2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

A Practical Trajectory Planning Framework for Autonomous Ground Vehicles Driving in Urban Environments

Xiaohui Li, Zhenping Sun, Zhen He, Qi Zhu, and Daxue Liu

Abstract— **This paper presents a practical trajectory planning framework towards fully autonomous driving in urban environments. Firstly, based on the behavioral decision commands, a reference path is extracted from the digital map using the LIDAR-based localization information. The reference path is refined and interpolated via a nonlinear optimization algorithm and a parametric algorithm, respectively. Secondly, the trajectory planning task is decomposed into spatial path planning and velocity profile planning. A closed-form algorithm is employed to generate a rich set of kinematically-feasible spatial path candidates within the curvilinear coordinate framework. At the same time, the velocity planning algorithm is performed with considering safety and smoothness constraints. The trajectory candidates are evaluated by a carefully developed objective function. Subsequently, the best collision-free and dynamically-feasible trajectory is selected and executed by the trajectory tracking controller. We implemented the proposed trajectory planning strategy on our test autonomous vehicle in the realistic urban traffic scenarios. Experimental results demonstrated its capability and efficiency to handle a variety of driving situations, such as lane keeping, lane changing, vehicle following, and static and dynamic obstacles avoiding, while respecting traffic regulations.**

I. INTRODUCTION

Autonomous driving technologies have great potentials to improve driving safety, comfort and efficiency. The past three decades have witnessed the remarkable development of autonomous driving technologies, which have attracted considerable interest and efforts from both academia and industry. Particularly in the last decade, with the advent of advances in sensing, computer technologies, artificial intelligence, and so forth, the autonomous driving technologies have become an extraordinary active research field in the robotics community. During the DARPA Grand Challenges (DGC) and DARPA Urban Challenge (DUC), AGVs exhibited their abilities of autonomously driving in both off-road and on-road environments [1], [2], [3], [4].

When AGVs advance towards handling realistic dynamic urban traffic scenarios, they are required to be capable of handling a variety of complex maneuvers, such as lane keeping, vehicle following, lane changing, merging, while respecting traffic rules, avoiding static and dynamic obstacles, and interacting with other traffic participants [5]. To enable the vehicle to autonomously drive in realistic urban environments, various state-of-the-art technologies are required. As one of the core technologies, motion planning plays a critical role in coping with these challenges. Based upon amount of previous research on this topics [3], [4], [6], this paper focuses on addressing the trajectory planning problem for AGVs driving in realistic urban scenarios in a practical way rather than a theoretical way.

*This work was supported in part by the National Natural Science Foundation of China under Grant 61473303 and 91220301.

Xiaohui Li, Zhenping Sun, Zhen He, Qi Zhu, and Daxue Liu are with the College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha, P. R. China (e-mail: xiaohui_lee@outlook.com, sunzhenping@outlook.com, hezhen.nudt@gmail.com, zhq_cs@126.com).

II. RELATED WORK

A large number of motion planning algorithms have been proposed for AGVs driving in urban environments. These algorithms can be roughly categorized into two major groups. One focuses on computing a long-term collision-free path using deterministic or random graph-search algorithms [7], [8], [9], which are capable of handling complex environmental constraints and efficiently preventing reactive motions. Most of these powerful algorithms are especially suitable for AGVs driving in partially or completely unknown environments. However, the resultant paths are often comprised of concatenated short-term motion primitives, such as stop-and-redirect motions, which cannot meet the smooth and relatively high-speed requirements in typical urban driving scenarios. In addition, most of these search-based algorithms are too computationally demanding to deal with the dynamic traffic situations.

There has been substantial research on trajectory planning for AGVs driving in urban environments. A well-known and efficient strategy follows a discrete sampling and optimization scheme. In [3], a model-predictive motion planner is proposed using the curvature polynomials to parameterize the control input space based on the work in [10]. A similar work can be found in [11] and [12]. In [13], nudges and swerves laterally shifting from the base trajectory are employed to avoid obstacles along the reference path while being aligned with the reference path. A similar scheme is adopted in [5], but considering both the lateral and longitudinal movements within the Frenét framework. The authors in [14] considers both the vehicle model and closed-loop feedback control laws to generate drivable trajectory candidates aligning with the reference path. A similar strategy was applied in [9].

Based on the Boss work in DUC [3], a powerful trajectory planner for highway driving is proposed using spatiotemporal lattice conforming to the road geometry [15]. Gu et al. proposed a two-step motion planner aiming at addressing both urban and highway driving in one framework [16]. Most of these approaches generate a rich set of drivable trajectory candidates firstly. Then the best trajectory which minimizes a specified cost function is selected for execution. Instead of applying the discrete optimization scheme, very recent work in [6] formulates the trajectory planning problem as a nonlinear optimization problem with inequality constraints. Based on a reference corridor provided by the digital map, online perception data, and high-level decision process, the planner explicitly considers hard constraints imposed by both static obstacle and dynamic vehicles, as well as the maximal steering curvature. The Newton type method is applied to solve the optimization problem in the continuous state space.

Based upon previous work mentioned above, this paper focuses on addressing the trajectory planning problem for AGVs driving in urban environments. The reference path is extracted using the digital map relying on the accurate and robust LIDAR-based localization technology. We admit that the trajectory planner assumes more responsibility to handle many dynamic traffic situations in an reactive manner with explicitly considering the constraints imposed by vehicle motion model and dynamic environments. Even though, it is still necessary for the trajectory planner to take advantage of maneuver decisions from the behavioral planner. In this way, it does not only significantly reduce the search space for trajectory planning but also avoid over-reactive or aggressive maneuvers in most of non-imminent traffic situations.

III. Framework and Methodology

An overview of the system framework is described here. As shown in Fig. 1, the reference path is derived firstly. To meet realtime requirements, we apply the nonlinear optimization algorithm to refine the reference path locally rather than globally. After that, the cubic B-spline algorithms is used to interpolate the optimized path in order to obtain dense waypoints. Considering the road geometry constraints as well as handling both static and moving objects, we decompose the trajectory planning problem into two subtasks: spatial path planning and velocity planning. Instead of using the optimization scheme to produce a sole optimal trajectory in each replan cycle [6], the trajectory planner is capable of generating a rich set of sub-optimal candidates, which could efficiently overcome noises in the perception and localization systems. Besides, it ensures that the vehicle could safely stop in the imminent situations.

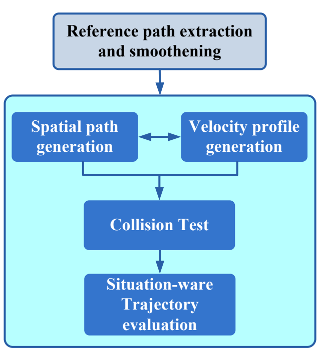


Fig. 1: System framework overview.

To guarantee driving safety and improve driving comfort, the collision test is performed to trim the trajectory collide with static and moving objects. Then, a objective function with a set of cost terms is carefully designed to select the best trajectory candidate. To handle various realistic driving situations in urban environments, a situation-aware policy is introduced to adaptively tune the weights of the cost terms in the objective function. The best trajectory is transformed into actuator commands by the trajectory tracking controller subsequently.

A. Online perception and localization for trajectory planning

Many researchers have proposed efficient trajectory planning algorithms for AGVs driving in highways and urban environments. Most of them assume that lanes could be obtained as prior information. To the best of the authors' knowledge, even in urban environments, the lanes cannot be robustly detected in many areas. Besides, the lane detection performance is highly dependent on weather and light conditions. A possible method to mitigate these issues is using digital maps combined with GPS-based inertial navigation System. Nevertheless, it may still suffer from the localization noises resulting from the block and reflection effects caused by tall buildings or forests. All of these negative effects make it hard to obtain robust localization information of lane-level accuracy. To overcome these problems, many researchers take advantage of LIDAR or vision-based localization technologies [17], [18].

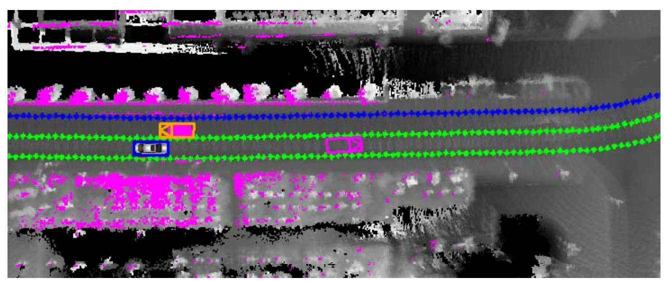


Fig. 2: Local trajectory planning map based on reference information from the digital map.

In this study, the trajectory planner uses the reference path information based on the LIDAR-based localization approach, which is similar to that proposed in [17]. Combining the data from GPS, IMU, and Velodyne HD-LIDAR, the high-resolution grid-map are created offline. Then we use the online LIDAR perception data for real-time localization. The experiments demonstrate its ability to provide reliable localization information for realtime autonomous driving in the urban environments with narrow roads. Based on the localization method, we are able to use lane information provided by the digital map. As shown in Fig. 2, the green and blue waypoints represent the right and left lanes respectively. The grey-scale mother map is created offline with high resolution ($20cm \times 20cm$ per cell) using cloud point data of the 64-beam HD-LIDAR. The grey-scale value represents the height value of each grid. The cells with magenta color indicates online detected obstacles, including dynamic and static obstacles. The rectangles represent the detected dynamic obstacles with arrows indicating their moving directions.

B. Reference path smoothing

Based upon the aforementioned localization approach, rich information of the digital map can be utilized for AGVs. For trajectory planning, the reference path could be the centerline of the desired lane, which could be obtained from behavioral commands. However, the centerline may not be smoothly enough to be tracked in many situations, e.g. when the vehicle negotiates a curvy turn. To ensure that the desired path could be easily tracked by the vehicle, the conjugate gradient method is applied to refine the reference path. A similar method was referred in [8]. It is able to smoothen the centerline in a continuous space. Giving discrete waypoints of the centerline as seed vertices $\mathbf{x}_i = (x_i, y_i), (i = 1, ..., N)$, which can be slightly tuned to minimize the following objective function:

$$\arg\min(\omega_1 \sum_{i=1}^{N-1} (\Delta\mathbf{x}_{i+1} - \Delta\mathbf{x}_i)^2 + \omega_2 \sum_{i=1}^{N-1} (\frac{\Delta\theta_i}{|\Delta\mathbf{x}_i|})^2) \quad (1)$$

where $\Delta\mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_{i-1}, (i >= 2)$ is the displacement vector at the vertex $\mathbf{x}_i = (x_i, y_i)$; $\Delta\theta_i = \left|\tan^{-1}\frac{\Delta y_{i+1}}{\Delta x_{i+1}} - \tan^{-1}\frac{\Delta y_i}{\Delta x_i}\right|$ is the vector direction change at the vertex; ω_1, ω_2 are weights. The first term minimizes the the vector change between the adjacent vectors, while the second term minimizes the cumulative curvature along the path.

After applying the aforementioned optimization algorithm, a much smoother path could be obtained. In practice, to achieve realtime performance, the waypoints taken for smoothing are often sparse and the distances between the waypoints are often large. To solve this problem, we take the sparse point as the control points and use the cubic B-spline to interpolate the refined reference path. In the meanwhile, four dimensional states (x, y, θ, κ) of each point along the smooth path could be obtained in an analytical manner. The tangent angle $\theta(i)$ and the curvature $\kappa(i)$ of the waypoints along the path are defined as

$$\theta = \tan^{-1}(\frac{\dot{x}}{\dot{y}}) \quad (2)$$

$$\kappa(t) = \frac{\dot{x}\ddot{y} - \ddot{x}\dot{y}}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \quad (3)$$

As shown in Fig.3, the green curve with equidistant waypoints is the optimized reference path. To simplify the nonlinear optimization problem and achieve realtime performance, environmental constraints are not explicitly considered during the path smoothing process. Obstacle avoidance will be taken into account by the trajectory planning algorithms discussed later.

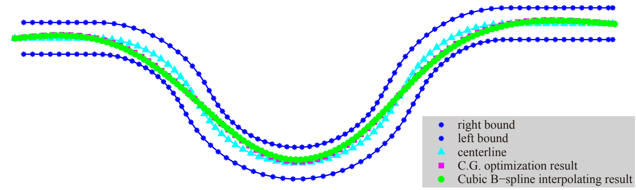


Fig. 3: Conjugate gradient algorithm is applied to smooth the centerline. Then dense waypoints are obtained using cubic B-spline interpolating method.

C. Spatial path generation based on the curvilinear coordinate framework

When AGVs drive in urban environments, in order to safely and friendly interact with other traffic participants like human drivers while respecting traffic regulations, it is necessary to generate trajectories explicitly considering both road geometry and speed constraints. To achieve this, instead of using the Cartesian coordinate framework, the road geometry model is described using the curvilinear coordinate framework. More precisely, the reference path is parameterized as a function of the arc length. In this way, vehicles' movements could be naturally divided into two dimensions: one is the lateral movement, and the other is the longitudinal movement. Relying on this framework, the trajectory planning are inherently decoupled into spatial path generation and velocity profile generation. The spatial path generation problem is solved firstly.

After using the smoothing algorithm mentioned above, the reference path is obtained and taken as the base frame of the curvilinear system. To ensure the consistency between consecutive replans, the generated path is required to be aligned with the base frame. In [3], a model-predictive motion planner is proposed, which sampled a finite set of terminal states laterally shifting from the base frame firstly. Then, the path generation problem was reformulated as a two-point boundary value problem. To reduce the solution space while guaranteeing the representativeness of output maneuvers, curvature polynomials were employed to parameterize the control input space. The approach considered constraints imposed by the sampling terminal configurations. However, it ignored the road geometry. As a consequence, when the reference path is curvy, it is difficult to generate long-term paths being aligned with the road geometry.

To overcome this disadvantage, we refer to the trajectory planning strategy mentioned in [5]. The well-known sampling-based motion planning strategy is employed to efficiently generate a rich set of path candidates, which are capable of avoiding static obstacles as well as being aligned with the reference path geometry. The details are described as follows.

Based on the curvilinear coordinate framework, the sampling terminal states could be represented by two parameters, one is the preview distance s_f along the reference path and the other is the lateral offset l_f. This is different from the sampling method in the Cartesian coordinate [3]. The preview distance s_f determines how fast the vehicle is regulated to be aligned with the base frame. In practice, it could be adaptively tuned according to the vehicle speed without violating the maximal lateral acceleration limit. The lateral offset l_0 is the perpendicular distance between the vehicle current position and the corresponding point the nearest point along the base frame.

Base on the differential flatness control theory referred in [19], polynomial splines can be applied to represent the smooth transition from the current lateral offset l_0 to the sampling terminal lateral offset l_f. To ensure the path could be analytically solved and cubic polynomial is applied as referred in [20], quintic polynomial could also be applied to guarantee the continuity of the first-derivative of the lateral offset. But in practice, we found that it is also more sensible to the initial conditions.

$$l(s) = c_0 + c_1 s + c_2 s^2 + c_3 s^3 \tag{4}$$

Hence, it can be obtained

$$l(s_0) = l_0, l(s_f) = l_f \tag{5}$$

Besides, the heading difference $\theta(s)$ between the vehicle and the base frame could be derived by the first derivative of the later offset $l(s)$ with respect to the arc length s.

$$\theta(s) = \arctan(\frac{dl}{ds}) \tag{6}$$

For simplicity, the terminal heading of the generated path is set to be same as that of the base frame at the preview distance s_f. The vehicle's initial heading angle may be different from the tangent direction of the base frame at the current position, so we can obtain:

$$\theta(s_0) = \theta_0, \theta(s_f) = 0 \tag{7}$$

θ_0 is the initial heading difference between the vehicle and the base frame. Since the number of unknown parameter-s is equal to that of equations, the unknown parameters $\{c_0, c_1, c_2, c_3\}$ could be analytically solved.

The curvature of the generated spatial path $\kappa(s)$ can be derived from the curvature κ_b of the base frame [21], [20], [5].

$$\kappa(s) = \frac{1}{Q}(\kappa_b + \frac{(1 - l\kappa_b)(d^2l/ds^2) + \kappa_b(dl/ds)^2}{Q^2}) \tag{8}$$

with

$$Q = \sqrt{(dl/ds)^2 + (1 - l\kappa_b)^2}$$

To guarantee the kinematical-feasibility of the generated path, $(1 - l\kappa_b)$ is required to be greater than zero. In other words, using this method is not able to generate drivable paths when the lateral offset l_s is greaterer than the radius of the corresponding point on the base frame. The motion planning approach referred in [3] could be applied to overcome the issue. Since the curvature state is the invariant intermediate variable between the curvilinear coordinate framework and the Cartesian coordinate framework, the spatial path in the Cartesian coordinate could be numerically solved using the vehicle kinematic equations as follows.

$$\begin{aligned}
\dot{x}(s) &= Q\cos(\theta(s)) \\
\dot{y}(s) &= Q\sin(\theta(s)) \\
\dot{\theta}(s) &= Q\kappa(s)) \\
\kappa(s) &= max\{\kappa_{\min}, \min\{\kappa(s), \kappa_{\max}\}\}
\end{aligned} \tag{9}$$

where $\kappa_{\min}$ and $\kappa_{\max}$ indicate steering angle constraints imposed by the vehicle. When vehicles drive at high speed, it is necessary for the local planner to generate long-term path to avoid reactive maneuvers. To achieve this, $d^2l/ds^2 = 0$ and $dl/ds = 0$, when $(s >= s_f)$. In this way, the generated paths are set to be aligned with the reference path after they reach the sampling terminal states.

As shown in Fig. 4(a), the generated spatial path candidates are able to connect the current vehicle state with the sampling terminal states. Figure 4(b) illustrates that when the vehicle has the initial lateral deviation (curves with dark green color) or a certain initial heading difference (curves with red color) from the base frame. The path generation approach is capable of generating spatial path candidates which are aligned with the base frame.

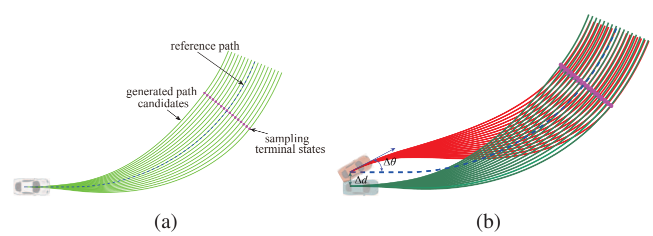


Fig. 4: Path candidates generation results. (a) $l_0 = 0$ and $\theta_0 = 0$,(b) $l_0 = \Delta d$ and $\theta_0 = \Delta\theta$

D. Velocity profile generation

Velocity planning has a significant impact on the driving safety and comfort. Especially when vehicles drive in the dynamic road traffic, where velocity planning is required to be explicitly considered. In this paper, velocity profile generation is performed considering a variety of constraints, such as commands from the behavioral planner, maximal allowed lateral and longitudinal accelerations. These constraints could considerably reduce the solution space for the velocity planning as well as allow the velocity planner to concentrate on the space where the optimal solution is most likely to exist.

In this study, a hierarchical velocity profile generation scheme is developed. The maximal speed is determined firstly. Then, the trapezoidal profile is employed to generate the initial velocity law assigned along the generated path. After that, to improve smoothness of the longitudinal control, polynomial splines are applied to smoothen the linear velocity profiles and generate acceleration-continuous velocity curves.

Firstly, in order to guarantee safety and improve driving comfort, several constraints are taken into account to determine the maximal speed as follows.

(i) The maximal allowed speed V_{limit1}. It could be derived by considering the high-level behavioral planner reasoning about the road conditions, traffic regulations and the safe distance of the generated path to the obstacles.

(ii) The maximal allowed lateral acceleration Acc_{MaxLat}

$$V_{\text{limit2}}^2 \, |\kappa| \leq Acc_{\text{MaxLat}}$$

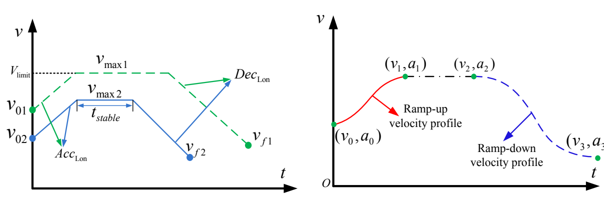
To prevent lateral force acting on tires from entering into the nonlinear or saturated zone, alleviate sideslip effects as well as ensure the yaw stability, the Acc_{MaxLat} is required to be carefully determined considering vehicle dynamics.

(iii) The maximal longitudinal velocity acceleration limit $Acc_{\text{Max\_Lon}}$ and deceleration limit $Dec_{\text{Max\_Lon}}$.

$$\frac{V_{\text{limit3}}^2 - v_0^2}{2Acc_{\text{Max\_Lon}}} + \frac{V_{\text{limit3}}^2 - v_f^2}{2Dec_{\text{Max\_Lon}}} + D_{\text{safe}} = S$$

where v_0 is the current velocity, v_f is the terminal speed, and S represents the path length. The terminal speed v_f is determined by the high-level behavioral planner, which considers the surround moving vehicles and traffic regulations. For instance, when the vehicle executes a vehicle following maneuver, the terminal speed is set to be the same as the front vehicle. When the vehicle approaches a stop sign/line or meets the passing pedestrians, v_f is set to be zero. The details will not be discussed in this paper. Besides, the safety distance D_{safe}, which involves the response time delay t_{delay} and imminent braking distance $D_{imminent}$. The imminent braking distance $D_{imminent}$ can be represented as a function with respect to v_f, e.g. the well-known constant time gap law can be employed.

$$D_{\text{safe}} = t_{\text{delay}} v_0 + D_{\text{imminent}}(v_f)$$



(a) (b)

Fig. 5: Velocity profile generation results. (a) trapezoidal velocity profile generation,(b) velocity profile smoothing

As shown in Fig. 5, the trapezoidal velocity profile is employed to generate the initial velocity profile. For simplicity, the linear velocity profile with constant acceleration and deceleration values is used. Base on our experimental experience gained in field tests, to avoid oscillations of the longitudinal control, the velocity is required to be kept at the maximal speed v_{max} for a certain period of time, which is longer than t_{steady}. Therefore, v_{max} does not necessarily reach the maximal allowed speed V_{limit}, such as the blue solid velocity curve shows in Fig. 5. Given the initial speed v_0, terminal speed v_f, the path length S, the maximal allowed speed V_{limit}, and user-specified parameters $\{Acc_{\text{lon}}, Dec_{\text{lon}}, t_{\text{stable}}\}$, the closed-form velocity profile can be obtained.

For the smooth tracking purpose, we smoothen the initially generated trapezoidal velocity profiles to guarantee the continuity of the longitudinal acceleration. Inspired by the parametric velocity profile generation method in [16], the velocity is smoothed using cubic polynomial.

$$v(t) = v_0 + at + bt^2 + ct^3$$

Correspondingly, the acceleration $acc(t)$ and path length $s(t)$ can be derived through the first derivative and integration of the velocity function respectively. Given initial velocity v_0 and acceleration a_0, the terminal velocity v_f, acceleration a_f, and path length s_f. The unknown parameters $\{a, b, c, t_f\}$ in (10) could be analytically solved via the following equations.

$$\begin{aligned}
&acc(0) = a = a_0 \\
&v(t_f) = v_0 + at_f + bt_f^2 + ct_3^2 = v_f \\
&acc(t_f) = a + 2bt_f + 3ct_f^2 = a_f \\
&s(t_f) = v_0 t_f + \frac{1}{2}at_f^2 + \frac{1}{3}bt_f^3 + \frac{1}{4}ct_f^4 = s_f
\end{aligned} \tag{10}$$

As shown in Fig. 5, the acceleration-continuous S-shaped ramp-up velocity profile (the red solid line) and ramp-down velocity profile (the blue dash line) are solved respectively.

IV. Trajectory Candidates Evaluation

The collision test is performed to trim the part of trajectories colliding with obstacles instead of the entire trajectory. To achieve this, the local trajectory planning map is represented as an occupancy grid map based on the online perception information. To reduce the computational complexity of the collision test, the rectangular shape of the vehicle is approximated by a set of circles with the same radius [6].

Noticed that avoidance of the moving objects which run in the same or opposite directions of ours are handled through the longitudinal velocity planning using the lane information from the digital map. To avoid the pedestrians or vehicles runs in the traffic intersection, we conservatively consider the prediction of the their movements within a finite time horizon (e.g 3s). The research in [22] provides a promising approach to handle moving obstacle avoiding in dynamic traffic scenarios.

After the collision test, the remaining trajectories are evaluated via an objective function comprised of four weighted cost terms as

$$i^* = \underset{i=1}{\overset{N}{argmin}}(\omega_p J_p(\tau_i) + \omega_s J_s(\tau_i) + \omega_d J_d(\tau_i) + \omega_c J_c(\tau_i)) \tag{11}$$

where $J\{p, s, d, c\}$ are cost terms, $\omega\{p, s, d, c\}$ are the corresponding weighted factors, and $\tau_i (i = 1, ..., N)$ is the evaluated trajectory candidate. The cost value is normalized to be within [0, 1]. The cost term J_p prefer longer trajectory in order to prevent over-reactive maneuvers.

$$J_p = (S_{\max} - S(\tau_i))/S_{\max} \tag{12}$$

where $S(\tau_i)$ is the path length of the trajectory candidate τ_i along the base frame and $S_{\max}$ is the path length threshold. To avoid aggressive lateral movements, smoothness criterion is considered by the cost term J_s, which is the integration of curvature along the trajectory candidate.

$$J_s = \frac{1}{s_f}\int_0^{s_f} \frac{\|\kappa(\tau_i(s))\|}{\kappa_{\max}} ds \tag{13}$$

To ensure the vehicle track the base frame accurately, the cost term J_d penalizes path which deviates far from the base frame.

$$J_d = \frac{1}{s_f}\int_0^{s_f} \frac{|D(\tau_i(s))|}{D_{\max}} ds \tag{14}$$

where $D(\tau_i(s))$ is the deviation distance and D_{max} is the deviation distance threshold.

The cost term J_c penalizes the path inconsistency during consecutive replans. The inconsistency between consecutive replans can easily result in abrupt steering actions, control overshoots or even control instability. To solve the problem, we minimize the sampling terminal lateral offset variations between consecutive replans.

$$J_c = \frac{|l(\tau_i) - l_p(\tau_{i^*})|}{l_{\max}} \tag{15}$$

where $l(\tau_i)$ is the sampling terminal lateral offset of the current path, $l_p(\tau_{i^*})$ is the sampling terminal lateral offset of the best path in the previous plan.

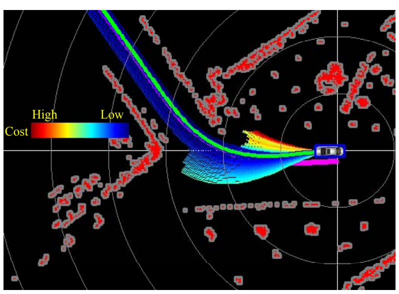


Fig. 6: Trajectory candidates evaluation.

As shown in Fig. 6, the color illustrates the cost of the generated trajectory candidates, the lowest cost trajectory candidate with green color is selected for execution. Currently, the weights are empirically determined to make a compromise between the cost terms mentioned above, the machine learning technique could be introduced to adaptively tune these parameters.

V. Experimental Results

Our test autonomous vehicle is shown in Fig.7. The low-level steer and speed control systems have been modified to be drive-by-wire. Field experiments have been carried out on this platform to verify the efficiency and applicability of the proposed trajectory planning algorithms in the realistic urban environments.

The proposed local trajectory planning algorithms are implemented in C++, which runs on a PC (Intel(R) Core(TM)

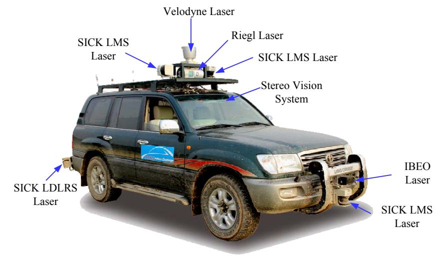
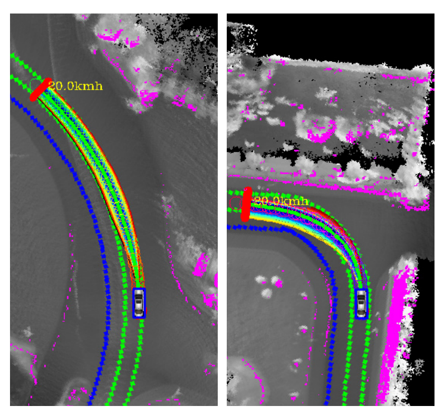


Fig. 7: Our test autonomous vehicle.

i7@2.70GHz, 4.0GB memory) with Linux system. The replan cycle is set to 100ms. The trajectory planning consumes around 20ms, while the time spent on path smoothing is around 50ms. In practice, this can meet the real-time requirements.

The maximal speed is set to 25km/h limited by the traffic regulation of the urban area, where we carry out the experiments. The terminal speed of the trajectory is set to 20km/h, when it does not collide with obstacles. The maximal path length is limited to 80m limited by the perception information. The maximal lateral acceleration limit is set to 3.0 m/s^2 to ensure the driving comfort. The total number of the generated trajectory candidates is 500, which could be enhanced to cope with the complex environments. For instance, to improve the probability of the existence of the collision-free path candidates, sampling terminal states, such as lateral offset and preview distance, can be sampled more densely.

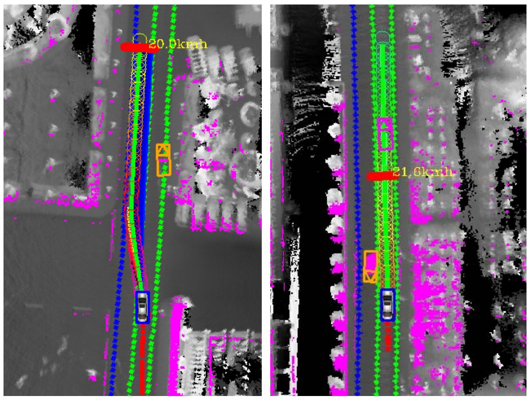
Figure 8 shows the trajectory planning results when the AGV negotiates a smooth turn and a curvy turn, respectively. It can be seen that vehicle is capable of generating the trajectory candidates which are aligned with the road geometry while avoid stationary obstacles.



(a) (b)

Fig. 8: Trajectory Planning results for Lane keeping. (a) a smooth turn, (b) a curvy turn

Figure. 9 depicts the trajectory planning results when the AGV interacts with other traffic participants. As Fig. 9 (a) shows, the vehicle executes a lane-changing maneuver when the current lane is occupied by a slowly moving vehicle in front. It can be seen In Fig. 9(b) that although the current lane is taken up by a vehicle in front, which has a relative higher speed than the AGV. In this case, the AGV executes a vehicle following maneuvers. Besides, the trajectory planner is also capable of planning a trajectory which keeps a certain safe distance from the oncoming vehicle in the adjacent lane.



(a) (b)

Fig. 9: Interact with the other traffic participants. (a) lane changing, (b) vehicle following

VI. Conclusions

This paper has proposed a practical and efficient trajectory planning framework for autonomous driving in urban environments. The presented trajectory planning strategy employs a hierarchical strategy. A reference path is extracted from the digital map relying on the accurate and robust localization information using LIDAR-based localization technology. The reference path is refined and interpolated via the nonlinear optimization algorithm and parametric algorithm respectively. Dividing the trajectory planning into spatial path generation and velocity profile generation has significantly reduced the solution space and enabled the planner to generate a rich set of drivable trajectory candidates, which are aligned with road geometry and capable of handling a variety of dynamic traffic situations, such as lane keeping, lane changing, vehicle following, static and moving objects avoidance. Although the presented planning framework focuses on solving the motion planning in on-road environments, it can easily be extended to integrate with the graph-search algorithm in off-road environments. Further research will consider uncertainty issues resulting from localization, perception and control systems to interact with the other traffic participants more naturally in urban environments.

References

[1] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, and G. Hoffmann, "Stanley: The robot that won the darpa grand challenge," *Journal of field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[2] U. Ozguner, C. Stiller, and K. Redmill, "Systems for safety and autonomous behavior in cars: The darpa grand challenge experience," *Proceedings of the IEEE*, vol. 95, no. 2, pp. 397–412, 2007.

[3] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, and C. Geyer, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[4] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke *et al.*, "Junior: The stanford entry in the urban challenge," *Journal of Field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.

[5] M. Werling, S. Kammel, J. Ziegler, and L. Groll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.

[6] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for berthała local, continuous method," in *Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, Conference Proceedings, pp. 450–457.

[7] M. Likhachev and D. Ferguson, "Planning long dynamically feasible maneuvers for autonomous vehicles," *The International Journal of Robotics Research*, vol. 28, no. 8, pp. 933–945, 2009.

[8] D. Dolgov, S. Thrun, M. Montemerlo, and J. Diebel, "Path planning for autonomous vehicles in unknown semi-structured environments," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 485–501, 2010.

[9] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. How, and G. Fiore, "Real-time motion planning with applications to autonomous urban driving," *Control Systems Technology, IEEE Transactions on*, vol. 17, no. 5, pp. 1105–1118, 2009.

[10] T. M. Howard and A. Kelly, "Optimal rough terrain trajectory generation for wheeled mobile robots," *The International Journal of Robotics Research*, vol. 26, no. 2, pp. 141–166, 2007.

[11] X. Li, Z. Sun, A. Kurt, and Q. Zhu, "A sampling-based local trajectory planner for autonomous driving along a reference path," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, Conference Proceedings, pp. 376–381.

[12] X. Li, Z. Sun, D. Liu, Q. Zhu, and Z. Huang, "Combining local trajectory planning and tracking control for autonomous ground vehicles navigating along a reference path," in *Intelligent Transportation Systems*. IEEE, 2014, Conference Proceedings, pp. 725–731.

[13] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, and G. Hoffmann, "Stanley: The robot that won the darpa grand challenge," *Journal of field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[14] U. Schwesinger, M. Rufli, P. Furgale, and R. Siegwart, "a sampling-based partial motion planning framework for system-compliant navigation along a reference path," in *Intelligent Vehicles Symposium Proceedings*. IEEE, 2013, Conference Paper, pp. 391–396.

[15] M. McNaughton, C. Urmson, J. M. Dolan, and J.-W. Lee, "Motion planning for autonomous driving with a conformal spatiotemporal lattice," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, Conference Proceedings, pp. 4889–4895.

[16] T. Gu, J. Snider, J. M. Dolan, and J.-w. Lee, "Focused trajectory planning for autonomous on-road driving," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, Conference Proceedings, pp. 547–552.

[17] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, and V. Pratt, "Towards fully autonomous driving: Systems and algorithms," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, Conference Proceedings, pp. 163–168.

[18] J. Ziegler, H. Lategahn, M. Schreiber, C. G. Keller, C. Knoppel, J. Hipp, M. Haueis, and C. Stiller, "Video based localization for bertha," in *Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, Conference Proceedings, pp. 1231–1238.

[19] M. J. Van Nieuwstadt and R. M. Murray, "Real time trajectory generation for differentially flat systems," *International Journal of Robust and Nonlinear Control*, vol. 8, pp. 995–1020, 1996.

[20] K. Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," *IEEE Transaction on Intelligent Transportation System*, vol. 13, no. 4, pp. 1599–1616, 2012.

[21] T. D. Barfoot and C. M. Clark, "Motion planning for formations of mobile robots," *Robotics and Autonomous Systems*, vol. 46, no. 2, pp. 65–78, 2004.

[22] F. Damerow and J. Eggert, "Predictive risk maps," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 703–710.