ANNUAL OF NAVIGATION 24/2017

DE GRUYTER OPEN

DOI: 10.1515/aon-2017-0011

MARCIN ŻYCZKOWSKI
Gdańsk University of Technology, Poland

METHOD OF ROUTING SHIPS SAILING IN DEDICATED ENVIRONMENT

ABSTRACT

The method for determining the suboptimal route of sailing vessels operating in a restricted area of the sea are proposed in the paper. The dynamics of the environment including weather conditions and speed characteristics of ships sailing are considered. As optimization criterion, measure sailing time T, and the number of maneuvers performed ω, are taken into account. An heuristic algorithm, and the appropriate application routing for fixed starting points and targets is designed in the method. In addition author analyzed the behavior of the algorithm, depending on the number of direction changes of the course, and granularity of the description of the surface of area.

Keywords:
sailing ship's routing, route optimization, weather routing, unmanned sailing vessel.

INTRODUCTION

The problem of route optimization is known in sailing, but not extensively analyzed by scientists. The first support for vessels sailing, in finding the optimal path was when articles in the scientific literature appeared in the late 70s. In [Kerwin, 1978] innovative way of searching for the optimal path of a sailing vessel using sets of nonlinear equations based on the polar diagram (sailboat behavior) and design parameters has been introduced. The big regatta yachts such as in the America's Cup, Cowes Races whether Mug Races determine sailing route optimization science development. Many scientists based on design data and polar diagram (sailboat behavior) to create models to assist in decision-making of foreign exchange by presenting various scenarios for decision-making with high a probability of success, as in the article [Philpott et al., 1993; Philpott et al., 2001]. The review of the state of the knowledge on this field can refer only to general notions and solutions of the outline without detailed results of the test and the attempt of the evaluation. This is due to business reasons and competitiveness between sailing vessels crew, which do not provide parameters for yachts. In [Philpot, 2001] authors noted that due to the changes of the weather, two approaches to this problem are considered: deterministic and stochastic. Deterministic algorithms to search the optimal sailing route are developed for the actual and constant during a longer period of time. Otherwise, when the weather is uncertain and inconstant in time, an algorithm is built on the basis of the stochastic approach, taking into account different scenarios of weather.

Interesting achievements relate to behavior problems of unmanned sailing ships. For example, in the article [Langbein et al., 2011], the authors suggest solving the problem of travel planning for an unmanned sailing vessel using algorithms based on graph theory, taking into account the polar diagram (sailboat behavior). If a polar diagram of a specific sailing vessel and parameters of the wind in a specific area are known, it is possible to determine prediction of the speed of the sailing vessel depending on the direction of movement. According to the concept of e-navigation as described in [Weintrit, 2007] to undertake this type of research it is important for the navigator of the sailing vessel, because it leads to optimization of travel depending on the selected criteria and limitations and increases sailing journey safety level. Proposals from scientific literature of an optimal route for a sailing vessel, (mainly from the communities involved in unmanned ships) have some limitations. For example, methods proposed in [Langbein et al., 2011] apply:

- optimize the movement of the ship sailing with only one criterion the time;
- considering only static data (available wind parameters are navigable only before the start);
- discretization takes into account the motion of the ship usually 8 possible directions.

Increased discretization of the navigable area and the inclusion of a greater number of directions of heading [Daniel et al., 2010; Philpott et al., 1995] requires modification of known algorithms and performance of large-scale computations.

The proposed method involves the inclusion of the 32 directions of change in the movement of the ship sailing, as well as providing dynamic weather data in the course of overcoming the route. In addition to routing it uses a multi--criteria approach based on the travel time and the number of maneuvers performed. Voyage time T is the time to cross the distance from start to finish. The number of maneuvers ω is the number of direction changes of shipping made by this facility during the voyage. In order to assess this approach author built a simulator travel SailingAssistance, which allows to check various alternative solutions and gives evaluation of each of them.

ENVIRONMENT NAVIGATION MODEL

For performing the studies, it is necessary to have discretization of the sailing area, as well as a set of routing optimization criteria. Conveniently it is present regular grid of same part of a sea. In practice the grid does not correspond to such shape. However in further consideration we consider only concrete points on such a grid. Therefore are consideration do not depend on the shape of cells and in consequence are valid.

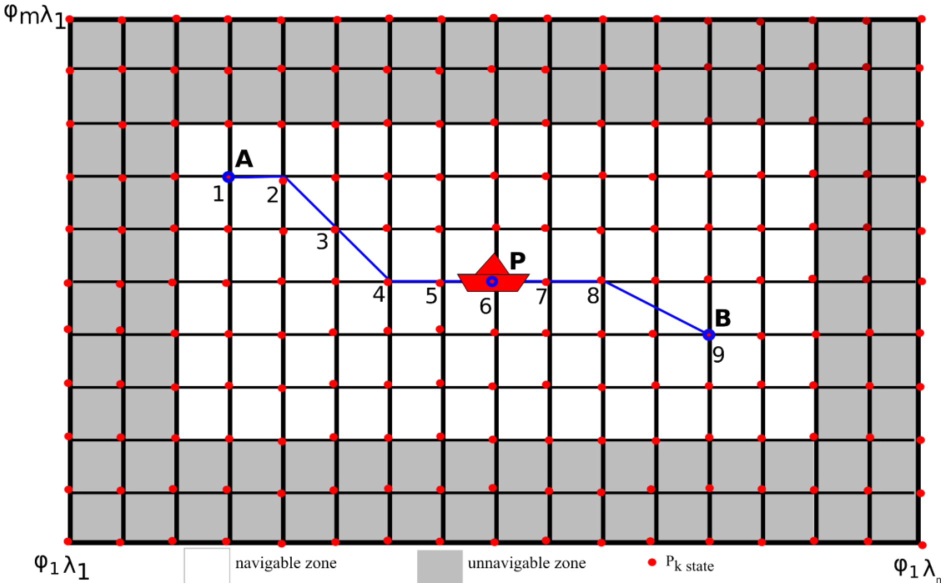

Fig. 1. Environment model for the vessel sailing routing

Figure 1 shows a limited area of sailing in which are the marked navigable zone and unnavigable zone, and applied to the grid, points which describe the state of the area and the vessel navigable route. As follows from the route the vessel made maneuver three times to get from point *A* to point *B* (at 2, 4, 8), where *A* is the starting point, *B* is the final point. Navigation chart of the actual sailing area is replaced by a grid described by a set of points P_k, selected at regular intervals determined by the geographical coordinates. P_k is determined by the latitude φ_i and longitude λ_i, ie.

$$P_k = P_{ij} = P(\varphi_i, \lambda_j),\ \ i = 1, 2, \ldots, m; j = 1, 2, \ldots, n. \qquad (1)$$

The granularity of area of the map is defined by surface area *OB* per one grid defined by a set of points P_k, ie.

$$z = \frac{OB}{m \times n} = \frac{(\varphi_{\max} - \varphi_{\min}) \times (\lambda_{\max} - \lambda_{\min})}{m \times n}, \qquad (2)$$

where:

$\varphi_{\max}$ ($\lambda_{\max}$), $\varphi_{\min}$ ($\lambda_{\min}$) — indicate the minimum and maximum latitude (longitude) in degrees,
$\varphi_{\min} = \min \{\varphi_i, i = 1, 2, ..., m\}$,
$\lambda_{\min} = \min \{\lambda_i, i = 1, 2, ..., n\}$,
$\varphi_{\max} = \max \{\varphi_i, i = 1, 2, ..., m\}$,
$\lambda_{\max} = \max \{ \lambda_i, i = 1, 2, \ldots, n\}$.

If z is smaller than the granularity is larger. Each point P_k in the area sailing is determined by some data describing the state of the analyzed area $S(P_k)$. In general, this state depends on the time and at any time t can be different, so it is denoted by $S(P_k, t)$. In turn, considered navigable object parameters such as the size of the speed or direction of heading, determine the state of the object, which may be different at each point of sailing. We assume that these values define $O(P_k, t)$. So we can define the problem of route selection depending on the adopted granularity of the navigable area, as well as time-dependent sailing values $S(P_k, t)$ and $O(P_k, t)$ $k = 1, 2, \ldots, m \cdot n$, $t \in <0, T>$ where *T* is the time of sailing from the start to the finish.

It can be assumed that the values of $S(P_k, t)$ and $O(P_k, t)$ are known or estimated before shipping. It can also be assumed that they are determined or read by the available services on a regular intervals during the sailing. In both cases, it is important that voyage time T is as small as possible. Another criterion may be the number of maneuver ω (or change direction of heading of the ship), because it affects significantly the time of shipping.

SAILING METHOD

The sailing area discretization could be transformed so it is an undirected graph $G(V, E)$, where V represents a set of points $P_k = P_{ij}$, ie.

$$V = \{P_{i,j};\ i = 1, 2, ..., m, j = 1, 2, \dots, n\}. \tag{3}$$

The set of edges E defines all the possibilities for sailing between points P_k

$$e_x \in E \text{ then } e_x = (P_k, P_l), \tag{4}$$

ex. navigable object can flow directly from P_k point to P_l point. As the defined graph $G(V, E)$ determines all the possible routes from point A to point B.

If states of the navigable area $S = \{S(P_k, t)\}$, and the state of an object $O = \{O(P_k, t)\}$ are considered for all the points P_k then graph $G(V, E)$ is changed to the weighted graph $G(V, E, S, O)$. So $G(V, E, S, O)$ allows one to realize route from A to B for conditions when S and O parameters are known, or constant during a specific time or estimated or predicted up to a time. In both cases, a modified Dijkstra algorithm [Dijkstra, 1959] can be used with adopted criteria for optimization. Another challenge is obtaining specific parameters of the area and the ship. In addition, due to the safety of navigation should be considered the unnavigable zone, which does not have to be as regular as shown in Figure 1.

In the following discussion, we assume that $S(P_k, t)$ is described by weather conditions, so each P_k consist of wind force $W_k(t)$ and true direction $Kw_k(t)$, the formula is below

$$S(P_k, t) = \{W_k(t), Kw_k(t)\}. \tag{5}$$

While the $O(P_k, t)$ is determined by the size of the object velocity $V_k(t)$ and its direction $Kr_k(t)$

$$O(P_k, t) = \{V_k(t), Kr(t)\}. \tag{6}$$

We assume that the object movement can be according with the directions rule as shown in Figure 2. It is said that $wr \in \{8, 16, 32\}$ is a possible direction of sailing. Moreover, due to the strength and direction of the wind, not all the directions of the maneuver will be acceptable for security reasons.

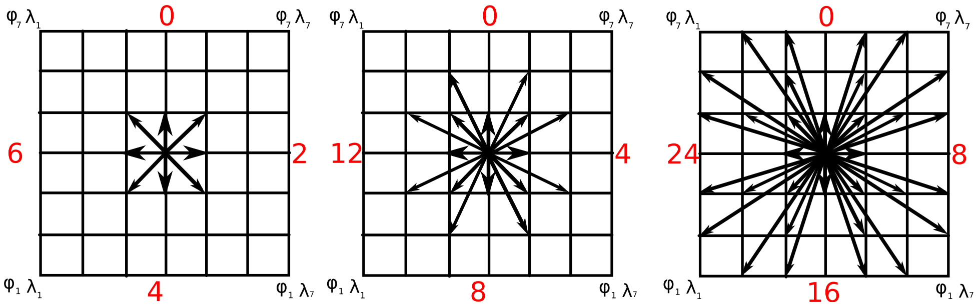

Fig. 2. Possible directions of sailing *wr* ϵ { 8, 16, 32}

Permissible directions maneuver are determined by the polar diagram (sailboat behavior) shown in Figure 3.

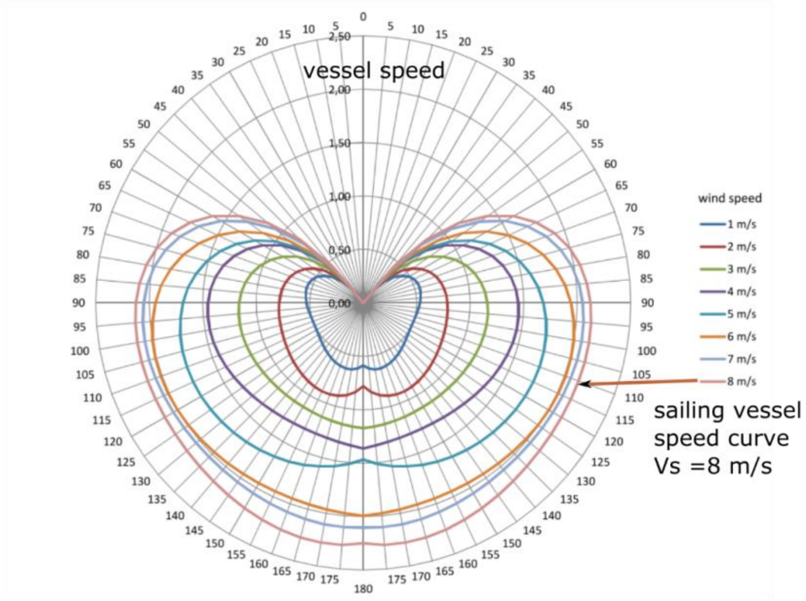

Fig. 3. Polar diagram (sailboat behavior) [Stelzer, 2012]

Polar diagram may be different for each sailing ship and is developed experimentally by its designers or manufacturers. In further discussion we also perform discretization of the polar diagram depending on possible directions of sailing *wr*. For example *wr* = 8 possible directions of ship sailing is *Kr* = [N, NE, E, SE, S, SW, W] and the actual reading speed W_k^r and wind direction K_k^r at the point P_k is discretized according to the formula

$$W_k = \lfloor W_k^r + 0.5 \rfloor; \tag{7}$$

$$Kw_k = \left\lfloor \frac{Kw_k^r}{\frac{360}{wr}} + 0.5 * \left(\frac{360}{wr}\right) \right\rfloor, \tag{8}$$

where

$Kw_k = 0$ — the direction of the wind blowing from the N,
$Kw_k = 1$ — wind from NE etc.

Polar diagram is changed to C_{ab} matrix

$$C = [c_{ab}] \quad a = 0, 1, \ldots, a_{\max} \quad b = 1, 2, \ldots, b_{\max}\,. \tag{9}$$

Indexes of present acceptable traffic directions, $wr \in \{8, 16, 32\}$ (see Fig. 3); b indexes, discrete wind speed, b_{max} is a maximal wind speed contour line in the diagram (Fig. 3), C_{ab} corresponds to the permissible speed value of the sailing vessel (in knots) under the given conditions, it means the selected direction of Kr_k, and the speed W_k and direction Kw_k in point P_k in time *t*. These values can read the polar diagram shown in Figure 3 when setting the shield according to the current wind direction (default setting — the wind blows from the N). That is why, the matrix *C* presented the polar diagram of Figure 3, with 8 directions of the sailing ship possibility adopts the form as shown below:

$$C = \begin{vmatrix} 0 & 0.41 & 0.52 & 0.53 & 0.58 & 0.53 & 0.52 & 0.41 \\ 0 & 0.45 & 0.77 & 0.86 & 0.79 & 0.86 & 0.77 & 0.45 \\ 0 & 0.52 & 1.16 & 1.13 & 1.17 & 1.13 & 1.16 & 0.52 \\ 0 & 0.57 & 1.45 & 1.33 & 1.35 & 1.33 & 1.45 & 0.57 \\ 0 & 0.65 & 1.68 & 1.70 & 1.47 & 1.70 & 1.68 & 0.65 \\ 0 & 0.69 & 1.93 & 1.97 & 2.00 & 1.97 & 1.93 & 0.69 \\ 0 & 0.81 & 2.04 & 2.05 & 2.12 & 0.81 & 2.04 & 0.81 \\ 0 & 0.83 & 2.11 & 2.16 & 2.26 & 0.83 & 2.11 & 0.83 \end{vmatrix}. \tag{10}$$

Assuming that the ship is at the point P_k, it is necessary to determine the next point P_l. To point P_l the ship reaches with the appropriate speed V_k and accepted Kr_k direction. Let $P_k = P_{ij}$ and $P_l = P_{xy}$ then depending on the adopted direction of flow $P_{xy} = P_{i+p,j+q}$, where $p, q \in \{-3, -2, -1, 0, 1, 2, 3\}$ as shown in Figure 4.

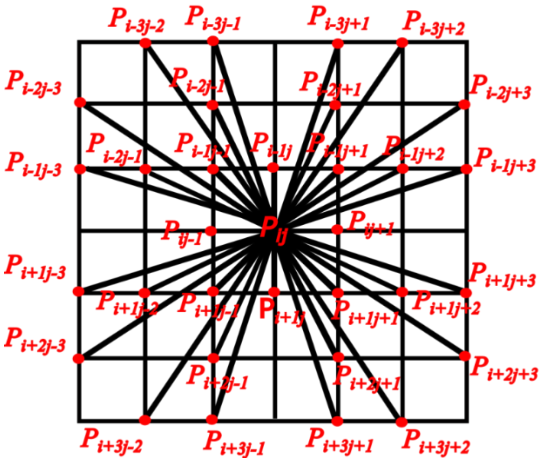

Fig. 4. Possible movement from P_k point to P_l point

For example, the length of the edge from P_{ij} to $P_{i-1j+3+1}$ is direction EbN will be calculated using the formula

$$d_e = \frac{\sqrt[2]{(3\Delta\lambda\ \cos\varphi_2)^2+\Delta\varphi^2)}}{ship\ velocity}. \tag{11}$$

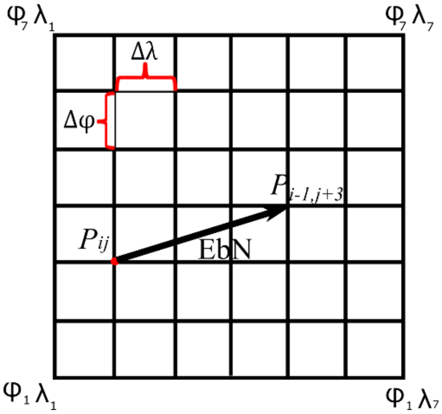

Fig. 5. Edge length from P_{ij} to P_{i-1j+3}

Assuming that the vessel route is held in a discrete area navigation dimension $m \cdot n$, so the complexity of the algorithm is $(wr \cdot m \cdot n)^2$ you will notice that their number increases significantly for smaller granularity (because $m \cdot n$ increases).

ROUTING PROCEDURE

Sailing Assistance procedure has been developed for determining an efficient sailing vessel route, using the following functions and input data:

- A, B — A is start point, B is end route point, indicated by user;
- create (OB, m, n) — function result is $m \cdot n$ grid of OB specific area;
- change (PD) — polar diagram PD transformation to matrix C_{ab} = change (PD);
- area (P_{ij}, t) — read area state OB in P_{ij} point and t time, ie. $S(P_{ij}, t)$ = area(P_{ij}, t) for $i = 1, 2, ..., m$; for $j = 1, 2, ..., n$;
- object (P_{ij}, t) — determine object state in point $P_k = P_{ij}$ and determine next points P_l (see Fig. 4);
- Dijkstra (A, B, wr, t) — implementation of Dijkstra's algorithm for graph $G(V, E, O, S)$ with fixed parameters (a variant of movement wr, and with a predetermined criterion of minimum travel time T); the result is a list of waypoints, where the first point is $P_1 = A$, and the last point to $P_{lst} = B$. Route = $\{P_1, P_2, \ldots, P_{lst}\}$ = Dijkstra (A, B, wr, t);
- Timeto (P_k, P_l) — time from P_k to P_l, (next point in defined route).

```
Procedure SailingAssistance ($OB, m, n, PD, A, B, wr, t$):
        create ($OB, m, n$);
        change ($PD$); % create matrix $C_{ab}$ allows you to calculate time from $P_k$ to $P_l$
        state ($t$)
                for $i = 1:m$
                        for $j = 1:n$
                                $S_{ij} (P_{ij}, t)$ = area ($P_{ij}, t$);
                                $O_{ij} (P_{ij}, t)$ = object ($P_{ij}, t$);
                        end
                end
        end
        $k = 1$; time = $t$;
        route = Dijkstra ($A, B, wr, time$); % create list of points $P_x$ of route, where $x \in \{1, 2, ..., lst\}$, $P_1 = A$, $P_{lst} = B$;
        $k = k+1$;
        while ($P_k+1 \neq B$)
                $t = time$;
```

```
            time = time + Timeto ($P_k$, $P_{k+1}$);
            if (area ($P_{k+1}$, $t$) ≠ area ($P_{k+1}$, $time$)
                  state ($time$); % state $OB$ in time $t = time$
               route = Dijkstry ($P_{k+1}$, $B$, $wr$, $time$); % create list of points $P'_x$
               of route, % where $x \in \{1, 2, ..., lst\}$, $P'_1 = A$, $P'_{lst} = B$,
                  $P_k = P'_2$; % get from new list of route second point
            else
                  $k = k+1$; % next point of route list
         end
     end
print route ($A$, $B$);
end
```

TESTING

The SailingAssistance simulator is located on a server sail.niwa.gda.pl. This solution is embedded in the application, which is part of the project CD NIWA [Krawczyk, 2015] in the sub-project Koala [Goluch et al., 2015]. CD NIWA is a platform for the production of various types of applications (parallel, distributed, and mobile). Koala is a developed C ++ library that implements a number of structures, operations and algorithms known from the theory of graphs and networks. CD NIWA provides server space in domain: sail.niwa.gda.pl, Koala library provides implementation of Dijkstra's algorithm for searching the optimal route for ships sailing. In the current version of the software application the user will have a choice of the number of possible directions of sailing *wr*, the selection of the starting point and end in a defined rectangular area (φ_1 = 54.217º N, λ_1 = 18.069º E; φ_n = 54.936º N, λ_m = 19.508º E). In this defined area are collected meteorological data from an external service AccusWebApi.

In the environment described, tests were conducted for the procedure SailingAssistance depending on the granularity of grid and permissible variations choice of sailing directions for fixed points *A* and *B*. Figure 5 shows the selected possible directions of sailing $\in \{8, 16, 32\}$ and grid resolution $z = 73 \cdot 10^{-8}$ and Grid resolution $z = 293 \cdot 10^{-9}$. The best overall time of the trial *T* for data on weather conditions turned out to be a variant of $wr = 32$ and $z = 293 \cdot 10^{-9}$ selected red frame. Selected parameters designated route and time of the procedure SailingAssistance are given in Table 1. For more selected directions *wr*, and greater granularity area shipping time is decreasing, but increasing uptime for procedures of SailingAssistance.

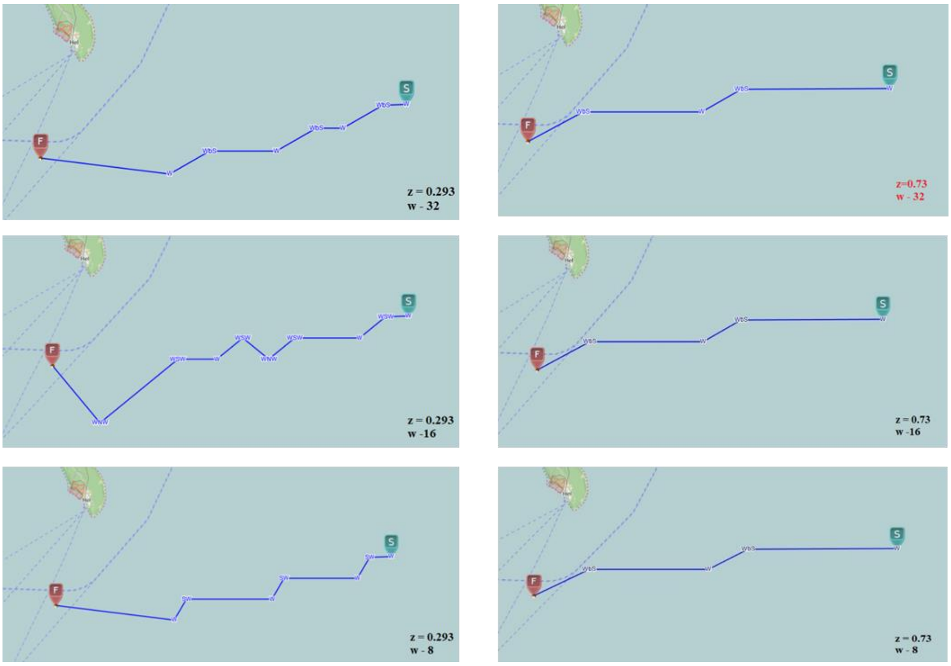

Fig. 6. Display effective methods of seeking suboptimal route: starting from the left in the front column for resolution granulity $z = 73 \cdot 10^{-8}$ variants movement $wr = \{32, 16, 8\}$ in the second column for $z = 293 \cdot 10^{-9}$ variants movement $wr = \{32, 16, 8\}$

Tab. 1. Comparison of options 8, 16, 32 movement from the starting point *A* to end point *B*

$\varphi_A = 54.56°$ N $\lambda_A = 19.23°$ E $\varphi_B = 54.52°$ N $\lambda_B = 18.76°$ E	Numbers of						
	Vertices $\lvert V\rvert = m \cdot n$	edges e_x $\lvert E\rvert$	changes direction ω	points on route P_k	Overall distance D [Nm]	Overall time T [h]	Work procedure time [s]
Grid resolution $z = 73 \cdot 10^{-8}$							
W – 8	595	7514	8	29	15	6.06	0.19
W – 16	1178	21 794	10	18	45	2.16	0.31
W – 32	1335	46 730	8	21	15	1.99	0.57
Grid resolution $z = 293 \cdot 10^{-9}$							
W – 8	1886	30 918	10	55	25	3.82	1.12
W – 16	2653	91 156	3	50	25	1.20	1.45
W – 32	3222	200 236	5	45	25	1.15	1.74

CONCLUSIONS

Realizing the route in the graph $G(V, E, O, S)$ is much more difficult than for a graph $G(V, E)$, as the inclusion of additional data increases the number of possible variants of navigation. Developed SailingAssistance simulator uses the functions described in the procedure SailingAssistance in paragraph 4. These functions can be used in certain sequences of actions depending on the formulated problem.

Extending the route optimization criteria for additional conditions leads to a modification of the procedure for routing and requires increased computing power used. This forces parallelization to perform certain functions, for example: discretization area and determining the matrix C_{ab}, or state areas and objects. In the next study plan will be multi-criteria analysis of selected criteria. In the procedure can be parallel determination of variations on a branch of shipping. However, this requires further study in order to take into account the results of route changes of navigation in real time, and ahead of time sailing. An Increases granularity grid leads to a discrete approximation to the real model, but requires more processing power. In subsequent studies we will provide an evolutionary process of approaching the real model, so that the method would be approaching in a sense to the optimum.

REFERENCES

[1] Daniel K., Nash A., Koenig S., Felner A., Theta*: Any-Angle Path Planning on Grids, ‘Journal of Artificial Intelligence Research’, 2010, Vol. 39, pp. 533–579.

[2] Dijkstra E., A Note on Two Problems in Connexion with Graphs, ‘Numerische Mathematik’, 1959, Vol. 1, No. 1, pp. 269–271.

[3] Goluch T., Ocetkiewicz K., Giaro K., Koala graph theory internet service, ‘TASK Quarterly’, 2015, Vol. 19, No. 4, pp. 455–470.

[4] Kerwin J., A velocity prediction program for ocean racing yachts revised to February 1978, M.I.T. Ocean Eng. Rep., No. 78-11, MIT, Cambridge, MA, 1978.

[5] Krawczyk H., C2 NIWA: The Centre of Competence for Novel Infrastructure of Workable Applications, ‘TASK Quarterly’, 2015, Vol. 19, No. 4, pp. 357–369.

[6] Langbein J., Stelzer R., Fruhwirth T., A Rule-Based Approach to Long-Term Routing for Autonomous Sailboats, Springer, Proceedings of the 4th International Robotic Sailing Conference, Lübeck 2011, pp. 193–204.

[7] Philpott A., Sullivan R., Jackson P., Yacht velocity prediction using mathematical programming, ‘European Journal Operational Research’, 1993, Vol. 67, No. 1, pp. 13–24.

[8] Philpott A., Mason A., Optimising yacht routes under uncertainty, Proc. 15th Chesap. Sail. Yacht Symp., Annapolis, MD, 2001.

[9] Philpott A., Henderson S., Teirney D., A Simulation Model for Predicting Yacht Match Race Outcomes, ‘Operations Research’, 2004, Vol. 52, No. 1, pp. 1–16.

[10] Stelzer R., Autonomous Sailboat Navigation — Novel Algorithms and Experimental Demonstration, PhD Thesis, Centre for Computational Intelligence, De Montfort University, Leicester 2012.

[11] Weintrit A., Wawruch R., Specht C., Gucma L., Pietrzykowski Z., Polish Approach to e-Navigation Concept, ‘International Journal on Marine Navigation and Safety of Sea Transportation’, 2007, Vol. 1, No. 3, pp. 261–269.

Received November 2016
Reviewed June 2017

MARCIN ŻYCZKOWSKI
Gdańsk University of Technology
Narutowicza 11/12 Str., 80-233 Gdańsk, Poland
e-mail: marzyczk@pg.gda.pl

STRESZCZENIE

W artykule zaproponowano suboptymalną metodę określania trasy dla statków żaglowych poruszających się w ograniczonym akwenie morskim. Uwzględniono przy tym dynamikę tego środowiska, między innymi warunki pogodowe oraz charakterystyki prędkościowe statków żaglowych. Jako kryterium optymalizacji przyjęto czas żeglugi T oraz liczbę wykonanych manewrów ω. Zaprojektowano heurystyczny algorytm oraz odpowiednią aplikację wyznaczania trasy dla ustalonych punktów startowych i docelowych. Przeanalizowano zachowanie się algorytmu w zależności od liczby zmian kierunków żeglugi oraz przyjętej ziarnistości opisu akwenu.