Scientific Journals
Maritime University of Szczecin

Zeszyty Naukowe
Akademia Morska w Szczecinie

2013, 36(108) z. 2 pp. 143–147
ISSN 1733-8670

2013, 36(108) z. 2 s. 143–147

The idea of using the A* algorithm for route planning an unmanned vehicle "Edredon"

Krzysztof Naus, Mariusz Wąż

Naval Academy
ul. Śmidowicza 69, 81-103 Gdynia

Key words: A* algorithm, navigation system, route planning, electronic chart, vehicle „Edredon"

Abstract

This article presents the concept of algorithm A* functioning in a navigation system equipped with electronic navigational chart for autonomous planning the shortest and safest route crossing an offshore unmanned vehicle "Edredon". The first part describes the general technical architecture and functionality of the vehicle's navigation system. In the second part shows in detail the modules of the system responsible for the planning of the road and how to implement them in the A* algorithm. The third part describes the proper operation of testing whether the A* algorithm in the navigation system, while the task of planning the route of the autonomous vehicle. Final part is a discussion of the results obtained from tests carried out in order to evaluate the applicability of the A* algorithm for route planning in autonomous navigation systems.

Introduction

Latterly, there arouse a great interest in unmanned autonomous vehicles (UV), whereof the unmanned land vehicles and miniature flying devices are the most popular ones. However, there are also constructed unmanned underwater vehicles (UUV) and unmanned surface vehicles (USV). In many fields the unmanned vehicles slowly supplant those traditional, manned ones, and it take place in both – military and civilian spheres. They are used wherever human – an operator's assistance is unnecessary and taking into consideration that a mission to perform by UV may be danger.

UUV and USV are used increasingly for exploration of sea bottom, also to acquire and to collect geodesic, hydrographic, hydrometeorological and geologic data [1, 2, 3].

At present many European research centres and design offices conduct works on new structures of USV. The examples can be **Basil** (ACSA ALCEN, France), **Rodeur** (Sirenha, France), **Sentry** (Atlas Elektronik, UK), **Inspector MK1** and **MK2** (ECA Robotics, France), **Piraya** and **SAM 3** (Kockums, Sweden), **STIPS** and **Seawiesel** (Veers Elektronik, Germany), **Mariner** (Maritime Robotics, Norway).

Poland also launched out into works on a design of unmanned surface vehicle – USV **Edredon**. It was worked out within the frameworks of a development project "Integrated system of planning perimetric protection and monitoring of sea ports and critical objects based on autonomous unmanned vehicles" by the Naval Academy (AMW), Polish-Japanese Institute of Information Technology and Sprint SA. It is the first such vehicle constructed in Poland.

One of the AMW tasks, realized within the frameworks of the mentioned project, was to design and perform a program module of the navigational system, appropriate for planning an AUV route autonomously.

General characteristics of the navigational system architecture

Any autonomous surface vehicle, manoeuvering on sea water areas and performing a task of designing its route and safe passage thereof, requires information about the natural marine environment and in addition, about the traffic of other vessels. Therefore, the data concerning the vehicle surroundings can be divided to two groups (in this case – due to variation of the information in a time frame):

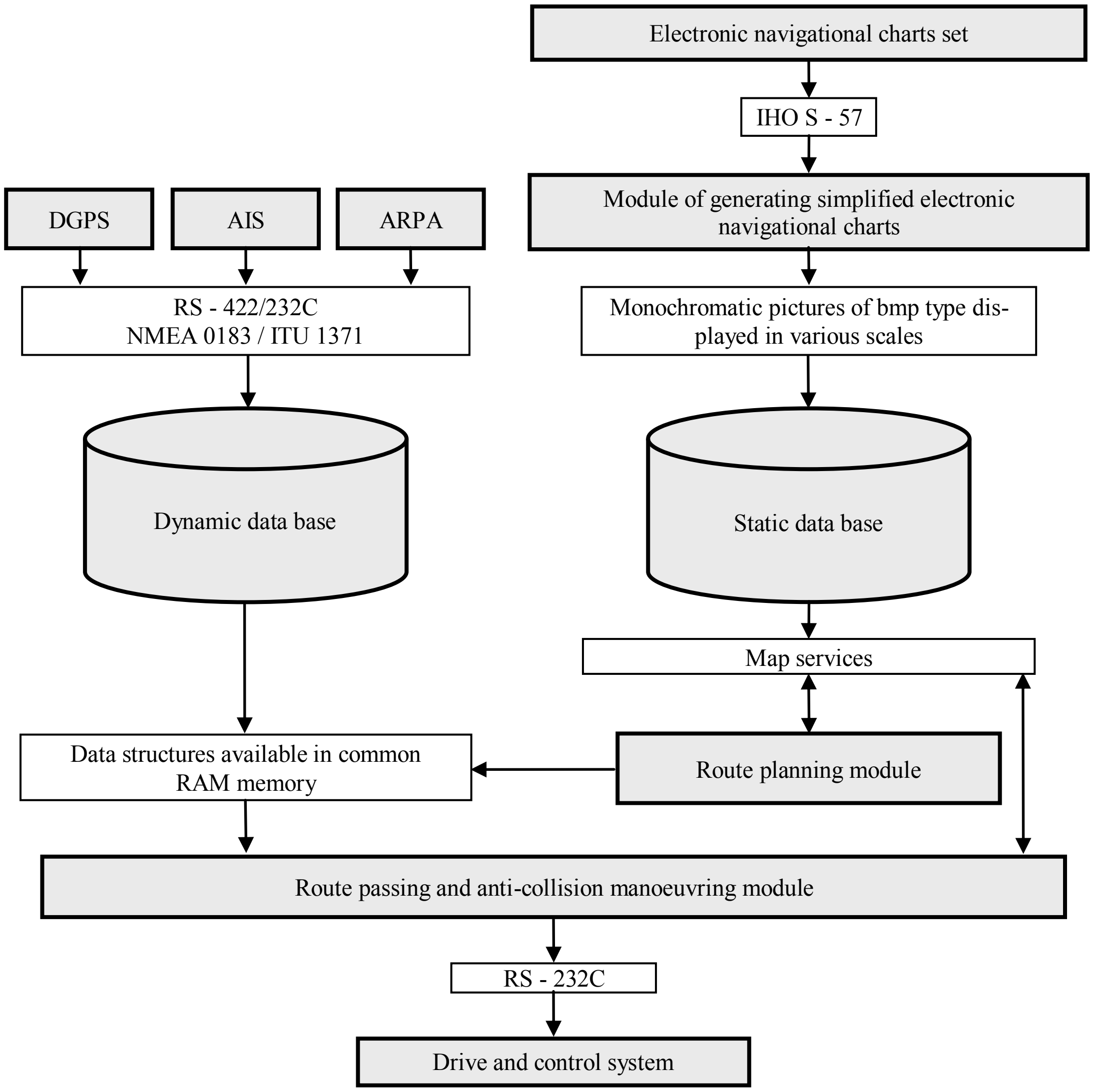

Fig. 1. Architecture of the software part of the USV autonomous navigational system

the static data and dynamic data. The dynamic data group comprises selected data acquired from the navigational tools in the vehicle (as, for example, positions of other ships manoeuvering close by and parameters of their motions), whereas the static data group contains data referring to the natural marine environment included in the electronic navigational chart (such as location of safety contours, setting out a vehicle's safe manoeuvering area).

The appropriately processed data from the both groups can be stored in data bases. Then, through the bases, they can become available in a form of aggregated and filtered out data sets – prepared suitably to the way of their future application. One sort of such data sets is required for working out the route (on a vehicle standby time, for example), the other for performance of anti-collision manoeuvering function within a time of vehicle passage along the designed route.

Having in mind this way defined model of data referring to the surroundings, its functionality set, also the assumed navigational equipment installed in the vehicle, an architecture of the software part of the USV autonomous navigation system can be designed (Fig. 1).

The static data base is supplied with monochromatic 256×256px pictures generated in Mercator's cartographic projection (EPSG:3395 [4]), using specially prepared software application. All pictures collected in the base are arranged by scales to layers, creating a so-called hierarchical pyramid of tile maps. The pyramid base is constructed of tiles of port maps scale, and at the top of tiles it corresponds to the coastal maps scale (Fig. 2).

The map pictures are of the simplified form in respect of:

- format, limited to the tables of pixels of one bite depth of colour;

Fig. 2. Monochromatic map pictures with their names chosen from hierarchical layer of tile map pyramid, conforming to $S = 10$ scale

- content – limited to safe areas (drawn up white) and unsafe (drawn up black) for carrying out vehicles' manoeuvering [5, 6].

Their pictured white and black content are composed of aggregated, suitably selected spatial objects, coded in the electronic navigational charts set (complying with S-57 standard, 3.0 or 3.1 version), covering a region of the intended vehicle activity [4]. The program module, which builds them up, plots at first the objects of an area character, then the linear and point ones; these which do not cover one another, however, together create a so-called "map skin". Such objects include [7, 8]: depth area, deepened area, floating dock, land area, navigational obstruction, wreck, ship permanently docked, pontoon, coastal construction, unmeasured marine area, ice cover area, docking or towing devices, bridge column/cantilever. Secondly, it puts on already plotted "map skin" the objects of point character symbolizing sea marks and dangerous hydrotechnical structures, it means [7, 8]: beacon (cardinal, of isolated danger, lateral, of save water, of special / general purpose), buoy (cardinal, of isolated danger, lateral, of save water, of special / general purpose), day mark, floating light construction, light vessel, land mark.

Route planning module

One of the most essential elements of the designed navigational system was the route planning module. Its operation was connected with setting out the shortest and safe vehicle passage route from the position actually occupied by the vehicle (position coordinates were taken from the dynamic data base, precisely from the common memory RAM) to the position of destination, earlier indicated for possible emergency return (performed in the autonomous mode of operation). The module exercised the assignment basing on the information about the natural marine environment, stored in the static data base. Therefore, at the very beginning, the static data base "demanded" preparing and sending the composed of tiles monochromatic map picture, covering the area between the both points (map scale corresponds to the most accurate source scale ENC). Then, after obtaining the map, it transformed its raster picture into a non-directed (straight) graph, which finally was used directly to set out the route applying algorithm A* [9].

Transformation of the raster map picture to a form of the graph (succeeding simplification of already simplified map) was aimed at optimization of algorithm A* in respect of its functioning speed.

Thus, it was optimization through simplification of so-called search space. The graph representing a search space was built only for safe areas, and its nodes were based on the grid of squares of various mesh dimensions, systematized to so-called quaternary trees. The trees were constructed in effect of recursive division of grid mesh to four trees of smaller size. Dividing was started from a singular mesh of map dimensions and was carried out long enough to get the grid mesh matching the settled precision to a shape of danger areas boundaries (it was assumed that such matching is to correspond to singular pixel plotted on ENC picture of a large scale).

The decision on such form of search space division was an effect of experiences gained in previous research on application of A* in planning the route with a use of ENC. The above research were aimed mainly at verification of potentiality of using the search space, composed of convex polygons formed basing on raw ENC data (spatial objects coded in accordance to S-57 standard). The above study revealed significant problems connected with solving algorithmically a question of selecting the path points inside the polygons, especially those which are laid on a boundary of danger areas, as well as problems joined with frequently obtained, very complicated division of space (composed of a large number of polygons) of a nodes number close to a number of nodes of an ordinary squares grid. Those arguments denied application of the studied search space division methods in the vehicle autonomous navigation system and thereby promoted the quaternary tree division method.

Finding out the passage path by means of algorithm A*

The worked out planning module, at the time of searching the shortest route with algorithm A*, is using the graph $G = (V, E)$ composed of vertexes V and edges E, defined in a form of a quaternary tree. The most important calculations of algorithm A* are carried out applying a function of evaluation (of a cost) of the vertex $f(V_n)$:

$$f(V_n) = g(V_n) + h(V_n, V_{n-1}) \tag{1}$$

where:

$g(V_n)$ – cost of the previous path;

$h(V_n, V_{n-1})$ – the predicted residual cost of the remaining road distance from n vertex to the destination point.

$h(V_n, V_{n-1})$ is a sum of jest Euclidean distance from n vertex to the destination point and the absolute value of a difference of the directional coefficients toward the target point from n vertex and $n - 1$ vertex assumed in the previous step of calculations to the path (difference of the coefficients accepts the least values for the directions neighbouring, reducing to minimum the cost of passing to the vertex):

$$h(n) = \sqrt{(x_n - x_k)^2 + (y_n - y_k)^2} + \left| \frac{y_n - y_k}{x_n - x_k} - \frac{y_{n-1} - y_k}{x_{n-1} - x_k} \right| \tag{2}$$

where:

(x_n, y_n) – stand for coordinates of the calculated n vertex;

(x_{n-1}, y_{n-1}) – are coordinates of the vertex assumed in the last step of calculations;

(x_k, y_k) – stand for coordinates of k vertex which is the point of destination.

Application of the directional coefficient caused that at first the algorithm searched the vertex compatible to the direction toward the destination point and additionally increased the so-called heuristic cost. Then a slight overestimation of $f(V_n)$ vertex costs resulted in a considerable increase of the path finding speed.

This way obtained path (broken line) was then generalized with the modified method, known as the "anchor" and "float" method [10, 11].

Modification of the method consisted in replacement of its main computational parameter, it means the constant permissible perpendicular distance between the points under research ("anchor" and "float") and a distance variable, which guarantees bypassing any dangers to navigation at a safe distance. In effect of this change, the process of generalization ("straightening") of the path was carried out taking into account the set error consisting in the vehicle cross tracking (Cross Track Error – XTE) and the information about the natural environment provided from the map.

In result of generalization there was obtained a sub-set of so-called visibility points ("seeing one another") selected from the set of all points which form the primary path. The changed path, however, set up on this set basis, was of more straight geometrical shape (less number of sections), anyhow, it still passed by all danger areas (Fig. 3).

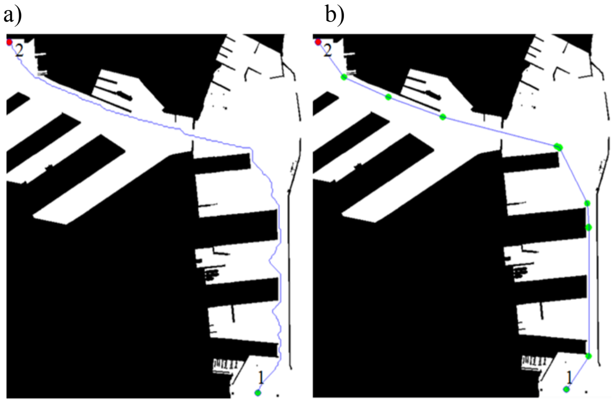

Fig. 3. Route of passage from the point No. 1 to the point No. 2 (places of destination) planned applying algorithm A*: a) before generalization, b) after generalization

Carrying out the generalization operation in this form was of considerable significance for safety in vehicle manoeuvering, in its travel along the planned route. The vehicle had not to change directions as often as before, what is considered to be the most danger element in manoeuvering, especially in bad hydrometeorological conditions and in narrow regions (such situation would happen if the vehicle was travelling along the primary route, usually of very complicated geometrical shape).

Conclusions

The problem of searching the shortest and safest route for USV passing a water area may be defined as one-status. Its solution is a sequence of actions (path in status space) leading from an initial status to the final one, where the main elements of initial and final state vector are position coordinates corresponding thereto, whereas the action (operations in the status space) is interpreted as a function of passage (valuated with a cost) to the determined position.

The space of status representing the search problem may be defined as a simplified (abstract) model of navigation environment. In the simplest form they can be locations on monochromatic map picture ENC, constructed of tile maps pyramid. The digitized space of this way described problem can be expressed in a form of non-directed graph, dividing recurrent safe areas, situated on white and black map picture, to so-called quaternary trees. Owing to such manipulations, functioning of algorithm A* used for setting out the shortest and safe route for USV passing (searching in the graph in effect of carrying so-called sequence of actions) is optimized in respect of computational timing complexity and demand for the computer storage resources.

References

1. CACCIA M., BIBULI M., BONO R., BRUZZONE GA., BRUZZONE GI., SPIRANDELLI E.: Unmanned Surface Vehicle for Coastal and Protected Waters Applications: The Charlie Project. Marine Technology Society Journal 41, 2007, 62–71.
2. MAJOHR J., BUCH T., KORTE C.: Navigation and Automatic Control of the Measuring Dolphin (Messin™). Proceedings of 5th IFAC Conference on Manoeuvring and Control of Marine Craft, Aalborg, Denmark, 2000.
3. YAN R.J., PANG S., SUN H.B., PANG Y.J.: Development and Missions of Unmanned Surface Vehicle. Journal of Marine Science and Application 9, 2010, 451–457.
4. http://spatialreference.org/ref/epsg/3395/
5. NAUS K., WĄŻ M.: S-57 standard as a data carrier for a simplified navigational char. Polish Hyperbaric Research Vol. 40, No. 3, 2012, 57–79.
6. NAUS K., WĄŻ M.: A simplified navigational chart pyramid dedicated to an autonomous navigational system. Polish Hyperbaric Research Vol. 40, No. 3, 2012, 139–161.
7. Special Publication No. S-57, Appendix A, Chapter 1 – Object Classes. Published by the International Hydrographic Bureau, Monako 2000.
8. Special Publication No. S-57, Appendix A, Chapter 2 – Attributes. Published by the International Hydrographic Bureau, Monako 2000.
9. HART P., NILSSON N., RAPHAEL B.: A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Sci. Cybernet. 4(2), 1968, 100–107.
10. DOUGLAS D.H., PEUCKER T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer 10, Issue 2, 1973, 112–122.
11. NAUS K., FRANCZAK D.: Generalizacja danych przestrzennych przy wykorzystaniu algorytmu Douglasa–Peuckera. Forum Nawigacyjne, Gdynia 2010, 71–76.