup node corresponding to cargo i as $[T_{Ai}, T_{Bi}]$ and call this an *outer time window.* The outer time window for the corresponding delivery node is $[T_{A(n+i)}, T_{B(n+i)}]$. Since these time window widths are normally much wider than the operating hours during the day, we get a ship planning problem with multiple time windows within the outer time window. Figure 1 shows an example of a time window construction for the pickup node corresponding to cargo i, and the grey shaded areas show the operating hours each day.

To our knowledge, the introduction of multiple time windows has scarcely been discussed in the literature, though it is interesting both from a theoretical and practical point of view. Desrosiers et al. [9] give some short comments on this issue in the context of their column generation approach for vehicle routing problems. They suggest considering the multiple time window aspects in the generation of the columns. Pesant et al. [26] adapt a constrained programming algorithm for the traveling salesman problem with time windows so that it can handle multiple time windows. Frizzell and Giffin [20] consider a variation of the vehicle routing problem in which the customers have up to two time windows. They use a construction heuristic based on a dynamic urgency classification of customers followed by simple node-exchange improvement heuristics.

Since the loading/discharging time in our ship planning problem normally exceeds the daily operating hours of the port, we have kept the outer time windows and defined a function for the total time in a port, called the *service time.* This service time function, $T_{SERVi}(t_i)$, is dependent on the arrival time of the ship carrying cargo i. Here, we define the arrival time, t_i, as the time when the ship is ready to start the loading or discharging of cargo i. The actual service may start later due to nonoperating hours at night and during the weekend. Each cargo i has its own service time function depending on the size of the cargo and the loading/discharging capacity for the actual port. This function may also be ship dependent if the ship is the restricting factor for the loading/discharging capacity. If two cargoes, i and j, are carried by the same ship and have to be loaded at the same physical port, the loading starts at t_i for the first cargo i whereas the service of cargo j, t_j, cannot start before $t_i + T_{SERVi}(t_i)$, $t_j \geq t_i + T_{SERVi}(t_i)$, given that i has to be loaded before j.

3.2.1. Numerical Example

In Figure 2, we give the service time function, $T_{SERVi}(t_i)$, for a given cargo i that is going to be loaded at a specified port. The outer time window starts on a Sunday at 24:00 and ends on

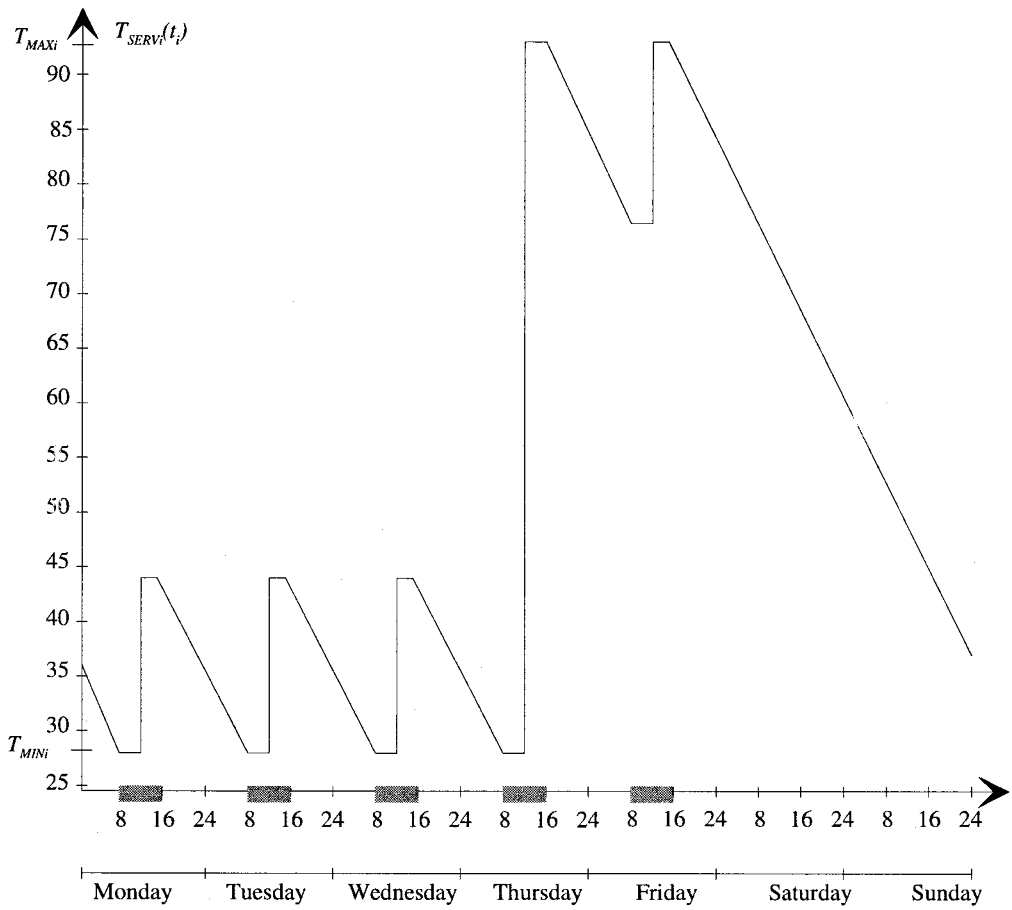

Figure 2. A service time function for cargo i with an outer time window of 1 week.

the next Sunday at 24:00. The operating hours for the port are between 8:00 and 16:00 from Monday to Friday. For this port, the expected loading time for cargo i is 12 hours. From the figure, we see that the total time in port varies between 28 and 92 hours, depending on the arrival time. $T_{MINi}(=28)$ is the minimum service time at port, while $T_{MAXi}(=92)$ is the maximum service time in port and includes a lot of idle time.

If a ship arrives on Monday morning at 8:00, the ship will be loading for 8 hours the first day and 4 hours the next. This gives a total of 28 hours in port.

In the other extreme situation, the ship arrives at 14:00 on Thursday. It loads for 2 hours that day, stays idle for 16 hours during Thursday evening and night and continues loading on Friday morning. Then it loads for 8 hours, but the ship does not finish its loading before the start of the weekend. It has to continue loading on Monday morning at 8:00 and finishes at 10:00. This means that the ship stays idle in port for 64 hours during the weekend. This gives a total of 92 hours in port. □

3.3. Robust Ship Schedules

The uncertainty at sea and ports makes it unfavorable to plan the completion of service in port at the end of the operating hours for the port and especially close to the weekends. In such a case, the reality is perhaps another night or even a whole weekend idle in the port instead of sailing to the next port. The consequences of not finishing the service just before the weekend are much larger than on weekdays, so in this paper we limit ourselves to try to avoid planned completion just before the weekends. A schedule containing such completion may be considered a nonrobust or "risky" schedule. In Section 3.3.1, we define the term risky arrivals, and Section 3.3.2 describes the introduction of penalty costs to avoid risky arrivals.

3.3.1. Risky Arrivals

For a ship loading or discharging cargo i, let us define T_{RENDi} as the arrival time of the ship resulting in a scheduled departure time equal to the weekend closing time for the corresponding port. This arrival time and the arrival times close up to T_{RENDi} are regarded as risky. So, in addition, we define an arrival time for cargo i, T_{RBEGi}, representing the first arrival time that is considered as risky. The scheduled arrivals in this interval, $[T_{RBEGi}, T_{RENDi}]$, are called *risky arrivals.* This interval can be defined in view of the planner's risk attitude, the uncertainty level, and the planning horizon of the actual shipping problem. The length of the risky arrival time interval is denoted by $T_{RISKi} = T_{RENDi} - T_{RBEGi}$. If the scheduled arrival time of a ship at a port is T_{RBEGi}, the ship may in reality be delayed up to T_{RISKi} hours and still finish the port service before the weekend starts. As the scheduled arrival time gets closer to T_{RENDi}, the schedule gets more risky.

3.3.2. Penalty Cost Function

In order to find alternative schedules that are more robust with respect to delays due to the restricted port operating hours, we impose an *artificial model cost* for risky arrivals. The model costs can be regarded as penalties by arriving at a port at risky times. These costs are therefore called *penalty costs.* The penalty costs increase as the arrival time gets closer to T_{RENDi}. Hence, this cost function is modeled as an increasing function of arrival time in the risky arrival time interval; $[T_{RBEGi}, T_{RENDi}]$. The solution method that is developed in Section 4 can handle linear as well as nonlinear penalty cost functions. For real planning problems of this type one would expect that the penalty costs grow faster as the weekend closing port time is approached since the probability of the delay happening is much higher. The penalty cost function should reflect the total costs that occur if the ship stays idle in port. In reality, the costs are not just related to the actual cargo at that port but to the whole activity of the ship in the remaining planning period. However, these penalty costs are not possible to predict. We do not aim to develop the correct penalty cost function, rather a function that gives good plans according to the risk profile of the users. In our approach the penalty cost function is user-specified as the calibration of an appropriate form of the function has to be done individually for the actual planning problem. Normally, the process of developing a suitable penalty cost function takes some time.

Both the form and the gradient of the penalty function reflect the uncertainty level and the risk attitude of the planner, together with the length of the risky arrival time interval, T_{RISKi}. A high value of the gradient means that the planner wants to drive the solution away from the risky arrival time interval to a large extent.

In order to make the penalty cost function more general and flexible, we may include a penalty factor, $P \in [0, 1]$, to the penalty cost function, such that the planner can easily choose how to weight the uncertainty aspects from run to run.

Both the penalty factor P and the risk interval length $T_{RISKi} = T_{RENDi} - T_{RBEGi}$ have to be set by the planner depending on the actual situation. It is natural to have larger values on these parameters for a problem run in the winter than in the summer, because it is more important to drive the optimal solution away from the most risky arrivals in the winter with more unpredictable weather conditions. In general, it is expected that with high values on these parameters, the optimal solution will contain less risky arrivals while the operating costs will increase, and vice versa.

We have presented the penalty parameters, P and T_{RISKi}, independently of the physical ports and ships. These could have been set individually for each combination of ship and port. For instance, it would be natural to increase the values of these parameters for a port with bad reputation or an unstable and old ship.

Numerical example continues: For simplicity, we model the penalty cost function as a linear function of arrival time in the risky arrival time interval; $[T_{RBEGi}, T_{RENDi}]$. Within $[T_{RBEGi}, T_{RENDi}]$ the penalty costs increase from 0 up to a maximum value, R_{MAXiv}, when $P = 1$.

This penalty cost function, $C_{Piv}(t_i)$, becomes

$$C_{Piv}(t_i) = \begin{cases} P \cdot R_{MAXiv} \frac{t_i - T_{RBEGi}}{T_{RISKi}}, & t_i \in [T_{RBEGi}, T_{RENDi}], \\ 0, & t_i \in [T_{Ai}, T_{Bi}] \backslash [T_{RBEGi}, T_{RENDi}], \end{cases} \qquad \forall i \in N, \quad v \in V. \tag{1}$$

Here, we define an arrival for the loading of cargo i as risky if the scheduled arrival time is 24:00 on Wednesday (T_{RBEGi}) or closer. The latest arrival time within the time window is 12:00 on Thursday; T_{RENDi}. This means that the risky arrival time interval, T_{RISKi}, is 12 hours.

In reality, we have more flexibility than is modeled. In the model, we operate with fixed average speed. For a given practical situation, the ship's master may, to some extent, increase this speed to reach a port in time at an additional cost. This issue is not explicitly modeled, but the introduction of penalty costs can approximately cover the increased costs in order to finish a port service before a weekend starts.

In this model, we have assumed no use of overtime for the port workers. In some ports, it is possible to negotiate with the port workers about extending the operating hours. This aspect can easily be incorporated into the model by extending the use of penalty functions in overtime intervals.

Some of the cargoes may be serviced by spot carriers. These spot cargoes are assumed to be serviced in nonrisky time intervals, i.e., at no penalty cost.

4. SOLUTION APPROACH FOR THE SHIP SCHEDULING PROBLEM

The objective of this ship planning problem is to determine robust schedules for each ship in the fleet, such that the sum of the operating costs, penalty costs, and spot costs are minimized. Here a schedule is defined as a visiting sequence of nodes including the arrival time at each node.

In Section 4.1, we give a set partitioning formulation of the ship planning problem. We find all columns for the set partitioning formulation *a priori.* These columns correspond to complete ship schedules. In Section 4.2, we describe how to find these schedules including information about the operating and penalty costs. Here, we focus on how to incorporate the special port operating time structure and the robustness aspects within a set partitioning approach.

4.1. The Ship Scheduling Problem Formulated as a Set Partitioning Problem

The described ship scheduling problem can be formulated as a set partitioning problem with variables that correspond to feasible ship schedules. At this time, it is assumed that the visiting sequence for a given set of nodes and the arrival time at each node are found such that the sum of the operating costs and the penalty costs for the given schedule is minimized. Let R_v be the set of candidate schedules for ship v, indexed by r. C_{Tvr} is the sum of the operating cost for sailing schedule r by ship v and the corresponding penalty cost. C_{SPOTi} is the cost for cargo i to be serviced by a spot carrier. B_{ivr} is a constant that is equal to 1 if schedule r for ship v services cargo i and 0 otherwise. Let x_{vr} be a binary variable that is equal to 1 if ship v sails schedule r and 0 otherwise. s_i is a binary variable which is equal to 1 if cargo i is serviced by a spot carrier and 0 otherwise. Now, the ship scheduling problem can be formulated as follows:

$$\min z = \sum_{v \in V} \sum_{r \in R_v} C_{Tvr} x_{vr} + \sum_{i \in N_P} C_{SPOTi} s_i, \tag{2}$$

$$\sum_{v \in V} \sum_{r \in R_v} B_{ivr} x_{vr} + s_i = 1, \qquad \forall\, i \in N_P, \tag{3}$$

$$\sum_{r \in R_v} x_{vr} \leq 1, \qquad \forall\, v \in V, \tag{4}$$

$$x_{vr} \in \{0, 1\}, \qquad \forall\, v \in V, \quad r \in R_v, \tag{5}$$

$$s_i \geq 0, \qquad \forall\, i \in N_P. \tag{6}$$

The objective function (2) minimizes the sum of the costs of operating the fleet, the penalty costs for arriving too close to a weekend and the costs of the spot shipments. Constraints (3) ensure that all cargoes are serviced, either by a ship in the fleet or by a spot carrier. Constraints (4) ensure that each ship in the fleet sails at most one of its candidate schedules, while constraints (5) impose binary requirements on the x_{vr} variables. According to (3), the s_i variables do not need to be defined as binary since both the B_{ivr} constants and the x_{vr} variables are binary.

4.2. Generation of the Ship Schedules

The column vector $[B_{1vr}, B_{2vr}, \ldots, B_{nvr}, 1]^T$ in the set partitioning formulation given in Section 4.1 corresponds to one of the feasible schedules belonging to the set R_v for ship v. First we describe how to select the nodes in each schedule and then how to sequence these nodes.

Often, ship scheduling problems are well restricted, meaning that it is possible to enumerate all feasible candidate schedules in a set partitioning approach. This is the case for the real planning problem considered in this paper. The main reason for this is the long duration of each vessel voyage compared with the planning horizon. In practice, it is hardly possible for a ship schedule planner to make plans for more than three voyages ahead for each ship. Most often, this results in "well-restricted" ship scheduling problems, which again makes the set partitioning approach well suited for solving such problems. This is further discussed in Fagerholt [14].

However, for some types of ship scheduling problems, the number of feasible schedules for the fleet may be too large to allow an exhaustive enumeration. For such problems, it is possible to generate only the promising schedules by use of heuristic rules. Examples of the use of such heuristic rules can be found in Fagerholt and Christiansen [16] and Fisher and Rosenwein [19].

Alternatively, we can use a column generation approach. In [5], Christiansen and Nygreen generate columns as needed for a complex ship routing and inventory management problem.

Here we generate all candidate schedules for ship v, and we start with schedules containing one cargo i. These initial schedules consist of a sequence of three nodes, $0 - i - (n + i)$, the initial point for ship v, the pickup node and the delivery node for cargo i, respectively. We generate new schedules by extending these initial schedules by the available cargoes in turn, one at a time, given that this new schedule is not considered before. The algorithm for selecting nodes for new candidate schedules proceeds by adding new cargoes to the already feasible, sequenced schedules until there are no cargoes that can be added to the schedules.

Suppose an existing schedule servicing cargoes 1 and 2 has the optimal visiting sequence $0 - 2 - 1 - (n + 1) - (n + 2)$, with node 0 as the starting node. Here, cargo 2 has first been loaded on board the ship, then cargo 1. Further on, cargo 1 has been discharged before cargo 2. This existing schedule is going to be extended by cargo i, with the corresponding nodes i and $(n + i)$. To find the optimal visiting sequence for the new extended schedule, we have to solve an optimization problem containing all the nodes. This problem consists of designing a visiting sequence for a given set of nodes such that the sum of the operating costs and the penalty costs in the sequence is minimized. Further, each node has to be serviced within its specified time window, while the service time in each port is calculated based on the arrival time at the port. Finally, the pickup node has to be visited before its corresponding delivery node. This problem corresponds to a Traveling Salesman Problem with Capacity, Penalized multiple Time Window and Precedence Constraints (TSP-CPmTWPC) for the nodes $0, 1, 2, i, (n + 1), (n + 2)$ and $(n + i)$.

Here, the TSP-CPmTWPC can be solved directly by use of a forward dynamic programming (DP) algorithm due to the small sized problems. Both Psaraftis [27] and Desrosiers, Dumas, and Soumis [10] present a forward DP algorithm for solving a single-vehicle pickup and delivery problem with time windows, where the time and total distance are respectively minimized. In contrast, the DP method presented here aims at minimizing the total costs. This is a somewhat more difficult optimization task than when the minimum schedule time is sought. In Fagerholt and Christiansen [17], we give a detailed description of the algorithm for finding the minimum schedule time for a sequence of nodes for a TSP with allocation, time window, and precedence constraints. As for this problem, the TSP in [17] occurs as a subproblem of generating ship schedules. For that problem, neither the total port costs nor the total time in the ports are influenced by the visiting sequence of the nodes in the schedule. Here, different visiting sequences consisting of the same nodes may result in considerably different amounts of time spent in the various ports.

To present the DP formulation of the TSP-CPmTWPC, we define $W = \{0, 1, 2, \ldots, w\}$ as the set of nodes to be sequenced and A_{vr} as the set of arcs in the network for ship v and schedule r. In A_{vr}, we retain only the arcs, which satisfy *a priori* restrictions based on capacity, time, and precedence. The time windows are reduced *a priori* for each network by applying the rules described by Desrosiers et al. [9]. Define d to be a dummy ending node for the TSP-CPmTWPC. Then $W' = W \cup \{d\}$ is the total set of nodes to be sequenced. Associate each arc with a sailing time T_{ijv} and an operating cost C_{Oijv}. The penalty cost function is given by $C_{Piv}(t)$. For each ship schedule combination (v, r), define the function $f_{vr}(S, j, t)$ as the least costs (operating and penalty) of a path originating at node 0, visiting every node of $S \subseteq W$ exactly once, ending at node $j \in S$, and ready to service node j at time t or later. In addition, if j is a delivery node, the function $f_{vr}(S, j, t)$ does only exist if the corresponding pickup node is in the set S. $f_{vr}(S, j, t)$ can be obtained by solving the recursions

$$f_{vr}(S, j, t) = \min_{(i,j)\in A_{vr}} \left\{ \min_{\substack{T_{Ai}\le\tau\le T_{Bi} \\ \tau+T_{SERVi}(\tau)\le t-T_{ijv}}} \{f_{vr}(S\backslash\{j\}, i, \tau) + C_{Piv}(\tau) + C_{Oijv} \mid i \in S\backslash\{j\}\} \right\},$$

$$\forall\, T_{Aj} \le t \le T_{Bj}, \quad j \in S, \quad S \subseteq W. \quad (7)$$

Note that the multiple time windows are handled by the service time function, $T_{SERVi}(\tau)$, within the minimization in the recursive formula (7). In (7), the port costs at port i are excluded as the total port costs for a set of nodes are fixed. The recursion formula is initialized by $f_{vr}(\{0\}, 0, t_0) = 0$, and the optimal solution to the TSP-CPmTWPC is given by

$$\min_t f_{vr}(W', d, t) = \min_{(i,d)\in A_{vr}} \left\{ \min_{\substack{T_{Ai}\le\tau\le T_{Bi} \\ \tau+T_{SERVi}(\tau)\le t-T_{idv}}} \{f_{vr}(W, i, \tau) + C_{Piv}(\tau) + C_{Oidv} \mid i \in W\} \right\}. \quad (8)$$

By backtracking, we get the schedule including information about the arrival times. Indirectly, each schedule contains information about the time spent at the visited ports. The cost term C_{Tvr} can now be included in the objective function (2). It consists of the number from (8) and the port costs for the visited ports in the schedule.

During the solution process, we make use of elimination tests to reduce the state space and hence the computational time. Examples of such elimination tests are described in Dumas et al. [13] and Fagerholt and Christiansen [17].

5. COMPUTATIONAL RESULTS

The proposed solution algorithm described in Section 4 is tested on data from a real problem considering the transportation of dry bulk cargoes in northern Europe. This problem is faced by Norsk Hydro ASA, which is engaged in waterborne transportation of its products. The problem is described in Section 3. The algorithms for generation of the candidate schedules and schedule optimization are written and compiled in Borland Pascal 7.0. The set partitioning problem is implemented and solved using GAMS/CPLEX Version 5.0. Data instances of the problem are run on a PC with a Pentium 166 MHz processor having 64 MB of RAM.

In Tables 1 and 2, we give the results from four different cases of the planning problem. Case 1 consists of 18 cargoes (36 nodes) distributed in a 2-week planning period. Cases 2 and 3 consist of 26 cargoes (52 nodes) and 35 cargoes (70 nodes), respectively. Both cases correspond to a 3-week planning period. Finally, in Case 4 the fleet of ships has to load and discharge 40 cargoes (visiting 80 nodes) within the planning period of 4 weeks. The fleet available to perform the transportation tasks consists of five ships, varying in their capacities and initial positions. There is no requirement for a specified position for any ship at the end of the planning period. The ships are assumed to be empty in the beginning and at the end of the planning period. The outer time windows for Cases 1 and 2 are 3 days on average, while for Cases 3 and 4 they are 2 days on average. The cases were solved with a time discretization of 1 hour in the schedule optimization phase.

For all cases, the problems were solved by generating all feasible schedules, thus ensuring optimal solutions. The number of schedules generated and the CPU-times for generating the schedules are given in Tables 1 and 2. We note that the number of schedules rises with an increased number of cargoes and time window widths. The number of generated schedules is reduced to less than half of the schedules from Case 2 to Case 3 due to a reduced time window width, even though the number of cargoes is increased from Case 2 to Case 3.

Table 1. Running information for Case 1 and Case 2.

Problem	Case 1			Case 2		
Planning period (weeks)	2			3		
Number of nodes	36			52		
Time window width (days)	3			3		
Schedule generation						
Number of schedules	3272			26663		
CPU-seconds	5.9			147.8		
Set partitioning problem						
	I1a	**I1b**	**I1c**	**I2a**	**I2b**	**I2c**
Penalty factor P	0.0	0.5	1.0	0.0	0.5	1.0
CPU-seconds	2.1	2.2	2.6	24.1	98.8	28.8
Number of risky arrivals	4	3	1	4	3	3
Avg. time of risky arrivals	5.8	8.0	10.0	3.0	8.7	8.7
Min. time of risky arrival	3	3	10	0	6	6
Total transportation cost (%)	100.0	100.4	107.3	100.0	107.5	107.5
Penalty cost (%)	—	4.9	1.6	—	1.1	2.2
Calculated risk cost (%)	20.4	9.8	1.6	19.5	2.2	2.2

For each of the four cases, we give three sets of results; Instances **I**s**a**, **I**s**b**, and **I**s**c** for case s. For the first instance, **I**s**a**, no penalty cost function is used to avoid risky arrivals. However, for the next two instances different penalty factors P are used. For both of these instances, we have used a linear penalty cost function in the risky arrival time interval. The length of this interval is 12 hours, $T_{RISKi} = 12, \forall i \in N$.

Further, Tables 1 and 2 show the resulting number of arrivals that are denoted as risky. We see that the number of risky arrivals decreases with an increasing penalty factor, P. With increased number of nodes and decreased time window width, it becomes more difficult to avoid risky arrivals.

Table 2. Running information for Case 3 and Case 4.

Problem	Case 3			Case 4		
Planning period (weeks)	3			4		
Number of nodes	70			80		
Time window width (days)	2			2		
Schedule generation						
Number of schedules	11597			25516		
CPU-seconds	19.6			50.9		
Set partitioning problem						
	I3a	**I3b**	**I3c**	**I4a**	**I4b**	**I4c**
Penalty factor P	0.0	0.5	1.0	0.0	0.5	1.0
CPU-seconds	6.0	6.5	7.4	19.8	17.6	14.7
Number of risky arrivals	8	6	5	8	6	5
Avg. time of risky arrivals	7.5	8.3	9.0	7.3	8.5	8.6
Min. time of risky arrival	5	5	8	0	8	8
Total transportation cost (%)	100.0	100.3	102.0	100.0	101.5	102.3
Penalty cost (%)	—	3.3	4.5	—	2.7	4.4
Calculated risk cost (%)	10.8	6.6	4.5	9.8	5.4	4.4

Next, we present the information about the "average time of risky arrivals." This means the average time interval from the scheduled arrival time of the ships, t_i, to the end of the risky arrival time interval, T_{RENDi}; $\sum_{i \in \text{"risky arrivals"}} [T_{RENDi} - t_i]/|\text{risky arrivals}|$. This can be considered as the average slack built into the solution. We see that this time increases with increased penalty factor.

In addition, we give the time interval from the scheduled arrival time of the ships to the end of the risky arrival time interval for the most risky arrival; "Min. time of risky arrival" = $\min_i[T_{RENDi} - t_i]$. For Cases 2 and 4 with no penalty, a ship is calculated to arrive at the latest possible time in order to complete the service in port before the weekend starts. When the penalty is considered, the most risky arrivals are scheduled to finish the service 6 and 8 hours before the weekend starts for Cases 2 and 4, respectively. This means that the introduction of penalties both reduces the number of risky arrivals and increases the time interval from the scheduled departure time to the end of the port operating hours on Friday. Clearly, we see that the introduction of penalty costs manages to drive the optimal solution away from the arrivals that are considered as most risky.

The total transportation costs include the operating costs and the cost of the spot cargoes. They are given in percentages compared with instance **I***s***a** for each case s. For these **I***s***a** instances, no penalty costs are included. We see that the total transportation costs increase when the penalty cost function is incorporated. The larger the penalty factor P is, the more expensive and robust is the route pattern found. This is mainly because the number of spot cargoes, and hence the spot costs, increases with increasing penalty cost. Most often the operating costs even decrease as fewer cargoes are carried by the fleet. Next, we give the penalty costs as they appear in the objective function as a percentage of the total transportation costs for **I***s***a**. These penalty costs have been multiplied by the penalty factor P. As we see, the penalty costs do not necessarily increase with increasing P. However, the total costs including transportation costs and penalty costs are non-decreasing. For Case 1, the transportation pattern is changed considerably from instance **I1b** to **I1c** in order to minimize the total costs. This minimum cost objective resulted in increased transportation costs and reduced penalty costs from **I1b** to **I1c**. Finally, we give the calculated penalty costs for all instances with a penalty factor P equal to 1. In general, the calculated penalty costs decrease with increased penalty factor value, as the penalty costs are given relatively more weight in the objective function.

As expected, with higher penalty costs, the optimal solution will be more robust, while the total transportation costs will increase, and vice versa. However, the results show that a significant increase in schedule robustness can be achieved at relatively little increase in transportation costs.

6. CONCLUDING REMARKS

In many types of shipping operations, the ports are closed for service at night and during the weekends. This is often the case for vehicles visiting customers as well. In contrast to vehicle routing problems, the loading or discharging of a ship is often very time-consuming. For the ship scheduling problem studied here, a ship will often be unable to finish the service during a day. This means that the ship will stay idle for much of the time in port. For ship schedule planners, it is of great importance to avoid long idle times in port and especially waiting time in port during weekends. We consider this issue for a ship scheduling problem concerned with the pickup and delivery of cargoes within given service time windows. Since the ports have restricted operating hours, the cargo time windows can be regarded as multiple time windows, when the cargo time window widths exceed 1 day. However, we have kept the cargo time windows and defined a function for the total time at a port. Due to the restricted port operating hours, this service time function is dependent on the arrival time at the particular port.

Ship scheduling is associated with a high degree of uncertainty, so ship planners want to avoid scheduled departures just before closing time on Friday afternoon. Such schedules are considered risky, and the planners want to find alternative schedules that are more robust with respect to delays. To handle this robustness aspect, we have introduced the possibility of imposing a penalty cost for risky arrival times. The penalty cost function is given as a non-decreasing function in an interval, defined as particularly risky. The proposed solution approach handles linear as well as nonlinear functions. According to the actual situation and the risk attitude of the planners, the gradient and form of the function as well as the length of the risky arrival time interval can be defined by the planners.

The problem is solved by using a set partitioning approach. First, all feasible schedules are generated. We have used DP to find the optimal sequence of the ports in each schedule. Here, the uncertainty issue is incorporated into the solution approach by the use of a penalty cost function and the arrival-dependent service times. Further on, the feasible schedules are brought into the set partitioning model and solved by commercial optimization software.

The solution approach is tested on data from a real ship planning problem and run on a standard PC. We obtained optimal solutions for all problem cases presented, which are all of real size or larger.

Using the computational results, we see that with high penalty costs, the optimal solution will appear more robust. This means that the number of risky arrivals decreased, and the average time from the scheduled departure time of the ships to the end of the port operating hours on Friday for risky arrivals increased. In addition, the calculated penalty costs were reduced. The total transportation costs increased with high penalty costs, and vice versa. However, the results show that significant increase in schedule robustness can be achieved at relatively little increase in transportation costs.

In practice, it may be difficult to construct an appropriate penalty function. However, by running several computations with varying penalty cost functions, one can perform analysis on the tradeoff between total transportation costs and the risk or uncertainty of the schedules.

As the computational results showed, the CPU-time is much higher for the schedule generation part of the solution approach than for solving the set partitioning problems. This is because we solve a Traveling Salesman Problem (TSP) with a number of side constraints optimally by a DP algorithm in the generation of each new schedule. Alternatively, we could have used a heuristic method for solving these TSPs, which would have reduced the CPU-time for the schedule generation. However, we could no longer guarantee that the visiting sequences for the schedules were optimal, and therefore the whole approach would become a heuristic one. Laporte [24] presents a survey on both exact and heuristic methods for the TSP.

REFERENCES

[1] N. Balakrishnan, Simple heuristics for the vehicle routeing problem with soft time windows, J Oper Res Soc 44(3) (1993), 279–287.
[2] D.O. Bausch, G.G. Brown, and D. Ronen, Scheduling short-term marine transport of bulk products, Maritime Policy Manage 25(4) (1998), 335–348.
[3] G.G. Brown, C.E. Goodman, and R.K. Wood, Annual scheduling of Atlantic fleet naval combatants, Oper Res 38(2) (1990), 249–259.
[4] G.G. Brown, W. Graves, and D. Ronen, Scheduling ocean transportation of crude oil, Manage Sci 33(3) (1987), 335–346.
[5] M. Christiansen, Decomposition of a combined inventory and time constrained ship routing problem, Transport Sci 33(1) (1999), 3–16.

[6] M. Christiansen and B. Nygreen, A method for solving ship routing problems with inventory constraints, Ann Oper Res 81 (1998), 357–378.
[7] M. Christiansen and B. Nygreen, A ship routing problem with soft inventory constraints, Working Paper, Section of Operations Research, Norwegian University of Science and Technology, Trondheim, Norway, 2000.
[8] K. Darby-Dowman, R.K. Fink, G. Mitra, and J.W. Smith, An intelligent system for US Coast Guard cutter scheduling, Eur J Oper Res 87 (1995), 574–585.
[9] J. Desrosiers, Y. Dumas, M.M. Solomon, and F. Soumis, "Time constrained routing and scheduling," Handbooks in operations research and management science 8, Network routing, M.O. Ball, T.L. Magnanti, C.L. Monma, G.L. Nemhauser, (Editors), North-Holland, Amsterdam, 1995, pp. 35–139.
[10] J. Desrosiers, Y. Dumas, and F. Soumis, A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows, Am J Math Manage Sci 6 (1986), 301–325.
[11] Y. Dumas, J. Desrosiers, and F. Soumis, The pickup and delivery problem with time windows, Eur J Oper Res 54 (1991), 7–22.
[12] Y. Dumas, F. Soumis, and J. Desrosiers, Optimizing the schedule for a fixed vehicle path with convex inconvenience costs, Transport Sci 24(2) (1990), 145–152.
[13] Y. Dumas, J. Desrosiers, E. Gélinas, and M.M. Solomon, An optimal algorithm for the traveling salesman problem with time windows, Oper Res 43 (1995), 367–371.
[14] K. Fagerholt, Optimisation based methods for solving ship routing and scheduling problems, Ph.D. dissertation, Department of Marine Systems Design, Norwegian University of Science and Technology, Trondheim, Norway, 1999.
[15] K. Fagerholt, Ship scheduling with soft time windows—an optimisation based approach, Eur J Oper Res 131 (2001), 559–571.
[16] K. Fagerholt and M. Christiansen, A combined ship scheduling and allocation problem, J Oper Res Soc 51 (2000), 834–842.
[17] K. Fagerholt and M. Christiansen, A travelling salesman problem with allocation, time window and precedence constraints—an application to ship scheduling, Int Trans Oper Res 7(3) (2000), 231–244.
[18] K. Fagerholt and H. Lindstad, Optimal policies for maintaining a supply service in the Norwegian Sea, OMEGA Int J Manage Sci 28(3) (2000), 269–275.
[19] M.L. Fisher and M.B. Rosenwein, An interactive optimization system for bulk cargo ship scheduling, Nav Res Logistics 36 (1989), 27–42.
[20] P.W. Frizzell and J.W. Giffin, The split delivery vehicle scheduling problem with time windows and grid network distances, Comput Oper Res 22 (1995), 655–667.
[21] M. Gendreau, G. Laporte, and R. Séguin, Stochastic vehicle routing, Eur J Oper Res 88 (1996), 3–12.
[22] I. Ioachim, S. Gélinas, F. Soumis, and J. Desrosiers, A dynamic programming algorithm for the shortest path problem with time windows and linear node costs, Networks 31(3) (1998), 193–204.
[23] Y.A. Koskosidis, W.B. Powell, and M.M. Solomon, An optimization-based heuristic for vehicle routing and scheduling with soft time window constraints, Transport Sci 26(2) (1992), 69–85.
[24] G. Laporte, The traveling salesman problem: An overview of exact and approximate methods, Eur J Oper Res 59 (1992), 231–247.
[25] G. Laporte and I.H. Osman, Routing problems: A bibliography, Ann Oper Res 61 (1995), 227–262.
[26] G. Pesant, M. Gendreau, J.-Y. Potvin, and J.-M. Rosseau, On the flexibility of constraint programming models: From single to multiple time windows for the traveling salesman problem, Eur J Oper Res 117 (1999), 253–263.
[27] H.N. Psaraftis, An exact algorithm for the single-vehicle many-to-many dial-a-ride problem with time windows, Transport Sci 17 (1983), 351–357.
[28] D. Ronen, Cargo ships routing and scheduling: Survey of models and problems, Eur J Oper Res 12 (1983), 119–126.
[29] T. Sexton and L. Bodin, Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling, Transport Sci 19(4) (1985), 378–410.
[30] T. Sexton and Y. Choi, Pickup and delivery of partial loads with time windows, Am J Math Manage Sci 6 (1986), 369–398.
[31] É. Taillard, P. Badeau, M. Gendreau, F. Guertin, and J. Potvin, A tabu search heuristic for the vehicle routing problem with soft time windows, Transport Sci 31(2) (1997), 170–186.