DOI: 10.1109/WEA.2015.7370124

Local and Global Path Generation for Autonomous Vehicles Using Splines

randerson.lemos@gmail.com, olmerg@gmail.com, janito@fem.unicamp.br

Randerson Araujo de Lemos, Olmer Garcia, Janito Vaqueiro Ferreira
Faculty of Mechanical Engineering, Autonomous Mobility Laboratoty - LMA
University of Campinas - UNICAMP, Rua Mendeleyev, 200, Campinas Brazil

***Abstract*—Autonomous vehicles soon will be a reality for many daily situations. Currently, the main barrier preventing the development of autonomous vehicle is its high cost. In agreement with the objectives of the Autonomous Mobility Laboratory from the Mechanical Engineering Faculty at Unicamp of developing inexpensive autonomous robots, the presented article exposes relevant aspects from the implementation process of a real time path planning technique for terrestrial vehicles. The studied technique can be divided into four stages, which are: global route construction, car localization, paths generation and path selection. The construction stage plays an important role in the robustness of the presented technique. The adequate number and distribution of waypoints are highly correlated to well global route construction and to a satisfactory arc length reparameterization. Unfortunately, the actual aspects of the waypoints provided by on-line services are inadequate to its directly usage in the construction of the global route curve, making necessary its prior processing. There is also a trade-off between the numbers of waypoints and the quality of the desired arc length parameterization that needs to be tuned.**

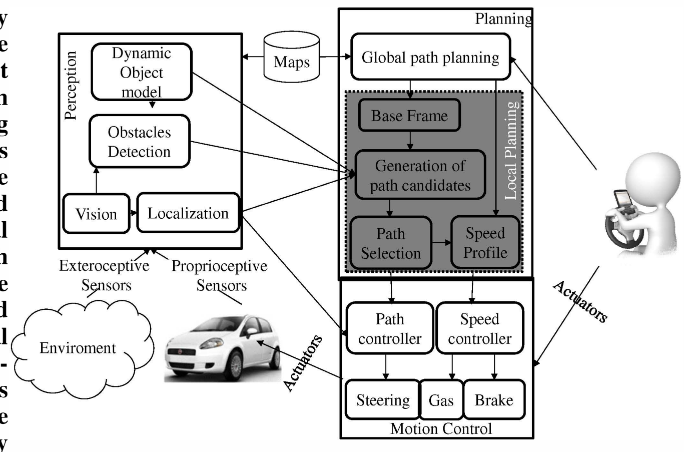

Figure 1. Architecture of the autonomous vehicle

I. Introduction

Recent technological advances have allowed the implementation of autonomous vehicles and advances driver assistance systems (ADAS) at both research and commercial level to provide increasing safety and comfort. Despite these advances, however, safety and liability problems are far to be completely solved.

The autonomous vehicle, which for instance can be assumed as a mobile robot, has its specific aspect of cognition directly linked to a robust mobility known as navigation [1]. In the literature the navigation has two components: planning and reacting. The planning component, also known as global path planing, generates the global route, which guides the vehicle from an initial position to a target. The reacting component, also known as local path planning, generates several local paths, which allows the vehicle to avoid obstacles. To optimize the distance and the speed, the global path planning is performed over previous known environment information (map) and user's choices. This approach for automotive applications has more recently been studied by [2], [3], [4], [5], [6].

The robotic vehicle design of the Autonomous Mobility Laboratory (LMA) from the Mechanical Engineering Faculty at Unicamp is developed on a FIAT-PUNTO platform. According to [7], autonomous vehicles systems can be organized into six major functional groups: interface sensors, perception, planning, control, vehicle interface and user interface. In this context, and taking into account the model of [1], which consists in perception, planning and motion control, the autonomous vehicle system blocks that are tackled on the article is shown in figure 1 and is in gray color.

The presented paper is divided into three section. The first section presents the local path planning algorithm for terrestrial vehicles adopted by LMA, the second section explains the techniques employed in the generation of the global routes from off-the-shelf software and the solutions adopted to solve problems with the global route arc length parameterization ([2]). Finally, conclusions are presented.

II. Local path algorithm

The main goal of the implemented algorithm is to generate smooth obstacle free paths, which start from an initial position and extend themselves in direction of the global route. In addition, the paths have high dynamic executability. Figure 2 shows an example of a global route and of candidates paths set with the features mentioned above.

The technique presented by [2] is didactically divided into four stages, which are: construction, localization, generation and selection.

A. Construction

The construction stage, which belongs to the global planning, is responsible to construct the global route curve. The global route is a curve that extends itself over the road center

978-1-5090-0228-3/15/$31.00 ©2015 IEEE

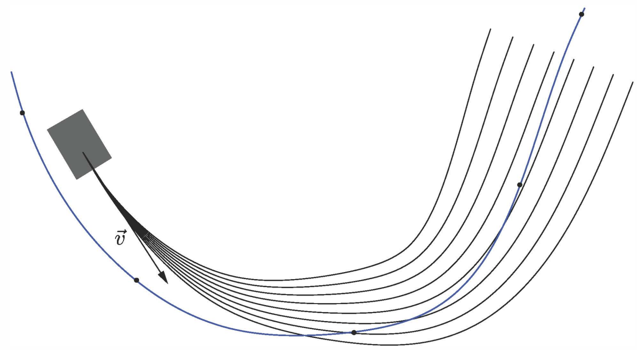

Figure 2. Global route and candidates paths

from an initial position until a target. The global route curve is parameterized in the arc length s and it is constructed by cubic splines from points localized in the center of the road, known as waypoints. The parameterized equation of the global route curve constructed by cubic splines in terms of the arc length s is shown in equation 1.

$$\begin{aligned} x_b(s) &= a_{x,i}(s-s_i)^3 + b_{x,i}(s-s_i)^2 + c_{x,i}(s-s_i) + d_{x,i} \\ y_b(s) &= a_{y,i}(s-s_i)^3 + b_{y,i}(s-s_i)^2 + c_{y,i}(s-s_i) + d_{y,i} \end{aligned} \tag{1}$$

From equation 1 it is possible to obtain the Cartesian positions (x_b, y_b) of the global route curve associated to any value of the arc length s.

1) Parametrization of the global route curve with respect to the cumulative distance between the waypoints: Parameterized curves in the arc length s can not be obtained directly from the knots (waypoints) by spline interpolation. In the literature there are different approaches that seek solutions for this problem ([8], [9], [10]).

In the present work, the strategy adopted to obtain an arc length parameterized global route curve by spline interpolation from the waypoints is divided into two steps: Firstly, it is constructed a global route curve with a generic parameter and, in sequence, it is performed its arc length s reparameterization. The generic parameter chosen to construct the global route curve is the cumulative distance between the waypoints d. Figure 3 shows the cumulative distance parameter d, in which $d_n = \sum_{i=1}^{n} d_i$.

The equations of the cumulative distance parameterized global route curve constructed by spline interpolation are shown in equation 2.

$$\begin{aligned} x_{b,i}(d) &= m_{x,i}(d-d_i)^3 + n_{x,i}(d-d_i)^2 + o_{x,i}(d-d_i) + p_{x,i} \\ y_{b,i}(d) &= m_{y,i}(d-d_i)^3 + n_{y,i}(d-d_i)^2 + o_{y,i}(d-d_i) + p_{y,i} \end{aligned} \tag{2}$$

The cumulative distance parameterized global route curve is illustrated by the blue line in Figure 3. This curve was constructed by spline interpolation using the Cartesian position (x, y) and cumulative distance d of each waypoint.

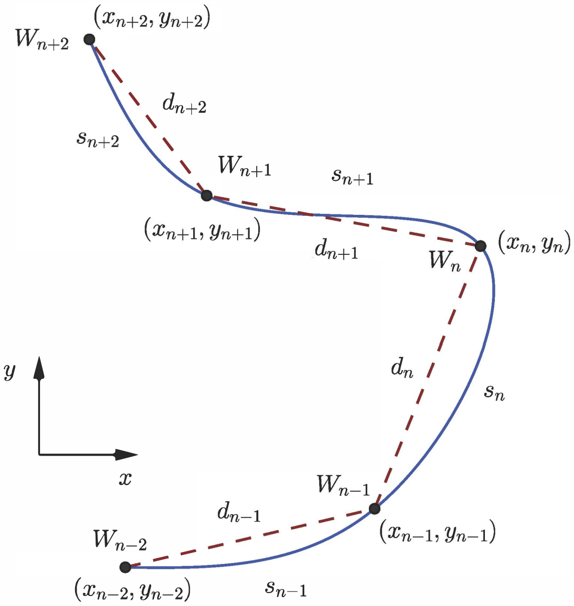

Figure 3. Global route in blue, cumulative distance parameters d in red and arc length parameter s

2) Global route curve reparameterization with recpect to the arc length distance: Constructed the global route curve parameterized in the cumulative distance between the waypoints d as illustrated by the blue line in Figure 3, it is performed its reparametrization according to [11], [9].

B. Localization

Dedicated to localize the vehicle with respect to the global route curve, the localization stage computes the distance traveled on the global route s_c and lateral offset q_c of the vehicle according the technique described by [12]. The parameters s_c and q_c are illustrated in Figure 4.

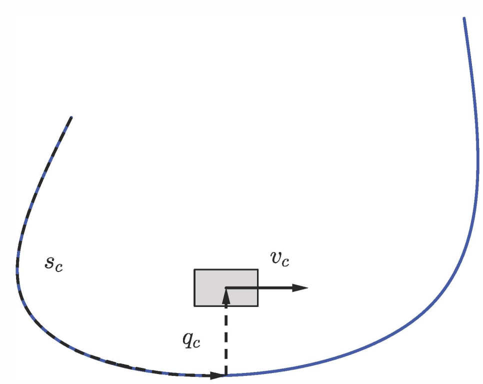

Figure 4. Vehicle localization

C. Generation

Constructed the global route curve parameterized in the arc length s and localized the vehicle, the candidates paths generation is performed. The candidates paths generations is executed in real-time and it is known as local planning. The candidates paths are initially generated in a curvilinear coordinate system defined by the parameters s_c and q_c and subsequently mapped to the Cartesian coordinate system of the global route. The curvilinear coordinate system $s0q$ and a set of candidates paths represented in the Cartesian coordinate system are shown in Figure 5.

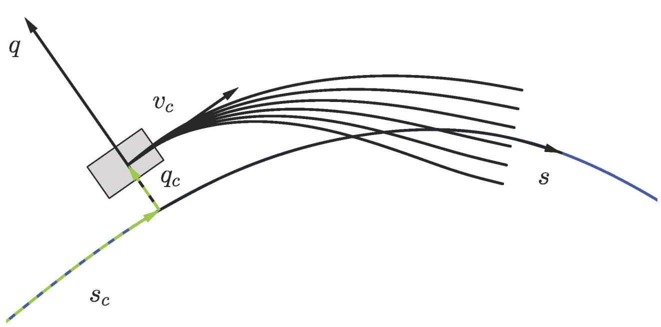

Figure 5. Curvilinear coordinate system, global route and candidates paths represented in the Cartesian coordinate system

1) Path candidates generation in the curvilinear coordinate system: The set of equation used to generate the candidates paths in the curvilinear coordinate system are shown in equation 3.

$$\begin{aligned} q(s) &= \begin{cases} a\Delta s^3 + b\Delta s^2 + c\Delta s + d & s_i \leq s < s_f \\ q_f & s_f \leq s \end{cases} \\ \frac{dq}{ds}(s) &= \begin{cases} 3a\Delta s^2 + 2b\Delta s + c & s_i \leq s < s_f \\ 0 & s_f \leq s \end{cases} \\ \frac{d^2q}{ds^2}(s) &= \begin{cases} 6a\Delta s + 2b & s_i \leq s < s_f \\ 0 & s_f \leq s \end{cases} \end{aligned} \tag{3}$$

where

$$\Delta s = s - s_i$$

The equations above are designed to provide smooth candidates paths as a function of the arc length s. The boundary conditions necessary to determine the coefficients a, b, c and d are shown in equation 4.

$$\begin{aligned} q(s_i = s_c) &= q_c \quad \frac{dq}{ds}(s_i = s_c) = tan\,\theta_c \\ q(s_f) &= q_f \quad \frac{dq}{ds}(s_f) = tan\,\theta_f = tan(0) = 0 \end{aligned} \tag{4}$$

The angle θ_c is defined as the difference between the vehicle orientation and the global route curve orientation at the position specified by the arc length $s_i = s_c$, ([13], [14]).

2) Mapping the candidates paths from the curvilinear coordinate system to the Cartesian coordinate system: The mapping approach is divided into two stages. Firstly, it is determined the curvature k of each candidate path in the Cartesian coordinate system. Subsequently, with the information of the curvature k and the equations of the vehicle model, the position of the candidates paths in the Cartesian system of the global route is computed. The curvature k of the candidates paths in the Cartesian system of the global route curve is determined by equation 5 ([15], [16]).

$$k = \frac{S}{Q}\left(k_b + \frac{(1-qk_b)\frac{d^2q}{ds^2} + k_b\frac{dq}{ds}^2}{Q^2}\right) \tag{5}$$

with

$$S = sgn(1-qk_b) \qquad Q = \sqrt{\left(\frac{dq}{ds}\right)^2 + (1-qk_b)^2}$$

With the candidates paths curvature k and using the bicycle vehicle model ([15], [17]) the differential equations 6 that determine the positions in the Cartesian system of the candidates paths are reached

$$\frac{dx}{ds} = Qcos\theta \qquad \frac{dy}{ds} = Qsin\theta \qquad \frac{d\theta}{ds} = Qk \tag{6}$$

D. Selection

From the set of candidates paths one is selected to be followed by the autonomous vehicle. The selection is made by the search of the candidate path that minimizes a linear combination of cost functions that represents the intrinsic and extrinsic safety of the vehicle. The cost functions are designed to quantify specific path candidates features that are judged important to a safe autonomous navigation, ([2]).

III. Global Route

For the well performance of the technique presented by [2] it is essential to obtain a global route curve that are both representative of the real route and properly parameterized in the arc length s. The construction of a representative global route curve from cubic splines is strongly correlated with the number and the distribution of the waypoints used. Likewise, an appropriate parameterization of the global route curve is also strongly related with the number and distribution of the points used in the reparameterization process, that does not need to be the same numbers of the waypoints used in the construction stage.

A. Global route construction

In a market scenario of autonomous vehicles commercialization to the general public, it is coherent to assume that the waypoints utilized in the global route construction will be obtained from off-the-shelf platforms that will integrate maps and route planning systems, as Google Maps ([18]) and Open Street Maps ([19]) do. Unfortunately, the current characteristics of the waypoints provided by those platforms do not allow its direct usage in the global route construction, making necessary a processing stage. The processing stage seeks to eliminate too close waypoints that will contribute to the generation of splines segments with high curvature and to add new waypoints in regions that lack of them, because in those regions the cubics splines will unwanted distance themselves from the real route. The processing stage analyses the euclidean distance between the waypoints, removing or adding waypoints according to the design criterion of maximum and minimum distance. A set of waypoints obtained from the platforms cited are shown in Figure 6.

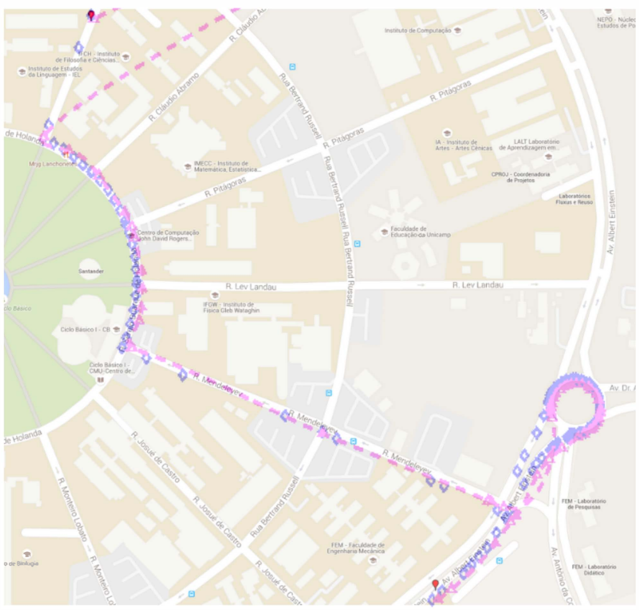

Figure 6. In magenta original waypoints provided by Open Street Maps and in blue waypoints provided by Google Maps. Global route curve constructed with Open Street Maps original waypoints in dashed line

In the Figure above, the blue markers represent the waypoints obtained from Google Maps and the magenta markers represent the one obtained from Open Street Maps. The dashed line represents the global route curve constructed from the Open Street Maps waypoints. In Figure 7, in black were represented the original waypoints provided by Open Street Maps and the associated global route curve constructed and in magenta were represented the processed waypoints and associated curve.

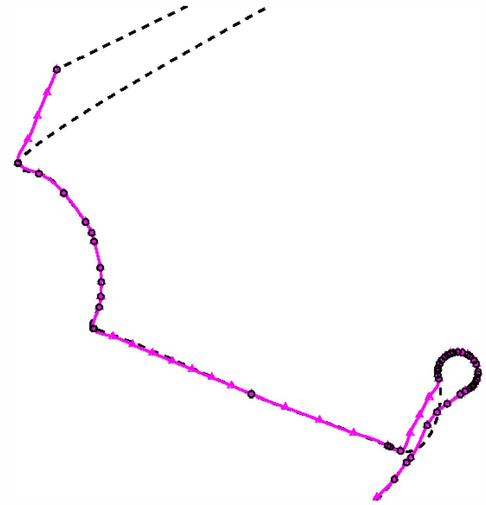

Figure 7. In black, original waypoints provided by Open Street Maps and associated global route curve constructed, and, in magenta, processed waypoints and associated global route curve constructed

In Figure 8, it is illustrated in details the first roundabout of the real route. In black, the original waypoints provided by Googles Maps and associated global route curve constructed and in blue the waypoints processed and associate curve.

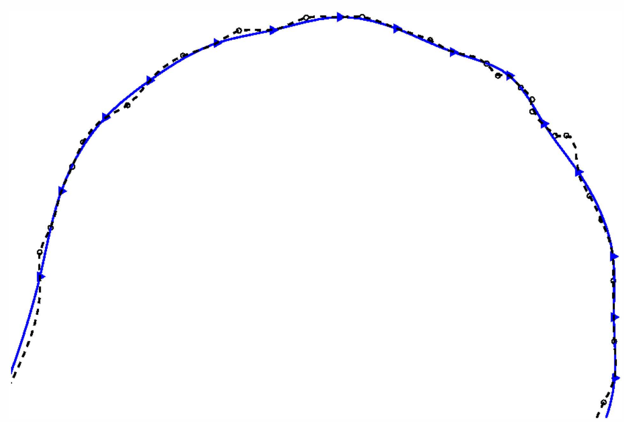

Figure 8. In black, original waypoints provided by Google Maps and associated global route curve constructed, and, in blue, processed waypoints and associated global route curve constructed

From Figures 7 and 8 it is observed a significant improvement in the global route curve constructed with the processed waypoints, magenta and blue colors, when compared to the ones constructed with the original set of waypoints provided, black colors. The new waypoints added to the original set of waypoints provided by Open Street Maps allowed the construction of a representative global route curve of the real route, Figure 7. The remotion of some waypoints from the original set provided by Google Maps allowed the construction of a smoother curve in the roundabout, Figure 8.

B. Arc length distance reparameterization problem

Once constructed a global route curve representative of the real route, the reparameterization to the arc length distance is performed. An adequate reparaterization, as well as the representative construction discussed above, is essential to the well performance of the technique discussed by [2] because its equating was developed considering a global route curve parameterized in the arc length s. To analyze the reparameterization quality it will be used the global route curve first derivative, known as unit velocity or Frènet tangent vector. The equation 7 shows the unit velocity vector.

$$\vec{T}(s) = \left(\frac{dx_b(s)}{ds}, \frac{dy_b(s)}{ds} \right) \tag{7}$$

For a perfect arc length parameterized curve the norm of the vector $\vec{T}$ is always unitary, distancing itself from the unit value according to the reparametezitaion quality. The following graphics show the results of the norm of the vector $\vec{T}$ subtracted by one as function of the arc length distance s.

Figure 9 shows the values of the norm of the vector $\vec{T}$ subtracted by one of three different reparemeterizarion performed upon a global route curve constructed by the waypoints from Open Street Maps. The blue curve represents the reparameterization quality performed with the 107 processed waypoints. In magenta, the quality of the reparameterizarion performed with 200 points upon the global route curve constructed and in blue the quality of the reparameterization performed with 300 points.

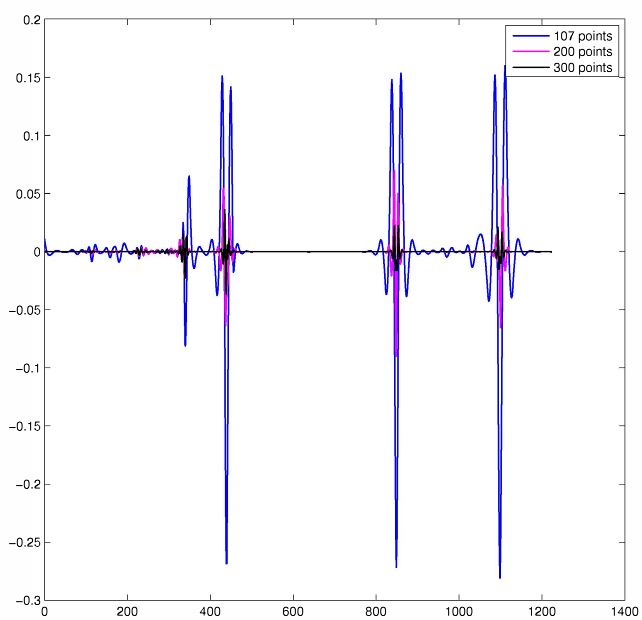

Figure 9. The norm of the vector $\vec{T}$ subtracted by one as function of the arc length distance s of the global route curve constructed with the waypoints provided by Open Street Maps

As the previous Figure, Figure 10 shows the values of the unity velocity vector norm subtracted by one of three different reparameterization performed upon the global route curve constructed by Google Maps waypoints. In blue, the behavior of the norm of the vector $\vec{T}$ subtracted by one of the performed reparameterization using the google's processed waypoints. In magenta, the quality of the reparameterizarion performed with 200 points on the global route curve constructed and in blue the quality of the reparameterization performed with 300 points.

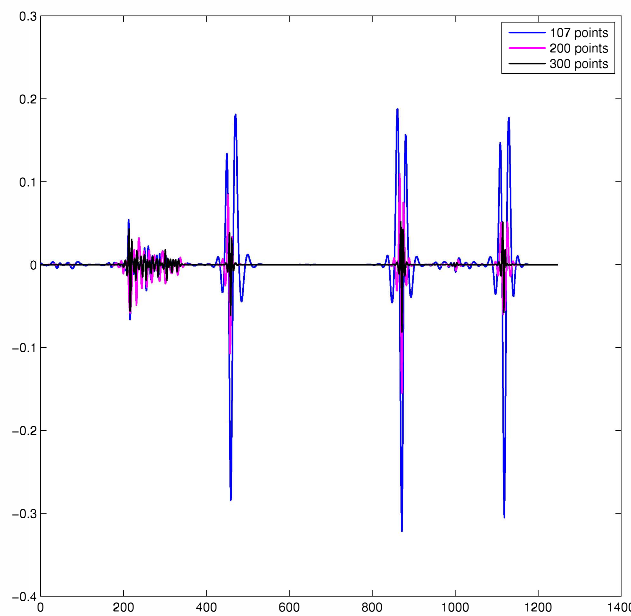

Figure 10. The norm of the vector $\vec{T}$ subtracted by one as function of the arc length distance s of the global route curve constructed with the waypoints provided by Google Maps

From the graphics above, it is possible to note a significant approximation of the values of the norm of the vector $\vec{T}$ subtracted by one to the number zero, which is the expected result for a perfect parameterized curve, as the number of points used in the reparameterization increase. These approximation indicates that the reparameterization quality was improved.

From the results shown in Figures 9 and 10 it is clear the necessity of find a good trade off between the arc length reparameterizarion quality sought and the number of waypoints used to do it. If, for one side, a global route curve badly arc length distance parameterized is an instability source, decreasing the robustness of the local path planning presented by ([2]), for other side, the use of high waypoints number can overload the communications between planning algorithm and the *real time* control system process.

IV. Conclusion

By the presented in this article, it is possible to conclude that is necessary a prior processing of the original waypoints provided by the on-line maps and route planning Open Street Maps and Google Maps in order to achieve a representative global route curve of the real route and to perform a satisfactory reparameterization to the arc length s of the constructed curve. The prior processing stage consists in adding, removing and redistributing the original waypoints. In specific to the arc length s reparameterization process, it was shown that perform it with the same waypoints used in the construction stage of the global route curve is likely an inappropriate approach because it generates a global route curve badly arc length parameterized. This condition makes necessary an engineering study to reach a good balance between the points utilized in the reparameterization process and the necessary quality of the arc length s parameterized global route curve. It is worth noting that a global route curve badly parameterized in the arc length s will inevitably lead the presented technique or any other based in arc length parameterized curves to a potentially dangerous failure. The arc length s parameterization quality problem can be softened by reducing the size of the paths' lengths, which will lead to smaller integration steps and, consequently, smaller errors.

The presented technique has two main interesting features. First, it works in the arc length space s instead the time space t, which allows the generation of the candidate paths without the profile velocity. Second, it works in the curvilinear coordinate system instead of the Cartesian system, which allows generate the candidates paths with simple cubic polynomials, what would just be possible in the Cartesian system with more complex equation.

ACKNOWLEDGMENT

This work has been cofounded by CNPq and CAPES. Garcia O. is sponsor by a Scholarship PEC/PG CAPES/CNPq-Brazil

REFERENCES

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*, 2nd ed. The MIT Press, Febraury 2011.

[2] S. Keonyup Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," vol. 13, no. 4, pp. 1599–1616, 2012.

[3] X. Li, Z. Sun, Q. Zhu, and D. Liu, "A unified approach to local trajectory planning and control for autonomous driving along a reference path," in *Mechatronics and Automation (ICMA), 2014 IEEE International Conference on*, Aug 2014, pp. 1716–1721.

[4] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 987–993.

[5] U. Lee, S. Yoon, H. Shim, P. Vasseur, and C. Demonceaux, "Local path planning in a complex environment for self-driving car," in *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2014 IEEE 4th Annual International Conference on*, June 2014, pp. 445–450.

[6] 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.

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

[8] J. W. Peterson, "Arc length parameterization of spline curves," *Journal of Compu ter Aided Design*, 2006.

[9] H. Wang, J. Kearney, and K. Atkinson, "Arc-length parameterized spline curves for real-time simulation," 2002.

[10] R. J. Sharpe and R. W. Thorne, "Numerical method for extracting an arc length parameterization from parametric curves," *Computer-aided design*, vol. 14, no. 2, pp. 79–81, 1982.

[11] B. Guenter and R. Parent, "Computing the arc length of parametric curves," vol. 10, no. 3, pp. 72–78, 1990.

[12] H. Wang, J. Kearney, and K. Atkinson, "Robust and efficient computation of the closest point on a spline curve," in *Proceedings of the 5th International Conference on Curves and Surfaces*, 2002, pp. 397–406.

[13] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer Science & Business Media, 2009.

[14] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. Wiley New York, 2006, vol. 3.

[15] 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.

[16] Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenét frame," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 987–993.

[17] J. M. Snider, "Automatic steering methods for autonomous automobile path tracking," *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08*, 2009.

[18] GoogleMaps, "Google company," https://www.google.com.br/maps/@-22.8195851,-47.0631774,15z, 2015.

[19] OpenStreetMap, "The free wiki world map," http://www. openstreetmap. org, 2015.