Advancement of pathfinding algorithms

Marek Vlček

Faculty of Mechanical Engineering, Brno University of Technology
Institute of Automation and Computer Science
Technicka 2896/2, Brno 616 69, Czech Republic
209495@vutbr.cz

Abstract: *Main focus of this paper is explaining beginnings of pathfidning in computer science. How graph theory turned from a curiosity project into it's own field of science and how pathfinding became incorporated into our daily lives.*

Keywords: *pathfinding, algorithms, RRT, Djisktra, searching, graph theory.*

1 Introduction

Pathfinding algorithms are fundamental components in modern fields of computer science, especially robotics and artificial intelligence. They are primally used to find the optimal path from starting point to a destination in a predefined space, although they can be altered to suit any need regarding searching and path planning.

This paper is focused on development of pathfinding algorithms. The first section is focused on early research and history of this scientific field. That includes first official path finding algorithm and it's creator. After that, the second section summarize advancements and different approaches to space searching. There are algorithms divided into groups and their functionality is explained. In the last, third section there are highlighted the best, current algorithms, that are being used or developed.

2 History

2.1 Graph theory

Even though the development of pathfinding algorithms is closely intertwined with the development of computer science in 20th century, its roots started about 200 years before that. Foundational concepts for graph theory which can be considered a main pilar of pathfinding algorithms, were first proposed by Leonhard Euler in year 1736 as part of a problem known as the Seven Bridges of Königsberg.

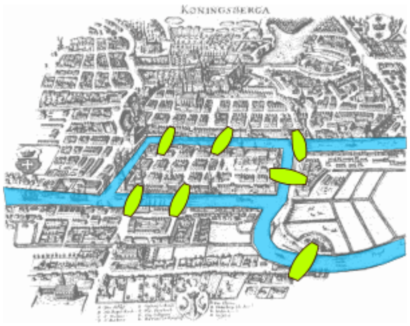

Figure 2.1: Seven Bridges of Königsberg

This mathematical problem delved into a problem of best way to walk through a city of Königsberg which was devided by a river Pregel without crossing all the bridges once. In his work Euler proved that there is mathematicaly no path and created a foundation for the graph theory in the process. After that in 19th century graph theory evolved from simple curiosity into rigid and important mathematical study which gained more popularity in 20th century with rising of the computer science.

2.2 Dijkstra's algorithm

One of the first algorithms to explicitly address the problem of finding the shortest path between two points was Edsger Djikstra's algorithm. It was created in 1959 and from our current understanding simple but in his time revolutionary algorithm is often credited as being the grandfather of the pathfinding algorithms that we use today.

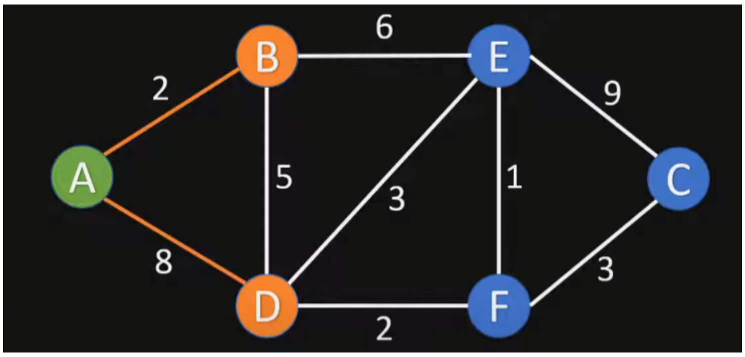

Figure 2.2: Djikstra's algorithm 1

In figure 2.2 you can see a weighted, undirected graph. That means all the nodes have a value that you can interpret as for example a real distance between the points and undirected means that the connections can be use in both ways.

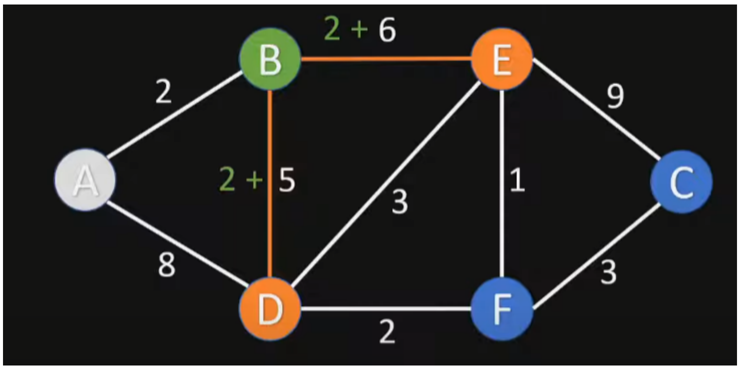

Figure 2.3: Djikstra's algorithm 2

Djikstras algorithm starts in a single point, in our case its point A. In the first iteration the algorithm checks weight of the connection to all the node that are connected to the node A.

That means now we know that shortest way from B to A is 2 and shortest way from D to A is 8. The algorithm now continues with next node that has the lowest value. That means now it checks all the neighbors of the node B. That means node E has value 8 and D has value 7, as you can see in figure 2.3. Because now there are two values for the node D, the algorithm chooses the lower one. Now the algorithm learned that connection A-B-D is actually shorter than connection A-D. This continues until the desired node is reached or all the shortest paths from node A to every other node in the graph was calculated.

3 Development

3.1 Search based algorithms

Into this category we can also count in Djikstra's algorithm. Search based algorithms use a heuristic function that is used to search a discrete, pre-defined space. Another famous search based algorithm is A*. It's an improved version of Djikstra, that is trying to always move in the direction of a goal. Because of this improvement the algorithm is able to find the goal way faster. In figure 4 you can see the difference between their aproach.

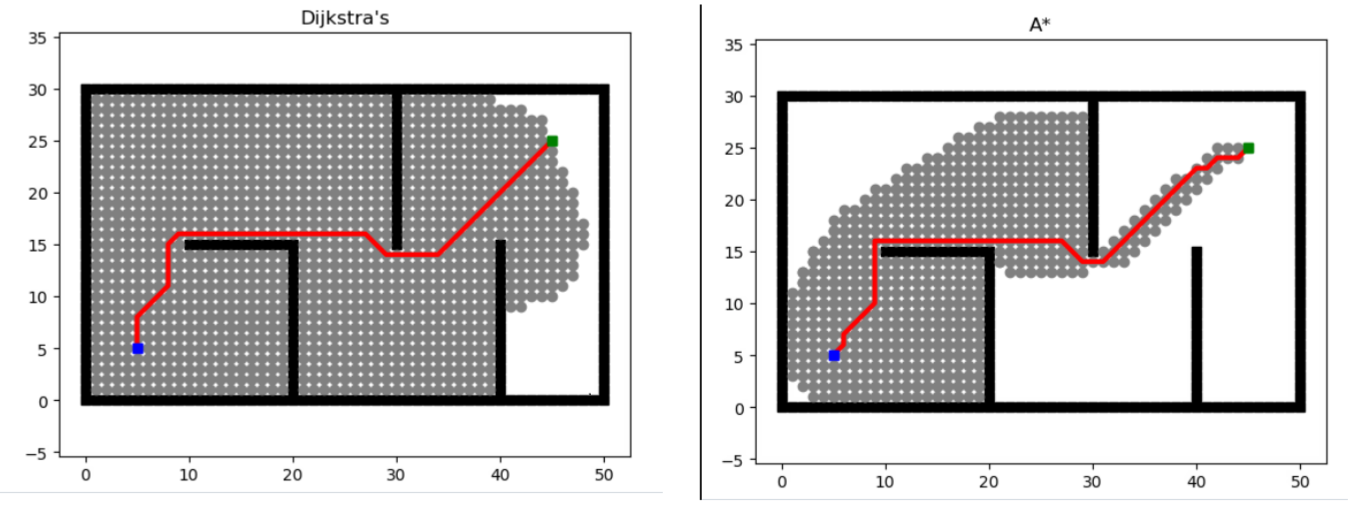

Figure 3.1: Djikstra vs. A-Star

From this point there were a lot of improvements and applications of these algorithms. To shorten the search time there are solutions which use two A* algorithms, one starting from the start node and the other from the goal node. These two paths then connect when they meet. With this trick the search time is cut by half, sometimes even more. The hardest problem for A* to solve was the speed. Even though it creates the most optimal path, checking every node is a time costly task. Solution to this was creation of RRT algorithms in 1990s.

3.2 Sample based algorithms

RRT algorithm, also known as Rapidly Exploring Trees, that was mentioned in a previous subsection is the original sample based algorithm. These algorithms were introduced as a solution to complex problems that came with robotics and other modern technologies. The main issue is that when the robot is controlled, every degree of freedom creates a dimension in the search space. That means robotic arm with six degrees of freedom creates a 6D. For these applications search based algorithms are too slow and most of the time they won't find a path at all. The reason for this is sample based algorithms use random samples to create a map, in case of RRT a tree structure, that is used to search the space. The RRT can be seen in figure 3.2. Sample based algorithms on it's own do not create the most optimal path but can find a usable one in a very short time.

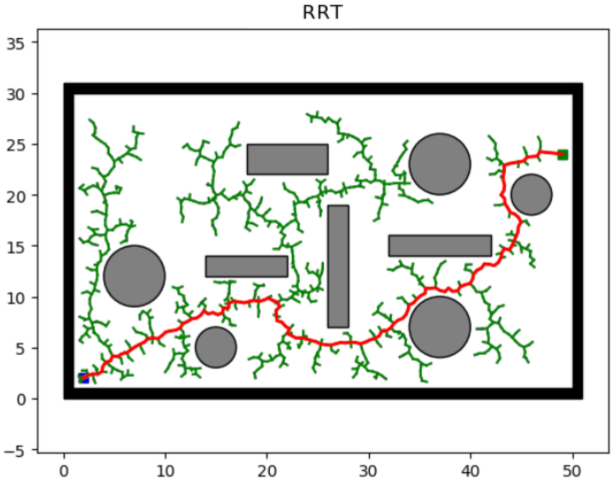

Figure 3.2: RRT

The algorithm creates one random node at the time and checks the space to find it's closest neighbor and connects them. This process is repeated until a tree structure is created that is searching the space until the goal is found. Because of it's random nature RRT can grow to unwanted directions or create paths that are unnecessarily long. The successor of RRT called RRT* is bringing values that represent distances just like in djikstra into the algorithm. Whenever a new iteration of the algorithm happens, the nodes are reconnected based on their values to create a shorter and more straightforward path. Because doing so takes computation time, the re-connection happens only in predetermined radius around the node, which makes it an adjustable parameter. This option let's you trade time for cleaner path. How this algorithm looks in practice can be seen in figure 3.3.

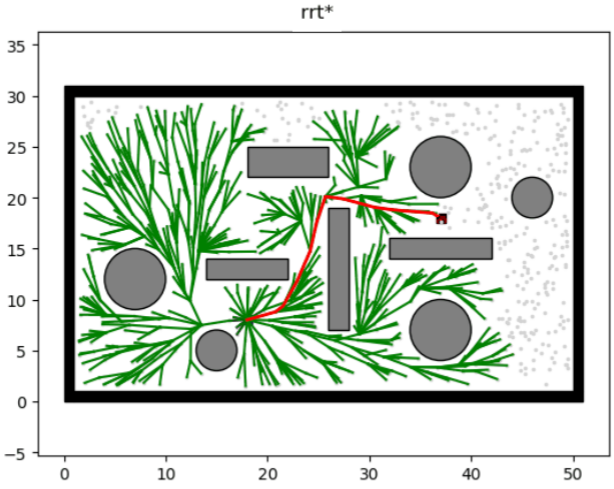

Figure 3.3: RRT Star

4 Current state

In our modern world we use pathfinding algorithms in a lot of advanced technologies. That includes robotic, GPS, selfdriving cars and more. The need for quick response resulted in development of very fast pathfinding algorithms. These algorithms are based on principles already presented in this paper. Combination of search and sample based algorithms, searching from both start and goal and many other upgrades over the original version were used to create new, improved algorithms.

One of them is RRT CONNECT. This is one of the most universal and used based algorithm today. It is a bidirectional algorithm which means it grows two search trees, one from the start and one from the goal. It also uses same system to optimize the path just like RRT*. The most beneficial feature of this algorithm is the way the two trees connect. Every time a new node is created, there is a check if this new node can be connected to other tree. That means if there is clear, collision free path, the trees are connected instantly. Visualization of RRT CONNECT is in the figure 4.1.

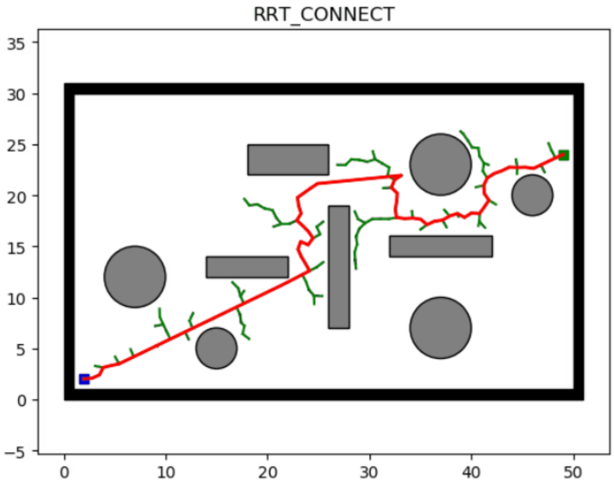

Figure 4.1: RRT CONNECT

5 Conclusion

This paper was focusing on explaining history, early development and current state of pathfinding algorithms. In the first section were explained roots of pathfinding algorithms as a study and then explained function of the first pahtfinding algorithm, Djikstra. The next section focused on differences between search based and sample based algorithms and in the last section there was a quick presentation of the modern pathfinding solution. This part of the paper is greatly shortened and didn't explore nearly enough ways we developed to search a state space using computer technology.

List of sources

[1] ZHOU, HUIMING. PathPlanning. GitHub repository, 2020.
Link: https://github.com/zhm-real/PathPlanning.

[2] NEEDHAM, Mark, HODLER, Amy .*Graph Algorithms*,O'REILLY MEDIA, 2019.

[3] DIESTEL, Reinhard.*Graph Theory*,Graduate Texts in Mathematics, 2017.

[4] LAWANDE, Sharmad. A Systematic Review and Analysis of Intelligence-Based Pathfinding Algorithms in the Field of Video Games, [online]. [cit. 18.4.2024]. Link: https://www.mdpi.com/2076-3417/12/11/5499

[5] SAI VEMPRALA (2023). 2D/3D RRT* algorithm. MATLAB Central File Exchange. Link: https://www.mathworks.com/matlabcentral/fileexchange/60993-2d-3d-rrt-algorithm

[6] FELIXTECHTIPS. (2021). Djikstras Shortest Path Algorithm Explained — WIth Example — Graph Theory [Video]. Youtube. [cit. 18.4.2024]. Link: https://www.youtube.com/watch?v=bZkzH5x0SKU&ab_channel=FelixTechTips

[7] VLČEK, Marek. TRAJECTORY PLANNING OF INDUSTRIAL MANIPULATOR WITH OBSTACLES IN THE WORKSPACE. Brno, 2023.
Link: https://www.vut.cz/studenti/zav-prace/detail/149881.

[8] Dynamic Target Tracking and Obstacle Avoidance using a Drone - Scientific Figure on ResearchGate. [Online]. ResearchGate. [Cit. 24.5.2023]. Dostupné z: https://www.researchgate.net/figure/An-example-of-a-traditional-potential-field-which-can-be-used-for-navigating-toward-a_fig2_294645105

[9] GIBBONS, Alan.*Algorithmics Graph Theory*,King's College London, 1985.

[10] POPOVIĆ, MARIJA. 5-Sampling-based Methods, [online]. [cit. 24.5.2023]. Dostupné z: https://mpopovic.io/uploads/dmar2021/L5-sampling-based-methods.pdf

List of figures

2.1 Seven Bridges of Königsberg 1
2.2 Djikstra’s algorithm 1 2
2.3 Djikstra’s algorithm 2 2
3.1 Djikstra vs. A-Star 3
3.2 RRT 4
3.3 RRT Star 4
4.1 RRT CONNECT 5