Journal of the Operational Research Society

ISSN: 0160-5682 (Print) 1476-9360 (Online) Journal homepage: http://www.tandfonline.com/loi/tjor20

Reducing fuel emissions by optimizing speed on shipping routes

K Fagerholt, G Laporte & I Norstad

Journal of the Operational Research Society (2010) 61, 523–529

www.palgrave-journals.com/jors/

Reducing fuel emissions by optimizing speed on shipping routes

K Fagerholt[1], G Laporte[2,]* and I Norstad[1,3]

[1]Norwegian University of Science and Technology, Trondheim, Norway; [2]Canada Research Chair in Distribution Management, HEC Montréal, Canada and [3]Norwegian Marine Technology Research Institute (MARINTEK), Trondheim, Norway

Fuel consumption and emissions on a shipping route are typically a cubic function of speed. Given a shipping route consisting of a sequence of ports with a time window for the start of service, substantial savings can be achieved by optimizing the speed of each leg. This problem is cast as a non-linear continuous program, which can be solved by a non-linear programming solver. We propose an alternative solution methodology, in which the arrival times are discretized and the problem is solved as a shortest path problem on a directed acyclic graph. Extensive computational results confirm the superiority of the shortest path approach and the potential for fuel savings on shipping routes.

Journal of the Operational Research Society (2010) **61,** 523–529. doi:10.1057/jors.2009.77
Published online 19 August 2009

Keywords: shipping routes; fuel emissions; speed optimization; shortest paths

Introduction

In the transportation industry and in several other sectors, the beginning of the 21st century has been characterized by an increased awareness of the effect of fuel usage on operation costs and on CO_2 emissions. This is particularly the case in maritime transportation, which is the object of this paper. The effects of optimization in this industry can be significant given the very large and still growing volume of goods that are transported by sea (Christiansen *et al*, 2004, 2007). According to UNCTAD (2007), more than seven billion tons of goods are carried by ships each year.

A recent news release by Oceana (www.oceana.org) states: 'Reducing commercial ship speeds, by only a few knots, yields salutary results for both shipping companies and the environment. "Slower steaming" saves on fuel consumption and cost while also releasing fewer greenhouse gas emissions. The commercial shipping industry is beginning to catch on to the benefits of slower steaming and many companies are now enjoying the rewards of more sustainable emissions through monetary savings.' Then follows a long list of quotes from representatives of the shipping industry on the environmental and economic benefits of slower speeds.

Fuel emissions are proportional to consumption, which is in turn a convex function of speed. Manning (1956), Ryder and Chappell (1979) and Ronen (1982) show that a cubic function provides a good approximation of this relationship after a certain speed level. Below this speed level the fuel consumption is approximately the same for any speed. Several authors (eg, Perakis and Papadakis, 1987a, 1987b; Brown *et al*, 1987; Papadakis and Perakis, 1989; Bausch *et al*, 1998; Fagerholt, 2001) have combined the determination of ship routes with speed decisions, albeit in contexts different from ours. Other authors (eg, Sexton and Bodin, 1985; Dumas *et al*, 1990; Ioachim *et al*, 1998) have investigated the more general problem of schedule optimization on a fixed route, but with different objectives and constraints.

Our study applies to fixed shipping routes, such as the one depicted in Figure 1. In this example there are six port calls (excluding the initial port) and port visits must start within set time windows. The distances in nautical miles between two consecutive port locations and the time windows associated with each port k are provided in Table 1. Consider a ship with a service speed of 18.5 knots leaving Antwerp at time 0 to sail the route given in Figure 1. The fourth column of Table 1 gives the arrival times at each port along the route, where 'w' indicates that ship must wait for the start of the time window. For simplicity, we have assumed that no time is spent in the ports in this example. If this ship travels at its service speed, it will consume about 30 thousand tons fuel on this route at a cost of about one million USD (although fuel costs fluctuate significantly over time). Now, if the ship reduces its speed by 10% it can still arrive at each destination within its time windows, thus reducing both the fuel costs and consumption, and hence the environmental emissions, by 19.4%. If the optimal speed for each sailing leg was used, the savings would be 24.3%.

*Correspondence: *G Laporte, HEC Montréal, 3000 chemin de la Côte-Sainte-Catherine, Montréal H3T 2A7, Canada.
E-mail: gilbert@crt.umontreal.ca*

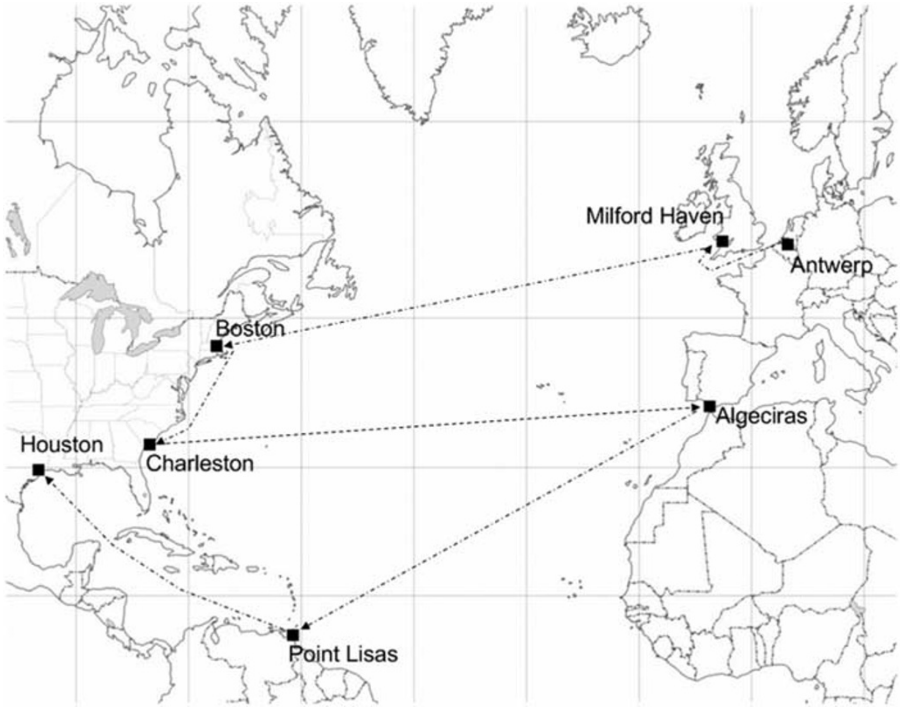

Figure 1 Illustration of a shipping route.

Table 1 Distances of shipping route, time windows and arrival time when using service speed.

Port	*Distance (nautical miles)*	*Time window (days)*	*Arrival time (days)*
Antwerp	0	[0, 0]	0
Milford Haven	510	[1, 5]	1.1
Boston	2699	[9, 13]	7.2 (*w*)
Charleston	838	[11, 15]	11.9
Algeciras	3625	[20, 24]	22.2
Point Lisas	3437	[32, 36]	29.9 (*w*)
Houston	2263	[35, 39]	37.1

Our aim is precisely to optimize the speed on each segment of a route in order to respect the time windows. For practical reasons, we consider only the speed interval where the fuel consumption can be approximated by a cubic function. We show that although this problem is a continuous non-linear optimization problem, it can be discretized and solved efficiently by means of a shortest path algorithm on a directed acyclic graph.

The remainder of this paper is organized as follows. In the next section we provide three models for the problem at hand. This is followed by a computational study and by conclusions.

Mathematical models for optimizing speed on a ship route

Figure 2 illustrates the optimal speed selection problem along a given single ship route with four ports calls (excluding the initial position of the ship), which we will refer to as nodes in the following.

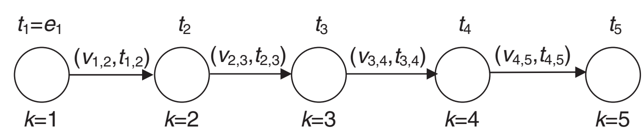

Figure 2 Example of a single ship route with decision variables.

The nodes along the route are indexed by $k = 1, 2, \ldots, n$. Let the parameter $d_{k,\,k+1}$ represent the sailing distance along arc $(k,\ k+1)$, defined for $k = 1, 2, \ldots, n-1$. Each node has a given time window, in which the service must start, given by $[e_k, l_k]$. We define $v_{k,\,k+1}$ and $t_{k,\,k+1}$ as the decision variables for speed and sailing time along arc $(k,\ k+1)$, respectively. The speed and sailing time are clearly interrelated, which will be illustrated further below. Let t_k be the time at which service starts at node k, which is interrelated to $v_{k,\,k+1}$ and $t_{k,\,k+1}$. Departure from the start node can always be as early as possible, which is indicated by $t_1 = e_1$ in Figure 2. In practice there will also be a specified service time at each node. However, this is independent of sailing speed decisions, and hence we do not include this feature in the following models.

For a given arc $(k, k+1)$ with the distance $d_{k,\,k+1}$, the relation between speed, $v_{k,\,k+1}$, and sailing time, $t_{k,\,k+1}$, is given by

$$d_{k,\,k+1} = v_{k,\,k+1} t_{k,\,k+1}, \qquad k = 1, \ldots, n-1. \tag{1}$$

The ship sailing the route has maximum speed, $v_{\max}$, and for practical reasons we can also assume a minimum speed, $v_{\min}$. The fuel consumption per time unit for a cargo ship lies between these upper and lower speed limits, as previously mentioned, and can be approximated by a cubic function of speed. By using (1), we can easily see that the sailing consumption per distance unit (or for a given arc) is a quadratic convex function $f(v)$. Using real data for a specific ship, we found the empirical relationship for fuel consumption per nautical mile to be $f(v) = 0.0036v^2 - 0.1015v + 0.8848$, which is valid for the speed interval from 14 to 20 knots. This function and speed interval were used in our experiments.

The problem we consider is to determine the ship speeds along the route that minimize fuel consumption, while satisfying the time window constraints. In the following, we present three alternative models.

Model 1: Speed as primary decision variable

In Model 1, we use speed as our primary decision variable. From (1), we can represent the sailing time on an arc as a function of speed, $t_{k,\,k+1} = d_{k,\,k+1}/v_{k,\,k+1}$. The speed selection problem can now be modelled as follows:

$$\text{Minimize} \sum_{k=1,\,\ldots,\,n-1} d_{k,\,k+1} f(v_{k,\,k+1}) \tag{2}$$

subject to

$$t_{k+1} - t_k - d_{k,\,k+1}/v_{k,\,k+1} \geqslant 0, \qquad k = 1, \ldots, n-1, \tag{3}$$

$$e_k \leqslant t_k \leqslant l_k, \qquad k = 1, \ldots, n, \tag{4}$$

$$v_{\min} \leqslant v_{k,\,k+1} \leqslant v_{\max}, \qquad k = 1, \ldots, n-1. \tag{5}$$

The objective function (2) is to minimize the fuel consumption for the given route. Constraints (3) ensure that the ship does not start service before it arrives at the node. Constraints (4) are the time window constraints, while (5) ensure that the speed on a given arc is within the lower and upper limits. For the start node, the time window can be given by a single value, which will represent the earliest possible start time, e_1. We note that the objective function is a convex non-linear function, and that constraints (3) also are non-linear.

Model 2: Sailing time as primary decision variable

Since, from (1), speed can be formulated as $v_{k,\,k+1} = d_{k,\,k+1}/t_{k,\,k+1}$, we can use the sailing time variables instead of the speed variables to formulate the problem. Then, fuel consumption per nautical mile along a given sailing arc $(k, k+1)$ can be given as a function of sailing time and distance. Now, the problem, given by (2)–(5), can be modelled in the following alternative way:

$$\text{Minimize} \sum_{k=1,\,\ldots,\,n-1} d_{k,\,k+1} g(t_{k,\,k+1}) \tag{6}$$

subject to

$$t_{k+1} - t_k - t_{k,k+1} \geqslant 0, \qquad k = 1, \ldots, n-1, \tag{7}$$

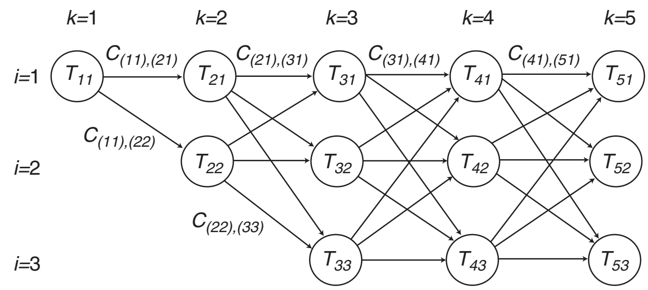

Figure 3 Example of discretized problem resulting in a directed acyclic graph.

$$e_k \leqslant t_k \leqslant l_k, \qquad k = 1, \ldots, n, \tag{8}$$

$$d_{k,\,k+1}/v_{\max} \leqslant t_{k,\,k+1} \leqslant d_{k,\,k+1}/v_{\min}, \qquad k = 1, \ldots, n-1, \tag{9}$$

where $g(t_{k,\,k+1})$ is obtained from $f(v_{k,\,k+1})$ using (1). The objective function (6) is equivalent to (2), while (7)–(9) are equivalent to (3)–(5). We may note that the objective function still is a nonlinear function, while all constraints are now linear.

Model 3: Discretized arrival time

We can also discretize the arrival time (within the time window) at each node, and solve the problem as a shortest path problem on a directed acyclic graph, following the ideas outlined by Fagerholt (2001). Let us consider the example ship route illustrated in Figure 2, and discretize the arrival times, see Figure 3.

Each node in the route corresponding to Figure 2 will now be replicated into a number of discretized feasible arrival times at that node. Each of the new replicated nodes will represent a state, given by (k, i), where k is the node index of the route and i represents the discretized arrival time T_{ki} at the node (shown inside the nodes in Figure 3). It may be noted that $T_{k,i+1} > T_{ki}$ ($i + 1$ always refers to a later arrival time than i). Each state (k, i) will represent a node in the graph. Every arc $((k, i), (k + 1, j))$ in the graph connects a discretized arrival time for node k to a discretized time for node $k + 1$, resulting in a directed acyclic graph. Each arc will correspond to a given sailing speed, where $c_{(k,i),(k+1,j)}$ represents the cost on arc $((k, i), (k + 1, j))$, equal to the fuel consumption based on the speed. Arcs corresponding to sailing speeds that are outside the feasible region for the ship (see (5) in Model 1) will not be generated. For instance the arcs pointing upwards in Figure 3 will often represent sailing speeds that are higher than the maximum speed. States without any incoming arcs will not be generated.

It may be noted that $c_{(k,i),(k+1,j)} \geqslant c_{(k,i),(k+1,j+1)}$, since the latter allows for slower sailing speed with less fuel consumption. We may also note that for $k = 1$, there is only one node because postponing the departure from the first node will only result in a poorer solution. We can also use the same argument to delete all replicated nodes for node n, except for the one representing the latest start of service time (node (5,3) in Figure 3). Although this is not done here, it is easy to introduce a penalty cost for finishing the route later than the earliest possible time because this may affect the routing options in following planning periods. This penalty can easily be included in $c_{(n-1,i),(n,j)}$.

The shortest path problem consists of finding a minimum cost path from node (1, 1) to one of the replicated nodes (n, i) for any i. Define m as the number of time discretizations for each node in the route, ie $i = 1, \ldots, m$. The flow variable $x_{(k,i),(k+1,j)}$ is equal to 1 if arc $((k, i), (k+1, j))$ is used, and 0 otherwise. We can now model the problem as the following shortest path problem on a directed acyclic graph:

$$\text{Minimize} \sum_{k=1,\ldots,n-1} \sum_{i=1,\ldots,m} \sum_{j=1,\ldots,m} c_{(k,i),(k+1,j)} x_{(k,i),(k+1,j)} \tag{10}$$

subject to

$$\sum_{i \in N_k} \sum_{j=1,\ldots,m} x_{(k,i),(k+1,j)} = 1, \qquad k = 1, \ldots, n-1, \tag{11}$$

$$\sum_{i=1,\ldots,m} x_{(k-1,i),(k,j)} = \sum_{i=1,\ldots,m} x_{(k,j),(k+1,i)}, \quad k = 1, \ldots, n, \; j = 1, \ldots, m, \tag{12}$$

$$x_{(k,i),(k+1,j)} \in \{0, 1\}, \qquad k = 1, \ldots, n-1, \quad i = 1, \ldots, m, j = 1, \ldots, m. \tag{13}$$

The objective function (10) minimizes the fuel consumption along the given route, while (11) ensure that each node in the original formulation is visited. (12) are the continuity constraints, while (13) impose binary restrictions on the flow variables. As the shortest path problem possesses the integrality property, the flow variables could also have been defined as continuous variables between 0 and 1.

We can also discretize the ship's sailing in two other ways: the speed on each arc and the sailing time on each arc. However, if we choose any of these two discretizations, the graph will not look like the one in Figure 3 (except for only a very careful selection of discretized speeds/sailing times). Then, the graph will become a tree where the number of nodes will grow exponentially with the number of discretizations and the number nodes in the route. Therefore, we choose to discretize the arrival times (within the time window) for each node. Then, the sailing speed and time on each arc (and hence the fuel consumption) in the graph will entirely be determined by the arc's departure and arrival time.

Computational study

We now describe the generation of test instances and the computational results obtained solving these, first by applying a non-linear programming solver to Model 1 and Model 2, then by discretizing the arrival times and solve the problem as a shortest path problem on a directed acyclic graph (Model 3).

Generation of test instances

In order to test the effect of optimizing speed along a single ship route, we have developed an instance generator coded in C#. The instances should resemble real shipping routes, so we have based the instance generation on realistic ship data. The ship we have relied on has a defined *service speed* of 18.5 knots, while the minimum ship speed is $v_{\min} = 14$ knots, and the maximum ship speed is $v_{\max} = 20$ knots. We let the sailing distances between the nodes in the route be randomly chosen between 850 and 8500 nautical miles, corresponding to about 2–20 sailing days when using the ship's service speed.

In order to produce a varied set of instances, different values are considered for the following three parameters:

- The number of nodes in the route (excluding the start node), with label settings, $n4$, $n8$, $n12$ and $n16$, corresponding to $n = 4$, 8, 12 and 16, respectively.
- The number of nodes in the route that will have a waiting time for the start of the time window when using the ship's service speed, with label settings $w0$, $w20$ and $w40$, corresponding to 0, 20% and 40% of the nodes (rounded to the nearest integer), respectively. The nodes with waiting are then randomly chosen, and the value of the waiting time is randomly chosen between 1 and 6 days, which are quite common figures in shipping routes.
- The width of the time windows at the nodes, with label settings tw5 and tw10, corresponding 5 and 10 days, respectively. We want to distribute the time windows in a way that resembles typical shipping routes. For the nodes with waiting time, the start of the time window will be given by the amount of waiting time. For the other nodes, we define u_k as the arrival time when using service speed. Then, we define $e_k = u_k - 0.75b + \text{random}\ (0, 0.5b)$ where b is the time window width (five or 10 days, respectively). The end of the time window will then be given by $l_k = e_k + b$.

By combining the possible values for these three parameters we obtain 24 *cases*. For each case, we generate five different *instances*, so there are 120 instances in total. Each instance is labelled by using the value of the three parameters, plus the instance number for each case. This means that for instance $n8$tw10$w20$_2 is instance number 2, with eight nodes, time window widths of 10 days, and 20% of the nodes will have a waiting time when using service speed.

Computational results

Each of the 120 instances was solved using a freely available non-linear programming solver, IPOPT from COIN-OR on

Table 2 Percentage savings over the service speed.

n	*tw*	*w*	*NLP (optimal)*	*Heuristic*
4	5	0	6	0
		20	12	6
		40	22	12
	10	0	16	0
		20	21	4
		40	25	11
8	5	0	2	0
		20	9	2
		40	18	10
	10	0	10	0
		20	14	4
		40	22	8
12	5	0	2	0
		20	8	4
		40	17	10
	10	0	5	0
		20	12	4
		40	20	9
16	5	0	2	0
		20	9	5
		40	15	9
	10	0	4	0
		20	13	4
		40	20	10

Model 1 and Model 2. As the non-linear problem is convex the solutions are proven optimal. We found that Model 2 was faster as it required fewer iterations to reach the optimal solution. Table 2 shows the reduction in fuel consumption that can be obtained by optimizing the speed for each leg on a sailing route. For each of the 24 cases, we report the average reduction in consumption over the five instances as a percentage of fuel consumption with the service speed (18.5 knots). The three columns to the left in Table 2 provide input data, showing the number of nodes in the route (n), the width of the time windows (tw) and the percentage of the nodes with waiting time using service speed (w). The column denoted NLP (optimal) shows the results obtained from solving Model 2. We also report the results from applying a simple heuristic way to introduce slow steaming, where sailing is normally at service speed, but speed is reduced in situations where the ship will arrive too early and have to wait for the start of the time window. We then calculate and use the speed so that the ship will arrive just in time for the start of the time window. This method is commonly used in operational shipping. The results from using this simple heuristic are given in the column denoted by heuristic in Table 2.

We see, as expected, that when the routes involves more waiting time, the potential for reducing fuel consumption increases. We also see the same effect for the cases with wide time windows. These effects are more clearly illustrated in the aggregated results presented in Table 3. For instance, for routes having time windows equal to 10 days and 40% of the nodes with waiting, it is possible to reduce the fuel consumption by 21% on average. This will have a significant impact both on environmental emissions and fuel costs for the shipping company.

Table 3 Aggregated percentage savings extracted from Table 2 (NLP optimal solution).

tw (days)/w (%)	0	20	40
5	3	10	16
10	8	15	21

Table 4 Average optimality gap in percent for different discretization levels for Model 3.

n	*tw*	*w*	5	10	20	50	100
4	5	0	0.31	0.06	0.03	0.01	0.01
		20	0.20	0.04	0.02	0.01	0.01
		40	0.32	0.04	0.01	0.01	0.01
	10	0	1.91	0.36	0.07	0.02	0.01
		20	0.41	0.10	0.02	0.01	0.01
		40	0.24	0.05	0.01	0.01	0.01
8	5	0	0.65	0.12	0.03	0.01	0.01
		20	0.58	0.12	0.02	0.01	0.01
		40	0.16	0.03	0.01	0.01	0.01
	10	0	1.51	0.22	0.08	0.02	0.01
		20	1.62	0.22	0.06	0.01	0.01
		40	0.66	0.12	0.04	0.01	0.01
12	5	0	1.87	0.28	0.04	0.01	0.01
		20	0.64	0.10	0.03	0.01	0.01
		40	0.21	0.08	0.02	0.01	0.01
	10	0	2.13	0.35	0.09	0.02	0.01
		20	1.27	0.29	0.06	0.02	0.01
		40	1.11	0.26	0.05	0.01	0.01
16	5	0	1.25	0.14	0.05	0.01	0.01
		20	0.56	0.07	0.03	0.01	0.01
		40	0.24	0.06	0.02	0.01	0.01
	10	0	3.88	0.58	0.13	0.03	0.01
		20	2.22	0.29	0.07	0.02	0.01
		40	1.28	0.25	0.06	0.01	0.01
Average over all instances			1.05	0.18	0.04	0.01	0.01

Table 2 also shows that when the number of nodes in the shipping route increases, the effect of optimizing the speed is reduced. For instance, for case $n4\text{tw}10w40\_x$ the average reduction in fuel consumption is 25%, whereas for case $n16\text{tw}10w40\_x$ the reduction is only 20%. This is not surprising as the average speed for the whole route in most instances cannot be reduced much while respecting the final time window.

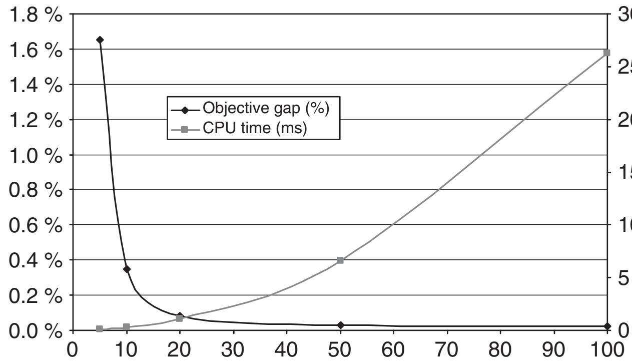

Figure 4 Percent gap with respect to the optimal solution value and CPU-time as a function of the number of discretization points.

Each of the 120 instances was also solved by using five different numbers of discretization points: 5, 10, 20, 50 and 100. For each instance, a shortest path network was generated (Model 3). Each network is a directed acyclic graph, and as the graph is generated in a topological order, finding the shortest path can easily be achieved during the graph generation. Table 4 shows the results for the different settings compared with the optimal solutions found by using the non-linear programming solver. The three columns to the left in Table 4 provide the input data. The last five columns show the average percentage gap for each of the different number of discretization points, compared with the optimal solution.

Figure 4 shows the percentage gap between the solution values of Model 3 and the optimal solution and CPU time in milliseconds as a function on discretization points. We only report the average results for the $n=16$ instances, as the CPU time is negligible for the smaller ones. We see that the solution value converges rapidly to the optimum as a function of the number of discretization points. For all practical purposes it seems that 20 points suffice. For this value, the objective gap is only 0.4% and the solution time also is about two milliseconds. In contrast, the CPU time needed to solve the same problem using the non-linear programming solver was on average 430 milliseconds, more than 200 times longer than the solution time required for the shortest path method with 20 discretization points. This difference in solution time can be critical if the speed selection problem is solved as a subproblem within a local search based heuristic for a ship routing and scheduling problem.

We have also performed computations on a few real planned ship routes to which we had access. The results from these experiments are consistent with those of Table 2.

Conclusion

We have modelled and solved the problem of determining optimal speeds along a shipping route, where each port call has a given time window, in which the service must start. As fuel consumption is typically a cubic function of speed within certain speed limits, the optimal speed problem becomes a non-linear problem. We have proposed a method where the arrival time within the time window of each node is discretized and the problem then is solved as a shortest path problem on a directed acyclic graph. Based on computational tests on a number of randomly generated test instances as well as a few real shipping routes, we see that the potential for reducing fuel consumption, and hence environmental emissions, is substantial. We have also compared our solution method with a freely available non-linear programming solver. The results show that our solution method is much faster and provides quasi-optimal solutions. Another advantage with the shortest path method over the non-linear programming solver is that is does not assume any property of the objective function. In particular, convexity is not required.

It is clear that as the effect of optimizing the speed along a shipping route can be high, this may also influence the routing. Hence, the problem considered in this paper should also be solved as a subproblem when planning routes and schedules for a fleet of ships, for instance as a move evaluator in a local search-based algorithm.

Acknowledgements—This research was carried out with financial support from the DESIMAL project, funded by the Research Council of Norway and from the Canadian Natural Engineering Research Council under grant 39682-05. This support is gratefully acknowledged. We are also grateful for valuable comments and suggestions from two reviewers.

References

Bausch DO, Brown GG and Ronen D (1998). Scheduling short-term marine transport of bulk products. *Marit Pol Mngt* **25**(4): 335–348.

Brown GG, Graves GW and Ronen D (1987). Scheduling ocean transportation of crude oil. *Mngt Sci* **33**: 335–346.

Christiansen M, Fagerholt K, Nygreen B and Ronen D (2007). Maritime transportation. In: Barnhart C and Laporte G (eds). *Handbooks in Operations Research and Management Science: Transportation*. North-Holland: Amsterdam. pp 189–284.

Christiansen M, Fagerholt K and Ronen D (2004). Ship routing and scheduling: Status and perspectives. *Transport Sci* **38**: 1–18.

Dumas Y, Soumis F and Desrosiers J (1990). Optimising the schedule for a fixed vehicle path with convex inconvenience costs. *Transport Sci* **24**: 145–152.

Fagerholt K (2001). Ship scheduling with soft time windows–an optimization based approach. *Eur J Opl Res* **131**: 559–571.

Ioachim I, Gélinas S, Soumis F and Desrosiers J (1998). A dynamic programming algorithm for the shortest path problem with time windows and linear node costs. *Networks* **31**: 193–204.

Manning GC (1956). *The Theory and Technique of Ship Design*. MIT Press: Cambridge, MA, p 45.

Papadakis NA and Perakis AN (1989). A nonlinear approach to the multiorigin, multidestination fleet deployment problem. *Nav Res Log* **36**: 515–528.

Perakis AN and Papadakis NA (1987). Fleet deployment optimization models. Part 1. *Marit Pol Mngt* **14**: 127–144.

Perakis AN and Papadakis NA (1987). Fleet deployment optimization models. Part 2. *Marit Pol Mngt* **14**: 145–155.

Ronen D (1982). The effect of oil price on the optimal speed of ships. *J Opl Res Soc* **33**: 1035–1040.

Ryder SC and Chappell D (1979). *Optimal Speed and Ship Size for the Liner Trades*. Maritime Transport Centre: University of Liverpool, p 17.

Sexton TR and Bodin LD (1985). Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transport Sci* **19**: 378–410.

UNCTAD (2007). *Review of Maritime Transport, 2007*. United Nations, New York and Geneva.

Received December 2008;
accepted May 2009 after one revision