A note on ship routing between ports

Shuaian Wang

Department of Logistics & Maritime Studies, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
Email address: wangshuaian@gmail.com

Abstract

To minimize bunker fuel consumption, ship captains choose suitable speed and heading to take advantage of ocean currents, wind, and wave when routing a ship between two ports. Conventional approaches discretize the space and time and then apply dynamic programming to find the optimal speed and heading of a ship at each time. Nevertheless, the resulting solution to the discretized problem may not converge to the optimal solution to the original continuous problem, even when the sizes of the discretization grids approach 0. To overcome this deficiency, we propose an improved dynamic programming approach. The novelty of the improved method lies in that we repeatedly re-discretize the space in both controlled and random manners to obtain better solutions. The improved dynamic programming approach provides significantly better solutions than the conventional approaches.

Keywords: sea transport, shipping, ship routing, bunker fuel, dynamic programming

1. Introduction

Shipping is vital to international trade (Zhen and Wang, 2015). Bunker fuel costs may represent more than 75 per cent of the operating costs of ships due to the high bunker prices (Ronen, 2011). To reduce fuel consumptions, shipping lines take measures at three planning levels (Bell et al., 2011, 2013; Fagerholt et al., 2013; Halvorsen-Weare and Fagerholt, 2013). At the strategic level, they order mega-ships that are more fuel-efficient. At the tactical level, the ships are steamed at speeds lower than the design speeds as the daily fuel consumption of a ship is roughly proportional to the speed cubed (Fagerholt et al., 2010; Norstad et al., 2011; Wang et al., 2013). At the operational level, ship captains choose suitable speed and heading to take advantage of ocean currents, wind, and wave when routing a ship between two ports.

The operational-level ship routing problem can be described as follows. Let ϕ_1, λ_1, and ϕ_2, λ_2 be the geographical latitude and longitude of the origin port and the destination port, respectively. A ship leaves the origin port at time 0 and must arrive at the destination port at time T. Represent by $x(t)$ and $y(t)$ the latitude and longitude of the ship at time t, respectively, $0 \leq t \leq T$. The two functions $x(t)$ and $y(t)$ could uniquely determine the speed and heading of the ship at each time. Assuming that $x(t)$ and $y(t)$ are differentiable, we can denote by $f(x(t), y(t), \frac{dx(t)}{dt}, \frac{dy(t)}{dt}, t)$ the fuel consumption rate (ton/hour) of the ship at time t. The functional form of f depends on the characteristics of the ship and its cargo load; the current, wind, and wave at the location $(x(t), y(t))$ at time t; and the sailing speed and heading of the ship. The model that minimizes the fuel consumption is:

$$\min \int_{t=0}^{T} f\left(x(t), y(t), \frac{dx(t)}{dt}, \frac{dy(t)}{dt}, t\right) dt \tag{1}$$

subject to:

$$\begin{aligned} x(0) &= \phi_1, \\ y(0) &= \lambda_1, \\ x(T) &= \phi_2, \\ y(T) &= \lambda_2. \end{aligned}$$

Of course, there should also be constraints regarding the speed, acceleration, and possible navigational regions. The functional form of f can be very complex and there is no reason to assume nice properties of f such as convexity.

2. A discretized dynamic programming approach and its deficiency

To address the fuel consumption problem (1), Lo et al. (1991) and Lo and McCord (1995) proposed a discretization-based dynamic programming approach. In particular, Lo and McCord (1995) discretized the latitude, longitude, and time. They considered the longitude as the stage variable, the combination of latitude and time as the state variable, and then applied dynamic programming to find the shortest path from the origin node $(\phi_1, \lambda_1, 0)$ to the destination node (ϕ_2, λ_2, T).

The rationale behind discretization is the implicit assumption that when the discretization grids are fine enough, i.e., when the sizes of all grids approach 0, the optimal solution to the discretized problem converges to the optimal solution of the original continuous problem (1). However, we find that this is not the case for the ship routing problem.

Example 1: Suppose that the fuel consumption rate is proportional to the speed cubed and is independent of the location, time, and heading. Then, the optimal

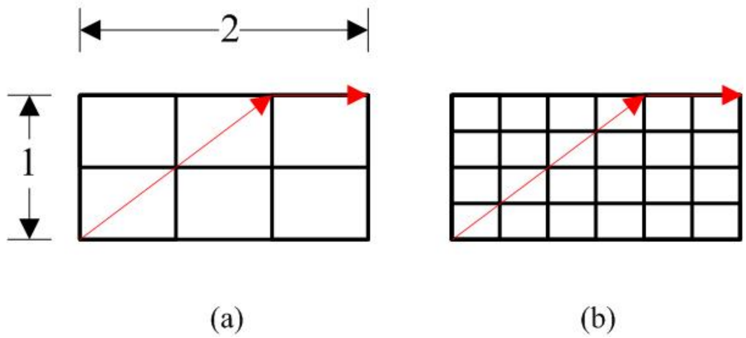

Figure 1: The shortest paths with different numbers of discretization points

sailing speed should be constant over the voyage (Hvattum et al., 2013) and the ship routing problem thus becomes one that finds the shortest path from the origin port to the destination port. Suppose further that the distance between the two ports is small (e.g., in coastal shipping), and hence the surface of the earth can be considered as a plane (Fagerholt et al., 2000). Now consider the 1×2 rectangle in Figure 1 where the bottom-left and upper-right corners are the origin and destination ports, respectively. The shortest distance is $\sqrt{1^2 + 2^2} \approx 2.2361$. We can discretize the latitude into two segments of equal length, and the longitude into three segments of equal length, as shown in Figure 1a. The shortest path is represented by the arrows, and its length is $\sqrt{1^2 + (\frac{4}{3})^2} + \frac{2}{3} \approx 2.3333$. If we divide the rectangle into 4×6 smaller ones, as shown in Figure 1b, the shortest path is still 2.3333. In fact, if the rectangle is divided into $2n \times 3n$ smaller ones of the same size, then the shortest path is always 2.3333 even when n approaches infinity.

Example 2: Suppose that the fuel consumption rate is proportional to the speed cubed and there is no obstacle between the origin and destination ports. Hence, the optimal route is the shortest path, which is the great circle distance. The great circle distance $d = r\Delta\sigma$, where $r = 6371$ km is the radius of the earth, and $\Delta\sigma$ is the central angle between the two ports, which can be calculated by the Haversine formula (Haversine, 2014):

$$\Delta\sigma = 2\arcsin\left[\sqrt{\sin^2\left(\frac{\phi_1-\phi_2}{2}\right)+\cos\phi_1\cos\phi_2\sin^2\left(\frac{\lambda_1-\lambda_2}{2}\right)}\right]. \tag{2}$$

Suppose $\phi_1 = 0$, $\lambda_1 = 0$, and $\phi_2 = \frac{\pi}{6}$, $\lambda_2 = \frac{\pi}{4}$. Eq. (2) calculates that the shortest distance is $0.911738291r$. We uniformly discretize the latitude and longitude into the same number of segments, and then use dynamic programming to find the shortest path. The results are shown in Table 1. We can see that there is a $\frac{0.914869682-0.911738291}{0.911738291} \approx 0.34\%$ gap even for very fine grids.

Table 1: The shortest distances with different numbers of discretization points

Discretization	Shortest distance
2×2	$0.913999751r$
5×5	$0.914726881r$
10×10	$0.914833858r$
20×20	$0.914860721r$
50×50	$0.914868251r$
100×100	$0.914869327r$
200×200	$0.914869596r$
500×500	$0.914869672r$
1000×1000	$0.914869682r$

3. An improved dynamic programming approach

As there is no special property of function f in Eq. (1), we could not expect to devise an efficient method to obtain an optimal solution. Example 1 and Example 2 demonstrate that simply refining the grids in the dynamic programming approach may not be useful. We thus propose an improved dynamic programming approach that incorporates a different discretization method to obtain better solutions.

For preciseness, we use the shortest path problem in Example 2 to describe the algorithm.

Improved dynamic programming approach to Example 2

Step 0. (Initialization): Define M as the number of discretization segments for the longitude and N the one for the latitude. Consider the longitude as the stage variable. Stage $m = 1, 2, \cdots, M$ corresponds to longitude $\frac{\pi}{4}\frac{m-1}{M}$. The maximum and minimum latitudes at stage m are $l_m^{\max} = \frac{\pi}{6}$ and $l_m^{\min} = 0$, respectively. The latitude of state n at stage m, i.e., node (m, n), is $l(m, n) = \frac{\pi}{6}\frac{n-1}{N}$, $n = 1, 2, \cdots, N+1$. Define $\pi(m)$ as the optimal state at stage m. The value of $\pi(m)$ is to be obtained. Define $\kappa = 0$ as the number of iterations that have been implemented and K as the maximum number of iterations.

Step 1. (Dynamic programming): Use the conventional dynamic programming approach to find the shortest path from node $(1, 1)$ to the destination port. The shortest path is recorded by the vector $(\pi(1), \pi(2) \cdots \pi(M))$.

Step 2. (Re-discretize):

Step 2.1. (Truncate the latitude interval at each stage): For each stage m, let $\gamma_m := \frac{l_m^{\max} - l_m^{\min}}{4}$, and update $l_m^{\max} \leftarrow \min(l_m^{\max}, l(m, \pi(m)) + \gamma_m)$ and $l_m^{\min} \leftarrow \max(l_m^{\min}, l(m, \pi(m)) - \gamma_m)$.

Step 2.2. (Keep the optimal path for the next iteration) For each stage m, set $l(m, N+1) = l(m, \pi(m))$.

Step 2.3. (Uniformly discretize the latitude): For each stage m, uniformly discretize the interval $[l_m^{\min}, l_m^{\max}]$ into $\frac{N}{2}$ segments, and record the $\frac{N}{2}+1$ latitudes in $l(m,1), l(m,2) \cdots, l(m, \frac{N}{2}+1)$.

Step 2.4. (Randomly discretize the latitude): Randomly choose $\frac{N}{2}-1$ points from the interval $[l_m^{\min}, l_m^{\max}]$, and record them in $l(m, \frac{N}{2}+2), l(m, \frac{N}{2}+3) \cdots, l(m, N)$.

Step 3. (Check the stopping criterion): Set $\kappa \leftarrow \kappa+1$. If $\kappa \geq K$, return the shortest path and stop. Otherwise go to Step 1. $\square$

The idea of the algorithm is as follows. When $\kappa = 0$, we have no information on the optimal path. Therefore, we discretize the latitude uniformly (Step 0) and use the conventional dynamic programming method to find the optimal path to the discretized problem (Step 1). As shown in Example 1 and Example 2, the obtained path may not be optimal for the continuous problem but should not deviate from the optimal one too much. Therefore, we no longer consider the paths that considerably deviate from the obtained one (Step 2.1). We keep the current solution for the next iteration (Step 2.2) so that the solution will not deteriorate as the algorithm progresses. We then re-discretize the reduced latitude interval for each stage. To avoid the deficiency demonstrated by Example 1 and Example 2, half of the discretization points are generated in a uniform manner (Step 2.3) and the other half in a random manner (Step 2.4). We invoke the dynamic programming approach K times.

3.1. Numerical example

We apply the improved dynamic programming approach to Example 2, with $M = 50$, $N = 50$, and $K = 10$. The results are shown in Table 2. We can see that the gap is only $\frac{0.911738318-0.911738291}{0.911738291} \approx 3 \times 10^{-6}\%$ after ten iterations.

Table 2: The shortest distance in each iteration

Iteration	Shortest distance
1	$0.914868251r$
2	$0.913632967r$
3	$0.912151598r$
4	$0.911857860r$
5	$0.911763365r$
6	$0.911746361r$
7	$0.911740016r$
8	$0.911738735r$
9	$0.911738389r$
10	$0.911738318r$

3.2. Computational complexity

The improved dynamic programming approach is very efficient: its overall computational complexity is KMN^2. Therefore, when $M = 50$, $N = 50$, and $K = 10$, the computational time is comparable to a case with $M = N = 50 \times 10^{\frac{1}{3}} \approx 108$ in Table 1 solved by the conventional dynamic programming approaches.

3.3. Discussions

Without considering ocean currents, wind, wave, and obstacles between the origin and destination ports, the optimal routing of a ship simply involves sailing along the great circle. The improved dynamic programming approach outperforms conventional dynamic programming methods when such realistic factors are incorporated into making the optimal routing decisions. Note that since the grids in dynamic programming is very fine, the weather and sea conditions in one grid can be considered as constant and hence the fuel consumption from one node to another is calculated simply based on the speed, distance, and the weather and sea conditions.

Consider Example 2 again. We assume that the weather condition changes from good to adverse with the increase of longitude. In particular, we assume that the fuel consumption in the area whose longitude is between 0 and $\lambda_2/10$ is 1.1 times the distance, between $\lambda_2/10$ and $2\lambda_2/10$ is 1.2 times the distance $\cdots$ between $9\lambda_2/10$ and $10\lambda_2/10$ is 2 times the distance. For this case, we do not have analytical formulae to calculate the minimum fuel consumption. Nevertheless, we can still compare the improved dynamic programming approach with the conventional one. We apply the improved dynamic programming approach to minimize fuel consumption for this example with $M = 50$, $N = 50$, and $K = 10$. The results are shown in Table 3. We can see that the improved dynamic programming approach still improves over the conventional approach.

The uncertain nature of the weather and sea conditions may also be formulated. Suppose that we consider several possible future scenarios of weather and sea conditions with known probabilities. Then, in the improved dynamic programming approach we can use the expected value of the fuel consumption of the voyage from one node to another and thus minimize the total expected fuel consumption

Table 3: The minimum fuel consumption with the conventional and the improved dynamic programming approach

Approach	Discretization	Minimum fuel consumption
Conventional	10×10	$1.399122324r$
	20×20	$1.399178096r$
	50×50	$1.399193745r$
	100×100	$1.399195981r$
	200×200	$1.399196540r$
	500×500	$1.399196697r$
	1000×1000	$1.399196719r$
Improved	50×50	$1.393910091r$

from the origin port to the destination port. Moreover, since the improved dynamic programming approach is efficient, the routing decisions can be frequently updated based on the latest forecast of weather and sea conditions.

4. Conclusions

We have pointed out that when routing ships between two ports, discretizing the space and time may not lead to an optimal solution to the original continuous problem, even when the sizes of the discretization grids approach 0. This poses a challenge to the conventional discretization-based dynamic programming approaches, at least theoretically. To overcome this problem, we have proposed an improved dynamic programming approach. The novelty of the improved method lies in that we repeatedly re-discretize the space in both controlled and random manners to obtain better solutions. The improved dynamic programming approach provided significantly better solutions than the conventional approaches.

References

Bell, M.G.H., Liu, X., Angeloudis, P., Fonzone, A., Hosseinloo, S.H., 2011. A frequency-based maritime container assignment model. Transportation Research Part B 45, 1152–1161.

Bell, M.G.H., Liu, X., Rioult, J., Angeloudis, P., 2013. A cost-based maritime container assignment model. Transportation Research Part B 58, 58–70.

Fagerholt, K., Heimdal, S., Loktu, A., 2000. Shortest path in the presence of obstacles: An application to ocean shipping. Journal of the Operational Research Society , 683–688.

Fagerholt, K., Hvattum, L.M., Johnsen, T.A.V., Korsvik, J.E., 2013. Routing and scheduling in project shipping. Annals of Operations Research 207, 67–81.

Fagerholt, K., Laporte, G., Norstad, I., 2010. Reducing fuel emissions by optimizing speed on shipping routes. Journal of the Operational Research Society 61, 523–529.

Halvorsen-Weare, E.E., Fagerholt, K., 2013. Routing and scheduling in a liquefied natural gas shipping problem with inventory and berth constraints. Annals of Operations Research 203, 167–186.

Haversine, 2014. Haversine formula. Wikipedia. URL: http://en.wikipedia.org/wiki/Haversine_formula.

Hvattum, L.M., Norstad, I., Fagerholt, K., Laporte, G., 2013. Analysis of an exact algorithm for the vessel speed optimization problem. Networks 62, 132–135.

Lo, H.K., McCord, M.R., 1995. Routing through dynamic ocean currents: General heuristics and empirical results in the gulf stream region. Transportation Research Part B 29, 109–124.

Lo, H.K., McCord, M.R., Wall, C.K., 1991. Value of ocean current information for strategic routing. European Journal of Operational Research 55, 124–135.

Norstad, I., Fagerholt, K., Laporte, G., 2011. Tramp ship routing and scheduling with speed optimization. Transportation Research Part C 19, 853–865.

Ronen, D., 2011. The effect of oil price on containership speed and fleet size. Journal of the Operational Research Society 62, 211–216.

Wang, S., Meng, Q., Liu, Z., 2013. Containership scheduling with transit-time-sensitive container shipment demand. Transportation Research Part B 54, 68–83.

Zhen, L., Wang, K., 2015. A stochastic programming model for multi-product oriented multi-channel component replenishment. Computers and Operations Research 60, 79–90.