NaSTech

JOURNAL OF NATURAL SCIENCES AND TECHNOLOGIES
2024, 3 (1), pp. 282-289, DOI: 10.5281/zenodo.12731955

CONSTRUCTING AN OPTIMAL ROUTE FOR A MOBILE ROBOT USING A WAVE ALGORITHM

Igor Nevliudov[1] Svitlana Maksymova[1] Vladyslav Yevsieiev[1] Nur Uluhan[2]

[1]Kharkiv National University of Radio Electronics, Ukraine
[2]Eskisehir Technical University, Türkiye

ABSTRACT

The article proposes the construction of an optimal route for a mobile robot using a wave algorithm. A mathematical description of the use of the wave algorithm is presented. The development of a program in Python is described. The authors describe the developed prototype of a mobile robot. Experiments have been carried out. The analyzed results show a fairly good speed of completion of the developed route.

Keywords: Mobile Robot, Route Construction, Wave Algorithm, Python, Manufacturing Innovation, Industrial Innovation.

1. INTRODUCTION

In today's world, robotics plays an important role in solving complex problems, including research and restoration of damaged buildings. One of the key challenges in such research is planning the optimal route for mobile robots, which can be challenging due to the complexity of the area and limited robot resources. To achieve this goal, different path planning algorithms are often used [1-8], among which one of the most effective is the wave algorithm. Particular attention should be paid to the fact that small-sized robots that are used to examine damage to reinforced concrete buildings require accurate and optimal route planning through complex and unpredictable territory [9]. Using the wave algorithm allows you to build an optimal route that will allow the robot to effectively explore and restore damaged objects. The wave algorithm is a popular method for finding the shortest path in a graph; it is based on the idea of "waves" that propagate from the starting point to the ending point [10-]. This method takes into account the complexity of the terrain and the robot's limitations, such as limited speed or movement restrictions. The implementation of the wave algorithm based on the Python programming language allows for fast and efficient implementation for mobile robots. Thus, implementing the search for the optimal route for a mobile robot based on the wave algorithm in Python is a relevant and important task in robotics, especially in the context of exploring and restoring damaged buildings. This article will discuss the process of implementing the wave algorithm for this purpose and its effectiveness in real research conditions.

2. OPTIMAL ROUTE CONSTRUCTION BASED ON WAVE ALGORITHM MATHEMATICAL REPRESENTATION AND SOFTWARE IMPLEMENTATION

Wavefront algorithm is a pathfinding method that is used to determine the optimal path from a starting point to a target point on a map with obstacles. The algorithm is based on the propagation of a wave from a target point to a starting point, where each cell on the map has a value corresponding to the distance from this cell to the target point.

Let us represent a mathematical description of the wave algorithm principle (from left to right). Let S denote the starting point and G denote the end point. The terrain map is presented in the form of a grid M of size $N \times M$, where each cell M_{ij} can be either free or an obstacle. The starting point S is initialized with distance 0, and all other cells with distance ∞. Wave propagation, starting from the initial point S, occurs over neighboring cells. Each cell M_{ij} that is adjacent to cells that have the current value k receives the value $k+1$.

$$M_{ij} = f((M_{ij}),k) \tag{1}$$

where: M_{ij} - matrix M element in row i and column j

k - integer

It is worth noting that the propagation process continues until the wave reaches its end point G. At each step of the algorithm, each cell M_{ij} is checked for the possibility of wave propagation. If a cell is empty and its neighbours have a higher value, then it is assigned a value one greater than the maximum value among its neighbours.

$$M_{ij} = \begin{cases} \max(\text{neighbours}) + 1, \text{if } M_{ij} = 0 \\ M_{ij} \end{cases} \tag{2}$$

After the wave has completed propagation, each cell M_{ij} contains the distance to the nearest starting point. To build the optimal path, the cell with the smallest value is selected among the cell's G neighbours. The process is repeated until the starting point S is reached, the optimal path being a sequence of cells starting at S and ending at G .

To check the correctness of the reasoning, we will develop a program in Python in the development environment PyCharm 2022.2.3 (Professional Edition). Let us give an example of software implementation of the above described mathematical expressions.

```
# Function for visualizing a terrain map with obstacles and wave algorithm calculation results
def visualize_map(grid, distances, start, goal, path):
    plt.imshow(grid, cmap='binary', origin='lower')  # Фон белый, препятствия черные
    plt.imshow(distances, cmap='gray', origin='lower', alpha=0.5, vmin=0, vmax=np.max(distances))  # Wave algorithm results
    plt.colorbar(label='Distance')  # Distance scale
```

This function is used to visualize a map of the area with obstacles and the results of the wave algorithm calculation. Here's what each line in this function does:

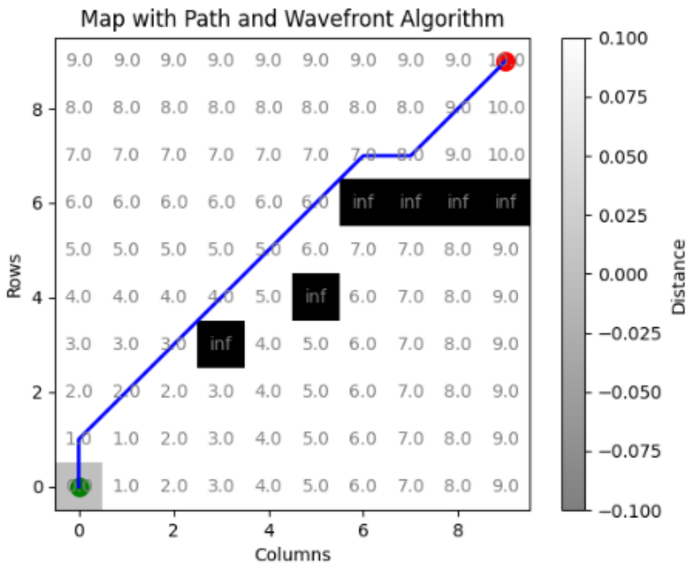
plt.imshow(grid, cmap='binary', origin='lower'): This line displays a map of the terrain with obstacles. grid is an array representing a map of the area, where a value of 0 indicates an empty cell and a value of 1 indicates an obstacle. The cmap='binary' parameter specifies the color scheme to display: white for free cells and black for obstacles. The origin='lower' parameter sets the origin to the lower left corner.

plt.imshow(distances, cmap='gray', origin='lower', alpha=0.5, vmin=0, vmax=np.max(distances)): This line displays the results of the wave algorithm calculation. distances is an array containing the distance values from the starting point to each cell on the terrain map. The cmap='gray' parameter specifies the color scheme to display, from black (minimum distance) to white (maximum distance). The alpha=0.5 parameter sets the transparency of the image to make the background easier to see. The parameters vmin=0 and vmax=np.max(distances) set the minimum and maximum values for the color map.

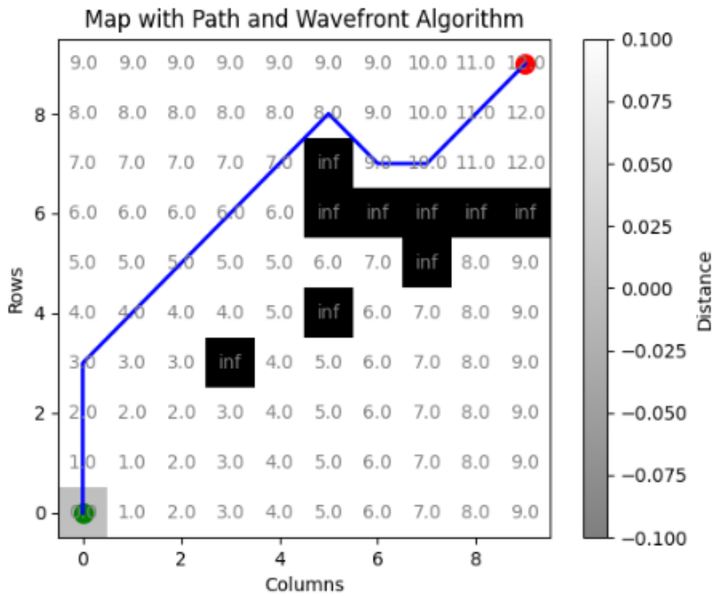
plt.colorbar(label='Distance'): This line adds a color bar labeled 'Distance' to the plot so that the user can understand the correspondence between the colors on the map and the distance values.

This function is an important part of visualizing the results of the algorithm and helps to understand how the distances from the starting point to other points on the terrain map are distributed.

The following hardware was used for research: CPU Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz, RAM 16 Gb, GPU NVideo GeForce GTX 1660Ti (Ram 8Gb), Web-camera HD WebCam, OS Windows 10 Pro (Version 22H2). The program for implementing the method of finding the optimal path using the wave algorithm was developed in the PyCharm 2022.2.3 (Professional Edition) environment in Python. The results of the program are presented in Figure 1.



aa)



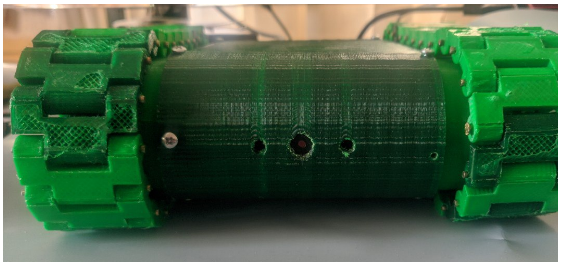
b)

Figure 1 - Results of the program for constructing an optimal route using the wave algorithm

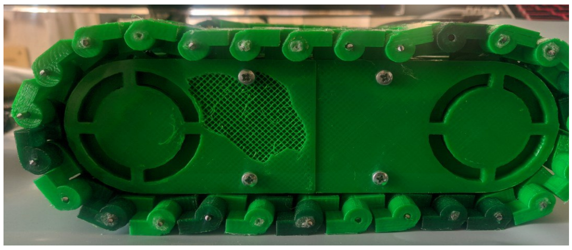
However, the wave algorithm also has its limitations. It assumes that the environment is static and does not take into account dynamic changes, such as the appearance of new obstacles or changing traffic conditions. This may result in suboptimal or unsafe routes if the environment changes. To successfully use the wave algorithm in movement from left to right, it is recommended to regularly update the obstacle map and recalculate the route when changes in the environment. It is also important to consider the speed and capabilities of the robot when choosing the optimal path.

3. EXPERIMENTAL RESEARCH AND ANALYSIS OF THE RESULTS OBTAINED

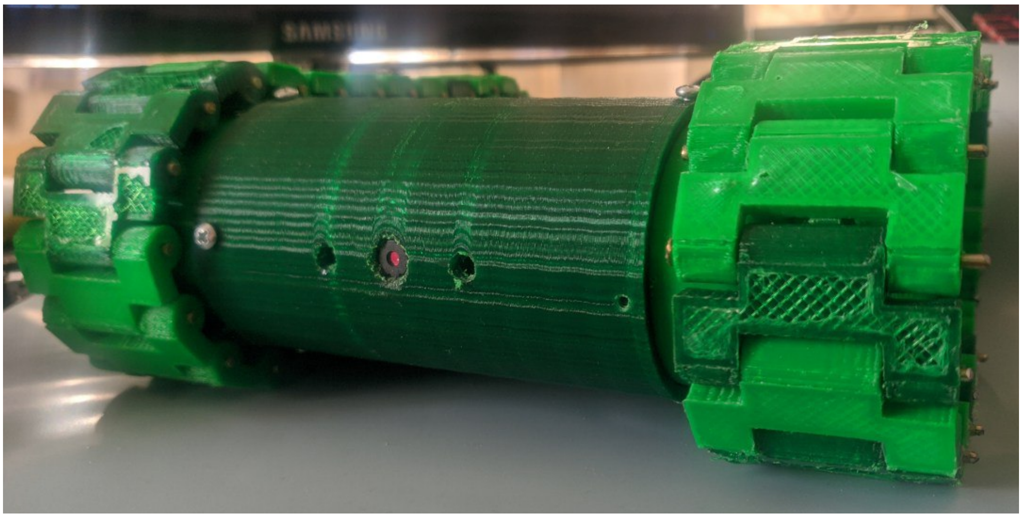
To carry out the experiment of constructing an optimal route based on the wave algorithm, a small-sized tracked mobile robot developed at KNURE at the CITAR department will be used, the general view of which is presented in Figure 2.



a)



b)



c)

a) front view; b) side view; c) isometry view

Figure 2 – General view of the model of a small-scale mobile robot for research.

The technical characteristics of the developed prototype of a small-sized mobile robot (Fig. 2) are presented in Table 1.

Table 1 - Technical characteristics of the developed prototype of a small-sized mobile robot

Characteristic	Value
Microcontroller module	ESP32-Cam
Motor control module	L298N
Power	4 batteries 18650
Number and type of engines	4 DC motors 4-6V
Control method	Wi-Fi
Control interface	"Thin client" on the base ESP32-Cam
Dimensions	Width: 19 cm, Length: 19 cm, Height: 7.5 cm

To conduct the experiment, a test zone was developed in accordance with the map of the area (Fig. 1) and the construction, time, speed and accuracy of the constructed route based on the developed program and in real conditions were checked. An example of the experiment is shown in Figure 3.

JOURNAL OF NATURAL SCIENCES AND TECHNOLOGIES
2024, 3 (1), pp. 282-289, DOI: 10.5281/zenodo.12731955

a) b)

a) passage of the section with coordinates (0,3)
b) passage of the section with coordinates (2,5)
Figure 3 – Test zone for conducting experiments

The results of the experiment with readings of the speed of the mobile robot when crossing the test zone are presented in Figure 4.

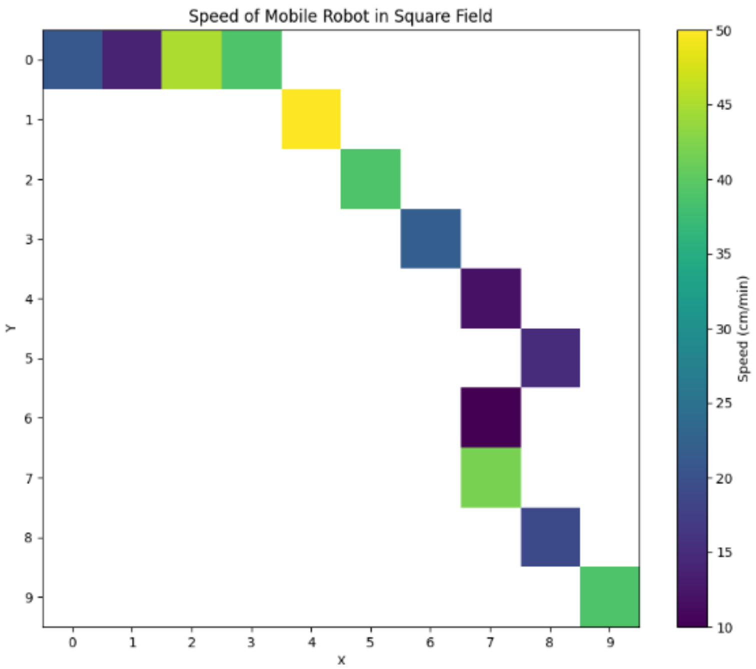


Figure 4 – Graph of the speed of the mobile robot when overcoming the test zone

The results of the experiment with readings of the time (s) of the mobile robot at the far end of each section of the test zone are presented in Figure 5.

JOURNAL OF NATURAL SCIENCES AND TECHNOLOGIES
2024, 3 (1), pp. 282-289, DOI: 10.5281/zenodo.12731955

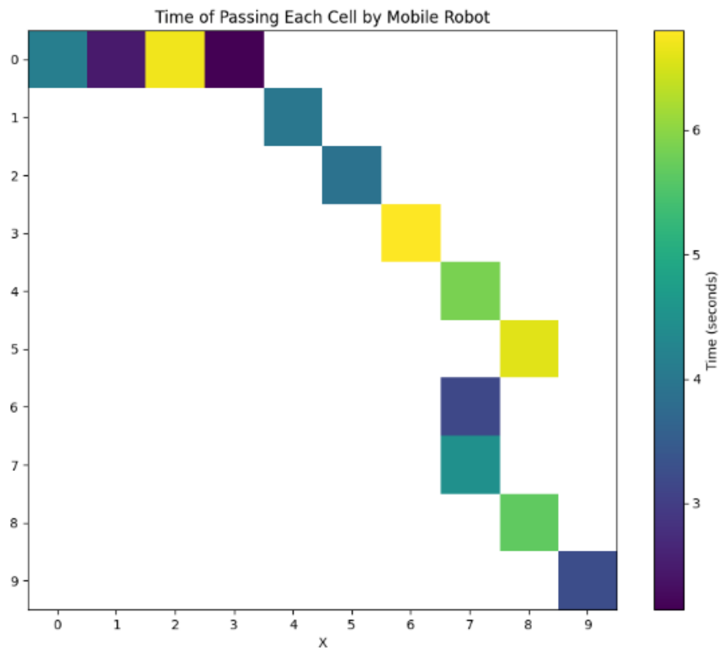


Figure 4 – Graph showing the time taken by the mobile robot to overcome the test zone

Analyzing the graphs of the time and speed of passage of each cell of the test zone, it is possible to draw some conclusions about the movement of the mobile robot:

- the travel time graph shows that for the given coordinates (0,0), (0,1), (0,2), (0,3), (1,4), (2,5), (3,6), (4,7), (5,8), (6,7), (7,7), (8,8), (9,9) the passage time of each cell ranges from 2 to 7 seconds. This may indicate a different degree of complexity of the passage of each of these cells or a different type of surface.
- the movement speed graph also indicates a large spread of values in the given coordinates, which can reflect different conditions for the movement of the robot in each of the cells. For example, the speed may depend on the condition of the surface (even or uneven), the presence of obstacles or other factors.

The graphs show that the time and speed of passage of cells do not always correlate with each other. This may be due to the fact that the speed of the robot is not the only reason for the duration of the passage. Other factors, such as the robot's maneuverability or exposure to external conditions, can also affect the transit time of each cell.

Some cells have the same travel time or speed, which may indicate similar conditions for robot movement in these areas of the test area. Therefore, the analysis of the graphs of the time and speed of the cells of the test area can help to understand the various conditions for the movement of the mobile robot and identify the factors that affect its movement.

4. CONCLUSION

The Wavefront algorithm is an effective method of finding the optimal path for mobile robots on a map with obstacles. It is based on the propagation of a wave from the target point to the starting point, where each cell on the map receives a value corresponding to the distance to the target point. This algorithm allows you to find optimal paths, bypassing obstacles and choosing the smallest distance values.

The mathematical description of the work of the wave algorithm is presented using a matrix and a formula for wave propagation from the starting point. The process of wave propagation continues until the wave reaches the end point. At each step of the algorithm, each cell is checked for the possibility of wave propagation, and if it is empty and has a higher value among its neighbors, then it is assigned a value one unit greater than the maximum value among its neighbors.

To visualize the results of the algorithm, the visualize_map function is used, which displays a map of the terrain with obstacles and the results of the wave algorithm calculation. This allows you to visualize how the distances from the starting point to other points on the map of the area are distributed.

Using hardware with an Intel Core i5 processor and a GeForce GTX 1660Ti video card, a Python program was developed to implement the method of finding the optimal path using the wave algorithm. The results of the program demonstrate the efficiency of the algorithm when searching for optimal paths on a map with obstacles.

REFERENCES

[1] Yevsieiev, V., Maksymova, S., & Abu-J, A. (2024). The Canny Algorıthm Implementatıon for Obtaınıng the Object Contour ın a Mobıle Robot’s Workspace ın Real Tıme. Journal of Universal Science Research, 2(3), pp. 7-19.

[2] Yevsieiev, V., Abu-Jassar, A., & Maksymova, S. (2024). Object Recognition and Tracking Method in the Mobile Robot’s Workspace in Real Time. Technical Science Research In Uzbekistan, 2(2), pp. 115-124.

[3] Yevsieiev, V., Samariddin, S. M., Starodubtsev, N., & Abu-Jassar, A. (2024). Active Contours Method Implementation for Objects Selection in the Mobile Robot’s Workspace. Journal of Universal Science Research, 2(2), pp. 135-145.

[4] Yevsieiev, V., Maksymova, S., & Demska, N. (2024). Using Contouring Algorithms to Select Objects in the Robots’ Workspace. Technical Science Research In Uzbekistan, 2(2), pp. 32-42.

[5] Yevsieiev, V., Abu-Jassar, A., & Maksymova, S. Buıldıng a Traffıc Route Takıng ınto Account Obstacles Based on the A-Star Algorıthm Usıng the Python. Technical Science Research In Uzbekistan, 2(3), pp. 103-112.

[6] Yevsieiev, V., Maksymova, S., & Abu-Jassar, A. Route Plannıng for a Mobıle Robot ın 3D Space Based on an Algorıthm Probabılıstıc Roadmap. Journal of Universal Science Research, 2(4), pp. 22-33.

[7] Maksymova, S., Yevsieiev, V., & Alkhalaileh, A. The Lucas-Kanade Method Implementatıon for Estımatıng the Objects Movement ın the Mobıle Robot’s Workspace. Journal of Universal Science Research, 2(3), pp. 187-197.

[8] Nevliudov, I., Maksymova, S., Klymenko, O., & Bilousov, M. (2023). Development of a Mobile Robot Prototype with an Interactive Control System. Системи управління, навігації та зв’язку. Збірник наукових праць, 3(73), 128-133.

[9] Nevliudov, I., Yevsieiev, V., Maksymova, S., & Chala, O. (2023). A Small-Sized Robot Prototype Development Using 3D Printing. FACULTY OF MECHANICAL ENGINEERING BIALYSTOK UNIVERSITY OF TECHNOLOGY.

[10] Tatino, C., Pappas, N., & Yuan, D. (2020). Multi-robot association-path planning in millimeter-wave industrial scenarios. IEEE Networking Letters, 2(4), pp. 190-194.

[11] Wang, X., Yao, X., & Zhang, L. (2020). Path planning under constraints and path following control of autonomous underwater vehicle with dynamical uncertainties and wave disturbances. Journal of Intelligent & Robotic Systems, 99(3), pp. 891-908.

[12] Xu, Z., Huang, W., & Wang, J. (2022). A wave time-varying neural network for solving the time-varying shortest path problem. Applied Intelligence, 52(7), pp. 8018-8037.

[13] Kim Geok, T., Zar Aung, K., Sandar Aung, M., Thu Soe, M., Abdaziz, A., Pao Liew, C., ... & Yong, W. H. (2020). Review of indoor positioning: Radio wave technology. Applied Sciences, 11(1), p. 279.

[14] Li, Y., Wang, Z., Zhang, W., Lu, Z., Xu, L., & Wang, Z. (2023). Joint correction method for ionospheric phase pollution of high- frequency sky- surface wave radar based on adaptive optimal path. IET Radar, Sonar & Navigation, 17(4), pp. 701-718.

[15] Lyridis, D. V. (2021). An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. Ocean Engineering, 241, 109890.

[16] Matrouk, K., Trabelsi, Y., Gomathy, V., Kumar, U. A., Rathish, C. R., & Parthasarathy, P. (2023). Energy efficient data transmission in intelligent transportation system (ITS): Millimeter (mm wave) based routing algorithm for connected vehicles. Optik, 273, 170374.

[17] Cai, X., Zhang, G., Zhang, C., Fan, W., Li, J., & Pedersen, G. F. (2020). Dynamic channel modeling for indoor millimeter-wave propagation channels based on measurements. IEEE Transactions on Communications, 68(9), pp. 5878-5891.

[18] Zhang, J., Huang, Y., Zhou, Y., & You, X. (2020). Beam alignment and tracking for millimeter wave communications via bandit learning. IEEE Transactions on Communications, 68(9), pp. 5519-5533.