

# Fast Real-Time Trajectory Planning Method with 3rd-Order Curve Optimization for Automated Vehicles

Ray Lattarulo<sup>1</sup>, and Joshue Perez<sup>1</sup>

**Abstract**—Automated driving (AD) is one of the fastest-growing tendencies in the Intelligent Transportation Systems (ITS) field with some interesting demonstrations and prototypes. Currently, the main research topics are aligned with vehicle communications, environment recognition, control, and decision-making. A real-time trajectory planning method for Automated vehicles (AVs) is presented in this paper; the contribution is part of AD's decision-making module. This novel approach uses the properties of the 3er order Bézier curves to generate fast and reliable vehicle trajectories. Online execution and vehicle tracking capacities are considered on the approach. A feasible trajectory is selected based on the criteria: (i) the vehicle must be contained by a collision-free corridor given by an upper decision layer, (ii) the vehicle must be capable to track the generated trajectory, and (iii) the continuity of the path and curvature must be preserved in the joints. Our approach was tested considering a vehicle length (automated bus) of 12 meters. The scenario has the dimension of a real test location with multiple roundabouts.

## I. INTRODUCTION

AVs are showing great potential to improve safety, passengers' comfort, and efficiency under cooperative scenarios. Applications such as pedestrian detection, lane departure warning and assistance, and driving through a lane are some examples of current impressive results. Nevertheless, real-time trajectory planning is still an issue in this field where optimal solutions are desired based on vehicle dynamics, road characteristics, and other participants involved.

This task is part of the vehicle decision module, as well as obstacle avoidance, optimal route calculation, speed profiles, among others. *Global*, *behavioral*, and *local planning* are common names used, in the AD literature, to group the tasks involved in this module [1].

The local planner includes the path planning generation and speed profile, which are sent to the vehicle controllers. Smoother and safer trajectories improve the general performance of the AVs. Sampling-based methods [2] and numerical optimization [3] can be summarized as the possible groups of methods to deal with this task.

Sampling-based methods demand an online or offline discrete set of precomputed motion primitives based on road geometry, vehicle dynamics, or a combination of both. The optimization-based methods generate a real-time trajectory under a continuous set. The main advantage of the optimization methods, for trajectory generation, is the capacity of considering the complete solution space with a

small compromise in the computation time [4]. In recent years, some researchers have been focused on the optimal trajectory generation, considering different constraints (i.e.: speed, acceleration, and jerk), as in [5], [6], [7]. However, real-time planning based on local information remains as a clear open challenge in urban and dynamic environments.

Consequently, this work has proposed a novel optimization-based solution for AVs' trajectory planning. It has used the Bézier curves' convex-hull property to generate a geometrically safe trajectory that fits into a *collision-free corridor* while including vehicle kinematic constraints and passengers' comfort. The *Bound Optimization BY Quadratic Approximation (BOBYQA)* method is used to optimize the curves. The total area of the vehicle is considered during the evaluation of the trajectory feasibility.

The rest of the paper is organized as follows. Section II presents a brief description of the properties of the Bézier curve, the optimization method, and the vehicle model. Section III explains the trajectory planning approach based on highly accurate map information. The proposed scenario, located in the Malaga's port, and the results obtained in this work are presented in sections IV and V, respectively. Finally, the last section presents some conclusions and future works.

## II. CONCEPTS AND BASIS

### A. Bézier curves

Bézier curves are parametric curves that are part of the family of the interpolated curves, as the splines [8]. In the robotics field, they have been used for trajectory generation of non-holonomic vehicles, and interesting results have been obtained [9], [10]. Some examples of these curves are presented in Fig. 1.

The Bézier curves are defined with the following equation:

$$\mathbf{B}(t|P_0, \dots, P_n) = \sum_{i=0}^n b_i(t) \mathbf{P}_i, \quad b_i = \binom{n}{i} t^i (1-t)^{n-i} \quad (1)$$

$\{b_i \in \mathbb{R}\}$  is the Bernstein polynomial,  $\{\mathbf{P}_i \in \mathbb{R}^2\}$  are the control points which define the curve in the plane  $x - y$ ,  $\{n \in \mathbb{N}^+\}$  is the Bézier order, and  $\{t \in \mathbb{R}, t = [0, 1]\}$  is the parameter used to interpolate the curve.

They have interesting properties which were summarized in previous work [11]. This approach considers the following ones:

- The starting point of the Bézier curve corresponds with the control point  $\mathbf{P}_0$ , and the ending point corresponds with  $\mathbf{P}_n$ .

<sup>1</sup>Ray Lattarulo, and Joshue Perez are with TECNALIA, Basque Research and Technology Alliance (BRTA), Astondo bidea 700, E-48160, Derio, Biscay, Spain. {rayalejandro.lattarulo and joshue.perez}@tecnalia.com

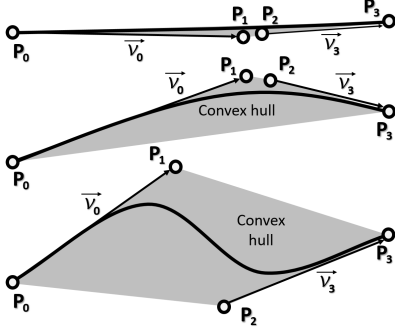


Fig. 1. 3rd-Order Bézier curves, its properties, and shape

- The direction at  $\mathbf{P}_0$  will be defined by the vector  $\overrightarrow{P_0P_1}$ , and the direction at  $\mathbf{P}_n$  will be defined by the vector  $\overrightarrow{P_{n-1}P_n}$ .
- The curve will lie in the convex hull formed by the control points.
- Bézier curves generates continuous curvature and path.

In this work, the Bézier curves define the trajectories to be tracked by the vehicle's controllers. Third-order curves are used to compute fast solutions while preserving the continuity in geometry and curvature. The four control points can generate trajectories that define the typical vehicle maneuvers, such as straight segment, soft and narrow curves, and S-shape movements (lane-change maneuvers). These examples are depicted in Fig. 1.

Moreover, the real-time trajectories use the property of the direction in the point  $\mathbf{P}_0$  to define the location of  $\mathbf{P}_1$ , preserving the continuity in geometry with the heading vector  $\vec{v}_0$ . Further details will be given in the following sections.

### B. Optimization method

The trajectory generation problem of the AVs, considering the total vehicle area contained into the lane boundaries, is a highly nonlinear problem with non-smooth, discontinuous, and non-convex objective functions. Derivative-based optimization methods are not capable to solve this type of problem with reliable results [12]. Although, modern derivative-free methods achieve better results.

Currently, the developments in derivative-free methods have suffered a significant improvement considering convergence proofs and theoretical background. This has derived in different packages for solving this type of complex problems, such as: *NOMAD*, *OQNLP*, *NEWUOA*, *BOBYQA*, among others [13]. Some of these methods have the following disadvantages: (i) the *NOMAD* method generates global optimal solution which demand a long computation time <sup>1</sup>, (ii) *OQNLP* demands smooth constraints and it is used to find a global optimal <sup>2</sup>, and (iii) the *NEWUOA* was developed to solve unconstrained optimization problems (long computation time) [14].

<sup>1</sup><https://www.inverseproblem.co.nz/OPTI/index.php/Solvers/NOMAD>

<sup>2</sup><https://tomopt.com/tomlab/products/oqnlp/>

In these terms, the *BOBYQA* method has been used to solve online optimization problems finding a quick and reliable local optimal value. It was originally a Fortran package used to obtain the minimum value of a function  $\{F(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n\}$  subject to the bound constraints  $\{a_i \leq x_i \leq b_i : i = 1, 2, \dots, n\}$  [15]. The method does not require precomputed derivatives and that is a major benefit in this type of highly nonlinear problem.

The procedure for solving the optimization problem is the following:

- 1) The system verifies the boundaries to avoid crossings or overlaps between the lower bound elements  $a_i$  and the upper bound elements  $b_i$ . Moreover, the initial condition  $\mathbf{x}^{(0)}$  must be into the bounds.
- 2) The *BOBYQA* algorithm generates a quadratic approximation of the objective function  $F(\mathbf{x})$  in the form of  $\{Q(x_k) = F(x_k) : k = 1, 2, \dots, m\}$ , with  $m$  the number of discretization steps.
- 3) The *truncated conjugate gradient method* is applied to solve the problem, which is a variant of the *conjugate gradient method* but with a limitation in the number of the iterations executed.

The *conjugate gradient method* obtains the optimal value  $\mathbf{x}^*$  in fewer iterations than traditional methods such as the *steepest descent*. Every iteration generates a possible value of  $\mathbf{x}^*$  based on a gradient that is A-orthogonal respect to the previous gradient [16].

In this work, the solution vector is in  $\mathbb{R}^3$ . The first element is the distance between the control point  $P_0$  and  $P_1$ , the distance between  $P_2$  and  $P_3$  is the second value, and the last one is the magnitude of the vector that goes from the right to the left bound and it is perpendicular to the direction of the point  $P_3$ .

The *DLib* toolkit was used to compute the *BOBYQA* method. It is distributed under a boost open source license and it has C++ and python interfaces. The optimization module was used for this approach. It has been integrated into the control architecture for AD, developed in the AD-team of Tecnia Research and Innovation [11].

### C. Vehicle model

The kinematic model was considered to ensure the correctness while tracking the generated trajectories. The reference point was set in the rear axle to avoid the complexity of adding the lateral slip component while the vehicle is rotating around the center of gravity. Moreover, the longitudinal slip is considered negligible due to the low speed and the flat surface of the scenario.

Fig. 2 presents the kinematic vehicle model. A relation between the turning angle  $\delta$  and the curvature  $\frac{1}{r}$  is extracted considering *similar triangles property* and the wheelbase  $l$ . This relation is given by  $\tan(\delta) = \frac{l}{r}$ . In these terms, the maximum value of the turning angle  $\delta_{max}$  is introduced, in the previous relation, to obtain the maximum curvature of the trajectory that the vehicle can track effectively.

$$k_{max} = \frac{1}{r_{min}} = \frac{\tan(\delta_{max})}{l}$$

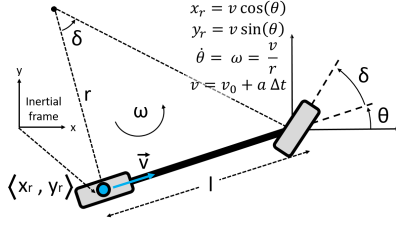


Fig. 2. Kinematic vehicle model

### III. TRAJECTORY PLANNING APPROACH

This work proposes a trajectory planning approach based on highly accurate map information, also known as *lane-level accuracy* [17]. A 3-levels automated vehicle's decision-module was considered. The first stage is a global planner that generates a route based on sensors and map information. The second part is the *behavioral planner* that defines a *collision-free corridor*. Lastly, the third level is the *local planner* that generates trajectories to drive through this *collision-free lane*.

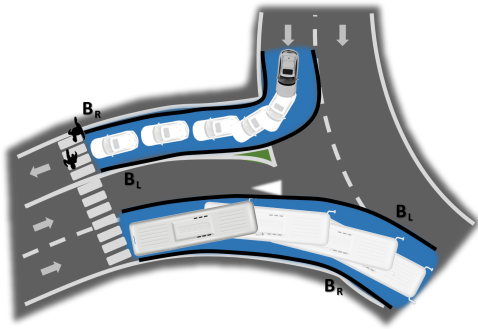


Fig. 3. Road and lane definition

Fig. 3 depicts the input of the trajectory planning approach which comes from the *behavioral module*. Those are the collision-free lanes (blue), which are defined by the right bound  $B_R$  and the left bound  $B_L$ . These lanes describe a collision-free area in terms of obstacles, other vehicles, pedestrians, vehicle space, and maneuvering capacity into the corridor. In this example, a first corridor is generated for the automated car defining a collision-free area that finishes right before the pedestrian crossing. A second corridor was generated considering two lanes to defines the drivable space of a long vehicle, such as an electric bus. This paper has tested the automated trajectory planning approach in a bus of 12 meters of length.

The trajectory and speed profile will be generated based on the corridor. This avoids possible unsafe conditions due to the violation of the limits given by the bounds and its maximum distance. These constraints and their use in the method are explained as follows:

#### A. Path planning

The lateral part of the trajectory has been developed based on a numerical optimization method, using the *BOBYQA*

optimization method and the Bézier curves, due to their computation time and the continuity in geometry and curvature. The method receives, as input, the lateral bounds  $B_L$  and  $B_R$  with a total length of  $D$  as Fig. 4 depicts.

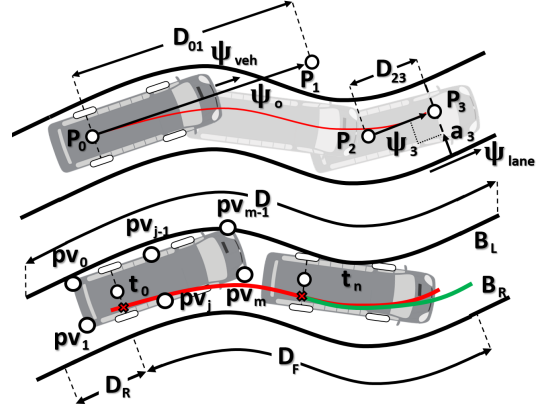


Fig. 4. Bézier control points positioning

The bounds are filtered to reduce the density of points, remaining the significant ones. The reduction considers the separation between consecutive points and the difference in angle. Moreover, the total length of the lane is truncated to a maximum frontal distance of  $D_F$ , boosting the responsiveness of the method. A segment of the lane, which measures  $D_R$ , is left behind the reference point of the vehicle and it measures two times the distance from the vehicle rear axle to the rear vehicle limit.  $D_R$  must be considered because the rear part of vehicle dimension has a considerable contribution in the total vehicle area (e.g. automated bus) and it must be evaluated to generate trajectories which fits into the collision-free corridor.

The first point of the trajectory  $P_0$  is set in the middle of the vehicle rear axle. This position is selected to reduce the slip angle contribution of the generated trajectory and the vehicle motion. The direction of the vehicle movement and the direction of the trajectory are equal at this point. Moreover, the direction of the trajectory  $\psi_0$  at  $P_0$  is equal to the current vehicle direction  $\psi_{veh}$ . The final control point of the trajectory  $P_3$  is set at a distance  $D_F$  from the first point and over the perpendicular lane axis  $a_3$ .

The optimization process generates variations of: (i) the distance  $D_{01}$  between the control point  $P_0$  and  $P_1$ , (ii) the distance  $D_{23}$  between the control point  $P_2$  and  $P_3$ , and (iii) the distance between the bound  $B_R$  and  $B_L$  of the control point  $P_3$  over the axis  $a_3$ . The location of the control point  $P_2$  is given by the position of  $P_3$ , the distance  $D_{23}$ , and the direction  $\psi_3$  which is equal to the direction of the lane  $\psi_{lane}$  at the point  $P_3$ .

The optimization process modifies the parameters  $D_{01}$ ,  $D_{23}$ , and  $a_3$  to generate an optimal trajectory. The lane generated considers the vehicle dimensions while the curvature is reduced under the maximum vehicle turning limits, which improves the comfort mitigating rough changes in the vehicle direction. After the generation of the control points  $P_n$ , they are interpolated using Bézier formulation which creates a

polyline of points  $p_i$  with a curvature associated  $k_i$ . They are evaluated in a cost function which has the following form:

$$\phi = \begin{cases} \sum_{i=1}^n \phi^-(\mathbf{p}_i, k_i), & \text{when feasible} \\ |\Delta k_0| + \sum_{i=1}^n \phi^+(\mathbf{p}_i, k_i), & \text{when unfeasible} \end{cases}$$

$$\phi^-(\mathbf{p}_i, k_i) = -\left(\sum_{j=1}^m \min\{dR_{pv_j}, dL_{pv_j}\}\right) - 1/|k_i|$$

$$\phi^+(\mathbf{p}_i, k_i) = \left(\sum_{j=1}^m \max\{dR_{pv_j}, dL_{pv_j}\}\right) + |k_{max} - k_i|$$
(2)

The trajectory is declared unfeasible when for one of the interpolated points  $p_i$ , at least one of the points in the set  $pv$  (description of the vehicle outer bound presented in Fig. 4) is out of the lane. Moreover, a curvature  $k_i$ , at the point  $p_i$ , over the maximum vehicle limit  $k_{max}$  means an unfeasible solution. Lastly, the deviation between the curvature at the starting point and the same location in the previous feasible trajectory  $\Delta k_0$  must be under a maximum limit. The distance of the points  $pv$  to the right and left bounds are  $dR$  and  $dL$  respectively. The best trajectory is obtained evaluating the optimization problem  $\min\{\phi\}$ , for a positive value of the cost function the trajectory is unfeasible and vice-versa.

In the case of an unfeasible solution, the optimization seed (the result of the last iteration) is randomly modified as well as the maximum lane distance. Both actions help to converge the optimization process.

### B. Speed profile

In some previous works, good results have been obtained on speed profiles generation using a *Linear Model Predictive Control* approach [18]. This model is a triple integrator chain that generates a fast and reliable response under vehicle dynamics constraints, as well as comfort and safety ones (road speed limit); for further details consult [19], [20]. Results related to the speed profile are not presented in the current publication because it does not add additional information respect to the previous tests.

## IV. PROPOSED SCENARIO

The approach has been tested in simulation environments considering an automated bus, the length was 12.0 meters and the width was 3.0 meters (including a safety lateral distance). The tests were executed in an industrial PC with a Intel Core i7-6820HQ processor and 32 GB of RAM. The proposed scenario was tested on the last segment of the Malaga's Port (Spain) which is the scenario of the EU AutoDrive project <sup>3</sup>.

The complete path is presented in Fig. 5. The algorithm has been validated in a segment with a large number of bends. Moreover, the selected section has two roundabouts with 180° turns.

The use cases have considered only the real-time trajectory generation based on collision-free lane information. In these



Fig. 5. The proposed scenario

terms, this approach is scalable to the stop-and-go, lane change, overtaking, obstacle avoidance, and lane-keeping based on map information maneuvers. The computation of the collision-free lane is a task of an upper layer of the vehicle decision module (behavioral planning) which is beyond the scope of this work.

## V. RESULTS

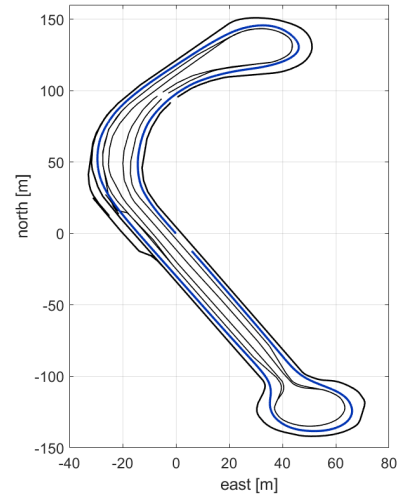


Fig. 6. The vehicle trajectory over the complete scenario

Fig. 6 presents the test scenario and the complete trajectory. It is obtained from the concatenation of all the generated trajectories in each timestep. The location is given in UTM coordinates, relative to the position east 373675.8174 and north 4063258.5712. These results can be used in real test cases as a pre-computed optimal trajectory which helps to drive through the scenario as a fallback strategy in case of a long delay or a failure in the decision module.

Fig. 7 shows a complete movement on the roundabout considering the vehicle area. In the entrance of the roundabout, the local planner generates a trajectory which minimized the vehicle changes passing between the right bound and the left bound of the inner part of the roundabout. A similar operation is presented at the exit of the roundabout. The vehicle is contained by the driveable area (lanes) due to the verification of this condition in the optimization process.

A sequence of generated trajectories is presented in Fig. 8. The blue line is the last feasible trajectory and the red one is the current evaluation. The dot-mark shows the location of the Bézier control points. The upper part of Fig. 8 represents a solution obtained from the optimization

<sup>3</sup><https://autodrive-project.eu/>

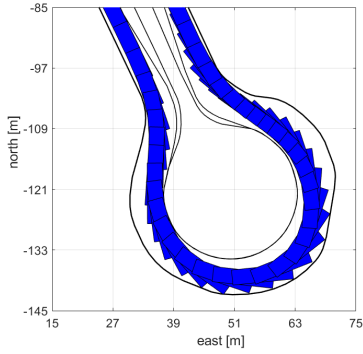


Fig. 7. Roundabout scenario at the Malaga's Port

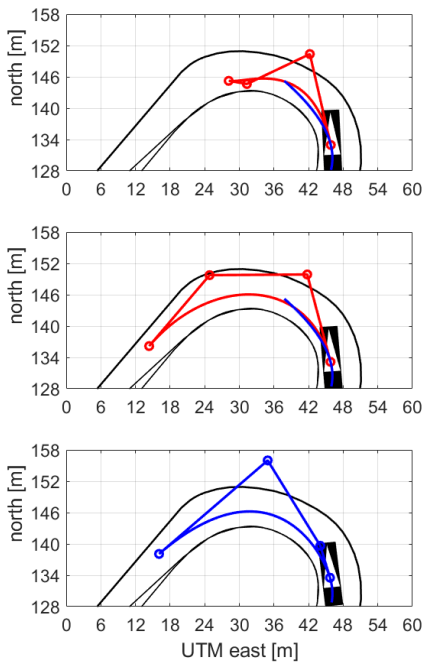


Fig. 8. Evolution of the trajectories

process, nevertheless, its curvature overpassed the maximum permissible value. In this case, the maximum lane distance and the optimization seed were randomly modified, and it generates the solution of the second picture. The middle part of the figure shows that the trajectory was unfeasible because the vehicle area was out of the lane. Lastly, another modification of the optimization seed was done to achieve a feasible solution.

Fig. 9 depicts four values of interest for each trajectory which are: (i) the trajectory distance, (ii) curvature of the first element, (iii) the maximum and the minimum value of the trajectory, and (iv) the angle of the initial element. The distance has a target value of 50 meters ahead. This is reduced if a feasible solution is not obtained. The minimum value, during the experiments, is around 25 meters. The second image presents a continuous response of the curvature in the joints of consecutive trajectories. In the third image, the gray area represents the maximum and minimum values of the curvature for each timestep which is into the limit of

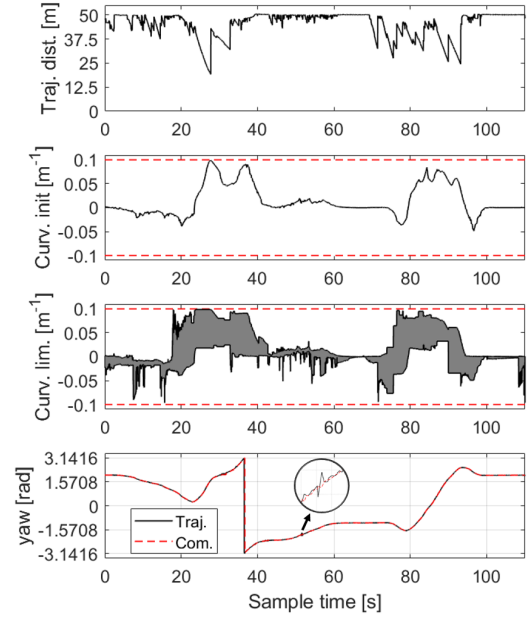


Fig. 9. Trajectory variables

$0.1 \frac{1}{m}$ . This value is related to the turning capacities of the vehicle. Lastly, the angle in the joints is similar to the angle rebuilt from the trajectory points (based on geometry), with some small variations due to numerical errors.

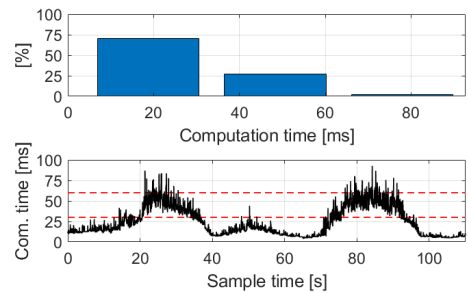


Fig. 10. Execution time

Finally, another value of interest was the time consume during the optimization process which is presented in Fig. 10. The histogram (upper part) concludes that around 72% of the samples during the experiments consumed less than 30 milliseconds, around 27% of the time was between this limit and 60 milliseconds and the rest of the samples were under the limit of 80 milliseconds. The lower part of the figure relates the long computation time with the most difficult parts of the experiment which were the  $180^\circ$  turns into the roundabout. The straight segments have the lowest computation time.

The efficiency of the algorithm was compared with other methods of the literature. The first method used a combination of optimal control (OCP) formulation using a kinematic vehicle model along with a spline parametrization [21] (the experiments reported the results with 8 knots for the interpolation). The second method was a convex optimization for-

TABLE I  
COMPARISON WITH METHODS OF THE STATE OF THE ART

Method	Exec. Time [ms]	Obstacles
OCP [21]	116 (avg.) - 229 (max.)	Static
Convex Opt. [22]	441	Static
RRT-GP [23]	647 (avg.) - 665 (max.)	Static

This work < 30 (72% samples) - 80 (max.)\* Dynamic\*\*  
 \* It doesn't consider the computation time of the collision-free corridor  
 \*\* This will be presented in future works

mulation which considered a previously computed collision-free tube [22]. The third method was a RRT approach considering a Gaussian process (GP) motion model [23]. The results are presented in table I. This table contains the execution time for the generation of one trajectory and the type of obstacles considered during the experiment.

## VI. CONCLUSION

This work has presented a novel real-time trajectory planning method based on Bézier curves. The best trajectory was generated with a nonlinear local optimization method named *BOBYQA*. It has used the principle of the Bézier curves' convex hull to generate a safe trajectory that contains the vehicle into the lane considering vehicle's width and length which is an improvement respect the typical evaluation of vehicle width made in the state of the art. The maximum trajectory distance is used to compute the speed profile with the distance restriction which keeps a vehicle safe state.

The method has presented a total computation time lower than 50 milliseconds more than 90% of the time, on different types of bend segments. The algorithm has been tested in simulation environments using a dynamic model of the bus, and modeling a real scenario context.

As future works, this algorithm will be tested with long non-holonomic vehicles as an automated bus, and maneuvers such as overtaking, obstacle avoidance, and automated parking will be evaluated. Experiments using real test vehicles are also considered. Moreover, the algorithm will be tested with dynamic obstacles, and more advanced state of the art algorithms will be collected and tested to extract the values of interest for benchmarking this approach.

## ACKNOWLEDGMENT

This work was supported by the European AutoDrive project from the ECSEL program under the grant agreement No 737469, and the European SHOW Project from the Horizon 2020 program under the grant agreement No 875530.

## REFERENCES

- [1] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2015.
- [2] Z. Kingston, M. Moll, and L. E. Kavraki, "Sampling-based methods for motion planning with constraints," *Annual Review of Control, Robotics, and Autonomous Systems*, 2018.
- [3] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for bertha - a local, continuous method," *IEEE Intelligent Vehicles Symposium Proceedings*, 2014.
- [4] W. Lim, S. Lee, M. Sunwoo, and K. Jo, "Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method," *IEEE Transactions on Intelligent Transportation System*, 2018.
- [5] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 187–210, 2018.
- [6] L. Yu, D. Kong, and X. Yan, "A driving behavior planning and trajectory generation method for autonomous electric bus," *Future Internet*, 2018.
- [7] Y. Zhang, H. Sun, J. Zhou, J. Hu, and J. Miao, "Optimal trajectory generation for autonomous vehicles under centripetal acceleration constraints for in-lane driving scenarios," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Oct 2019, pp. 3619–3626.
- [8] D. González, J. Pérez, R. Lattarulo, V. Milanés, and F. Nashashibi, "Continuous curvature planning with obstacle avoidance capabilities in urban scenarios," *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014.
- [9] L. Han, H. Yashiro, H. Tehrani Nik Nejad, Q. H. Do, and S. Mita, "Bézier curve based path planning for autonomous vehicle in urban environment," in *2010 IEEE Intelligent Vehicles Symposium*, June 2010, pp. 1036–1042.
- [10] X. Qian, I. Navarro, A. de La Fortelle, and F. Moutarde, "Motion planning for urban autonomous driving using bézier curves and mpc," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 826–833.
- [11] R. Lattarulo, L. González, E. Martí, J. Matute, M. Marcano, and J. Pérez, "Urban motion planning framework based on n-bézier curves considering comfort and safety," *Journal of Advanced Transportation*, 2018.
- [12] K. R. Fowler, J. P. Reese, C. E. Kees, J. Dennis Jr, C. T. Kelley, C. T. Miller *et al.*, "Comparison of derivative-free optimization methods for groundwater supply and hydraulic capture community problems," *Advances in Water Resources*, vol. 31, no. 5, pp. 743–757, 2008.
- [13] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: a review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, no. 3, pp. 1247–1293, 2013.
- [14] M. J. Powell, "The newuoa software for unconstrained optimization without derivatives," in *Large-scale nonlinear optimization*. Springer, 2006, pp. 255–297.
- [15] M. J. D. Powell, "The bobyqa algorithm for bound constrained optimization without derivatives," *Cambridge NA Report NA2009/06, University of Cambridge*, pp. 26 – 46, 2009.
- [16] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," *School of Computer Science Carnegie Mellon University*, 1994.
- [17] F. Poggenhans, J.-H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr, "Lanelet2: A high-definition map framework for the future of automated driving," *International Conference on Intelligent Transportation Systems (ITSC)*, 2018.
- [18] R. Lattarulo, D. He, and J. Perez, "A linear model predictive planning approach for overtaking manoeuvres under possible collision circumstances," *IEEE Intelligent Vehicles Symposium*, 2018.
- [19] D. Heß, R. Lattarulo, J. Pérez, J. Schindler, T. Hesse, and F. Köster, "Fast maneuver planning for cooperative automated vehicles," *IEEE International Conference on Intelligent Transportation Systems*, pp. 1625–1632, 2019.
- [20] D. Heß, R. Lattarulo, J. Pérez, T. Hesse, and F. Köster, "Negotiation of cooperative maneuvers for automated vehicles: Experimental results," *IEEE Intelligent Transportation Systems Conference*, pp. 1545–1551, 2019.
- [21] T. Mercy, R. Van Parys, and G. Pipeleers, "Spline-based motion planning for autonomous guided vehicles in a dynamic environment," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 6, pp. 2182–2189, 2018.
- [22] Zhijie Zhu, E. Schmerling, and M. Pavone, "A convex optimization approach to smooth trajectories for motion planning with car-like robots," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 835–842.
- [23] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How, "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns," *Springer Autonomous Robots*, vol. 35, pp. 51–76, 2013.