



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 8.423

# Path Finding Visualizer Using Shortest Pathfinding Algorithms

Vijay Kumar Pandey, Aryaman Kumar, Arav Singh, Shubham Kale

Department of Computer Engineering, Dr.D.Y Patil School of Engineering Academy Ambi, Pune, India

**ABSTRACT:** The development of efficient algorithms for path finding has been a significant area of research in computer science. A key aspect of this field is the visualization of the path finding process, which aids in understanding the underlying algorithms and their performance. In this project, we present a path finding visualizer for the shortest path problem, which aims to provide a user-friendly interface for visualizing the search process and determining the most efficient route. The motivation behind this research stems from the need for effective tools that facilitate the comprehension of path finding algorithms. While numerous algorithms exist, such as Dijkstra's algorithm and A\* search algorithm, their inner workings are not always apparent. Visualizing the path finding process enables users to grasp the algorithm's behaviour and make informed decisions about the chosen algorithm and input data.

**KEYWORDS:** Algorithm design and analysis, Graph algorithms, Pathfinding algorithms, Data structures

## I. INTRODUCTION

Pathfinding Visualizer is an interactive tool that allows users to explore and visualize various algorithms used in pathfinding and maze-solving. It provides a hands-on experience to understand the intricate processes involved in finding the shortest path between two points in a complex network. This presentation will introduce you to the key features and benefits of using Pathfinding Visualizer. The project's user interface presents users with a grid-based environment where they can define a start and end point, as well as place obstacles or walls to represent a real-world scenario. Users can then select from a range of pathfinding algorithms, such as Dijkstra's algorithm, A\* algorithm, or others, and witness how these algorithms navigate the grid to find the optimal path. Through dynamic visualization, the "Pathfinding Visualizer" showcases the exploration process, highlighting visited nodes and the final path found. Users can adjust the animation speed, observe the algorithm's efficiency, and analyze the resulting paths to gain insights into the trade-offs between different algorithms.

## II. RELATED WORK

That the tool provides a grid-based interface where users can place walls or obstacles, select a starting and ending point, and choose from a range of pathfinding algorithms such as Dijkstra's algorithm, A\* algorithm, breadth-first search, and depth-first search. As the algorithms run, the tool animates the exploration process, highlighting the visited nodes and the final path found. The "pathfinding visualizer" tool also includes additional features such as speed adjustment, allowing users to control the animation speed, and grid size customization. Users can experiment with different scenarios, observe how the algorithms behave in different grid configurations, and gain a better understanding of their inner workings. The tool is built using HTML, CSS, and JavaScript, making it accessible to users through a web browser. It provides an intuitive and visually appealing interface, allowing users to interact with the grid and algorithms seamlessly. This related work serves as a source of inspiration and a reference for implementing similar functionalities in the "pathfinding visualizer" project. It demonstrates how to design an interactive and educational tool for visualizing pathfinding algorithms, providing a valuable resource for learning and experimentation.

## III. METHODOLOGY

**Analyzing Requirements:** Begin by understanding the requirements and goals of the project. Identify the key features, such as the ability to create obstacles, specify start and end points, and visualize the pathfinding algorithms. Break down the problem into smaller components to gain a clear understanding of the project's scope.

**Designing the Grid:** Develop a grid-based system to represent the problem space. Determine the appropriate data structures and algorithms to efficiently represent and manipulate the grid. Consider factors such as grid size, node representation, and methods for creating obstacles and specifying start and end points.

**Implementing Pathfinding Algorithms:** Choose and implement various pathfinding algorithms, such as Dijkstra's algorithm or A\* algorithm. Understand the algorithms' logic and adapt them to work within the grid system. Pay attention to handling edge cases, such as walls or obstacles, and optimize the algorithms for efficiency.

#### IV. EXPERIMENTAL RESULTS

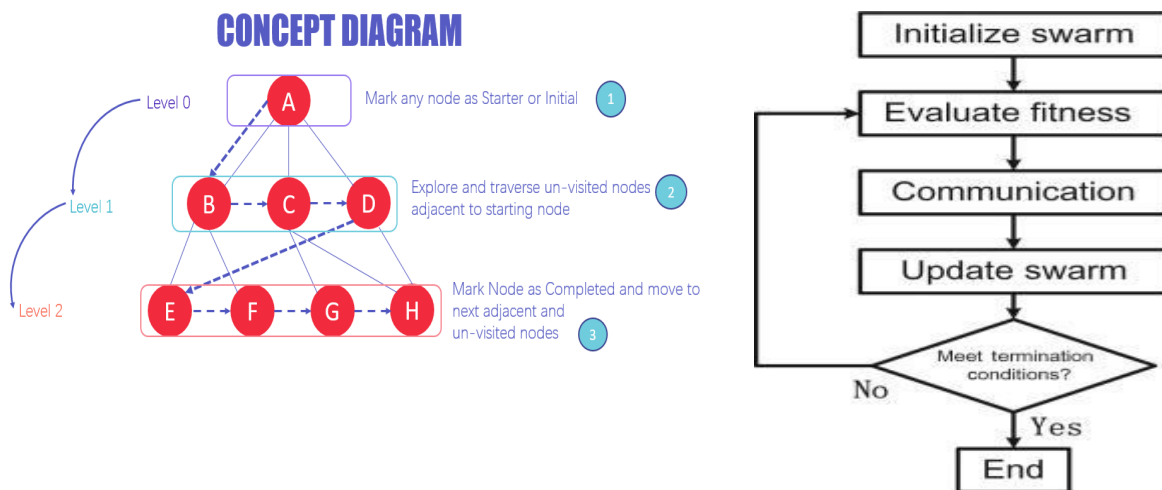
**Impact of Grid Size:** By varying the size of the grid while keeping other factors constant, you can measure the effect of grid size on algorithm performance. It could reveal that larger grids lead to increased runtime, especially for algorithms with higher time complexity.

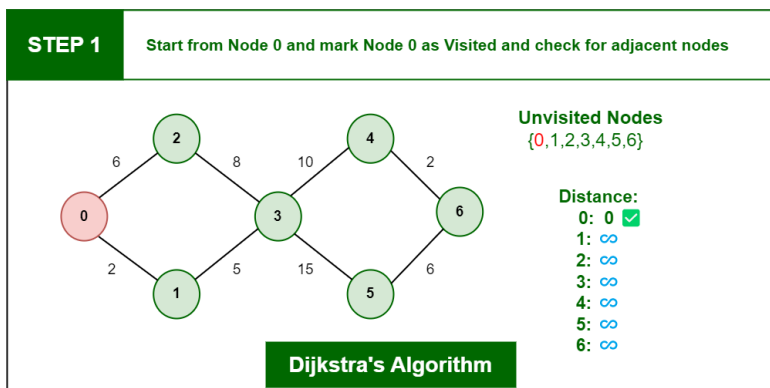
**Obstacle Density and Algorithm Efficiency:** By changing the density of obstacles on the grid, you can evaluate how it affects the efficiency of the pathfinding algorithms. It may indicate that highly dense obstacle configurations result in slower pathfinding due to increased search space.

**Algorithm Sensitivity to Heuristics:** If you have implemented heuristics in certain algorithms like A\*, you can experiment with different heuristic functions and analyze their impact on algorithm performance. This could help identify more effective heuristics for specific scenarios.

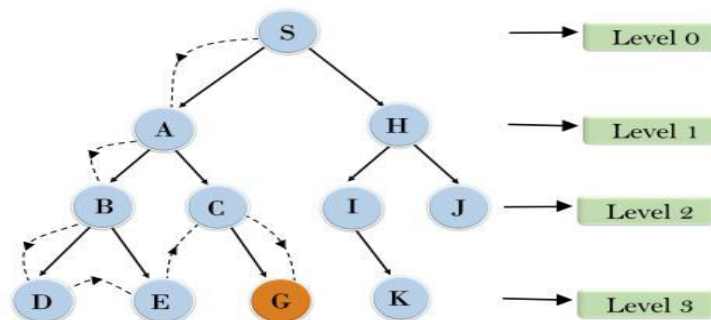
**User Interaction Feedback:** Conducting user surveys or feedback sessions can provide insights into the usability and intuitiveness of the application's interface. Collecting user feedback on the effectiveness of the visualization in understanding pathfinding algorithms can help identify areas for improvement.

These experimental results can provide valuable insights into the performance, efficiency, and user experience aspects of the "Pathfinding Visualizer" project. However, the specific results would depend on the implementation, the chosen algorithms, and the datasets used for testing.





### Depth First Search



### V. CONCLUSION

The Pathfinding Visualizer project provides a useful tool for understanding and visualizing various pathfinding algorithms. By implementing this project, we have successfully created an interactive application that allows users to explore different pathfinding algorithms in a visually appealing manner.

Throughout the development process, we have gained a deeper understanding of key pathfinding algorithms such as Dijkstra's algorithm, A\* algorithm, and others. We have also learned how to represent the problem space using a grid-based system and how to manipulate the data structures necessary for efficient pathfinding.

The project offers a user-friendly interface where users can interactively create obstacles, specify the start and end points, and observe how the algorithms navigate through the grid to find the optimal path. The visualization aspect of the project provides a clear representation of the algorithm's progress, helping users to grasp the underlying concepts more effectively. Additionally, the project can be expanded and customized in various ways. Additional pathfinding algorithms, heuristics, and grid variations can be implemented to further enhance the functionality. Furthermore, the project can serve as a learning resource, allowing students and enthusiasts to experiment with and compare different pathfinding algorithms.

### REFERENCES

1. "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.
2. This book provides a comprehensive introduction to algorithms, including pathfinding algorithms, with detailed explanations and pseudocode.
3. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig.





4. This textbook covers various artificial intelligence topics, including pathfinding algorithms, providing both theoretical background and practical examples.
5. "Pathfinding Algorithms in Grids: A Comparative Analysis" by J. Henkel and T. Filipek.
6. This research paper provides a comparative analysis of different pathfinding algorithms specifically applied to grid-based environments.
7. "A\* Search Algorithm" (Wikipedia article)
8. The Wikipedia article on the A\* search algorithm provides a detailed explanation of the algorithm, its implementation, and its variants.
9. "Dijkstra's Algorithm" (Wikipedia article)
10. *The Wikipedia article on Dijkstra's algorithm offers a comprehensive overview of the algorithm, its applications, and variations.*
11. *Open AI Gym (<https://gym.openai.com/>)*
12. *Open AI Gym is a popular open-source toolkit for developing and comparing reinforcement learning algorithms. It includes various environments that can be used for pathfinding tasks.*



**INNO SPACE**  
SJIF Scientific Journal Impact Factor  
**Impact Factor: 8.423**



**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

**9940 572 462** **6381 907 438** **ijirset@gmail.com**



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details