A Survey on Smooth Path Generation Techniques for Nonholonomic Autonomous Vehicle Systems

Omveer Sharma
School of Electrical Sciences
Indian Institute of Technology
Bhubaneswar, India
os10@iitbbs.ac.in

N C Sahoo
School of Elecrtical Sciences
Indian Institute of Technology
Bhubaneswar, India
ncsahoo@iitbbs.ac.in

N B Puhan
School of Elecrtical Sciences
Indian Institute of Technology
Bhubaneswar, India
nbpuhan@iitbbs.ac.in

***Abstract*— Autonomous vehicles (AVs) are at the heart of academic and industry research because of various advantages such as safety improvement, lower energy and fuel consumption, exploitation of the road network, reduced congestion and greater mobility. Obstacle avoidance, searching the safest path to follow, generation of suitable manoeuver, optimization based most comfortable trajectory along with road boundaries and traffic rules are important concerns in motion planning. The purpose of this paper is to review the existing approaches for trajectory planning in terms of their feasibility, dynamic constraints, handling obstacles, and optimality for comfort.**

Keywords*— *Autonomous vehicles, Path Planning, Trajectory planning, Obstacle handling.

I. Introduction

Traffic accidents are mainly caused due to driver's distraction and wrong decisions. These accidents can be reduced by using AVs. AVs can also improve the capacity of highway, increase safety and decrease traffic congestion [1]. The motion planning architecture of an AV is based upon the discrete parameters of driving environment and continuous parameters of traffic participants. The software architecture of an AV consists of three layers, (1) perception layer, (2) planning layer, and (3) trajectory control layer. In perception layer, estimation about surrounding scenarios is the main concern by combining information generated by real-time data from multiple sensors like camera, RADAR, LIDAR, LASER sensors etc. This information is used to define search space (where vehicle can move) or driving corridor for planning. Planning layer is the core layer of the architecture of AV. There are three main features of planning layer: (1) route planning, (2) behavior decision-making, and (3) motion planning (path planning and trajectory planning) as shown in Fig. 1. The route planner generates optimal global routes according to traffic and distance from starting point to destination by using sensor information which is provided by the perception layer. The behavior decision-making module provides reasonable abstraction to avoid collision and safe driving actions. According to action generated by the behavior decision-making module, path planning modules generate optimal local trajectories in terms of safety, comfort and accuracy.

The understanding of driving environment depends upon both discrete and continuous parameters. The sampling and searching based models are suitable for the discrete elements of the driving environment because of its advantage of the free-cost function. On the other hand, numerical optimization based models are mainly used for the continuous parameters' estimation for smooth trajectory generation. Path planning and trajectory planning play the main role in the motion of AVs. The path planner generates a path to avoid collision with static and dynamic obstacles by taking environmental constraints into account. The path generated by path planner works as a reference path for the trajectory planner. The trajectory planner considers the vehicle dynamics and associated constraints to generate safe and comfort trajectory. Continuous parameters and vehicle dynamic constraints (vehicle model) based cost function is optimized to generate smooth and comfort trajectory. This optimized trajectory is followed by AVs. The motion planning strategies along with the classification of path planning and trajectory planning are shown in Fig. 1.

This paper is organized as follows. An overview of path planning algorithms is presented in Section II. In Section III, the trajectory planning techniques are described. In Section IV, future research is highlighted. Finally, a conclusion is outlined in Section V.

II. Path Planning: State-of-the-Art

The path planner constructs a feasible path from initial configuration to the goal configuration with consideration of road boundary and geometry. The strategy to generate this path may be based upon the lowest length of the path or lowest time for traveling from initial stage to goal stage with considering the obstacles and time to collision (TC). The planner tries to find the best sequence of vehicle states. Path planning approaches can be divided into two categories: (1) sampling-based approach and (2) searching-based approach.

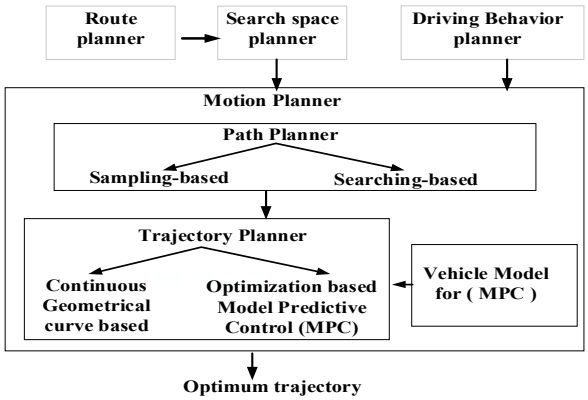

Fig. 1. System Architecture of the motion planning strategies

978-1-7281-4878-6/19/$31.00 ©2019 IEEE

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.

A. Sampling-based Approach

Sampling-based methods can be further divided into random sampling-based methods and deterministic sampling-based methods. Probabilistic road map [2]-[4] and rapidly exploring random tree (RRT) [5] techniques are widely used for robot motion. The RRT algorithm constructs a randomized data structure tree which is iteratively expanding by adding new configuration node with control input until it reaches goal configuration. The RRT algorithm and its variants quickly expand in the search space to generate collision free path. RRTs can be used in state space where the tree has grown with the sampled state. The RRT planner has been used in autonomous driving with particle-based extension like prediction step of a particle filter to address the issue of uncertainty [6]. RRT has quick exploration in free space, but it creates a jerky path [7] to solve this problem. RRT algorithm with B-spline is used in which random configuration state sample has orientation angle, sampling time, acceleration and acceleration duration [8].

To solve the time constraints, a Dual-Tree RRT (DT-RRT) method is implemented for path generation in case of two-wheeled mobile robots [9]. In this work, path generation scheme consists of two trees, one for workspace and another for state space. A closed loop controller is used with the RRT in CL-RRT algorithm. The RRT algorithm with a closed loop controller is able to generate smooth path with satisfying vehicle dynamics constraints. Each node in the tree is extended by using low-level controller with closed-loop dynamics so that each edge in the tree is dynamically feasible [10]-[12]. Improved CL-RRT approach (named RRT-Reach) has been implemented by combining Cl-RRT with threat assessment in game theory framework. Threat level can be defined as the remaining time to the earliest possible collision between host and errant vehicles [12]. Gaussian process (GP) is used to improve CL-RRT for obstacle handling for safe extension of the tree (named RR-GP) [13]. In this work, a motion pattern is generated to learn this model by combining the flexibility of GP with efficiency of RRT-Reach.

In RRT*, a new node will connect with low-cost node (it has a minimum cost path to each new node) rather than the nearest node (like RRT) [14]. Thus, RRT* algorithm gives guarantee of the low-cost path. A variant of RRT* with two modification in routines (one for extension and one for collision-free) is used for obstacle avoidance in [15]. The RRT* algorithm with half car dynamic model is implemented to generate dynamically feasible path for a circular road [16]. Minimum Violation RRT* algorithm (MVRRT*) is proposed in [17]. In this study, Kripke structure is used to form the road network and vehicle dynamics. Variants of RRT have some advantages like quick search of free space, collision checking by using advanced techniques with real-time feasibility etc. However, these approaches generate the jerky path with heavy dependency on the nearest nodes. Perfect knowledge is pre-supposed to check collision.

Deterministic sampling-based methods are considered as local search methods. Searching the complete graph in the time period is not efficient; so, a number of approaches use a restricted horizon in terms of time and space. Partial motion planning (PMP) can be used for path generation rather than searching entire state space. In [18], PMP is used with RRT and Inevitable Collision States (ICS) (a state where a collision cannot be avoided) concept. A bicycle model and non-holonomic continuous curvature (CC) metric are used in [19]. PMP and a Probabilistic Collision State (PCS) (probabilistic generalization of an ICS) concept is introduced to give guarantee about no collision [20].

B. Search-based Approach

Graph-based search methods such as A*, state lattice, hybrid A*, Dijkstra and their variants are considered as Search–based methods. A-star (A*) is a heuristic-based search algorithm in an informed environment with best first search concept to fulfill the limited time constraint [21]-[24]. These algorithms quickly generate high sub-optimal solution initially rather than improved solution within the limited deliberating time. Anytime Dynamic A* (AD*) and a heuristic function with an inflation weight factor (value greater than 1) work many times faster than normal algorithm [25]. ALT (A* search, landmarks, and the triangle inequality) algorithm and risk heuristics are combined to search path on a probabilistic grid cost map [7]. The dynamic graph is generated with fixed time differences and flexible distances between nodes in Flexible Unit A* (FU-A*) algorithm [26]. In this study, a cubic polynomial is used to connect the parent node with the child node in the adjacent lane. A hybrid A* algorithm is implemented for optimal speed generation by taking speed and acceleration constraints into account [27]. In this work, behavioral path planning (upper) generates the macro trajectory based on the discrete information of objects and traffic control device (TCD). On the other hand, the lower trajectory planning algorithm optimizes the local trajectory for energy, safety, and driving comfort by using numerical optimization.

To solve the parking problem, global planning is done by A* while the local planner uses adapted Hybrid A* (AHA*) algorithm in which two extra costs, changes in steering and changes in direction, are added in A* [28]. Integration of global and local planning has been done according to the distance between nodes. Hybrid A* algorithm and conjugate-gradient descent path smoother are used to generate smooth path by expanding four node patterns in circular manner [29]. An input sampled lattice is proposed by using coordinate transformation into chained form coordinates for vehicle model [30-31]. The state lattice is a discretization of the configuration space into a set of states. The connection between states represents a path. According to discretization property for a state lattice, two constituent paths split at a node or in other words we can say, two paths meet at a lattice node. The lattice creation for resolution can utilize parallel computation resulting in computation time savings [32]. Lattice edges can be pre-computed offline for online real-time planning. The best trajectory from a finite set of trajectories is then selected [33]. State Lattice is well suitable for a complex environment like road and non-holonomic system [25]. The approach is based on Anytime Dynamic A* with multi-resolution of dynamic

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.

feasible state lattice space. Fixed state lattices restrict the movement of vehicles and search time will increase because of small differentiation in lattice set trajectories. To overcome this issue, spatio-temporal state lattices are proposed [34] in which time and configuration space are combined into a single manifold. An algorithm to generate input & state space lattices is proposed to address the curvature issue [35]. Quartic curvature polynomials are used and both path & speed are iteratively optimized during trajectory optimization [36]. State lattice is also used for road coordinate system, in which spatio-temporal space is explored by dynamic programming algorithm to find trajectory that encodes manoeuvers [37].

A focused spatio-temporal search lattice planner is used to generate smooth and comfort trajectory in a safe space [38]. A new graph structure namely time-bounded lattice in which short term planning in time and long term planning without time are merged to generate two-stage path [25]. Lattice planners have some advantages like low computation power, feasibility of path in dynamic environments, guarantee of smoothness, capability to handle vehicle dynamic constraints, pre-computed offline lattice edges for online real-time planning etc. However, lattice planners may lead to time inefficiency and exhaustive oscillations.

III. Trajectory Planning: State-of-the-Art

A trajectory should fulfill the requirements of state constraints of vehicle dynamics and traffic environment with consideration of smooth trip & comfort for passengers after predicting best manoeuver and the best path to follow by the vehicle. Trajectory planning considers the continuous elements of the environment. The trajectory planning methods can be divided into two categories, (1) continuous geometric curve based methods and (2) Model Predictive Control (MPC).

A. Continuous geometric curve-based methods

First, a smooth geometric curve with required properties of smoothness & controllability is selected according to generated path and manoeuver. Then, a best trajectory is selected by cost function optimization and a sequence of control inputs is provided to the actuator for following the trajectory by AV. Most popular curves are Bezier curves [39]-[40], Spline curves [41]-[43], polynomial spirals [38, 44], Nelson polynomials [45], arcs [46], and clothoids [47].

(a) Bezier curves

The shape of brazier curves depends upon control point $\boldsymbol{P}_0$ through $\boldsymbol{P}_n$, where n is the order of the curve. The curvature of the curve depends upon the number of control points (n = 1 for linear, n=2 for quadratic, etc.). The first and last control points will always be the endpoints of the curve. Generally, intermediate control points do not lie on the curve.

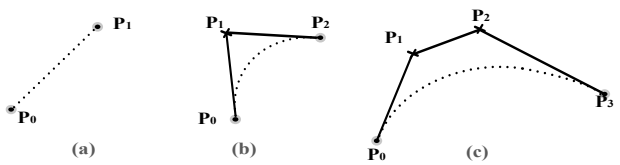

Fig.2. Bezier curves: (a) 1st order, (b) 2nd order and (c) 3rd order.

The equation of the curve is defined as:

$$B(t)=\sum_{i=0}^{n} C_i^n\, t^i (1-t)^{n-i} P_i, 0\le t\le 1; 1^{\text{st}}\text{order } B(t)=(1-t)P_0+tP_1(1-t)$$

$$2^{\text{nd}}\text{order } \mathrm{B}(t)=(1-t)^2 P_0+2(1-t)tP_1+\mathrm{t}^2P_2$$

$$3^{\text{rd}}\text{order } \mathrm{B}(t)=(1-t)^3 P_0+3(1-t)^2 tP_1+6(1-t)\mathrm{t}^2P_2+\mathrm{t}^3P_3 \quad \ldots (1)$$

Bezier curve is able to fulfill the properties of smoothness and controllability. The cost function of the curve takes into account of smoothness, curvature, arc length of the path along with distance between path and the lane centerline. Cubic Bezier curves have been used as a path generator in [40]. 4th-order Bezier curve for smooth trajectory generation with obstacle avoidance capabilities is used in [48]. Road constraints and static obstacles are considered to generate control points for the Bezier curve. A comparative analysis between 5th and 3rd order Bezier curves for single and double lane change manoeuvers is presented in [49]. In this study, 5th order curves give more feasible and smooth path in a dynamic environment. The curvature from the generated trajectory is used to find the steering rate control input.

(b) Spline curves

A minimum jerk level for desires path is the key to feasible trajectory for vehicle control and comfort. A spline path can be used to fulfill constraints and minimum jerk. The equations for cubic spline for segment (i) are given by [32]:

$$\frac{d^3y}{dx^3}=(-1)^{i+1}k, \frac{d^2y}{dx^2}=(-1)^{i+1}kx+C_{i,1}$$

$$\frac{dy}{dx}=\frac{(-1)^{i+1}kx^2}{2}+C_{i,1}x+C_{i,2},$$

$$y=\frac{(-1)^{i+1}kx^3}{6}+C_{i,1}\frac{x^2}{2}+C_{i,2}x+C_{i,3} \quad \ldots (2)$$

Third derivative of the position is piecewise constant with equal magnitude and alternate sign. While the second derivative is piecewise linear and continuous, the first derivative is also differentiable and continuous. The jerk is minimized by considering the specified constant. The cubic polynomial spline is used to generate multiple trajectories along these splines with the lateral shift and rollout capability [43]. The edge track is also considered to add information about the span of the spline. The cost function considers obstacles along the trajectory and gives them high penalty. The B-spline results in the Reproducing Kernel Hilbert Space (RKHS) provide a smooth and controllable trajectory [50]. The cubic spline based trajectory representation has been considered and cost function is associated with constraint of path within the safe driving corridor [41]. In this study, the trajectory is divided into shared path segment and contingency path with considering all constraints. Spline with distance and velocity constraints is implemented to generate a safe trajectory for autonomous driving in no lane road condition [42].

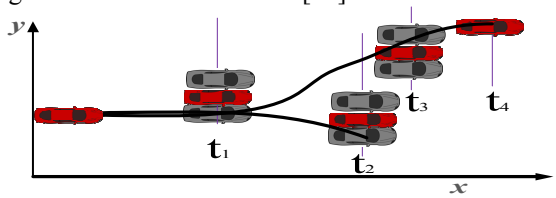

Fig. 3: Spline with objects and barrier [32]

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.

Cubic Splines and Quintic Splines are used to keep the jerk level low. After optimization, a minimum jerk trajectory is chosen [32].

(c) Polynomial spirals

For constant acceleration profile, polynomial spirals can be used. 4th order polynomial for position is represented as [51]:

$$x = a_1 t^4 + a_2 t^3 + a_3 t^2 + a_4 t + a_5$$
$$y = f(x) = f(a_1 t^4 + a_2 t^3 + a_3 t^2 + a_4 t + a_5) \qquad \ldots (3)$$

A path as a cubic polynomial spiral is defined in [44]. In this study, the cubic polynomial function is used to join spatio-temporal lattice. They consider curvature in the state lattice vector. Two-step motion planning system is also used for both urban and highway driving to compensate the planning time [38]. In first step, the reference trajectory according to road geometry is generated by using dense lattice sampling and optimization. Cubic interpolation here generates the seeding path. During the second step, further trajectories that satisfy dynamic constraints for execution are generated and evaluated.

Nonlinear optimization algorithm is used to calculate local minima in this work. 4th order polynomials of time based on differential flatness and dynamic bicycle model are used for autonomous driving in the presence of moving obstacles [51]. A tree graph with three classes is used in trajectory planning according to different conditions on highway driving. Maximum longitudinal & lateral acceleration and maximum driving force are considered as constraints. Overtaking and car following are carried out based on vehicle presence, lane availability, and distance. Constant velocity is assumed in this work which is not suitable in the real world scenario. A new trajectory generation scheme (named as TiJi) is presented in [52]. A second order polynomial is used as control inputs: (a (t) = $\alpha_2 + 2\beta_2 t + 3\gamma_2 t^2$ and ζ (t) = $\alpha_1 + 2\beta_1 t + 3\gamma_1 t^2$) to compute the trajectory from a start state to goal state. The trajectory optimization is based on an error between the current and final state. In this approach, first, acceleration profile is computed according to vehicle acceleration constraint. After that velocity profile is computed taking acceleration into account. Then velocity profile is also adjusted according to the vehicle constraint. Quintic polynomials on longitudinal and lateral axes respectively are used to trajectory generation for overtaking on the highway [53]. This optimization requires less computation power due to simplicity of polynomial functions.

(d) Arcs

An arc can be used for constant curvature. An arc is shown in Fig. 4 for the current period. Position 'A' is the start point, 'B' is the endpoint, 'θ' is the orientation of the vehicle and 'R' is the radius of the arc. Two symmetrical circulars are joined to make the trajectory for lane changing. So, Δy should be less than half of the lane width.

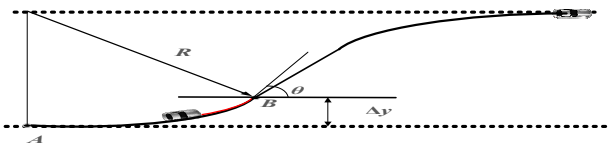

Fig. 4. Trajectory planning for lane change maneuver [46]

Two symmetrical circulars are used for trajectory planning mainly based on lane change maneuvers [46]. According to collision probability, best trajectory is chosen from a cluster of trajectories with consideration of safety and comfort. In this work, the uncertainty is based upon Gaussian distribution and prediction time. Arcs are more suitable to solve car parking problems; however the selection of start and end points is somewhat difficult. A combination of line and arc are used in searching of hybrid A* algorithm to solve car parking problem [28].

(e) Clothoids

If the curvature is constant, than the vehicle will follow circular trajectories. However, because of turning of the steering wheel, curvature variation is also introduced. Thus, the vehicle motion can be approximated as a clothoid path because of curvature variation. Clothoids are used to represent the path in [47]. The trajectory is generated from the evaluation of a subset of possible acceptable vehicle trajectories on a cost map. A new trajectory based on clothoids is used, which joins previously planned trajectory to avoid obstacles in [54]. The new trajectory is smooth up to second derivative (curvature) which provides the ability to track emergency manoeuver.

B. Optimization based Model Predictive Control (MPC)

Model Predictive Control (MPC) is a trajectory planning approach that combines different aspects of control engineering. The trajectory planning is considered as a nonlinear optimization problem with nonlinear constraints. In this approach, first, identify a safe driving corridor and then conveys the vehicle to move in this corridor. The approach considers all constraints during planning of each step rather than planning a certain path and controlling the vehicle along with it. A constraint planner predicts driving corridor over a horizon by using sensor data, predicted behaviors of the target vehicle and traffic participants. The driving corridor can be presented by lateral position vector with the dynamic constraint produced by the vehicle dynamic prediction model. It computes an optimal control input sequence according to the optimized trajectory. Three path planning methods are compared: a state lattice path planner, predictive constraint-based planning and spline-based search tree [32]. MPC is used to generate the sequence of control input in predictive constraint-based planning. A linear bicycle model with considering lateral and yaw dynamics is used for continuous trajectory generation [32]. MPC is also used to trajectory generation for lane changing manoeuver by selecting an appropriate inter-vehicle traffic gap and time instance [55]. A fast trajectory generated by MPC with Reference Horizon Generator (RHG) is used for racing car [56].

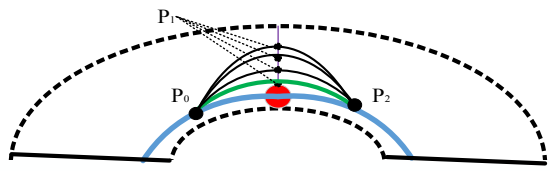

Fig. 5. Clothoids; (Blue) not feasible global trajectory due obstacle (red circle). (Green) The optimal escaping manoeuver, (black) the feasible candidates for different choices of P_1 [54].

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.

A new state space sampling-based model is introduced in predictive motion planner framework [57], in which four-dimensional state-space (position (x, y), heading θ, and curvature k) are used to represent continuous curvature of the path. Maximum speed, maximum lateral & longitudinal acceleration and curvature are considered as the constraints. Cubic polynomial spirals are used to represent the control inputs. The cost function considers distance to obstacles, distance from reference path, smoothness and prediction distance. Only static obstacles are tested in this study. A variant of the RRT based on particle filtering (PF) is also considered for generating complex trajectories. Afterwards MPC is used to generate control input sequence. The experiment is performed on a small robotic car [58]. In [59], a driving corridor to limit the action space is prepared. Sensor and map-based representations are used to describe the situation and trajectory. The trajectory is time parametrized with consideration of moving obstacles. The cost function has offset from the reference path, speed, acceleration, jerk, and yaw rate. Computational complexity to generate driving corridor becomes high with the increase of obstacles in this work. MPC is capable to solve optimization problem for best trajectory generation due to presence of dynamic model of vehicle. However, MPC is also sensitive to the number of constraints and variables during optimization process.

IV. Future Research Directions

AVs operate in a dynamic environment with uncertainty. This happens due to continuous movement of AVs and other vehicles at high speed. Road signals and disappearance & appearance of obstacles create a complex dynamic environment. Noise in measurement also creates computation time and prediction problem. Good amount of research has been done to determine status and position of the target vehicle to understand static environment with respect to the local environment. More research is required for tracking and determining the intention of traffic moving participants such as pedestrians, bikes, and other vehicles. Online vehicle to vehicle communication can be used to simplify these complex scenarios. Model predictive control has been used to ensure the required input data to follow desired trajectory under certain manoeuver with using feedback control. But the computation complexity increases in MPC because of large number of dynamic parameters. The sampling of state space can be quicker for efficient planning of complex manoeuvers.

V. Conclusions

The path which is followed by self-driving or autonomous vehicles should be comfortable and safe. The planning layer should put the vehicle in right behavior and motion with respect to constraints. This paper represents a critical review of different planning techniques which have been applied for path planning and trajectory planning to road driving of AVs. Most approaches to the trajectory level optimize a given geometric curve with respect to dynamic and static obstacles. The real world performance of current approaches can be improved by incorporation of dynamic modelling into planning.

ACKNOWLEDGEMENT

The research in this paper is supported by the KPIT Technologies Ltd, India under research grant for the project entitled “Driver behavior modelling for autonomous driving”.

Refrence

[1] Xu, Qing, et al. "Effects of vehicle-vehicle/roadside-vehicle communication on adaptive cruise controlled highway systems." *Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall. 2002 IEEE 56th*. Vol. 2. IEEE, 2002.

[2] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, ‘‘Probabilistic roadmaps for path planning in high-dimensional configuration spaces,’’ IEEE Trans. Robot. Autom., vol. 12, no. 4, pp. 566–580, Aug. 1996.

[3] L. E. Kavraki, M. N. Kolountzakis, and J.-C. Latombe, ‘‘Analysis of probabilistic roadmaps for path planning,’’ IEEE Trans. Robot. Autom., vol. 14, no. 1, pp. 166–171, Feb. 1998.

[4] S. Karaman and E. Frazzoli, ‘‘Sampling-based algorithms for optimal motion planning,’’ Int. J. Robot. Res., vol. 30, no. 7, pp. 846–894, 2011.

[5] LaValle, Steven M. "Rapidly-exploring random trees: A new tool for path planning." (1998).

[6] N. Melchior and R. Simmons, “Particle rrt for path planning with uncertainty,” in 2007 IEEE International Conference on Robotics and Automation, April 2007, pp. 1617–1624.

[7] Murphy, Liz, and Paul Newman. "Risky planning: Path planning over costmaps with a probabilistically bounded speed-accuracy tradeoff." *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011.

[8] Macek, K., Becker, M., Siegwart, R., 2006. Motion planning for car-like vehicles in dynamic urban scenarios. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4375–4380.

[9] Kwon, Hyunki, et al. "Trajectory planning for mobile robot with kinodynamic constraints." *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. IEEE, 2017.

[10] Kuwata, Yoshiaki, et al. "Real-time motion planning with applications to autonomous urban driving." *IEEE Transactions on Control Systems Technology* 17.5 (2009): 1105-1118.

[11] Aoude, Georges S., et al. "Sampling-based threat assessment algorithms for intersection collisions involving errant drivers." *IFAC Proceedings Volumes* 43.16 (2010): 581-586 [38] Georges S. Aoude (2010).

[12] Aoude, Georges S., et al. "Threat-aware path planning in uncertain urban environments." *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010.

[13] Aoude, Georges S., et al. "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns." *Autonomous Robots* 35.1 (2013): 51-76.

[14] Karaman, S., Frazzoli, E., 2011. Sampling-based algorithms for optimal motion planning. Int. J. Robot. Res. 30, 846–894.

[15] Garrote, Luis, et al. "An RRT-based navigation approach for mobile robots and automated vehicles." *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*. IEEE, 2014.

[16] Hwan Jeon, Jeong, et al. "Optimal motion planning with the half-car dynamical model for autonomous high-speed driving." *2013 American Control Conference*. IEEE, 2013.

[17] Castro, Luis I. Reyes, et al. "Incremental sampling-based algorithm for minimum-violation motion planning." *52nd IEEE Conference on Decision and Control*. IEEE, 2013.

[18] Benenson, Rodrigo, et al. "Integrating perception and planning for autonomous navigation of urban vehicles." *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006.

[19] Kaplan, Kemal, Caner Kurtul, and H. Levent Akin. "Fast lane tracking for autonomous urban driving using hidden Markov models and multiresolution Hough transform." *Industrial Robot: An International Journal* 37.3 (2010): 273-278.

[20] Althoff, Daniel, Dirk Wollherr, and Martin Buss. "Safety assessment of trajectories for navigation in uncertain and dynamic environments."

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.

Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011.

[21] P. E. Hart, N. J. Nilsson, and B. Raphael, “A formal basis for the heuristic determination of minimum cost paths,” IEEE Transactions of Systems Science and Cybernetics, vol. 4, no. 2, pp. 100–107, July1968

[22] N. Nilsson, Principles of Artificial Intelligence. Tioga Publishing Company, 1980.

[23] R. Zhou and E. Hansen, “Multiple sequence alignment using A*,” in Proceedings of the National Conference on Artificial Intelligence (AAAI), 2002, Student abstract.

[24] M. Likhachev, G. Gordon, and S. Thrun, “ARA*: Anytime A* with provable bounds on sub-optimality,” in Advances in Neural Information Processing Systems. MIT Press, 2003.

[25] Kushleyev, Aleksandr, and Maxim Likhachev. "Time-bounded lattice for efficient planning in dynamic environments." *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009.

[26] Boroujeni, Zahra, et al. "Flexible unit A-star trajectory planning for autonomous vehicles on structured road maps." *2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 2017.

[27] Lim, Wonteak, et al. "Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method." *IEEE Transactions on Intelligent Transportation Systems* 19.2 (2018): 613-626.

[28] Esposto, Francesco, et al. "Hybrid path planning for non-holonomic autonomous vehicles: An experimental evaluation." *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. IEEE, 2017.

[29] Zhang, Yu, et al. "Hybrid trajectory planning for autonomous driving in highly constrained environments." *IEEE Access* 6 (2018): 32800-32819

[30] A. Bicchi, A. Marigo, and B. Piccoli. On the reachability of quantized control systems. IEEE Transactions on Automatic Control, 4(47):546–563, April 2002.

[31] M. Pitvoraiko and A. Kelly, “Efficient constrained path planning via search in state lattices,” in *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*, 2005.

[32] Madås, David, et al. "On path planning methods for automotive collision avoidance." *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013

[33] Werling, Moritz, et al. "Optimal trajectory generation for dynamic street scenarios in a frenet frame." *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010.

[34] Ziegler, Julius, and Christoph Stiller. "Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios." *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009.

[35] Rufli, Martin, and Roland Siegwart. "On the design of deformable input-/state-lattice graphs." *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010.

[36] Xu, Wenda, et al. "A real-time motion planner with trajectory optimization for autonomous vehicles." *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012.

[37] Gu, Tianyu, and John M. Dolan. "On-road motion planning for autonomous vehicles." *International Conference on Intelligent Robotics and Applications*. Springer, Berlin, Heidelberg, 2012.

[38] Gu, Tianyu, et al. "Focused trajectory planning for autonomous on-road driving." *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013.

[39] Zhao, Pan, et al. "Dynamic motion planning for autonomous vehicle in unknown environments." *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011.

[40] Ma, Liang, Jing Yang, and Meng Zhang. "A two-level path planning method for on-road autonomous driving." *Intelligent System Design and Engineering Application (ISDEA), 2012 Second International Conference on*. IEEE, 2012.

[41] Hardy, Jason, and Mark Campbell. "Contingency planning over probabilistic obstacle predictions for autonomous road vehicles." *IEEE Transactions on Robotics* 29.4 (2013): 913-929.

[42] Kala, Rahul, and Kevin Warwick. "Motion planning of autonomous vehicles in a non-autonomous vehicle environment without speed lanes." *Engineering Applications of Artificial Intelligence* 26.5 (2013): 1588-1601.

[43] Wang, Miao, Tinosch Ganjineh, and Raúl Rojas. "Action annotated trajectory generation for autonomous maneuvers on structured road networks." *Automation, Robotics and Applications (ICARA), 2011 5th International Conference on*. IEEE, 2011

[44] McNaughton, Matthew, et al. "Motion planning for autonomous driving with a conformal spatiotemporal lattice." *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011.

[45] Xiu, CaiJing, and Hui Chen. "A behavior-based path planning for autonomous vehicle." *Intelligent Robotics and Applications* (2010): 1-9.Sun, H., Deng, W., Zhang, S., Wang, S., 2014. Trajectory planning for vehicle autonomou.

[46] Sun, H., Deng, W., Zhang, S., Wang, S., 2014. Trajectory planning for vehicle autonomous driving with uncertainties. In: 2014 International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS), pp. 34–38.

[47] Broggi, Alberto, et al. "Autonomous vehicles control in the VisLab intercontinental autonomous challenge." *Annual Reviews in Control* 36.1 (2012): 161-171.

[48] González, David, et al. "Continuous curvature planning with obstacle avoidance capabilities in urban scenarios." *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014erand tracking based on a Bezier curve for a steering rate controller of autonomous vehicles. In: IEEE Conference on Intelligent Transportation Systems, ITSC, pp. 436–441.

[49] Bae, I., Moon, J., Park, H., Kim, J.H., Kim, S., 2013. Path generation and tracking based on a Bezier curve for a steering rate controller of autonomous vehicles. In: IEEE Conference on Intelligent Transportation Systems, ITSC, pp. 436–441.

[50] Dong, Chiyu, and John M. Dolan. "Continuous Behavioral Prediction in Lane-Change for Autonomous Driving Cars in Dynamic Environments." *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018.

[51] Sawodny, O., Zimmermann, J., Lutz, A., 2010. Motion planning for an autonomous vehicle driving on motorways by using flatness properties. In: 2010 IEEE International Conference on Control Applications, pp. 908–913.

[52] Delsart, Vivien, Thierry Fraichard, and Luis Martinez. "Real-time trajectory generation for car-like vehicles navigating dynamic environments." *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009.

[53] He, Xu, et al. "A Human-like Trajectory Planning Method by Learning from Naturalistic Driving Data." *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018.

[54] Bertolazzi, Enrico, et al. "Efficient Re-planning for Robotic Cars." *2018 European Control Conference (ECC)*. IEEE, 2018.

[55] Nilsson, Julia, et al. "Lane change maneuvers for automated vehicles." *IEEE Transactions on Intelligent Transportation Systems* 18.5 (2016): 1087-1096.

[56] Caporale, Danio, et al. "A Planning and Control System for Self-Driving Racing Vehicles." *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*. IEEE, 2018.

[57] Li, Xiaohui, et al. "A unified approach to local trajectory planning and control for autonomous driving along a reference path." *2014 IEEE International Conference on Mechatronics and Automation*. IEEE, 2014.

[58] Berntorp, Karl, et al. "Control Architecture Design for Autonomous Vehicles." *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018.

[59] Ziegler, J., Bender, P., Dang, T., Stiller, C., Preliminaries, A., 2014a. Trajectory planning for BERTHA – a local, continuous method. In: 2014 IEEE Intelligent Vehicles Symposium (IV), pp. 450–457

Authorized licensed use limited to: Technische Hochschule Ingolstadt. Downloaded on August 10,2022 at 08:21:38 UTC from IEEE Xplore. Restrictions apply.