Journal of Heuristics (2020) 26:801–825
https://doi.org/10.1007/s10732-020-09449-7

A genetic algorithm for finding realistic sea routes considering the weather

Stefan Kuhlemann[1,2] · Kevin Tierney[1]

Received: 29 January 2019 / Revised: 9 July 2020 / Accepted: 11 July 2020 / Published online: 23 July 2020

Abstract

The weather has a major impact on the profitability, safety, and environmental sustainability of the routes sailed by seagoing vessels. The prevailing weather strongly influences the course of routes, affecting not only the safety of the crew, but also the fuel consumption and therefore the emissions of the vessel. Effective decision support is required to plan the route and the speed of the vessel considering the forecasted weather. We implement a genetic algorithm to minimize the fuel consumption of a vessel taking into account the two most important influences of weather on a ship: the wind and the waves. Our approach assists route planners in finding cost minimal routes that consider the weather, avoid specified areas, and meet arrival time constraints. Furthermore, it supports ship speed control to avoid areas with weather conditions that would result in high fuel costs or risk the safety of the vessel. The algorithm is evaluated for a variety of instances to show the impact of weather routing on the routes and the fuel and travel time savings that can be achieved with our approach. Including weather into the routing leads to a savings potential of over 10% of the fuel consumption. We show that ignoring the weather when constructing routes can lead to routes that cannot be sailed in practice. Furthermore, we evaluate our algorithm with stochastic weather data to show that it can provide high-quality routes under real conditions even with uncertain weather forecasts.

Keywords Weather routing · Ship routing · Genetic algorithm · Uncertain weather

✉ Stefan Kuhlemann
stefan.kuhlemann@uni-bielefeld.de

Kevin Tierney
kevin.tierney@uni-bielefeld.de

[1] Universität Bielefeld, Universitätsstraße 25, 33615 Bielefeld, Germany

[2] Universität Paderborn, Warburger Str. 100, 33098 Paderborn, Germany

1 Introduction

Adverse weather conditions pose a significant danger to ships, their crews, passengers, and cargo and represents one of the main causes of delays in the shipping industry (Notteboom 2006). Since over 90% of the world trade is carried by ships (Hoffmann and Sirimanne 2017), considering the weather when finding safe and efficient routes for ships is of utmost importance. Furthermore, finding good routes for ships lowers fuel consumption, leading to lower CO_2 emissions. Finding routes that are not only safe and efficient, but also ensure that the ship arrives on time is a complex task that is difficult to do by hand, especially when considering the weather.

In this paper, we present an approach for finding routes between two points on the globe considering the current weather along the route. For this, we use a real-coded genetic algorithm (GA) with specialized mutation and crossover operators for the weather routing of ships. Our goal is to minimize the fuel consumption of the vessel while respecting a constraint on the latest arrival time at the vessel's destination. Our GA finds realistic, smooth routes that are not restricted to an arbitrary grid and avoid adverse weather conditions, such as strong storms, considering spatial and temporal aspects of the weather.

Our approach contains the following novel components:

1. We provide a general routing algorithm that does not have preset values for the longitudes/latitudes of the waypoints along the path.
2. The speed of the vessel is variable, allowing for, e.g., slow-steaming.
3. The generated routes avoid sharp changes in direction to ensure they can be sailed by even large vessels.

We test our algorithm with instances based on many different geographies and weather conditions. Our results show that the minimization of fuel consumption leads to routes avoiding areas with adverse weather conditions, as weather and sea conditions largely effect the different factors travel time, safety and fuel consumption simultaneously. Due to the high bunker consumption of container vessels, even small percentage improvements lead to high cost savings. However, it is also possible to sail through areas with favorable weather conditions to increase the speed of the vessel. The weather data we use in our algorithm is provided by an industrial partner so that we have reliable, real-life information. Overall, the different instances show a wide range of possible savings potentials of up to 13.9% over ignoring the weather. Furthermore, our results also show that the consideration of weather conditions can have a high impact on the route depending on the weather intensity. Our algorithm is also faster than existing approaches, requiring only a single minute of CPU time even for large instances. For smaller instances, the runtime falls to under 30 s.

This paper is organized as follows. The weather routing problem is described in detail in Sect. 2. In Sect. 3, the main algorithms for weather-dependent routing of vessels are summarized. Our genetic algorithm is presented in Sect. 4. In Sect. 5, we show computational results for the algorithm for different instances. Section 6 summarizes the paper.

2 Weather routing of ships

Consider a decision maker who wants to determine a route for a vessel over the ocean between two ports. The route clearly must avoid landmasses, shallows, and any other undesirable area as specified by a planner [e.g., pirate zones, or Sulphur Emission Control Areas (SECA)]. SECA zones are areas in which the emission of sulphur and sulphur oxides by ships is restricted by law. Planners may also specify a maximum time duration (due date) for the voyage.

We allow vessels to vary their speed along their voyage, thus parameters related to the fuel consumption and impact of weather on the speed of the vessel (draft, displaced volume, waterline length, block coefficient and design speed; the design speed is a speed of the vessel for which the fuel consumption is known.) must be determined. The vessel also has a minimum and a maximum speed between which the speed can be chosen throughout the route. Typically, the problem of weather routing is modeled as a minimum time problem or a minimum fuel consumption problem subject to some constraints (avoiding unfavorable weather, etc.). We solve the problem as a minimum fuel consumption problem, since in the variable speed setting, minimizing only the time leads to expensive solutions with high CO_2 outputs. In this work, we use a model of speed consumption that takes into account wind and waves. It is possible to use a more sophisticated model for the calculation of the fuel consumption, but we note that this has no impact on the algorithmic aspects of our approach.

2.1 Computation of the fuel consumption

Accurately representing the fuel consumption of vessels is critical for making realistic weather-dependent routes. We use the formula

$$F(v) = (v/v^*)^3 \cdot f^* \tag{1}$$

from Brouer et al. (2013) to compute the fuel consumption (subsequently referred to as the *bunker consumption*), where v is the speed of the vessel in knots, v^* is the design speed and f^* is the fuel consumption at the design speed. We split the route into n segments each with a fixed speed v_i. The bunker consumption of the whole route is then calculated by summing the consumption of each segments.

The weather has a strong influence on the *realized speed* of the vessel. We calculate a speed loss coefficient that accepts the parameters of the vessel and the current weather to determine the actual speed. This speed is determined by the direction and speed of the wind and the direction and height of the waves. We use the approach from Larsson and Simonsen (2014) to calculate the speed loss that takes into account both waves and wind simultaneously. The percentage of speed loss is calculated with the following formula (Kwon 2008):

$$v_{loss} = \alpha_{corr} \cdot \mu_{red} \cdot \frac{\Delta v}{v} 100\%, \tag{2}$$

where v_{loss} is the speed loss in percent, α_{corr} is the correction factor for the block coefficient of the vessel and μ_{red} is the weather direction reduction factor. The block coefficient of a vessel is the ratio of the underwater volume of the ship to the volume of a rectangular block having the same length, breadth and depth. The factor α_{corr} depends on the block coefficient of the vessel and its loading condition. A full list of formulas for the calculation of α_{corr} for all different combinations of block coefficients and loading conditions can be found in Larsson and Simonsen (2014). The parameter μ_{red} changes with the intensity of the weather and the angle of the wind/waves with respect to the ship. The speed loss in head weather is given as $\frac{\Delta v}{v}100\%$. The speed loss is represented with Δv and the planned speed with v. The ratio is calculated by one of

$$\frac{\Delta v}{v}100\% = 0.5BN + \frac{BN^{6.5}}{2.7\nabla^{\frac{2}{3}}}, \tag{3}$$

$$\frac{\Delta v}{v}100\% = 0.7BN + \frac{BN^{6.5}}{2.7\nabla^{\frac{2}{3}}}, \text{ or} \tag{4}$$

$$\frac{\Delta v}{v}100\% = 0.5BN + \frac{BN^{6.5}}{22\nabla^{\frac{2}{3}}} \tag{5}$$

depending on the loading condition of the vessel. The parameter BN is the Beaufort number and ∇ is the displaced volume of the vessel. Equation 3 is used for vessels in laden condition and Eq. 4 for vessels in ballast condition. We use Eq. 5 for our experiments as this is the one for container ships and vessels in normal condition. The Beaufort number characterizes the strength of the waves and the wind.

3 Related work

Existing approaches in the literature for ship weather routing can be divided into three groups: exact approaches, single objective heuristic approaches, and multiobjective evolutionary approaches. Exact approaches to solve the problem are especially found in early works, whereas recent approaches are mainly heuristics. The following overview is divided into the three groups of approaches and summarizes the most important ones. We further identify differing objective functions for the approaches. Touati and Jost (2012) partitions the objective functions found in the ship routing literature into three groups: economic, climate/sustainability and regional fairness/health. The most frequent objective functions are the minimization of fuel consumption and distance traveled. The optimization of fuel consumption is also often combined with the minimization of risk related to a route, resulting in a multiobjective optimization problem. In the last part of this section, we present some papers related to the problem of weather routing.

3.1 Exact approaches

One of the first approaches for ship weather routing is the isochrone method (James 1957). It minimizes the travel time of the ship and allows the manual construction of a route. This method is improved by Hagiwara and Spaans (1987), who make it suitable for a computer and include fuel consumption in addition to travel time in the objective function. Another approach for the problem is the calculus of variations method proposed by Haltiner et al. (1962). It is an analytic approach to weather routing that determines the path and the engine power of the ship. There are also many approaches using dynamic programming such as De Wit (1990) and Motte and Calvert (1990) or based on shortest path algorithms such as Montes (2005), Panigrahi et al. (2012), Sen and Padhy (2015) and Mannarini et al. (2016).

3.2 Single objective heuristic approaches

Evolutionary approaches are well suited algorithms for solving path finding problems because they allow the inclusion of a wide range of constraints and objectives into problems that are hard to solve with exact algorithms. The optimization of only one objective makes these approaches additionally fast compared to the multiobjective ones. Walther et al. (2018) propose a genetic algorithm (GA) that supports variable ship speed and minimizes fuel costs. They compare a graph algorithm for a ship weather routing problem with a GA. The exact details of the GA are not clear, however to the best of our knowledge, our GA differs in its domain specific heuristics and variable length encoding.

Wang et al. (2018) propose a real-coded GA with fixed longitudes for the waypoints that make up the solution representation. This approach therefore has a problem finding paths around vertical obstacles, meaning circumnavigating landmasses can be very difficult or even impossible if the path must traverse a high range of latitude in only a small range of longitude. They integrate further constraints in their optimization including a restriction of the sailing area, avoidance of land obstacles and shallow waters, an interval for the ship's speed and weather alarm zones that are caused by severe weather conditions and wave heights exceeding a certain value. Furthermore, they introduce a general mathematical model for the problem. Yuankui et al. (2014) introduce a simulated annealing algorithm minimizing the travel time of the vessel.

3.3 Multiobjective evolutionary approaches

Multiobjective evolutionary approaches allow for the simultaneous optimization of several objectives. Evolutionary algorithms (EA) are the most common multiobjective solution procedures for weather routing, such as the work of Tsou (2010) and Azariadis (2017) who combine an EA with a modified A* algorithm. The A* algorithm is used to compute one-third of the initial population, and the remaining two-thirds are computed randomly. Tsou (2010) calculates routes avoiding obstacles that are given to the evolutionary algorithm as the first population. The EA then tries to find the optimal route incorporating the safety and economy of routes.

Li and Zhang (2017) propose a GA that minimizes the turning variation and fuel consumption of vessels while considering fixed orientations of the vessel at the start and end position for the vessel as their approach is used to find the optimal trajectory in close-range maneuvering. The real-coded GA presented by Maki et al. (2011) supports variable ship speed by varying the propeller revolutions for each segment of the route.

However, the longitudes of the route points are fixed and only the latitudes can be changed, severely limiting the possibilities of the algorithm. This leads to the same problems as the algorithm from Wang et al. (2018). Furthermore, the authors do not mention how to avoid obstacles (land masses, etc.) in the algorithm. Tsou and Cheng (2013) implement an ant colony algorithm that is combined with the crossover and mutation concept of GAs for finding routes while considering weather conditions. If ants pass the same waypoint, a crossover of both routes is conducted.

Krata and Szlapczynska (2012), Szlapczynska and Smierzchalski (2009), Szlapczynska (2013) and Szlapczynska (2015) use a multicriteria weather routing algorithm based on the concept of the Multi-objective Evolutionary Weather Routing Algorithm proposed by Szlapczynska (2007) to solve the problem of finding routes taking into account changeable weather conditions. All these approaches focus on minimizing the passage time, fuel consumption and the risk factor of routes. The constraints in Szlapczynska (2015) include regions to avoid and the variables of the algorithm are the coordinates of the waypoints and the settings of the ship's engine. There is no information given about how the mutation and crossover in this algorithm are performed and therefore the quality of the algorithm cannot be evaluated.

Li et al. (2017), Veneti et al. (2015,2018) apply the NSGA II approach. Veneti et al. (2018) compare it to an implementation of the SPEA (Zitzler and Thiele 1999). Veneti et al. (2015) define a non linear integer programming problem and present a modified version of the NSGA II (Deb et al. 2002). The algorithm uses nodes in a grid to find an optimal path from the origin to the destination. The velocity of the ship is fixed, meaning adverse weather can only be avoided through route changes. Vettor and Guedes Soares (2016) perform their search only with the Strength Pareto Evolutionary Algorithm 2 from Zitzler et al. (2001) to minimize the the fuel consumption, time of arrival and risk related to rough weather. The optimization of the initial routes is done with a version of the grid-based Dijkstra's algorithm and the speed may differ between two waypoints. Within the evolutionary algorithm the mutation only changes one waypoint at once strongly restricting the changes within one iteration. To select the most favorable route the hyperplane strategy distance method also presented by Vettor and Guedes Soares (2016) is used. It is used to select a solution out of a Pareto set according to user preferences specifying the importance of the different objectives.

3.4 Related problems

Several problems exist that are similar to the vessel weather routing problem. One of them is the routing of planes instead of vessels. The basis of the Flight Planning Problem (FPP) as presented by Knudsen et al. (2018) is a directed graph with nodes representing waypoints at different altitudes. The arcs are associated with a resource consumption and costs. These costs depend on the fuel consumed so far and on the weather conditions that depend on the time when the arc is traversed. Therefore, in contrast to our approach, the problem is solved with the help of a graph structure. The variant of the problem presented by Knudsen et al. (2017) assume a fixed altitude for the flight making the problem similar to ours. Normally, there are different flight levels and therefore a third dimension for each waypoint. Furthermore, variations in speed are often not considered in the FPP.

Shortest path problems over roads are also related to our problem [see Madkour et al. (2017) for a survey]. The key difference comes in the freedom of movement at sea as well as the varying speed of the ship. Although cars and trucks can vary their speed while driving, considering this in the planning phase is difficult since crowded roads and tight deadlines prevent speed variation in practice. Furthermore, the weather has much less impact on the quality of solutions and is therefore not included in approaches found in literature. One approach to solve these shortest path problems is the use of contraction hierarchies as presented by Geisberger et al. (2008) and extended in Geisberger et al. (2012). This approach is designed for road network specific structures where the visited points are directly linked, in contrast to the situation on oceans where the connections of the ports are flexible.

Another area of related problems is the path planning of Unmanned Aerial Vehicles (UAVs). Arantes et al. (2016), for example, use a genetic algorithm (GA) to plan the paths of drones and mention no-fly zones and obstacles that are similar to the land masses in the weather routing problem. They include position uncertainty of the aircraft due to turbulence, which does not occur in the weather routing problem. The multi-population genetic algorithm is combined with a visibility graph maintaining all feasible paths for the drone, which would not be possible for the weather routing problem because of the much larger solution space. Hasircioglu et al. (2008) also use a GA to plan paths offline for UAVs. The GA uses three different mutation operators that update, insert and delete control points visited by the UAV. Hence, there are no operators that are specific to the case of UAV path planning. In contrast to this, we are using customized operators involving the weather and the speed of the vessel when changing a solution to achieve better improvements than with general operators. Ragusa et al. (2017) also investigate a GA for "micro aerial vehicles". The algorithm is similar to the approach presented by this paper for finding routes for an initial population for our GA. However, for the problem of weather routing, intersections are infeasible, which is different than in the approach of Ragusa et al. (2017). In their approach, intersections are allowed and the algorithm focuses on minimizing the degree of intersection with obstacles. Furthermore, the routes used by Ragusa et al. (2017) have fewer waypoints than in the case of ship routing, because the routes are much shorter and need fewer waypoints.

4 A genetic algorithm for realistic weather-dependent routes

We use a genetic algorithm (GA) to find weather-dependent routes for vessels. GAs provide an ideal framework for weather routing for a couple of reasons. First, GAs allow us to combine parts of routes in a natural way in the hopes of forming a high quality solution. Second, the population of a GA offers diversity that is important for avoiding local optima that can be induced by large storms or weather systems. Our GA uses a variable length, real-valued solution representation in which each individual is composed of waypoints, each characterized by a longitude, latitude and the planned speed of the vessel on the segment preceding the waypoint. The set A defines the arcs between the waypoints.

We fill the initial population with feasible routes that have random waypoints added to ensure diversity (Sect. 4.1). The crossover and mutation operators used within our algorithm are adapted to the weather routing problem and we use multiple operators, unlike in standard GA implementations (Sects. 4.2 and 4.3). The fitness function used for evaluating each individual of the population is the bunker fuel required to sail the route under the given weather conditions, and the computations for this are given in Sect. 2. We further add penalty costs to the fitness function for sailing through undesirable areas (e.g., areas with pirates) or not meeting temporal constraints. The GA terminates when the improvement between two iterations falls below a threshold value or after a given number of iterations. The objective function used to evaluate a solution is represented as

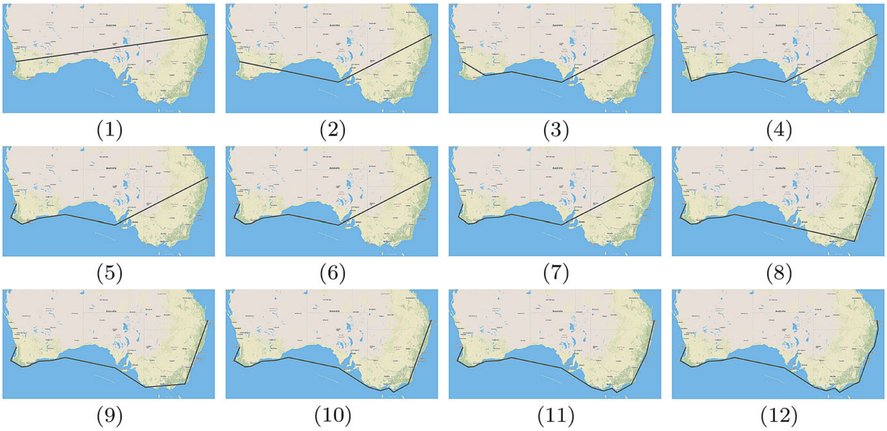
$$minimize \sum_{a\in A} \left(c^{a} \cdot F^{arc}(a)\right) + p^{pirate} \cdot y^{pirate} + p^{delay} \cdot t^{delay} \tag{6}$$

where $F^{arc}(a) = \begin{cases} \infty, & \text{if the wave height on arc } a \text{ exceeds } h \\ \frac{F(v_a^{plan})}{24} \cdot \frac{d_a}{v_a^{real}}, & \text{otherwise.} \end{cases}$

The set of all arcs used to construct the route from the start to the destination is denoted as A. The binary variable y^{pirate} is set to 1 if the route has at least one arc intersecting a pirate zone and the vessel sails slower than 18 knots. Furthermore, t^{delay} specifies the delay of the vessel in days if a time limit for the travel time exists. The penalty for additional safety costs is given by p^{pirate} and the parameter p^{delay} specifies the penalty for not meeting the arrival deadline, if one is specified. The function $F^{arc}(a)$ calculates the fuel consumption for a single arc between two way points of a complete route, where v_a^{plan} is the planned speed, d_a is the length and v_a^{real} specifies the real speed calculated with the loss factor v_a^{loss} for arc a. When the maximum acceptable wave height h is exceeded for an arc, the fuel consumption is set to ∞, because solutions containing such arcs are infeasible and must therefore be ignored. For feasible solutions, the fuel consumption is multiplied by the cost of fuel per ton c^a to determine its cost for the objective function.

We define two types of high cost zones: areas with increased risk of pirate attacks, and the SECA zone in the North Sea. We specify four different pirate zones based on public information about pirate encounters: the Caribbean Sea, the Gulf of Guinea, around Somalia/the Horn of Africa/the Gulf of Bengal, and the South China Sea. In these zones, it is necessary to sail at a speed of at least 18 knots as recommended by The Baltic and International Maritime Council (2011) or to pay additional safety costs of 50,000 USD, as for example suggested by Wrede (2013). Therefore, p^{pirate} is set to this value for our calculations. In the SECA zone, we assume more expensive fuel is used and adjust the bunker consumption function of the vessel accordingly.[1] The cost factor c^a is set to 450 USD except in the SECA zone where it is 850 USD. The penalty p^{delay} for arriving too late is set to 25,000 USD and is calculated by multiplying a

[1] With the adoption of scrubbers and low/no-sulpher engine technologies, the SECA zone may not be relevant to all ships. It can be easily ignored in the GA when this is the case.



(1) (2) (3) (4)

(5) (6) (7) (8)

(9) (10) (11) (12)

Fig. 1 Visualization of the generation of an initial route from Perth, Australia to Brisbane, Australia in 12 steps

penalty of 100 USD per container per day, as suggested by Li et al. (2015), with an assumed number of transported containers of 2500, which is the half of the capacity of the vessel we use. The parameter h is set to 9 m corresponding to a Beaufort number of 10 for our calculations.

4.1 Initial route generation

Our initial solution algorithm creates a route by first directly connecting the origin and destination and iteratively moving the midpoint of the line segments over land orthogonally into the water until no segments intersect with the land anymore. Note that a buffer zone could also be specified around land to ensure that ships travel further offshore. More specifically, a segment is divided into two parts by inserting and moving its midpoint. This is done until the distance between the start and end of a new segment falls below a set threshold, and therefore does not need be divided anymore. This process is visualized in Fig. 1. The left half of the route is arbitrarily chosen to be moved into the water before the right half (note that the order does not matter), leading to the evolution of a route as shown. To generate individuals for the initial population, this algorithm is used to find a route that has to visit one random point between the starting and the target point. The pseudocode for this algorithm is given in Algorithm 1. The algorithm accepts a start node s, an end node e, a threshold t for the length of segments and a distance d for the movement of middle points. The algorithm returns the node sequence from the start node to the end node, with the end node missing, which therefore still needs to be added. We use this algorithm to generate some initial solutions and then use the crossover and mutation procedures to expand the initial population so that the first generation has the full population size. Start and end points of connections generated with this algorithm can be chosen arbitrarily, however, we only use ports for our computational experiments.

Algorithm 1 Initial Route Generation

```
function IRG($s, e, t, d$)
    $N \leftarrow \emptyset$                                          ▷ Sequence of nodes along path
    if DISTANCE($s, e$) $> t$ then
        $m \leftarrow$ MIDPOINT($s, e$)
        $v \leftarrow$ vector orthogonal to the connection $s$ to $e$ with length 1
        $i \leftarrow 0$
        while $m$ is over land do
            for $j \in \{1, -1\}$ do
                $m' \leftarrow m + v \cdot i \cdot d \cdot j$
                if $m'$ is in water then $m \leftarrow m'$
            $i \leftarrow i + 1$
        $N \leftarrow N \cup$ IRG($s, m, t, d$) $\cup$ IRG($m, e, t, d$)
    else
        $N \leftarrow N \cup \{s\}$
    return $N$
```

4.2 Crossover operators

In this step of the GA, the routes of two random individuals from the current population are combined into a single, new route. The selection of the individuals is based on a roulette wheel selection with quadratic fitness scaling. We use two different crossover methods. The first combines the routes by choosing a point to join the routes somewhere near the middle of each route, preferably at a position in which both routes are near each other. This operator is similar to the one from Vettor and Guedes Soares (2016) who combine the first k waypoints of a route with the waypoints of a second route from $k + 1$ to the end, but do not search for a waypoint in the middle of the route first. The second combines the routes using a random position from the first half of one route and a random position from the second half of the other route. Veneti et al. (2015,2018) also use a crossover operator combining routes at random points, but it requires the same node to be contained in both parent routes to perform a recombination, meaning it is rarely applicable. Furthermore, for both of our operators we post-process the route to ensure that it is completely located in the water using the same procedure as for the initial routes. The result of one of our crossovers is visualized in Fig. 2 by means of the website geojson.io, in which the two black routes are combined to obtain the red one. We use the great circle distance[2] for the length of all routes presented in this paper, but in the visualizations the way points are connected with straight lines for ease of visualization.

4.3 Mutation operators

The mutation operators are used to make different changes to random individuals of the population. These operators delete or move waypoints or change the speed at

[2] The great circle distance specifies the shortest distance between two points on the surface of a sphere. The distance of two points on the earth is calculated by $d = 2r \cdot \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2-\phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2-\lambda_1}{2}\right)}\right)$, where ϕ_1 and ϕ_2 are the latitudes, λ_1 and λ_2 are the longitudes of the two points and r is the radius of the earth.

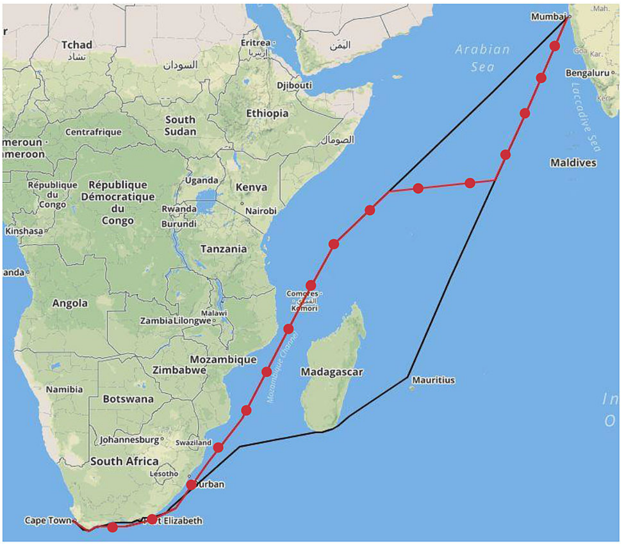


Fig. 2 Visualization of the crossover operator

waypoints and allow the GA to search for new solutions that are not just combinations of existing individuals. We use several different operators to introduce a domain-specific heuristic for dealing with weather conditions and obtain realizable routes. The operator applied to an individual is selected uniformly at random from the list of nine operators.

Deleting a single point/deleting multiple points We first introduce two simple mutation operators that delete a single point or multiple points within a given interval, respectively. The now disconnected parts of the route are reconnected with the initial route algorithm to avoid any landmasses.

Moving a single point/moving multiple points Our second set of operators tries to move one point or an interval of a route by a random distance limited by a parameter in a random direction. Should any part of the route now intersect with land we use the same procedure as in the initial route algorithm to repair the route.

Moving a point with the maximum wind This operator takes the current weather on the route into account. In the first step the position along the route with the maximum influence of the wind on the vessel is determined. In the second step this waypoint is then moved in a randomly determined direction. Afterwards, the route points around the moved one are recalculated with the initial route algorithm to smoothen the route.

Moving a point with the maximum angle/Moving multiple points with the maximum angle Routes with sharp angles are difficult to realize in practice, as the turning radius of large vessels is limited. This operator thus tries to remove points or multiple points with large angles from the route. We select the point with the maximum angle and remove a given percentage defined by a parameter of all nodes in each direction. Afterwards, the initial route algorithm is used to adjust the route should it end up over land. This process can be repeated for multiple points with large angles. This leads to the second operator of this group that moves the points around the several largest angles with one application.

Mutating the speed of the vessel Our final operator adjusts the speed of the vessel for a segment of the route. The new speed is randomly determined in the interval between 80 and 120% of the current speed in the segment.

4.4 Stochastic optimization

We adjust the objective function to allow the GA to handle stochastic weather. Thus, instead of optimizing the fuel consumption for only a single weather scenario, we evaluate a route given multiple scenarios, giving us an expected fuel consumption. The multiple weather scenarios are forecasts for the true weather and are generated by modifying the historical weather data. The generation of scenarios is discussed in detail in Sect. 5.5. We note that this could be adjusted by risk averse users to use, e.g., conditional value at risk or other functions.

5 Computational results

In this section, we present computational results for our approach tested on 15 instances that we have constructed. The scenarios model various times of departure, starting/ending locations and travel times. We present two different weather routing settings: one with perfect information about the weather data and one where we plan under uncertainty. The first setting shows the performance of our approach under perfect information. In the second case, we show how the algorithm performs in a more realistic situation. We are able to conclude from both settings that including weather is critical for generating realistic routes. We implement the GA in C# and run it on a computer with an Intel Core i7-7700K 4.2 GHz processor and 32GB of RAM. The computation time for the solutions for all instances is less than 1 min. Every instance has been run 5 times and we report the average of these runs. Furthermore, we provide the standard deviation of the values for the fuel consumption of the five solutions. We cut the algorithm off at 60 CPU seconds or when the improvements within a single iteration become too small after a minimum number of iterations have been completed. We note that letting the GA run longer than 60 s can sometimes lead to slightly improved results on long routes.

5.1 Parameters for the GA

Our GA includes a number of parameters that influence the quality of the solutions found. Table 1 lists these parameters and the values used. The values have been determined using the GGA algorithm configurator (Ansótegui et al. 2009, 2015). We tune our GA with GGA for 5 days using 45 instances in our training set covering different departure times and locations. We use a Panamax vessel for all of our experiments, but it is also possible to run our algorithm with other vessel types having different vessel-specific parameters.

Table 1 Parameters for the GA

Parameter	Value
Population size	194
Mutations per iteration	529
Number of chromosomes used for crossover	43
Number of chromosomes used for mutation	34
Crossovers per iteration	40
Number of initial routes	31
Minimal number of iterations	130

5.2 Weather data

The weather data used in our experiments is provided by an industrial collaborator. It includes three months of weather information in a 0.5° × 0.5° grid with a 3 h interval for the time from August to October 2017. For each latitude, longitude, and timestamp, we are provided the wind direction and the wave height.

5.3 Experimental results for weather data with perfect information

We create 15 problem instances and try to include coverage of routes in a variety of locations around the world. Our instances also have a number of different weather conditions, ranging from “normal” inclement weather to Hurricane Irma in the instance from USNYC (New York, USA) to SRPBM (Paramaribo, Suriname). The computational results for all 15 instances using perfect information about the weather (one weather scenario) are listed in Table 2. The average of the five runs ignoring the weather is compared to the runs including the weather in terms of route length in nautical miles (nm), duration in days (d) and fuel consumption (FC) in tons (t) of fuel. The results of the tests with and without weather optimization (WO) can be found in the two rows for each instance and the difference of the values is given below each pair of values. The duration and fuel consumption were calculated both with and without the influence of weather to evaluate the quality of the routes leading to the two columns for each of the two key figures. The instances with more extreme weather (Beaufort scale 9 or higher) are marked with a star (*). The solutions for all instances are visualized in Fig. 5. Weather-optimized routes are colored black and non-weather-optimized routes gray. The map is shaded based on the influence of the weather (dark red is worse weather, meaning stronger winds and higher waves) over the entire time period. The weather is visualized for the points in time when the ship following the weather-optimized route traverses the area or is near the visualized area.

In general, considering the weather leads to less costly routes with shorter travel times. In only one case, the instance from ZACPT (Cape Town, South Africa) to INBOM (Mumbai, India), including the weather leads to a route with less fuel consumption even when evaluating it without weather because the combination of the shorter route west of Madagascar and the strong wind and high waves along that path

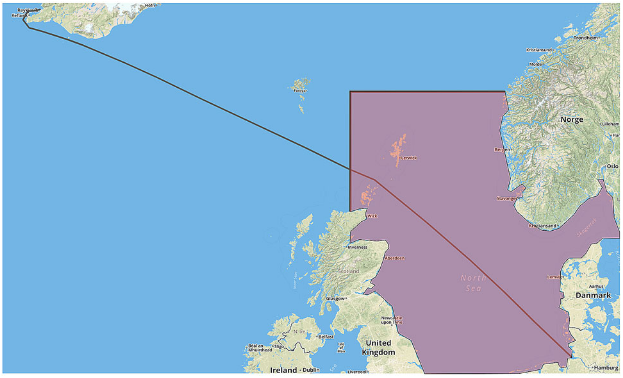


Fig. 3 Route from Hamburg to Reykjavík and the SECA-zone

make it hard for the GA to move its path to the east of Madagascar when generating a route without considering the weather.

For some scenarios there is only a small change in the vessel routing when weather is considered, usually when regions with adverse weather can be easily avoided. However, there are also instances where the weather has a decisive influence on the routing. The instances from USNYC (New York, USA) to SRPBM (Paramaribo, Suriname) and from AUPER (Perth, Australia) to AUBNE (Brisbane, Australia) show that a path without the influence of weather would lead to routes that are not feasible in reality due to very strong wind and high waves, leading to an infinite value for the fuel consumption and the duration. On the route from New York to Paramaribo we have the strongest weather conditions of all instances (11 on the Beaufort scale). Therefore, the longer routes are necessary here to guarantee safety for the vessel and lead to huge improvements. The instances located in the Mediterranean region (ITTRS to EGALY, ESALG to EGALY, and NLRTM to FRMRS.) show that taking the weather into consideration also makes sense for short/medium distance routes. Our three instances in this region show reductions of the fuel consumption of up to about 3%. The SECA-zone in the North Sea does not have a high impact on the routing in the affected instances because it is not possible to leave it to save costs.

In the instance from DEHAM (Hamburg, Germany) to ISREY (Reykjavík, Iceland) it is not efficient to leave the SECA-zone on a route other than the generated one because the zone reaches far north. The route and the zone are visualized in Fig. 3. One instance with a relatively small difference between the weather-optimized and the non-weather-optimized path is the connection from USDUT (Dutch Harbor, USA) to USLAX (Los Angeles, USA). Despite this small deviation in the routing, it leads to an improvement in travel time and fuel consumption of approximately 1.6% when comparing the weather-optimized route with the non-weather-optimized route. The weather-optimized route is visualized in Fig. 5a in black.

The routes running from ZACPT (Cape Town, South Africa) to INBOM (Mumbai, India) show one of the biggest differences as the weather-optimized routes run east of Madagascar while routes ignoring the weather run west of it. This is the case because the western routes are shorter, but the weather to the west is worse than to the east. This is visualized in Fig. 5d. This leads to an average improvement of fuel consumption by almost 7% considering the weather during optimization. The routes from USNYC (New York, USA) to DEHAM (Hamburg, Germany) show a similar observation with routes running north and south of the United Kingdom depending on the use of weather data. The weather-optimized routes run south of the United Kingdom although this path is longer because the weather conditions in the north are worse than in the south. Another important aspect for the routing in the Arabian Sea is the pirate zone that covers a large part of it. The weather-optimized routes, which already run east of Madagascar, take a right turn to bypass the zone as far as possible, while the routes running west of Madagascar run directly through it. This can be seen in Fig. 4, which shows the pirate zone. The routes from AEDXB (Dubai, UAE) to AUPER (Perth, Australia) and the routes from MYTPP (Tanjung Pelepas, Malaysia) to OMSLL (Salalah, Oman) are not visibly affected by the zone because the pirate zone covers the complete coast and cannot be avoided. The greatest savings potential can be found for the last instance from JPTYO (Tokyo, Japan) to GUGUM (Guam), in which an approximately 4.3% longer distance leads to savings savings of fuel and travel time of nearly 14%.

Overall, the results for the instances using perfect information about the weather show that there is a huge savings potential when including weather conditions into the optimization of routes for vessels. The improvements range from about 1% to almost 14% excluding the instances for which infeasible routes were generated when not considering weather.

5.4 Pirate zones and travel time limitations

In addition to higher speeds in pirate zones, the objective function can be penalized to account for security services (e.g., armed mercenaries). Increasing the costs within pirate zones by 10%, as well as requiring a higher speed leads to routes similar to the one in Fig. 4. This route has been generated without considering the influence of weather conditions to show the effect of the increased costs for a higher level of security. Our approach generates a solution that completely avoids the pirate zone marked in orange, and requires no modifications to our heuristics to identify such zones.

Another aspect that can be included into the routing problem is the addition of penalty costs for a delayed arrival at the destination port. Very high penalty costs can be used to force the algorithm to meet the deadline at any cost (if it is at all possible). Therefore, the course of the routes are changed for some instances and the speed on the routes must be increased in most cases.

5.5 Experimental results for stochastic weather data

In reality, reliable weather forecasts are only available for the beginning of a route. Thus, we must plan a route under uncertainty and replan whenever a new weather forecast is available. Table 3 shows the results for the previously given instances using multiple stochastic weather scenarios instead of planning under certainty for

Table 2 Comparison of routes computed considering weather to ones computed ignoring the weather regarding length in nautical miles duration in days and fuel consumption in tons, averages over five runs

			Evaluated duration		Fuel consumption	
		Length	Ignoring weather	Including weather	Ignoring weather	Including weather
USDUT	Without WO	2374.54	8.25	8.90	128.39	138.49
to	With WO	2376.28	8.26	8.76	128.49	136.29
USLAX	Difference	0.1%	0.1%	− 1.6%	0.1%	− 1.6%
USNYC	Without WO	2295.97	7.98	∞	124.14	∞
to	With WO	2324.02	8.07	8.74	125.78	136.24
SRPBM*	Difference	1.2%	1.2%	− 100%	1.3%	− 100%
USNYC	Without WO	3404.30	11.83	14.21	184.07	221.03
to	With WO	3537.17	12.29	13.76	191.26	214.02
DEHAM*	Difference	3.9%	3.9%	− 3.2%	3.9%	− 3.2%
ZACPT	Without WO	4631.28	14.36	16.63	351.80	396.48
to	With WO	4690.69	14.91	16.86	332.74	368.87
INBOM	Difference	1.3%	3.9%	1.4%	− 5.4%	− 7.0%
AUPER	Without WO	2494.42	8.67	∞	134.87	∞
to	With WO	2563.17	8.91	9.63	138.59	149.89
AUBNE*	Difference	2.8%	2.8%	− 100%	2.8%	− 100%
AEDXB	Without WO	4877.69	15.92	17.90	324.28	359.06
to	With WO	4954.76	16.20	17.32	327.34	346.78
AUPER	Difference	1.6%	1.8%	− 3.2%	1.0%	− 3.4%
MYTPP	Without WO	3172.52	8.57	9.72	314.87	349.00
to	With WO	3188.60	8.60	9.48	316.92	345.24
OMSLL	Difference	0.5%	0.4%	− 2.5%	0.7%	− 1.1%
BRSSZ	Without WO	3407.19	11.84	13.25	184.23	206.09
to	With WO	3433.05	11.93	12.68	185.68	197.32

Table 2 continued

			Evaluated duration		Fuel consumption	
		Length	Ignoring weather	Including weather	Ignoring weather	Including weather
ZACPT	Difference	0.8%	0.8%	− 4.3%	0.8%	− 4.3%
ITTRS	Without WO	1183.40	4.11	4.46	63.99	69.33
to	With WO	1197.44	4.16	4.32	64.75	67.26
EGALY	Difference	1.2%	1.3%	− 3.0%	1.2%	− 3.0%
ESALG	Without WO	1793.84	6.23	6.49	96.99	100.97
to	With WO	1800.52	6.26	6.37	97.38	99.11
EGALY	Difference	0.4%	0.4%	− 1.9%	0.4%	− 1.8%
NLRTM	Without WO	2019.50	7.02	7.88	109.20	122.60
to	With WO	2046.40	7.11	7.77	110.65	120.87
FRMRS	Difference	1.3%	1.4%	− 1.4%	1.3%	− 1.4%
NOOSL	Without WO	555.25	1.93	2.16	30.02	33.55
to	With WO	565.52	1.96	2.06	30.59	32.10
GBFXT	Difference	1.9%	1.8%	− 4.3%	1.9%	− 4.3%
DEHAM	Without WO	302.63	1.05	1.32	16.36	20.47
to	With WO	314.74	1.09	1.22	17.02	19.00
GBFXT	Difference	4.0%	4.0%	−7.1%	4.0%	−7.2%
DEHAM	Without WO	1173.09	4.08	4.83	63.43	75.08
to	With WO	1176.24	4.09	4.73	63.60	73.62
ISREY*	Difference	0.3%	0.3%	− 2.0%	0.3%	− 2.0%
JPTYO	Without WO	1329.21	4.62	6.32	71.87	98.31
to	With WO	1386.67	4.82	5.44	74.98	84.60
GUGUM*	Difference	4.3%	4.3%	− 14.0%	4.3%	− 13.9%

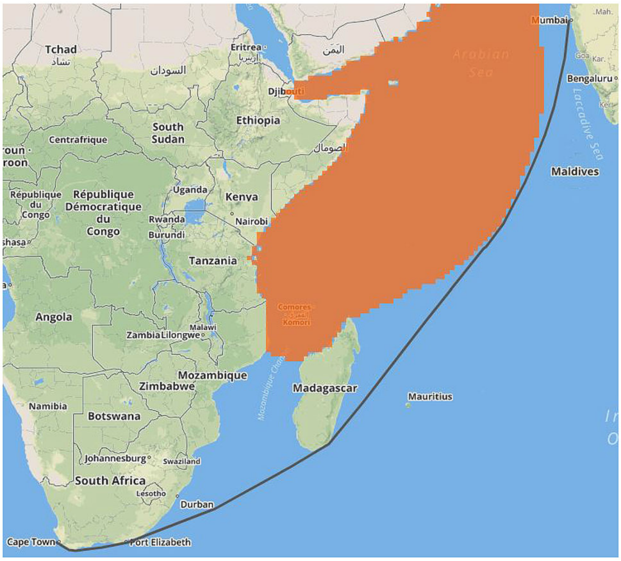
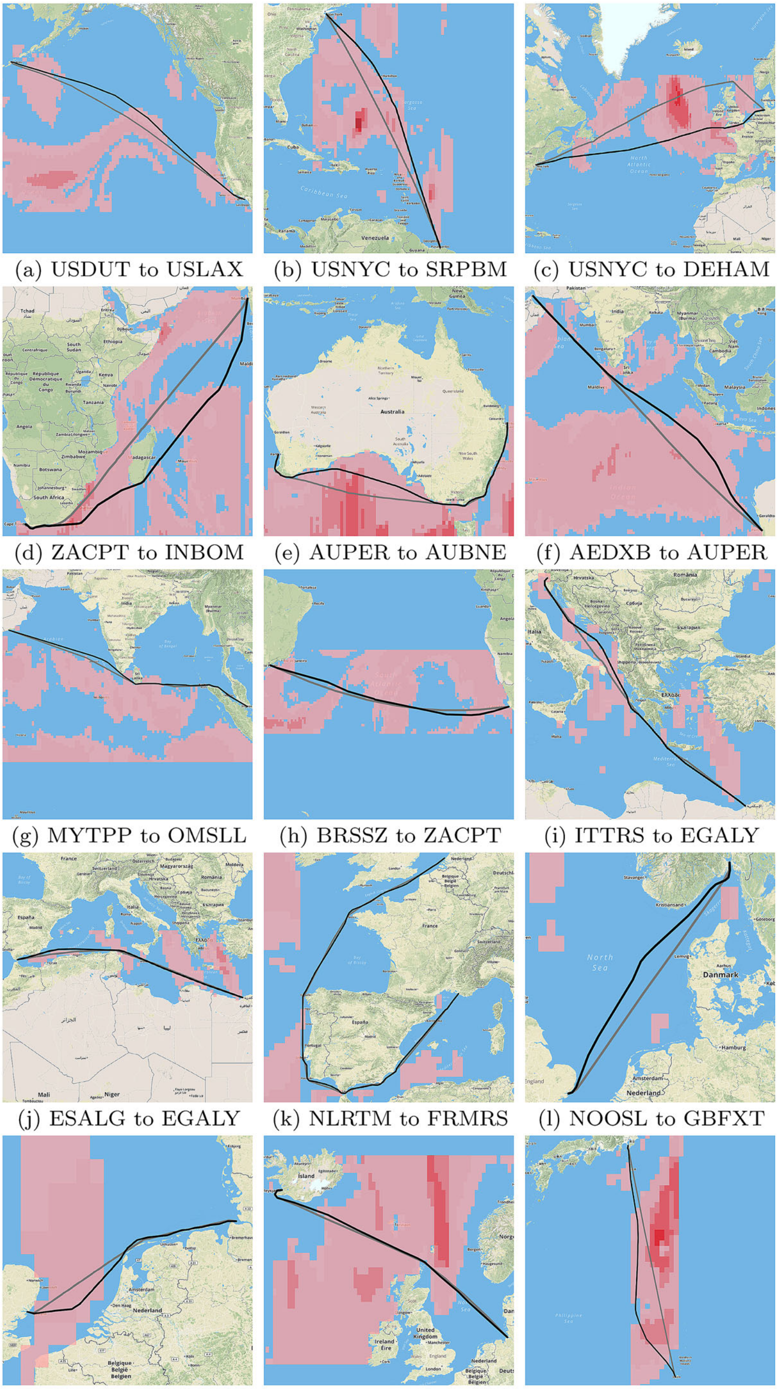


Fig. 4 Route optimized with additional costs for security

a single scenario as shown previously. The route length, duration and fuel usage are computed using the "true" weather, averaging over five different executions of our approach. The route planning uses either perfect information ("PI"), the expected value over five scenarios with replanning ("Stochastic"), the expected value over five scenarios without replanning ("Stoch. NR"), or plans without considering the weather ("No WO"). Percentage differences to the plan with perfect information are given for each value. Furthermore, the standard deviation for the calculated values for the fuel consumption of the five different solutions is given. Figure 6 shows an example for the four different routes for the second instance from USNYC (New York, USA) to SRPBM (Paramaribo, Suriname). The weather is left out for this figure as it differs too much for the different solutions and would be misleading regarding the quality of the routes.

We perform stochastic routing by first generating five potential forecasts of the weather from the "true" weather. This is done by modifying the components (direction, wind speed, wave height) by a random factor between -50% and $+50\%$. We plan a route from the starting location using these five forecasts and minimize the expected value over the scenarios. As the ship moves closer to a given location, the better the forecast becomes. Perfect information is provided for the exact location of the vessel and for all locations within seven days, a forecast is provided for other locations as an average of the "true" weather and randomized forecast weighted by the time needed to arrive at this point.[3] We therefore replan the route in regular intervals. These scenarios are then used to provide the algorithm with information about the possible weather on the route.

[3] We note that while we use real, historical weather data, we do not have detailed forecasts for different locations, hence we use the procedure described for stochastic data.



(a) USDUT to USLAX (b) USNYC to SRPBM (c) USNYC to DEHAM

(d) ZACPT to INBOM (e) AUPER to AUBNE (f) AEDXB to AUPER

(g) MYTPP to OMSLL (h) BRSSZ to ZACPT (i) ITTRS to EGALY

(j) ESALG to EGALY (k) NLRTM to FRMRS (l) NOOSL to GBFXT

(m) DEHAM to GBFXT (n) DEHAM to ISREY (o) JPTYO to GUGUM

Fig. 5 Visualization of the solutions for the instances

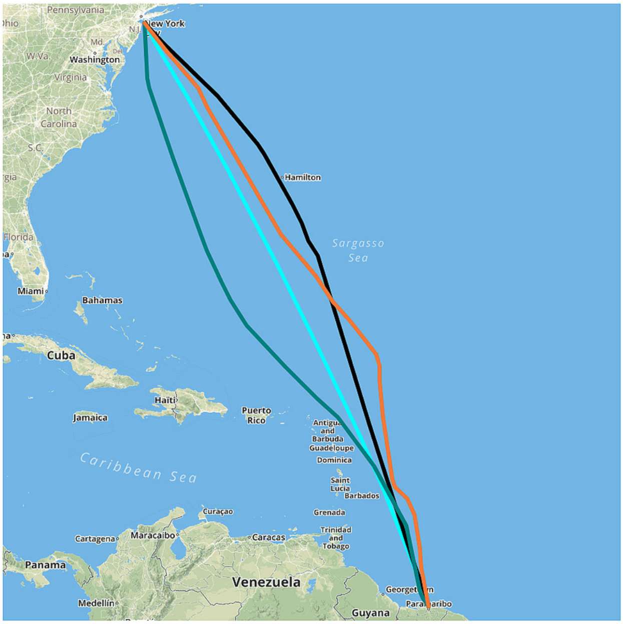


Fig. 6 The four different routes from USNYC (New York, USA) to SRPBM (Paramaribo, Suriname). “PI” in black, “Stochastic” in orange, “Stoch. NR” in green and “No WO” in blue (Color figure online)

Overall, the results presented in Table 3 show that the use of stochastic weather data and replanning the route when new data becomes available leads to lower total fuel consumption than only planning with stochastic data at the start of the route, or planning using no weather data at all. Our approach is generally only a few percent worse than the solution found planning with perfect information. Replanning the routes during the journey leads to shorter travel times for all instances. For two cases, not replanning the route even leads to infeasible routes (ZACPT to INBOM and AUPER to AUBNE) meaning replanning is essential in these cases. The fuel consumption is also improved and the second instance (USNYC to SRPBM) shows the biggest difference with a deviation of 21.2% (without replanning) compared to a deviation of 9.8% (with replanning). We observe that the best route found is often not much longer or shorter than the best route without replanning.

Comparing the results without the usage of weather data to the results with the usage of stochastic weather shows that there are two special cases. For the instance from ZACPT (Cape Town, South Africa) to INBOM (Mumbai, India), the route has a longer travel time caused by the fact that the fuel consumption is the optimization goal and a longer travel time leads to a lower fuel consumption in this case. The only case where the the stochastic results are worse than the results generated without weather is the instance from ESALG (Algeciras, Spain) to EGALY (Alexandria, Egypt), which is due to unstable weather in front of the coast of Egypt.

Using stochastic data leads to longer travel times and higher fuel consumption than using perfect information, but not necessarily to longer routes. For 10 of the 15 instances the routes are longer when using stochastic scenarios instead of the perfect information about the weather. The travel times increase by 0.6–9.3% and the fuel consumption increases by 0.6–9.8%, respectively. Uncertainties regarding the weather conditions on the remaining route make it difficult for the algorithm to find a route that is good for all weather scenarios.

The best example of difficulty in planning the speed and the route of the ship to avoid strong weather with very high waves is the instance from USNYC (New York, USA) to SRPBM (Paramaribo, Suriname). For this instance, the largest deviation of the stochastic results from the results using perfect information is observed. The uncertain forecasts result in the route partly passing through unfavourable areas. Nevertheless, areas that are impassable due to strong waves and wind are avoided. Furthermore, the replanning of the route has a very high impact on the fuel consumption, showing the importance of the adaption to current weather data.

It is essential to adapt the route during the journey, as is exemplified by the instance from ZACPT (Cape Town, South Africa) to INBOM (Mumbai, India). Here, the route runs east of Madagascar because of the strong weather conditions west of it, which we also observed when using perfect information. The same observation can be made for the instance from AUPER (Perth, Australia) to AUBNE (Brisbane, Australia). However, there are many instances where the replanning does not have a large impact, we note that since replanning is computationally cheap, there is no reason not to do it. Usually, replanning has little impact when there are no large storms or extreme weather along the planned route.

It can be concluded that our approach provides high quality solutions when provided stochastic data and can effectively replan the route in the face of adverse weather conditions. We are further able to show that the routes we find are not much worse than those generated with perfect information, meaning our algorithm could be used in a real system.

6 Conclusion

In this paper, we presented a GA for the weather-dependent optimization of routes for vessels. We introduced an algorithm for generating initial routes as a useful supplement for the GA and listed a variety of domain-specific mutation operators for the GA to find good routes adapted to the present weather. The combination of this algorithm and the GA is highly effective at finding weather-dependent routes. Overall, the solutions for the different instances indicate that there is a need for the consideration of weather when generating routes for vessels. Weather-optimized routes lead to lower costs, and in some cases these routes are the only feasible routes. The experimental results showed that the proposed GA is able to find high quality solutions in a short amount of time. We have shown that the algorithm can handle stochastic weather data and generate reasonable routes that are not much worse than routes generated with perfect information. Furthermore, it makes sense to recalculate the route when new weather data becomes available. For future work, we plan to include more external factors, for example tides. The function for the bunker consumption will have to be extended then as well to include the new factors.

Table 3 Comp. results for stochastic optimization with replanning along the route under perfect information ("PI"), stochastic optimization with replanning ("Stochastic"), stochastic optimization with no replanning ("Stoch. NR"), and without weather routing ("No WO")

		Length [nm]		Duration [d]		Fuel [t]		
		Avg.	%	Avg.	%	Avg.	%	σ
USDUT to USLAX	PI	2376		8.76		136.29		0.57
	Stochastic	2385	0.4	8.89	1.5	**138.33**	**1.5**	**1.02**
	Stoch. NR	2406	1.2	9.00	2.7	140.01	2.7	0.45
	No WO	2375	− 0.1	8.90	1.6	138.49	1.6	0.71
USNYC to SRPBM	PI	2324		8.74		136.24		0.61
	Stochastic	2356	1.4	9.56	9.3	**149.53**	**9.8**	**0.85**
	Stoch. NR	2353	1.3	9.34	6.9	165.07	21.2	5.65
	No WO	2296	− 1.2	∞	100	∞	100	0.00
USNYC to DEHAM	PI	3537		13.76		214.02		0.24
	Stochastic	3583	1.3	14.16	2.9	**220.35**	**3.0**	**1.00**
	Stoch. NR	3613	2.2	14.25	3.6	226.78	6.0	2.72
	No WO	3404	− 3.8	14.21	3.3	221.03	3.3	2.35
ZACPT to INBOM	PI	4691		16.86		368.87		9.57
	Stochastic	4938	5.3	18.25	8.3	**381.36**	**3.4**	**8.05**
	Stoch. NR	4631	− 1.3	15.50	− 8.0	∞	100.0	∞
	No WO	4631	− 1.3	16.63	− 1.4	396.48	7.5	5.43
AUPER to AUBNE	PI	2563		9.63		149.89		0.26
	Stochastic	2524	− 1.5	9.82	1.9	**152.71**	**1.9**	**1.17**
	Stoch. NR	2561	− 0.1	9.19	− 4.6	∞	100	∞
	No WO	2494	− 2.7	∞	100	∞	100	0.00
AEDXB to AUPER	PI	4955		17.32		346.78		2.8
	Stochastic	4927	− 0.6	17.84	3.0	**354.56**	**2.2**	**3.1**
	Stoch. NR	4888	− 1.4	17.25	− 0.4	365.56	5.4	2.1
	No WO	4878	− 1.6	17.90	3.3	359.06	3.5	3.5
MYTPP to OMSLL	PI	3189		9.48		345.24		1.58
	Stochastic	3195	0.2	9.61	1.5	**348.29**	**0.9**	**1.60**
	Stoch. NR	3225	1.1	9.65	1.8	352.98	2.2	2.18
	No WO	3173	− 0.5	9.72	2.6	349.00	1.1	2.70
BRSSZ to ZACPT	PI	3433		12.68		197.32		1.08
	Stochastic	3427	− 0.2	13.09	3.3	**203.74**	**3.3**	**3.95**
	Stoch. NR	3428	− 0.1	13.14	3.7	204.45	3.6	0.37
	No WO	3407	− 0.8	13.25	4.5	206.09	4.4	1.03
ITTRS to EGALY	PI	1197		4.32		67.26		0.11
	Stochastic	1195	− 0.2	4.46	3.1	**69.32**	**3.1**	**0.13**
	Stoch. NR	1194	− 0.3	4.46	3.1	69.38	3.2	0.21
	No WO	1183	− 1.2	4.46	3.1	69.33	3.1	0.37

Table 3 continued

		Length [nm]		Duration [d]		Fuel [t]		
		Avg.	%	Avg.	%	Avg.	%	σ
ESALG to EGALY	PI	1801		6.37		99.11		0.45
	Stochastic	1806	0.3	6.83	7.2	106.23	7.2	0.19
	Stoch. NR	1803	0.1	6.85	7.5	106.50	7.5	0.20
	No WO	1794	− 0.4	6.49	1.9	**100.97**	**1.9**	**0.08**
NLRTM to FRMRS	PI	2046		7.77		120.87		0.24
	Stochastic	2047	0.0	7.81	0.6	**121.55**	**0.6**	**0.22**
	Stoch. NR	2055	0.4	7.84	0.9	121.94	0.9	0.08
	No WO	2019	− 1.3	7.88	1.4	122.60	1.4	0.29
NOOSL to GBFXT	PI	566		2.06		32.10		0.14
	Stochastic	565	− 0.2	2.07	0.5	**32.72**	**1.9**	**0.55**
	Stoch. NR	564	− 0.3	2.11	2.2	32.75	2.0	0.65
	No WO	555	− 1.8	2.16	4.5	33.55	4.5	0.12
DEHAM to GBFXT	PI	315		1.22		19.00		0.10
	Stochastic	318	1.1	1.23	0.7	**19.18**	**0.9**	**0.36**
	Stoch. NR	323	2.8	1.24	1.5	19.37	1.9	0.43
	No WO	303	− 3.8	1.32	7.7	20.47	7.7	0.20
DEHAM to ISREY	PI	1176		4.73		73.62		0.11
	Stochastic	1178	0.2	4.78	1.1	**74.39**	**1.0**	**0.55**
	Stoch. NR	1176	0.0	4.78	1.1	74.45	1.1	0.18
	No WO	1173	− 0.3	4.83	2.0	75.08	2.0	0.21
JPTYO to GUGUM	PI	1387		5.44		84.60		0.22
	Stochastic	1380	− 0.4	5.74	5.5	**89.91**	**6.3**	**3.19**
	Stoch. NR	1409	1.6	5.82	7.0	90.54	7.0	0.41
	No WO	1329	− 4.1	6.32	16.2	98.31	16.2	0.77

The line with the smallest deviation from the value for perfect information is marked bold
Percentage gaps relate to perfect information and the standard deviation is given for the fuel

Acknowledgements Open Access funding provided by Projekt DEAL. We would like to thank Lukas Arendt, Tino Engelbrecht, and Stephan Volkmann for their contributions to a student project leading to this work. Furthermore, we thank the Paderborn Center for Parallel Computation (PC^2) for the use of the OCuLUS cluster.

References

Ansótegui, C., Sellmann, M., Tierney, K.: A gender-based genetic algorithm for the automatic configuration of algorithms. In: International Conference on Principles and Practice of Constraint Programming, Springer, pp 142–157 (2009)

Ansótegui, C., Malitsky, Y., Samulowitz, H., Sellmann, M., Tierney, K.: Model-based genetic algorithms for algorithm configuration. In: 24th International Joint Conference on Artificial Intelligence, pp 733–739 (2015)

Arantes, MdS., Arantes, JdS., Toledo, CFM., Williams, BC.: A hybrid multi-population genetic algorithm for UAV path planning. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, ACM, pp 853–860 (2016)

Azariadis, P.: On using density maps for the calculation of ship routes. Evol. Syst. **8**(2), 135–145 (2017)

Brouer, B.D., Alvarez, J.F., Plum, C.E., Pisinger, D., Sigurd, M.M.: A base integer programming model and benchmark suite for liner-shipping network design. Transp. Sci. **48**(2), 281–312 (2013)

De Wit, C.: Proposal for low cost ocean weather routeing. J. Navig. **43**(3), 428–439 (1990)

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)

Geisberger, R., Sanders, P., Schultes, D., Delling, D.: Contraction hierarchies: faster and simpler hierarchical routing in road networks. In: International Workshop on Experimental and Efficient Algorithms, Springer, pp 319–333 (2008)

Geisberger, R., Sanders, P., Schultes, D., Vetter, C.: Exact routing in large road networks using contraction hierarchies. Transp. Sci. **46**(3), 388–404 (2012)

Hagiwara, H., Spaans, J.: Practical weather routing of sail-assisted motor vessels. J. Navig. **40**(1), 96–119 (1987)

Haltiner, G., Hamilton, H., 'Arnason, G.: Minimal-time ship routing. J. Appl. Meteorol. **1**(1), 1–7 (1962)

Hasircioglu, I., Topcuoglu, HR., Ermis, M.: 3-D path planning for the navigation of unmanned aerial vehicles by using evolutionary algorithms. In: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, pp 1499–1506 (2008)

Hoffmann, J., Sirimanne, S.N.: Review of Maritime Transport 2017. United Nations Publication, New York (2017)

James, RW.: Application of wave forecasts to marine navigation. U S Navy Eydrographic Office, Washington DC SP-1 (1957)

Knudsen, AN., Chiarandini, M., Larsen, KS.: Constraint handling in flight planning. In: International Conference on Principles and Practice of Constraint Programming, Springer, pp 354–369 (2017)

Knudsen, AN., Chiarandini, M., Larsen, KS.: Heuristic variants of A* search for 3D flight planning. In: International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research, Springer, pp 361–376 (2018)

Krata, P., Szlapczynska, J.: Weather Hazard Avoidance in Modeling Safety of Motor-Driven Ship for Multicriteria Weather Routing. In: Methods and Algorithms in Navigation Marine Navigation and Safety of Sea Transportation, Weintrit and Neuman (ed), Gdynia Maritime University 6(1) (2012)

Kwon, Y.: Speed loss due to added resistance in wind and waves. Naval. Archit. **3**, 14–16 (2008)

Larsson, E., Simonsen, MH.: DIRECT Weather Routing. Master's Thesis, Chalmers University of Technology, Gothenburg (2014)

Li, C., Qi, X., Lee, C.Y.: Disruption recovery for a vessel in liner shipping. Transp. Sci. **49**(4), 900–921 (2015)

Li, G., Zhang, H.: A Bézier curve based ship trajectory optimization for close-range maritime operations. In: ASME 2017 36th International Conference on Ocean, Offshore and Arctic Engineering, American Society of Mechanical Engineers (2017)

Li, X., Wang, H., Wu, Q.: Multi-objective optimization in ship weather routing. In: Constructive Nonsmooth Analysis and Related Topics (dedicated to the memory of VF Demyanov)(CNSA), 2017, IEEE, pp 1–4 (2017)

Madkour, A., Aref, WG., Rehman, FU., Rahman, MA., Basalamah, S.: A Survey of shortest-path algorithms. arXiv preprint arXiv:1705.02044 (2017)

Maki, A., Akimoto, Y., Nagata, Y., Kobayashi, S., Kobayashi, E., Shiotani, S., Ohsawa, T., Umeda, N.: A new weather-routing system that accounts for ship stability based on a real-coded genetic algorithm. J. Mar. Sci. Technol. **16**(3), 311 (2011)

Mannarini, G., Pinardi, N., Coppini, G., Oddo, P., Iafrati, A.: VISIR-I: small vessels-least-time nautical routes using wave forecasts. Geosci. Model Dev. **9**(4), 1597–1625 (2016)

Montes, AA.: Network shortest path application for optimum track ship routing. Master's Thesis, Monterey, California. Naval Postgraduate School (2005)

Motte, R., Calvert, S.: On the selection of discrete grid systems for on-board micro-based weather routeing. J. Navig. **43**(1), 104–117 (1990)

Notteboom, T.E.: The time factor in liner shipping services. Marit. Econ. Logist. **8**(1), 19–39 (2006)

Panigrahi, J., Padhy, C., Sen, D., Swain, J., Larsen, O.: Optimal ship tracking on a navigation route between two ports: a hydrodynamics approach. J. Mari. Sci. Technol. **17**(1), 59–67 (2012)

Ragusa, VR., Mathias, HD., Kazakova, VA., Wu, AS.: Enhanced genetic path planning for autonomous flight. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp 1208–1215 (2017)

Sen, D., Padhy, C.P.: An approach for development of a ship routing algorithm for application in the North Indian Ocean region. Appl. Ocean Res. **50**, 173–191 (2015)

Szlapczynska, J.: Multiobjective approach to weather routing. TransNav Int. J. Mar. Navig. Saf. Sea Transp. **1**(3), 273–278 (2007)

Szlapczynska, J.: Multicriteria evolutionary weather routing algorithm in practice. TransNav Int. J. Mar. Navig. Saf. Sea Transp. **7**(1), 61–65 (2013)

Szlapczynska, J.: Multi-objective weather routing with customised criteria and constraints. J. Navig. **68**(2), 338–354 (2015)

Szłapczynska, J., Smierzchalski, R.: Multicriteria optimisation in weather routing. Mar. Navig. Saf. Sea Transp. **3**, 423 (2009)

The Baltic and International Maritime Council, et al: Best Management Practices for Protection Against Somalia Based Piracy. Edinburgh: Witherby 4 (2011)

Touati, N., Jost, V.: On green routing and scheduling problem. (2012) arXiv preprint arXiv:1203.1604

Tsou, M.C.: Integration of a geographic information system and evolutionary computation for automatic routing in coastal navigation. J. Navig. **63**(2), 323–341 (2010)

Tsou, M.C., Cheng, H.C.: An Ant Colony Algorithm for efficient ship routing. Pol. Marit. Res. **20**(3), 28–38 (2013)

Veneti, A., Konstantopoulos, C., Pantziou, G.: An evolutionary approach to multi-objective ship weather routing. In: Information, Intelligence, Systems and Applications (IISA), 2015 6th International Conference on, IEEE, pp 1–6 (2015)

Veneti, A., Konstantopoulos, C., Pantziou, G.: Evolutionary computation for the ship routing problem. In: Modeling, Computing and Data Handling Methodologies for Maritime Transportation, Intelligent Systems Reference Library, Springer, Cham, pp 95–115 (2018)

Vettor, R., Guedes Soares, C.: Development of a ship weather routing system. Ocean Eng. **123**((Supplement C)), 1–14 (2016)

Walther, L., Shetty, S., Rizvanolli, A., Jahn, C.: Comparing two optimization approaches for ship weather routing. In: Operations Research Proceedings 2016, Operations Research Proceedings, Springer, pp 337–342 (2018)

Wang, H.B., Li, X.G., Li, P.F., Veremey, E.I., Sotnikova, M.V.: Application of real-coded genetic algorithm in ship weather routing. J. Navig. **71**, 1–22 (2018)

Wrede, I.: Hohe Kosten für Reeder durch Piraterie. https://p.dw.com/p/18XHu, as of 16.03.2020 (2013)

Yuankui, L., Yingjun, Z., Feixiang, Z.: Minimal time route for wind-assisted ships. Mar. Technol. Soc. J. **48**(3), 115–124 (2014)

Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans. Evolut. Comput. **3**(4), 257–271 (1999)

Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK-report 103 (2001)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.