Journal of Engineering and Technology

ISSN 2180-3811 eISSN 2289-814X https://jet.utem.edu.my/jet/index

A COMPREHENSIVE REVIEW ON DIFFERENT PATH PLANNING METHODS FOR AUTONOMOUS VEHICLES

A. Vinayak[1,2], M. A. Zakaria*[1,2] and K. Baarath[1,2]

[1] IMAMS Laboratory, Faculty of Manufacturing and Mechatronic Engineering Technology, Universiti Malaysia Pahang, Pekan, Pahang 26600, Malaysia.

[2] Autonomous Vehicle Laboratory, Centre for Automotive Engineering, Universiti Malaysia Pahang, Pekan, Pahang 26600, Malaysia.

corresponding: maizzat@ump.edu.my

Article history:
Received Date: 15 November 2022
Revised Date: 31 March 2023
Accepted Date: 30 April 2023

Keywords: Motion Planning, Autonomous Vehicle, Path Planning

Abstract— Autonomous vehicles are an active field where research is going on to improve the vehicle's capability to travel autonomously from one place to another. Vehicle must progress through different levels of control structure to navigate through different environments. Among those path planning plays a major role in autonomous vehicles navigation as different planning methods need to be used for planning the path at different intersections for the vehicle. However, AVs still face some challenges in urban intersections such as roundabouts, obstacle avoidance, which need to be resolved for completely automated path planning in AVs. So, this paper presents an overview on different path planning methods implemented in autonomous navigation. A description on different path planning methods and implementation of these methods by different authors is presented.

I. Introduction

The intelligent transport system improvement has helped to decrease the rate of accidents caused by human drivers and increase the comfort and safety of the passengers. There has been development in both lateral and longitudinal control of the vehicle for travelling through a road. Longitudinal controls such as adaptive cruise control (ACC), Emergency braking (EB) are used in vehicles. The lateral control helps the vehicle to follow the planned path. The path needs to be planned depending on different factors such as the type of environment the vehicle has to travel, the traffic rules that need to follow and the vehicle steering constraints such that the vehicle is able to steer according to the required path. To create a path according to environment, the vehicle has to go through different control stages that is the acquisition, perception, Decision, Control, Actuation. Each stage has different tasks to be done.

This paper focuses on the decision module, where the path is created for the vehicle using different path planning methods. Depending on the information from the sensors the environment will be detected. There will be different types of road environments with varying rules of traffic. Therefore, to create paths through disparate environments, different path planning methods are used, enabling path generation according to the environment. Thus, this paper presents a comprehensive review on different path planning methods and their application by different authors in different environments.

II. Different Path Planning Methods

Researchers have introduced different techniques to plan paths across various obstacles or environments the vehicle has to travel through. Most path planning methods are extracted from mobile robots developed to adjust to the road conditions.

The path planning techniques are mainly classified into four types. Each of the four types has subdivisions that are improved to increase the efficiency of the previous methods. Depending on the type of environment, different properties of these methods help create a smooth path that can follow the traffic rules associated with the intersection.

III. Different Path Planning Methods

In graph search-based planner, the state space or the environment will be divided into small grids, and each grid will be assigned an occupancy if any object or obstacle will block the vehicle's path. The main goal is to locate a path from the starting position of the vehicle to the destination position while avoiding obstacles. Three different algorithms were developed to find paths based on the grids, which help in finding the path for autonomous vehicles moving from initial to final.

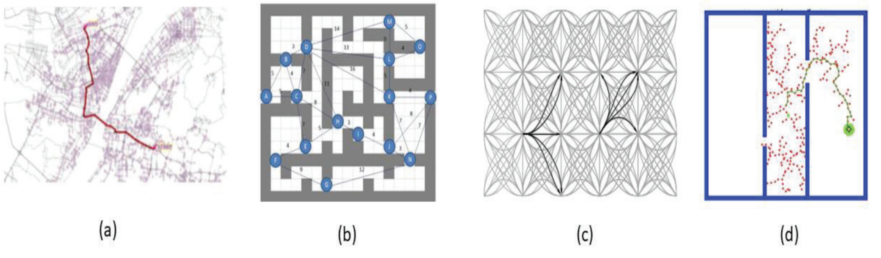

Figure 1: (a) Dijkstra[1] (b) A-star (c) State lattice[2] (d) RRT

A. Dijkstra Algorithm

The Dijkstra algorithm helps find the shortest distance from one point to another by comparing the distance between each node. The environment will be split into nodes and the distance between each node is compared with the neighboring nodes.

This algorithm finds the shortest distance without considering the shape of the path. Figure 1(a) shows Dijkstra based path planning. The path has the shortest distance between the vehicle's initial and final positions. The application of the algorithm can be found in [1–3]. The major advantage of this algorithm is that it creates the shortest path from one point to another using the nodes. It is useful for global planning in a structured and unstructured environment. The disadvantage is that the computation time and cost is high considering all the neighboring nodes for creating the path.

B. A- Star Algorithm

A* algorithm is a graph search-based planner which works similar to the Dijkstra algorithm except it uses a heuristic function while selecting the nodes to create a path. A* tries to look for a better path by using a heuristic function and cost function which gives priority to vertexes that are supposed to be better or closer to the final point. The heuristic function is the distance between the initial vehicle position and the final point. Depending on the distance between the starting node to the current node a cost function is determined which is used for determining the distance between each vertex. The shortest path will be calculated based on these functions to create a path for the vehicle to follow. Figure 1(b) shows A* based path planning.

The major advantage of using A* over the Dijkstra algorithm is that it reduces computation time. There are several applications in mobile robotics which is then further improved to form dynamic A* (D*), Field D* [4], Theta*, Anytime repairing A*(ARA*) [5] and Anytime D*(AD*) [6].

C. State Lattice Algorithm

In order to permit relevant state continuity, deterministically acquire target states, and satisfy the differential restrictions of the vehicle the state lattices build a discrete search space. The method uses a discrete hyper-dimensional grid of states to represent the planning region. The motion planning search is conducted over this grid, referred to as the state lattice. Vehicles can go from one starting state to multiple others due to the algorithm's path search, which is based on local queries from a set of primitives or lattices that contain all feasible features. A cost function determines the optimal route between the precomputed lattices.

Figure 1(c) shows the path planning using state lattice by creating lattices from starting point to goal point. Mooney and Johnson [7] reported a path planning approach in different environments such as straight roads and other intersections. The space was divided into state lattice with occupancy marked. The path is selected based on the cost function implemented. In other report by Kushleyev and Likhachev [8] and Eilers et al. [9] presented path planning using state lattice for lane change and overtaking in different environments and vehicle was able to follow the path without much error.

IV. Sampling - Based Planner

The sampling-based planner approach randomly samples the configuration or state space to look for connectivity inside it. The Corresponding space (c-space) or the bound area is created for selecting points to create path.

A. Rapidly Exploring Random Tree (RRT)

Rapidly exploring random trees belongs to sampling-based path planning applicable to online path planning. By doing a random search through the navigation area, it enables quick planning in semi-structured space. The RRT path planning method will find a path if there is any path that connects the starting point with the goal point. The path is created from the starting point to the goal point by creating branches like a tree. The branches will be extended in the navigational area by randomly selecting nodes at a specific distance. The path made by adding nodes from the starting point to the goal point is depicted in Figure 1(d). Kuwata et al. [10] demonstrated a path planning strategy in static and dynamic environments. The major disadvantage is that the resulting trajectory is not always continuous, so the path may be jerky. The path planning method was for the MIT team at DARPA urban challenge [10] but the path generated was jerky and has discontinuous curvature. The upgraded version of the RRT is the RRT* where the path is created by connecting nodes within a given radius thus improving the continuity and flow of the curve the example of the curve can be seen in [11, 12].

V. Interpolating Curve Planner

Interpolating curve planners are commonly used to create path having a predetermined set of waypoints. These allow the path planner to generate a more feasible, comfortable path that follows the vehicle dynamics.

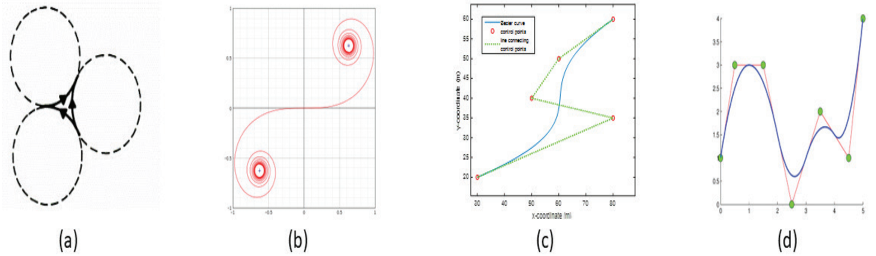

Figure 2: (a) Lines and Circles (b) Clothoid curves (c) Bezier curve (d) Spline curve[21]

A. Lines and Circles

The path planning using lines and circles depends on the different properties of both line and circle. It will be utilized to determine the shortest path for the car to travel. It could be used in both forward and reverse directions. A simple circular equation is used to create the circular path from one point to another.

Figure 2(a) shows the formation of different circles based on radius. The path for the vehicle for parking the vehicle in a parking lot is determined using the lines and circle method [13]. Other report has calculated the path planning for finding the shortest path for the vehicle to follow [14]. The main advantage of this type of curve is the computation time and costs are very less. It could be easily calculated, but it could not be used in sharp turns and intersections.

B. Clothoid Curves

Clothoid curves are parametric curves that change their curvature linearly with respect to the length of the curve. Path planning can be done using clothoid curve due to the curve is smooth and has continuity. The curve has different properties that help create a path for the vehicle to travel through different intersections.

The curvature of the curve increases as the length of the curve increases, as shown in Figure 2(b). The computation uses the clothoid curve's curvature and its derivative to determine the path's smoothness, removing the need for a subsequent stage of optimization to increase smoothness and bound the curvature. The curvature of the curve changes linearly with respect to its arc length.

Silva and Grassi [15] have presented a path planning method using clothoid curves and circular arcs to navigate vehicle through a roundabout. Vorobieva et al. [16] reported that they have used clothoid curves as a path planner for vehicle parking. Meanwhile, Ravankar et al. [17] also used clothoid curves for the parallel parking of the vehicle.

C. Bezier Curve

Bezier curves are parametric curves used in path planning, computer graphics and related fields. This curve is based on the Bernstein polynomial which is further developed. This curve is created using control points to control the curvature and shape of the curve [18]. The change in the placement of the control points will change the shape of the curve which helps in altering the shape of the curve.

Figure 2(c) shows a Bezier curve with five control points. The use of Bezier curve can be found in a report by Rastelli and Peñas [19]. The authors have used Bezier curve-based path planning for autonomous vehicles inside the roundabout.

Lattarulo and Perez [20] and Han et al. [21] also have used Bezier curve-based path planning techniques for the trajectory generation of autonomous vehicles as it can produce a smooth and continuous path for the vehicle to follow. The Bezier curve is combined with other kinds of curve fitting methods to easily achieve a continuous and smooth path for the vehicles to follow.

D. Spline Curve

A piecewise polynomial parametric curve is divided into sub-intervals that can be described as polynomial curves is referred to as a spline. Spline curves are useful for smooth and obstacle-avoiding path planning due to several characteristics. They have calculated closed-form functions effectively. The quantity of B-spline basis functions employed to define them affects both their capacity to approximate objects and their shape characteristics [28]. Figure 2(d) shows a spline curve with multiple control points.

Kogan and Murray [22] and Berglung et al. [23] have used spline curves for vehicle path generation. The spline curves are primarily used in junctions or to tackle a road curve. The main advantages of these curves are the computational cost is low and the curve formed is continuous and piecewise it can be easily adjusted. The main disadvantage of this curve is, that it tries to be continuous and smooth but does not consider the road shape and there are chances of collision with other vehicles present on the road.

VI. Conclusion

The improvement in the path planning for autonomous vehicles is an important step as it helps navigate the vehicle through different intersections obeying the traffic rules and road layout in different environments. In this paper, the path planning methods are studied by reviewing the properties of each path planning method. Path planning for autonomous vehicles in different environments is a concern. The vehicle must travel through different kinds of environment during travelling. To achieve such kind of variation in the path planning module, different concepts were introduced to create paths such as the graph search-based planner, sampling-based planner, and interpolating planner which is further explained in this paper. Among these methods, graph search-based and interpolating planners are widely used which helps to improve the path flow by comparing it with the previously determined points. All the different methods were enhanced by different researchers to improve the use of the planner. However, the need for efforts is important on the path planning stage to get a completely smooth path for the vehicle to follow through different environments.

VII. Acknowledgement

The author would like to thank for the financial support of the project provided under the fundamental research grant scheme (FRGS) No. FRGS/1/2021/TK02/UMP/02/2 (University Reference RDU210103) by Ministry of Education Malaysia.

VIII. References

[1] A. Bacha *et al.*, "Odin : Team VictorTango ' s Entry in the DARPA Urban," vol. 25, no. 8, pp. 467–492, 2008.

[2] J. Hwang, S. S. Lim, and K. H. Park, "A Fast Path Planning by Path Graph Optimization," no. February, 2003.

[3] R. Kala and K. Warwick, "Multi-level planning for semi-autonomous vehicles in traffic scenarios based on separation maximization," *J. Intell. Robot. Syst. Theory Appl.*, vol. 72, no. 3–4, pp. 559–590, 2013.

[4] W. Li, C. Yang, Y. Jiang, X. Liu, and C. Su, "Motion Planning for Omnidirectional Wheeled Mobile Robot by Potential Field Method," vol. 2017, 2017.

[5] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, "Anytime search in dynamic graphs," vol. 172, pp. 1613–1643, 2008.

[6] J. Ziegler, M. Werling, and J. Schröder, "Navigating car-like Robots in unstructured Environments using an Obstacle sensitive Cost Function," pp. 787–791, 2008.

[7] J. G. Mooney and E. N. Johnson, "A Comparison of Automatic Nap-of-the-earth Guidance Strategies for Helicopters," *J. F. Robot.*, vol. 33, no. 1, pp. 1–17, 2014.

[8] A. Kushleyev and M. Likhachev, "Time-bounded lattice for efficient planning in dynamic

environments," *Int. Symp. Comb. Search, SoCS 2009*, pp. 5–12, 2009.

[9] S. Eilers, J. Boger, and M. Fränzle, "A path planning framework for autonomous vehicles," *9th Int. Work. Robot Motion Control. RoMoCo 2013 - Work. Proc.*, pp. 203–208, 2013.

[10] Y. Kuwata, J. Teo, and S. Member, "Real-time Motion Planning with Applications to Autonomous Urban Driving," vol. 17, no. 5, pp. 1105–1118, 2009.

[11] S. Karaman, "Incremental Sampling-based Methods Optimal Kinodynamic Motion Planning using Incremental Sampling-based Methods," 2010.

[12] M. R. Walter, "Anytime Motion Planning using the RRT * Anytime Motion Planning using the RRT ∗," no. June, 2011.

[13] L. Lin and J. Zhu, "Path Planning for Autonomous Car Parking DSCC2018-9195 PATH PLANNING FOR AUTONOMOUS CAR PARKING," no. September, 2018.

[14] J. A. Reeds and L. A. Shepp, "Optimal paths for a car that goes both forwards and backwards," *Pacific J. Math.*, vol. 145, no. 2, pp. 367–393, 1990.

[15] J. A. R. Silva and V. Grassi, "Path planning at roundabouts using piecewise linear continuous curvature curves," *Proc. - 2017 LARS 14th Lat. Am. Robot. Symp. 2017 5th SBR Brazilian Symp. Robot. LARS-SBR 2017 - Part Robot. Conf. 2017*, vol. 2017-Decem, pp. 1–6, 2017.

[16] H. Vorobieva, S. Glaser, N. Minoiu-Enache, and S. Mammar, "Automatic parallel parking with geometric continuous-curvature path planning," *IEEE Intell. Veh. Symp. Proc.*, pp. 465–471, 2014.

[17] A. Ravankar, A. A. Ravankar, Y. Kobayashi, Y. Hoshino, and C. C. Peng, "Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges," *Sensors (Switzerland)*, vol. 18, no. 9, pp. 1–30, 2018.

[18] A. Vinayak, M. A. Zakaria, K. Baarath, and A. P. P. A. Majeed, "A novel Bezier curve control point search algorithm for autonomous navigation using N-order polynomial search with boundary conditions," *2021 IEEE Int. Intell. Transp. Syst. Conf.*, pp. 3884–3889, 2021.

[19] J. P. Rastelli and M. S. Peñas, "Fuzzy logic steering control of autonomous vehicles inside roundabouts," *Appl. Soft Comput. J.*, vol. 35, pp. 662–669, 2015.

[20] R. Lattarulo and J. Perez, "Fast Real-Time Trajectory Planning Method with 3rd-Order Curve Optimization for Automated Vehicles," *2020 IEEE 23rd Int. Conf. Intell. Transp. Syst. ITSC 2020*, 2020.

[21] L. Han, H. Yashiro, H. T. N. Nejad, Q. H. Do, and S. Mita, "Bézier curve based path planning for autonomous vehicle in urban environment," *IEEE Intell. Veh.*

Symp. Proc., pp. 1036–1042, 2010.

[22] D. Kogan and R. M. Murray, "Optimization-Based Navigation for the DARPA Grand Challenge," *System*, 2006.

[23] T. Berglund, A. Brodnik, H. Jonsson, M. Staffanson, and I. Söderkvist, "Planning smooth and obstacle-avoiding B-spline paths for autonomous mining vehicles," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 1, pp. 167–172, 2010.