



## Computerized Shortest Path Searching for Vessels

Gene Eu Jan

*Associate Professor, Department of Computer Science, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.,  
b0199@hpwsl.ntou.edu.tw*

Ming-Bo Lin

*Associate Professor, Electronic Engineering Department, National Taiwan Institute of Technology, Taipei, Taiwan,  
R.O.C., mblin@et.ntit.edu.tw*

Yung-Yuan Chen

*Associate Professor, Department of Computer Science, Chung Hua Polytechnic Institute, Hsin-Chu, Taiwan, R.O.C.*

Follow this and additional works at: <https://jmstt.ntou.edu.tw/journal>



Part of the [Electrical and Computer Engineering Commons](#)

### Recommended Citation

Jan, Gene Eu; Lin, Ming-Bo; and Chen, Yung-Yuan (1997) "Computerized Shortest Path Searching for Vessels," *Journal of Marine Science and Technology*. Vol. 5: Iss. 1, Article 11.

DOI: 10.51400/2709-6998.2542

Available at: <https://jmstt.ntou.edu.tw/journal/vol5/iss1/11>

This Research Article is brought to you for free and open access by Journal of Marine Science and Technology. It has been accepted for inclusion in Journal of Marine Science and Technology by an authorized editor of Journal of Marine Science and Technology.

---

## Computerized Shortest Path Searching for Vessels

### Acknowledgements

The authors wish to thank S. T. Chen and L. Y. Pan of Institute of Navigation Technology, National Taiwan Ocean University for providing them with an excellent technical support in modifying and coding this algorithm.

# COMPUTERIZED SHORTEST PATH SEARCHING FOR VESSELS

Gene Eu Jan\*, Ming-Bo Lin\*\* and Yung-Yuan Chen\*\*\*

**Keywords:** Electronic Chart, Geographic Information System, The Shortest Path, The Optimal Path, Raster, Search and Rescue, Vessel Traffic Management System.

## ABSTRACT

The solution of "computerized shortest path searching for vessels" is presented in the paper. In the core of the method, the shortest path algorithm on the raster plane with  $O(N^2)$  time was applied with dedicated database structure and searching rules. Furthermore, we can extend the method to the optimal path searching for Search and Rescue (SAR) and Vessel Traffic Management System (VTMS) by modifying the database and searching rules.

## INTRODUCTION

Computerized navigation is the most interesting issue in the modern maritime fields but what the computers can do for the navigators, and how are still unknown.

The paper is concerned with developing and implementing algorithm for the simplest case of maritime routing, that is, traveling between two points without crossing a land mass. The problem is entirely geometric; effects of current, wind, weather, water density etc. are simply ignored. The problem becomes that of how to determine the shortest rout between two points (assumed to be ports) on the surface of earth, subject to the constraint that the route does not cross any land mass. The location of each port is specified by latitude and longitude;

similarly the coastlines of the world are available as sequences of points of latitude/longitude points digitized at roughly equal intervals (say 1 km or more apart).

In Gould's thesis [1], he pointed out that by combining each land mass (represented by a polygon) with convex hull it is possible to find out the shortest rout between two ports. Recently, Chen applied the "Quadmap" which is a map based on the Quadtree data structure to the same problem in his thesis [2]. However, another method based on the *raster map* and *radiation scheme* is used to solve this problem in the paper. This method can speed up the computation and might be extended for some other maritime technical usage.

The rest of the paper is organized as follows. Section 2 gives a brief overview of the raster data structures and the rasterization of polygons. The fast algorithm for finding the shortest path between the source point and destination point will be described in Sections 3. The paper is concluded in Section 4.

## RASTER DATA STRUCTURES

Two of the simplest data structures in designing a geographic information system (GIS) are vector graphics (line drawings) and raster graphics (continuous-tone images) or cellular organization of spatial data. The raster technique is more attractive than the vector technique for several reasons. First, complete areas can be filled in easily with a raster technique by finding the pixels which are inside the boundry of an area and turn them "on". In vector graphics, shading must be simulated by letting the "pen" cross-hatch the area. Second, the full spectrum of color is easily obtained, while a vector display is limited to the number of pens available for a pen plotter [3].

The 2-D contour map is perhaps the most common example of a geophysical map. In this type of map, we develop a value for the parameter of interest

*Paper Received March, 1997. Revised June 1997. Accepted June, 1997.  
Author for Correspondence: Gene Eu Jan.*

*\*Associate Professor, Department of Computer Science, National Taiwan Ocean University, Keelung, Taiwan, R.O.C. E-mail: b0199@hpws1.ntou.edu.tw*

*\*\*Associate Professor, Electronic Engineering Department, National Taiwan Institute of Technology, Taipei, Taiwan, R.O.C. E-mail: mblin@et.ntit.edu.tw*

*\*\*\*Associate Professor, Department of Computer Science, Chung Hua Polytechnic Institute, Hsin-Chu, Taiwan, R.O.C.*

— elevation in meters above datum, plant biomass in grams per square meter etc— for every cell in a (frequently regular) array over space. For example, elevation in meters above mean sea level has been recorded at locations on a regular grid, that is, each point is simply a pair of coordinates and an associated elevation value. We can extract the contour lines from the original data of a topographic map and the raster array of elevations is derived from these contour lines using procedures described in [4-8].

When a simple rectangular raster structure is applied to the earth's surface, at least two limitations must be considered. First of all, there is a finite limit to our ability to specify location for a raster structure at a particular scale. Any position on the raster is in either one cell or another, nothing in between, because that the line separating adjoining cells is considered to be infinitely narrow. This is the case for any raster data structure, including the non-rectangular forms such as hexagonal and triangular raster structures. Second, all adjacent cells may not be evenly spaced, depending on their relative position with the center cell. For example, the cells above, below, and on the left/right of the center cell are 1 unit of distance from the center cell, while the cells on diagonal are approximately 1.4 (the square root of 2) unit of distance from the center.

For the 4-connected neighborhood, depicted in Fig. 1a, we are interested in the cells above, below, and on the left/right of a cell only when we search through the data, and all cells are equidistant from their neighbors. Thus, neighboring cells share an edge. If the four cells on the diagonal are included, we are working in an 8-connected neighborhood, depicted in Fig. 1b, and now cells are not evenly spaced. In this latter case, the diagonal cells in the neighborhood share only a vertex, while others share an edge. Since all the cells in these two connection styles have neighbors of the same size and shape, we say that we have *spatial neighborhood similarity* [9].

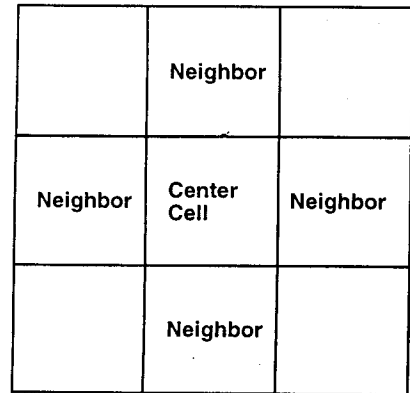
For a given black-and-white geometric map where black area and white area represents the land and the sea, respectively, we first convert it to raster using Pineda's algorithm which can rasterize the polygons (lands on the electronic chart) [10]. An example of illustration of the map conversion is shown in Fig. 2. After the geometric map in Fig. 2a has been rasterized in Fig. 2b, a special rectangular shaded border is attached to converge the distance from the detination cell. The final data stored in computer is depicted in Fig. 2c. We noticed that it should not have any significant distortion between the scanned original geometric map and the final

computerized raster since the size of cells are exactly the same as pixels.

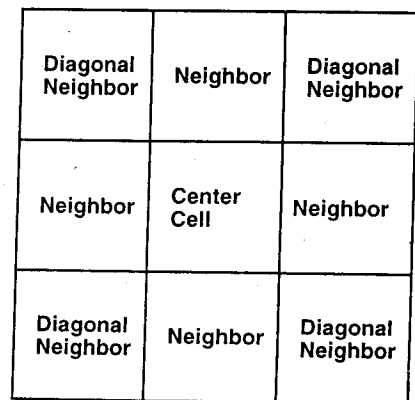
After the geometric map has been rasterized and stored in computer, we can then specify the source and destination through the Window 95's interface randomly, and the algorithm presented in the next section will generate the shortest path between these two ports automatically.

### THE SHORTEST PATH ALGORITHM

We need to predefine two factors for each cell before the description of algorithm. It is assumed that each cell has two parameters  $L$  and  $D$  where  $L$  represents sea ( $L=1$ ) or land ( $L=\infty$ ) and  $D$  is the distance from the detination cell, and its initial value is  $\infty$  which means the distance between this cell and the destination cell is infinite. The main idea behind the shortest path algorithm is to determine the distances between the destination cell and the rest of cells in the raster. The simplest way to



(a)

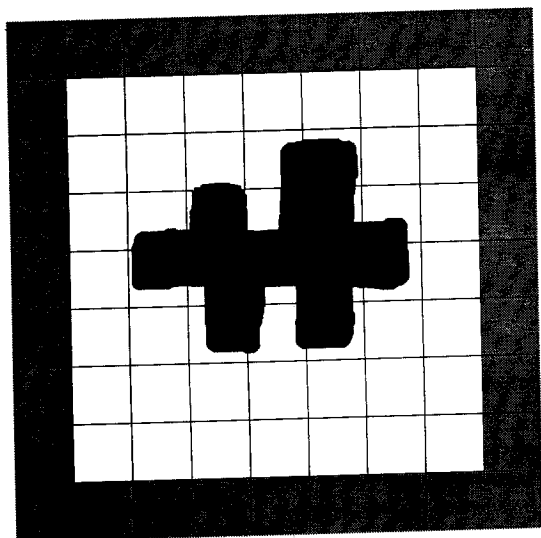


(b)

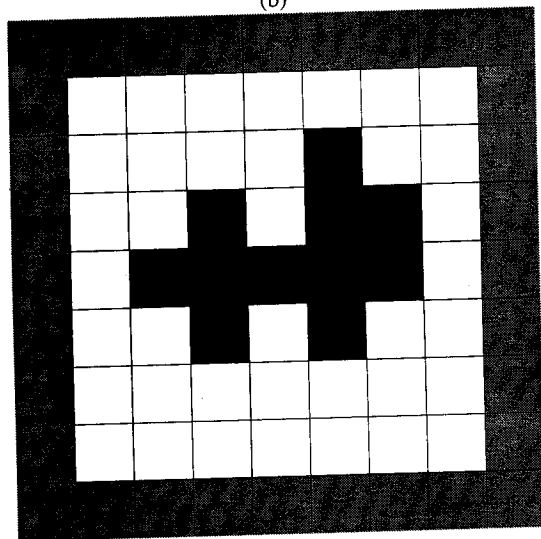
Fig. 1. Spatial neighborhoods. (a) a 4-connected neighborhood and (b) an 8-connected neighborhood.



(a)



(b)



(c)

Fig. 2. Illustration of the map conversion (a) a given black-and-white geometric map (b) rasterization (c) the final computerized raster.

find the distance of each cell is to calculate the distances of 8-connected neighbors of the destination cell first, then spread over the outer neighbors of the original 8-connected neighbors of the destination cell by applying the same scheme radially till all the distances of sea cells are determined. An example of illustration of the distance from the destination cell whose distance is zero is shown in Fig. 3.

There is one more concern that some cells "trapped in" the U-shape land from the aspect of destination cell may not be determined their distances due to the direction of radiation during the first radiation step, therefore, the radiation procedures from the destination cell are repeated until all the distances of sea cells remain unchanged during any two consecutive radiation steps.

We have thus the shortest path algorithm as follows.

**Algorithm: The Shortest Path**

**Step 1:** Initialization and error messages.

**Do**

**Step 1.1** For any cell  $C_{i,j}(L_{i,j}, D_{i,j})$  in a  $n \times n$  raster, the initial value of  $L_{i,j}$  is one if cell  $C_{i,j}$  is in the sea area or infinity if it is in the land area, and  $D_{i,j} = \infty$  for all cells, where  $1 \leq i, j \leq n$ .

**Step 1.2** If the source cell's  $L_{i,j} = \infty$  **then** return the error message "The source you chose is in the land area"

**Step 1.3** Calculate the  $D_{i,j}$  of the destination cell. If  $L_{i,j} = 1$  **then**  $D_{i,j} = 0$  **else** return the error message "The destination you chose is in the land area".

**Step 2:** Computation of distances between the destination cell and the rest of cells.

**Do**

2.8	2.4	2	2.4	2.8
2.4	1.4	1	1.4	2.4
2	1	0	1	2
2.4	1.4	1	1.4	2.4
2.8	2.4	2	2.4	2.8

Fig. 3. The distances from the destination cell whose distance is zero.

**Step 2.1** Calculate the distances of the 8-connected neighbors of destination cell.

$D_{i,j\pm 1} = D_{i\pm 1,j} = D_{i,j} + 1$  (the four 4-connected neighbors), and  $D_{i\pm 1,j\pm 1} = D_{i,j} + 1.4$  (that is, the four diagonal neighbors).

**Step 2.2** Extend the scheme described in Step 2.2 to the outer neighbors which surround the destination cell in a rectangular shape till the distances of the outermost neighbors are determined. The distance of a cell can be determined by the minimum of (the smallest distance of the four 4-connected neighbors) + 1 and (the smallest distance of the four diagonal neighbors) + 1.4.

**End**

**Step 3:** Recomputations

If any  $D_{i,j}$  at the  $k$ th iteration does not equal to  $D_{i,j}$  at the  $k+1$ th iteration for  $1 \leq i, j \leq n$ , then repeat **Step 2** until all the  $D_{i,j}$ s remain unchanged during any two consecutive iterations.

**Step 4:** Backtracking

If the source cell's  $D_{i,j} = \infty$  then return the message "There is no path between the source port and the destination port" else backtracking the shortest path from the source cell to the destination cell by selecting one of the 8-connected neighbors of the source cell with the smallest distance and so on so forth.

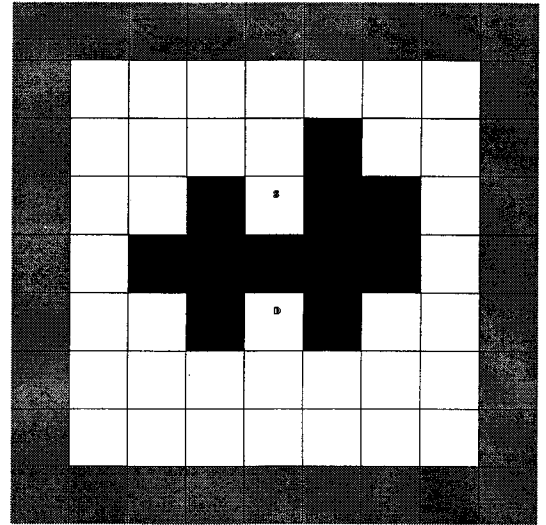
**END {Algorithm}**

An example that illustrates the steps of algorithm is depicted in Fig. 4 and the screen printouts of execution of program based on this algorithm are shown in Fig. 5.

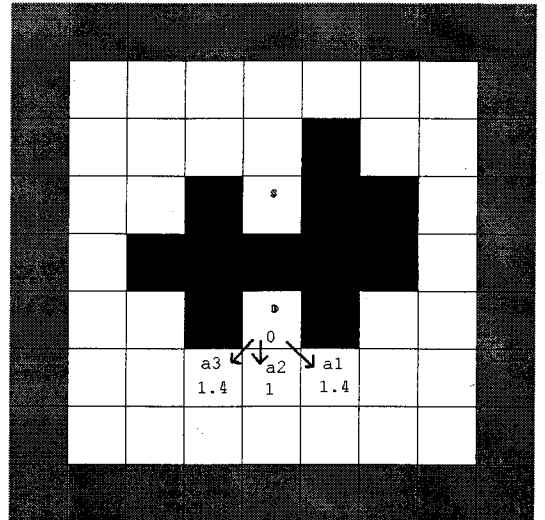
It is obvious that this algorithm has the time complexity of  $O(N^2)$ , where  $N$  is the total number of sea cells, since at most  $8N$  time units are required in step 2 and there are at most  $N$  iterations in step 3.

**CONCLUDING REMARKS**

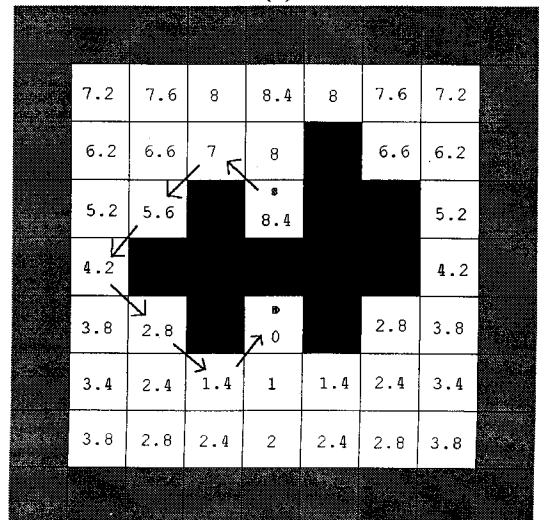
A planar shortest path searching algorithm on raster map based on the radiation scheme is presented in this paper. This algorithm has the time complexity of  $O(N^2)$  which could be reduced to  $O(N)$  furthermore by using another searching rule and extra data structure. As for future work, it can be extended from the 2-dimensional raster arrays (plane) to the 3-dimensional raster arrays (space) but also from the static to dynamic (time vary) situation by simply adding extra spatial and time parameters to the cell, respectively.



(a)



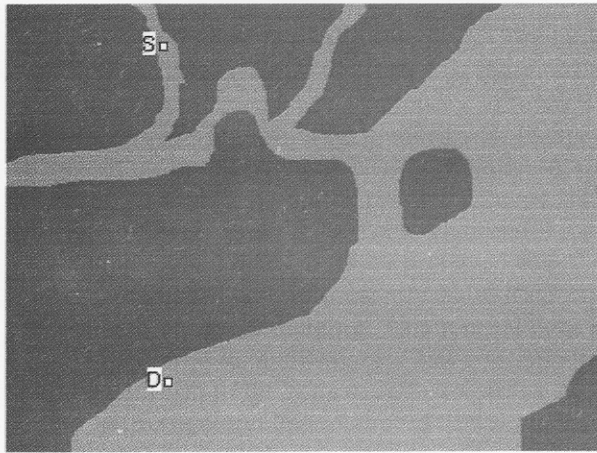
(b)



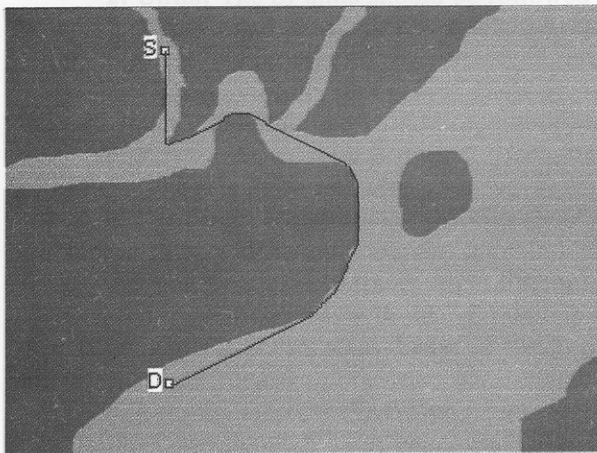
(c)

Fig. 4. An example of illustration of the algorithm (a) steps 2.1 and 2.2 (b) step 3 (c) step 4.

## Notes for Contributors



(a)



(b)

Fig. 5. Illustration of screen printouts (a) before, and (b) after the execution of program.

## ACKNOWLEDGEMENT

The authors wish to thank S. T. Chen and L. Y. Pan of Institute of Navigation Technology, National Taiwan Ocean University for providing them with an excellent technical support in modifying and coding this algorithm.

## REFERENCES

1. Gould, C., "A shipping route finder," *Master Thesis*, Dept. of Computer Science, Univ. of Newcastle, 1992.
2. Chen, S.T., "Optimum Maritime Routing System," *Master Thesis*, Dept. of Computer Science, Univ. of Newcastle, 1994.
3. Jern, M., "The Raster Graphics Approach in Mapping," *Computers and Graphics*, Vol. 9, No. 4, pp. 373-381 (1985).
4. Goodchild, M.F., "Fractals and the Accuracy of Geographic Measures," *Mathematical Geology*, Vol. 12,

No. 2, pp. 85-98 (1980).

5. Mark D.M. and Anderson, P.B., "Scale-Dependent Fractal Dimensions of Topographic Surfaces: An Empirical Investigation, with Applications in Geomorphology and Computer Mapping," *Mathematical Geology*, Vol. 16, No. 7, pp. 671-683 (1984).
6. Nagy, G. and Wagle, S., "Geographic Data Processing," *Computing Surveys*, Vol. 11, No. 2, pp. 139-181 (1979).
7. Burgess, T.M. and Webster, R., "Optimal Interpolation and Isarithmic Mapping of Soil Properties: I. The Semi-Variogram and Punctual Kriging," *Journal of Soil Science*, Vol. 31, pp. 315-331 (1980).
8. Webster R. and Burgess, T.M., "Optimal Interpolation and Isarithmic Mapping of Soil Properties: III. Changing Drift and Universal Kriging," *Journal of Soil Science*, Vol. 31, pp. 505-524 (1980).
9. Tobler, W.R., "Cellular Geography," in Gale, S. and Olsson, G., eds. *Philosophy in Geography*, Dordrecht, Holland: D. Riedel Publishing Company, pp. 379-386 (1979).
10. Pineda, J., "A Parallel Algorithm for Polygon Rasterization," *Computer Graphics*, Vol. 22, No. 4, pp. 17-20 (1988).

## 船舶最短航路搜尋之電腦化

詹景裕

國立臺灣海洋大學資訊科學系

林銘波

國立臺灣工技院電子工程系

陳永源

中華工學院資訊工程系

摘要

本論文是以一適用於網格平面之最短路徑搜尋演算法為核心，其時間複雜度為 $O(N^2)$ ，如果以適當之資料結構及搜尋法則，可以提出一套解決船舶最佳航路搜尋的方案。進而根據此最佳路徑搜尋演算法為基礎擴展資料庫及搜尋法則，可使之適用於搜索與救難(SAR)及海上交通管理系統(VTMS)之最佳路徑自動搜尋。

對於其他各種不同形式的交通工具，只要對演算法內的第三度空間與時間參數稍加修改即可適用，可論文將可提出一套通用的演算法來解決從地表到太空、從靜態到動態的航路選擇問題。