**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS**

Júnior Anderson Rodrigues da Silva

**Planejamento de trajetória com curvatura contínua e
linear por partes para veículos autônomos**

São Carlos

2018

Júnior Anderson Rodrigues da Silva

**Piecewise linear continuous-curvature path planning for
autonomous vehicles**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, para obtenção do título de Mestre em Ciências - Programa de Pós-Graduação em Engenharia Elétrica.

Área de concentração: Sistemas Dinâmicos

Orientador: Prof. Dr. Valdir Grassi Jr.

São Carlos

2018

Trata-se da versão corrigida da dissertação. A versão original se encontra disponível na EESC/USP que aloja o Programa de Pós-Graduação de Engenharia Elétrica.

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

Sp Silva, Júnior Anderson Rodrigues da
Planejamento de trajetória com curvatura contínua e
linear por partes para veículos autônomos / Júnior
Anderson Rodrigues da Silva; orientador Valdir Grassi
Jr.. São Carlos, 2018.

Dissertação (Mestrado) - Programa de Pós-Graduação
em Engenharia Elétrica e Área de Concentração em
Sistemas Dinâmicos -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2018.

1. Planejamento de caminho. 2. Veículos autônomos.
3. Clotóides. 4. Mapeamento. I. Título.

Eduardo Graziosi Silva - CRB - 8/8907

FOLHA DE JULGAMENTO

Candidato: Engenheiro **JÚNIOR ANDERSON RODRIGUES DA SILVA**.

Título da dissertação: "Planejamento de trajetória com curvatura contínua e linear por partes para veículos autônomos".

Data da defesa: 26/01/2018.

Comissão Julgadora:

Resultado:

Prof. Doutor **Valdir Grassi Junior**
(Orientador)
(Escola de Engenharia de São Carlos/EESC)

Aprovado

Prof. Associado **Denis Fernando Wolf**
(Instituto de Ciências Matemáticas e de Computação/ICMC)

APROVADO

Prof. Dr. **Luciano Cunha de Araújo Pimenta**
(Universidade Federal de Minas Gerais/UFMG)

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia Elétrica:
Prof. Associado **Luís Fernando Costa Alberto**

Presidente da Comissão de Pós-Graduação:
Prof. Associado **Luís Fernando Costa Alberto**

This work is dedicated to my parents Maria Ivone and José Maria, and to my sister Josiane.

ACKNOWLEDGEMENTS

I would like to express my very great appreciation to Professor Valdir Grassi Junior for accepting be my Professor Advisor in this Master's dissertation. His huge support and expert advice made possible the achievement of this work. I am particularly grateful for the assistance given by the Laboratory of Mobile Robotics (LRM), especially to Professor Denis Wolf, Carlos Massera and Tiago dos Santos, that provided the vehicle platform (CaRINA II) to collect data and help me a lot with import advices in the development of this work. Also, assistance provided by the Department of Electrical and Computer Engineering was greatly appreciated.

I would like to offer my special thanks to the Coordination for the Improvement of Higher Education Personnel (CAPES) and the Foundation for Engineering Technological Development (FDTE) for the financial support.

I wish to acknowledge the help provided by my colleagues of the Laboratory of Intelligent Systems (LASI). The time spent in the laboratory was much more pleasant because of their fellowship. My special thanks are extended to Professor Janison Rodrigues for the many advices that encouraged me to initiate this Master's course.

This project would have been impossible without the support of my family, friends and girlfriend, who stood by my side throughout all challenges on towards the conclusion of this Master's dissertation.

“Discipline is freedom.”

Renato Russo

ABSTRACT

SILVA, J. A. R. **Piecewise linear continuous-curvature path planning for autonomous vehicles**. 2018. 88p. Dissertation (Master) - São Carlos School of Engineering, University of São Paulo, 2018.

Autonomous vehicles have increasingly become an attractive field due its promising capabilities of improvements regarding safety, comfort, traffic flow etc. A required attribute for those vehicles is the ability of autonomously compute its path towards a destination point. The path must be planned considering the constructive aspects of the vehicle in order to guarantee the maneuver feasibility. This work consists on computing a feasible path for autonomous vehicles with non-holonomic constraints. Piecewise linear continuous-curvature paths constituted of clothoids, circular arcs, and straight lines are used for this purpose, providing passenger's comfort. The **road network** is modeled from GPS (Global Positioning System) vehicle trajectories by defining lanes, roundabouts and intersections. GPS points are used later to parameterize lanes using clothoids and to extract roundabout centers and radii. This approach provides a sparse road network model since GPS points are replaced by parameterized curves. The information about connections between roads coming from the model is used by a global path planner, which computes a minimal length route from the vehicle current position to the destination point. After that, path planners compute intersection and roundabout paths depending on the nature of connections between roads. Also, lanes changes are performed to obey traffic rules. These three path planners that connects adjacent roads use clothoids, circular arc, and straight lines as interpolating curves whose curvature is constrained to that the vehicle can perform: the **intersection path planner** uses only a minimal amount of steering to perform the maneuver, increasing the comfort level; the **roundabout path planner** takes the roundabout center and radius as well as parameters that defines the entrance and exit maneuvers to compute the path; the **lane change path planner** connects lanes belonging to the same road with a prescribed longitudinal traveled distance depending on whether this maneuver is required. In the end, an global continuous-curvature path is generated. As the result of this work, a real urban scenario is modeled and the proposed approaches are validated.

Keywords: Path planning, autonomous vehicles, clothoids, mapping.

RESUMO

SILVA, J. A. R. **Planejamento de trajetória com curvatura contínua e linear por partes para veículos autônomos**. 2018. 88p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2018.

Veículos autônomos têm cada vez mais se tornado um campo atraente de pesquisa devido às suas capacidades promissoras de melhorias em segurança, conforto, fluxo de tráfego, etc. Um atributo necessário para esses veículos é a capacidade de calcular, de forma autônoma, o seu caminho para um ponto de destino. O percurso deve ser planejado considerando os aspectos construtivos do veículo para que a viabilidade das manobras a serem executadas seja garantida. Este trabalho consiste no planejamento de trajetória para veículos autônomos com restrições não-holonômicas. Utilizam-se, para esse efeito, trajetórias cuja curvatura seja contínua e linear por partes, constituídas por clotóides, arcos de circunferência e retas, de forma a proporcionar conforto aos passageiros. A **topologia de vias** é modelada a partir de trajetórias definidas por pontos de GPS (Sistema de Posicionamento Global), definindo pistas, rotatórias e cruzamentos. Pontos de GPS são usados posteriormente para parametrizar as pistas usando clotóides a para extrair centros e raios das rotatórias. Essa abordagem proporciona um modelo esparsa de topologia de vias uma vez que pontos de GPS são substituídos por curvas parametrizadas. A informação a cerca das conexões entre vias advinda do modelo é usada por um planejador de caminho global, o qual calcula a rota mais curta da posição atual do veículo até seu ponto de destino. Após essa etapa, planejadores calculam caminhos em cruzamentos e rotatórias dependendo do tipo de conexão entre as vias. Também, trocas de faixa devem ser executadas para obedecer regras de trânsito. Esses três planejadores de caminho usam clotóides, arcos de circunferência e retas como curvas interpoladoras, cuja curvatura é restrita a valores que o veículo é capaz de executar: o **planejador de caminho em cruzamentos** usa apenas um mínimo de velocidade de rotação do volante do veículo para executar a manobra, melhorando o nível de conforto; o **planejador de caminho em rotatórias** requer as coordenadas do centro e o raio da rotatória, bem como parâmetros que definem as manobras na entrada e na saída da rotatória para calcular o caminho; o **planejador de caminho para troca de faixa** conecta pistas pertencentes à mesma via com uma distância longitudinal do caminho previamente determinada. Ao final, um caminho com curvatura globalmente contínua é gerado. Como resultado deste trabalho, um cenário urbano real é modelado e os métodos propostos são validados.

Palavras-chave: Planejamento de caminho, veículos autônomos, clotóides, mapeamento.

LIST OF FIGURES

Figure 1.1 – Example of path planning considering an urban scenario. The planned path is the resulting connections between lanes (blue lines) by link paths (yellow lines).	25
Figure 2.1 – Road stretch depicting the ICC position of a robot positioned at point P_r with heading θ_r	27
Figure 2.2 – Geometry of a bicycle model. The ICC is determined from the length of the bicycle body D_{bl} and the steering angle ϕ	28
Figure 2.3 – Path planning in the presence of obstacles. The robot $\mathcal{R}$ moves from $\mathbf{q}_{start}$ to $\mathbf{q}_{goal}$ as avoiding the obstacles $\mathcal{O}_i$	29
Figure 2.4 – Example of a Dubins's minimal length path.	31
Figure 2.5 – Representation of a sampled curve depicting the inputs used to compute the discrete curvature.	32
Figure 2.6 – Configuration change from $\mathbf{q}_1$ to $\mathbf{q}_2$	33
Figure 4.1 – Clothoids with different σ . Higher κ are reached for the same s as σ increases.	42
Figure 4.2 – Example of G^1 Hermite interpolation.	43
Figure 4.3 – Example of G^2 Hermite interpolation.	43
Figure 4.4 – Clothoid computation using the lookup table method.	45
Figure 4.5 – Clothoid trigonometry.	46
Figure 4.6 – Clothoid path using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta = \pi/4$ rad.	47
Figure 4.7 – Path made up of two symmetrical clothoids using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta = -\pi/4$ rad.	47
Figure 4.8 – Discontinuous path constituted of two clothoids using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta_1 = -\pi/4$ rad (blue clothoid); $x_{ct} = 10$ m and $-\delta_2 = \pi/6$ rad (red clothoid).	48
Figure 4.9 – Scheme depicting the trigonometric relation in order to compute x_{ct}	48
Figure 4.10–Path constituted of two linked clothoids using a minimal amount of steering with $\delta_1 \neq \delta_2$	49
Figure 4.11–Three examples of possible maneuvers at a intersection. Each lane is 4 m wide and the maximum curvature allowed is $ 0.2 \text{ m}^{-1}$	49
Figure 4.12–Scheme depicting the computation of the lane change parameters.	50
Figure 4.13–Lane change. Inputs: $\alpha_1 = 0.47$ rad, $\alpha_2 = -0.93$ rad and $h = 22.2$ m.	51
Figure 4.14–Example of lane change with parallel initial and final positions.	52
Figure 4.15–Example of a lane change with non parallel initial and final positions.	52
Figure 4.16–Scheme depicting the roundabout path planning approach.	53

Figure 4.17–Computation of β_i in order to reduce the error relative to the circulatory roadway.	55
Figure 4.18–Evaluation of the radius of curvature when the first exit is taken. . . .	56
Figure 4.19–Simulations considering a 11 m radius for the external lane. All lanes are 3 m wide.	57
Figure 4.20–Simulations considering a 25 m radius for the external lane. All lanes are 4 m wide.	58
Figure 5.1 – Original path to be reconstructed by the optimization process.	62
Figure 5.2 – Reconstructed path resulting from the optimization process.	63
Figure 5.3 – Parameters used to reconstruct the clothoids.	64
Figure 5.4 – Error in θ_0 caused by the distance Δs between the samples.	64
Figure 5.5 – Representative example of a road network representation. Roads are separated by solid lines, whereas dashed lines separate lanes.	65
Figure 5.6 – Curvature profiles of initial and destination lanes.	67
Figure 5.7 – Example of a graph representing connections between roads.	68
Figure 5.8 – Required lane changes.	69
Figure 5.9 – Scheme depicting the lane change path planning.	69
Figure 6.1 – CaRINAII: Intelligent Car for Autonomous Navigation.	71
Figure 6.2 – Road network considering a real map. The solid lines represent the center of the lanes. Two adjacent green dots define a clothoid segment.	72
Figure 6.3 – Intersection connecting three lanes in the road network model.	73
Figure 6.4 – Intersection path planning.	74
Figure 6.5 – Two lanes belonging to the same road in the road network model. . . .	74
Figure 6.6 – Lane change path planning considering tree longitudinal distances: 30 m, 65 m and 100 m.	75
Figure 6.7 – Roundabout path planning. The entrance road is double-lane and the external lane inside the roundabout is taken.	76
Figure 6.8 – Roundabout path planning. The entrance road is double-lane and the internal lane inside the roundabout is taken.	76
Figure 6.9 – Roundabout path planning. The entrance road is single-lane and the external lane inside the roundabout is taken.	77
Figure 6.10–Roundabout path planning. The entrance road is single-lane and the internal lane inside the roundabout is taken.	77
Figure 6.11–Roundabout path planning. Increase of r_{off} by an amount of 0.8 m in the roundabout planning when the entrance road is single-lane and the internal lane inside the roundabout is taken.	78
Figure 6.12–Global path planning. Three paths are planned from the current position to a destination point: 1 (path 1), 2 (path 2) and 3 (path 3).	79

LIST OF TABLES

Table 4.1 – Inputs used to compute the lookup table for the path in Figure 4.4. . .	44
Table 5.1 – Initial parameters of the path depicted in Figure 5.2.	64
Table 5.2 – Clothoids sharpness and length of the path depicted in Figure 5.2. . .	65
Table 6.1 – Elapsed times assuming different Δs in the path interpolation.	80

LIST OF ABBREVIATIONS AND ACRONYMS

USP	Universidade de São Paulo
EESC	Escola de Engenharia de São Carlos
Darpa	Defense Advanced Research Projects Agency
USA	United States of America
ICC	Instantaneous Center of Curvature
EKF	Extended Kalman Filter
GPS	Global Positioning System
CC	Continuous Curvature
DCC	Double Continuous Curvature
RTK	Real Time Kinematic

CONTENTS

1	INTRODUCTION	23
2	TECHNICAL BACKGROUND	27
2.1	Kinematic model of car-like robots	27
2.2	Configuration space	29
2.3	Computing the discrete curvature	31
2.4	Comfort level	32
2.5	Final considerations	33
3	RELATED WORK	35
3.1	Interpolating Curve Planners	35
3.2	Road network modeling	36
3.3	Dynamic paths	37
3.4	Final considerations	39
4	INTERSECTIONS, LANE CHANGES AND ROUNDABOUTS	41
4.1	Clothoids	41
4.1.1	Real-time approximation of clothoids	43
4.2	Turning	44
4.3	Intersections	48
4.4	Lane change	50
4.5	Roundabout path planner	51
4.6	Final considerations	59
5	GLOBAL PATH PLANNING	61
5.1	Piecewise linearization of the curvature	61
5.2	Lane parametrization	62
5.3	Lane-level road network model	65
5.4	Route computation	67
5.5	Lane change path	68
6	RESULTS	71
6.1	Road network model	71
6.2	Intersections	73
6.3	Lane change path	73
6.4	Roundabouts	75
6.5	Global path planning	78

7	CONCLUSION	83
	BIBLIOGRAPHY	85

1 INTRODUCTION

The use of mobile robots for transportation of people and goods has been studied intensively in the last years (VILLAGRA et al., 2012; FERNÁNDEZ et al., 2013; FERNANDES et al., 2014). Researchers's efforts have been focused on adapting four-wheeled automobiles to operate in partial or fully automated mode. Litman (2014) highlights that these autonomous robots, popularly called autonomous vehicles, self-driving cars or cybercars, could significantly reduce parking and traffic problems, accidents, pollutants emission and propitiate a better use of the travel time (the driver would be free to do other activities) once they achieve a high level of automation, and the maintenance costs are not prohibitive to final customers. In order to achieve this aim, autonomous cars must contain a set of attributes that ensure efficiency and reliability (PARENT, 2007).

The Darpa Grand Challenge, a competition promoted by the USA Defense Advanced Research Projects Agency in 2004, was an important step towards the technology development of self-driving cars. In this competition, four-wheeled automobiles should autonomously complete a predefined route of 225 km trough a desert. None of the competitors was able to complete the task. According to Buehler, Iagnemma and Singh (2007), the competition showed that algorithms that worked very well in simulations could not provide the same results in real scenarios. Indeed, there was a significant evolution from the first Darpa Grand Challenging to the subsequent ones: the second Darpa Grand Challenge (2005), when five competitors were able to complete the course of 212 km, and the Darpa Urban Challenge (2007), which imposed upon the competitors the execution of some tasks in urban traffic scenarios such as lane change, obstacle avoidance, parking and intersection negotiation (URMSON JOSHUA ANHALT, 2009). These competitions gave more visibility to self-driving cars and showed that a world where intelligent cars could really take place was not far to achieve.

Among many capabilities, self-driving cars must be able to autonomously define a path that leads it to its destination point. At the highest level in the decision-making hierarchy, the vehicle must find a route taking into account connections between the roads that compose the structured environment the vehicle is inserted in, the so-called road network. If two or more feasible routes are found, the best one is chosen considering the cost of traversing each route. This cost may be related to travel time, travel distance, fuel consumption, to name a few. After the route has been chosen, the task is to locally decide how to progress towards the destination. This kind of decision composes the behavioral layer in the decision-making hierarchy, e.g. staying in the current lane or moving to another one (lane change), merging in a highway, traversing a roundabout, etc. Then, the motion planning layer selects a continuous path, which is passed to the low-level control system.

If the evolution of the vehicle is parameterized with respect to time, the motion planning problem takes the form of a trajectory. Otherwise, this task takes the form of a path. Path planning is generally applied to static environments. On the other hand, trajectory planning is suitable to dynamic environments with the presence of other traffic participants, such as pedestrians, bicyclists and other vehicles, since their positions vary over time. Therefore, trajectory planning can be seen as a generalization of path planning in static environments (PADEN et al., 2016).

Just after defining the route, a reference path must be generated to guide the vehicle towards its destination. Path planners at low levels hierarchy that deal with static and dynamic obstacles, by sensing the environment and re-planning the path, can benefit from accurate and smooth reference paths, since the center of the lane is the desirable position for running cars in many cases (LI et al., 2017; OBAYASHI; UTO; TAKANO, 2016; ALIA et al., 2015). In this way, vehicle's physical limits must be considered such as minimum turn radius and maximum steering rate change, in order to produce a feasible reference path.

The knowledge about the connections between roads and the geometry of lanes, intersections and roundabouts can be sufficient to plan a path. In this sense, the path planning problem can be divided into the following tasks:

- Accurately representing the geometry of lanes: a sequence of coordinate points can be used to describe the center line of the lanes, where two adjacent points create an edge segment. The curve smoothness depends on the length of adjacent edges, meaning that more accurate representations require larger number of points. Consequently, there is a trade-off between accuracy and number of points representing each lane.
- Defining the connections between roads: this allows the computation of the route regarding some cost to traversing each road.
- Planning a continuous path towards the goal point: in an urban scenario, roads are connected either by intersections or roundabouts. These connections can be explicitly defined in the road network model, assisting the vehicle to move from a road to its subsequent one. Once this connection is known, it is used to plan link paths between roads. Although these paths may be previously represented in the road network model, it is preferred to compute only those paths that belonging to the computed route in order to reduce the number of points in the model. The task becomes to compute link paths based on intersection and roundabout features. Moreover, lane change maneuvers are necessary to obey traffic rules and must be executed whenever it is required.

Figure 1.1 shows an example of path planning in urban scenario where intersections

and roundabouts are considered. Firstly, the route from the start point to the destination point is computed. After that, the path is generated taking into account the geometry of lanes belonging to the chosen route and lanes connections. As it can be seen, path planning at roundabouts and intersections is required to accomplish the task.

Figure 1.1: Example of path planning considering an urban scenario. The planned path is the resulting connections between lanes (blue lines) by link paths (yellow lines).

Source: Prepared by the author.

This work deals with the problem of sparsely representing the road network in urban scenarios composed by roads, intersections and roundabouts. Also, a path planning approach for autonomous vehicles considering vehicle's constructive aspects and comfort level is developed, regarding the structured environment features given by the road network model. The geometry of lanes is described by parametric curves in order to reduce the number of points required for accurate representation. Clothoids, which are curves whose curvature varies linear with respect to the arc length, are used for this purpose. Connections between adjacent roads, made by either intersections or roundabouts, are used to build the road network model. This allows computing the best route considering the minimal travel distance to an assigned point. Intersection and roundabout features are used to plan link paths after the definition of the route. Also, lane change paths are planned to obey traffic rules. All link path planners take into account the maximum curvature the vehicle can perform and comfort level, using clothoids, arc segments, and straight lines as curve primitives. The resulting piecewise linear continuous-curvature path is smooth and feasible to be tracked.

In summary, the main contributions of this work are:

- A road network model describing connections between roads with a sparse representation of lanes, intersections and roundabouts. The model is used in a global path planning throughout a structured environment.
- A real-time lane change path planner connecting two lanes belonging to the same road with prescribed longitudinal travel distance, and parallel and non-parallel configuration change. The curvature is bounded according to the minimum turn radius that the vehicle can drive.
- A real-time roundabout path planner connecting the assigned entry and exit of a roundabout with a prescribed maximum curvature. The planner is based on parameters that describe the geometry of roundabout entry/exit as well as roundabout center and radius. The path planner is able to deal with single-lane and double lane roundabouts. Moreover, the proposed approach considers geometrically uniform and non-uniform roundabouts in the planning.
- A continuous-curvature path planning in an urban scenario, where traffic rules must be obeyed. This is made by connecting lanes by roundabout, intersection and lane change paths with prescribed maximum curvature, and low curvature derivative, improving the comfort level.

The remaining chapters are organized as follows: Chapter 2 gives important background concepts related to path planning for autonomous vehicles; in Chapter 3, works related to road network modeling and interpolating curve planners are discussed; Chapter 4 shows the process of path computation considering intersections, lane changes, and roundabouts using clothoids; a road network modeling is presented in Chapter 5, as well as the shortest length route planning; the methods developed in this work are applied to a real scenario and practical results are shown in Chapter 6; finally, important remarks and future work are discussed in Chapter 7.

2 TECHNICAL BACKGROUND

This chapter provides some background concepts applied to path planning for autonomous vehicles, detailing the requirements that must be fulfilled in order to achieve feasible paths and to increase comfort as the vehicle travels along the planned path.

2.1 Kinematic model of car-like robots

Figure 2.1 shows a road stretch on the plane, whose center is represented by the dashed line. At point P_r , the vector tangent to the path has an orientation angle θ_r .

Figure 2.1: Road stretch depicting the ICC position of a robot positioned at point P_r with heading θ_r .

Source: Prepared by the author.

(measured counter-clockwise from the x axis heading), considering that the path is followed from bottom to top. The circular arc that fits the path at P_r belongs to the circumference of radius R_r , whose center is the Instantaneous Center of Curvature (ICC). It can be seen that R_r and the ICC depend on the path geometry and may vary along it. The **curvature** κ_r at P_r is defined as

$$\kappa_r = \frac{1}{R_r}. \quad (2.1)$$

Considering that the path is discretized, its geometry is represented by sample points named **way-points** and, from them, values of tangent angles and curvature can be evaluated.

The constructive aspects of the robot are relevant and must be considered in order to know whether it is able to track the planned path. Figure 2.2 shows a bicycle model used

to represent **car-like robots**, which are characterized by Ackerman steering geometry (DUDEK; JENKIN, 2010), moving with longitudinal velocity v . The front wheel is able

Figure 2.2: Geometry of a bicycle model. The ICC is determined from the length of the bicycle body D_{bl} and the steering angle ϕ .

Source: Prepared by the author.

to turn and gives the steering angle ϕ , whereas the rear wheel is always aligned with the bicycle body, which is considered a rigid body. It should be noted in Figure 2.2 that θ gives the heading of the vehicle and P is the guided point, which is controlled in order to follow the assigned path (SHIN; SINGH, 1990). The intersection between the lines that pass through the rear and front wheels axes gives the ICC. The distance between the ICC and P represents the radius of curvature R . The curvature of the vehicle is given by $\kappa = 1/R$.

Considering that the wheels roll without slipping and the vehicle is traveling at low speeds, only the kinematic equations can be considered, since the lateral dynamic effects can be neglected (LIMA et al., 2015a). Hence, from Figure 2.2,

$$\begin{aligned}\dot{x} &= v \cos \theta \\ \dot{y} &= v \sin \theta,\end{aligned}\tag{2.2}$$

and

$$R = D_{bl} \tan(\pi/2 - \phi) = \frac{D_{bl}}{\tan \phi},\tag{2.3}$$

where D_{bl} is the distance between the two wheel axes, representing the length of the bicycle body. The angular velocity $\dot{\theta}$ can be computed as

$$\dot{\theta} = \frac{v}{R}.\tag{2.4}$$

Substituting (2.3) in (2.4) and making use of (2.1) yields

$$\dot{\theta} = \frac{\tan \phi}{D_{bl}} v = \kappa v.\tag{2.5}$$

Assuming $\tau = \dot{\kappa}$, then

$$\tau = d \left(\frac{\tan \phi}{D_{bl}} \right) / dt = \frac{\dot{\phi}}{D_{bl} \cos^2 \phi}. \quad (2.6)$$

where t represents time.

The following kinematic model results from the considerations made so far:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\kappa} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \kappa \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tau. \quad (2.7)$$

2.2 Configuration space

Four different states are defined in the kinematic model (2.7): x and y coordinates on the plane, the heading θ , and the curvature κ . This set of states, which represents the **pose** or **configuration** of the robot, will be represented by $\mathbf{q}$. The **configuration space** or **C-space** represents the different configurations that the robot can assume, whose dimension equals to the dimension of $\mathbf{q}$. Therefore, car-like robots defined by (2.7) has a four-dimension configuration space.

Planning a path means to determine a set of configurations that leads the robot towards its goal. However, $\mathbf{q}$ can not assume arbitrary values. An example is depicted in Figure 2.3. In order to reach its goal configuration $\mathbf{q}_{goal}$ starting from the initial

Figure 2.3: Path planning in the presence of obstacles. The robot $\mathcal{R}$ moves from $\mathbf{q}_{start}$ to $\mathbf{q}_{goal}$ as avoiding the obstacles $\mathcal{O}_i$.

Source: Prepared by the author.

configuration $\mathbf{q}_{start}$, the robot $\mathcal{R}$ is not allowed to cross the obstacles $\mathcal{O}_i$. This suggests the definition of **C-obstacle** $\mathcal{CO}_i$,

$$\mathcal{CO}_i = \{\mathbf{q} \in \mathcal{C} | \mathcal{R}(\mathbf{q}) \cap (\cup \mathcal{O}_i) \neq \emptyset\}, \quad (2.8)$$

as the portion of space occupied by obstacles, and the free configuration space $\mathcal{C}_{free}$,

$$\mathcal{C}_{free} = \{\mathbf{q} \in \mathcal{C} | \mathcal{R}(\mathbf{q}) \cap (\cup \mathcal{O}_i) = \emptyset\}, \quad (2.9)$$

as the portion of space free from obstacles. In this way, the robot is constrained to be in $\mathcal{C}_{free}$.

Other constraints arise from the constructive aspects of the robot. **Holonomic constraints** are defined in function of $\mathbf{q}$ in the form (DUDEK; JENKIN, 2010)

$$G(\mathbf{q}) = 0. \quad (2.10)$$

Consider the point P_{bm} , lying on the middle of the bicycle body, in Figure 2.2. It can be shown that once the coordinates of P_{bm} ($x_{P_{bm}}$ and $y_{P_{bm}}$) are determined, the states x , y and θ can be computed from

$$\begin{aligned} x + (D_{bl}/2) \cos \theta - x_{P_{bm}} &= 0 \\ y + (D_{bl}/2) \sin \theta - y_{P_{bm}} &= 0, \end{aligned} \quad (2.11)$$

meaning that x , y and θ are constrained to (2.11), since the bicycle body is assumed as a rigid body.

Whereas holomic constraints are regarded to $\mathbf{q}$, **non-holonomic constraints** are also considered in its derivatives as well (DUDEK; JENKIN, 2010), i.e.,

$$G(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \dots) = 0. \quad (2.12)$$

These constraints make the planning task more difficult since the robot is not allowed to move to any direction. Equation (2.2) through (2.6) describe these constraints for car-like robots. Another way to define non-holonomic constraints is analyzing the **degrees of freedom** of the robot. Car-like robots can assume positions on the 2-D plane, different headings and steering angles, thus adding up to four degrees of freedom. But it has two kinematic constraints: a) the vehicle is allowed to move only forward and backward and b) the steering angle is bounded (KATRAKAZAS et al., 2015). In fact, there is only control over the longitudinal velocity and the steering wheel angle in order to control four states. Parallel parking between two cars may be a good example of the difficulty in reaching desired configurations of typical four-wheeled automobiles even for human drivers.

Thus, planning a path from the start configuration to the goal one is to search in $\mathcal{C}_{free}$ a path that connects these two configurations considering the presence of holonomic and non-holonomic constraints.

Suppose a vehicle traveling along the path illustrated in Figure 2.4, composed by two straight lines and a circular arc. Also, consider that R_{min} is the minimum turn radius that the vehicle can perform. Dubins (DUBINS, 1957 apud FRAICHARD; SCHEUER,

Figure 2.4: Example of a Dubins's minimal length path.

Source: Prepared by the author.

2004) showed that the minimal length path is achieved joining straight lines, and circular arcs with radius R_{min} . However, $\dot{\kappa} \rightarrow \infty$ at the joins between the circular arc, and the straight lines (κ abruptly varies from 0 to $1/R_{min}$). Since the steering angle can not instantaneously change, this path is not feasible to be tracked by car-like robots with the non-holonomic constraints previously discussed. Otherwise, the vehicle would have to stop to reorient its front wheels.

2.3 Computing the discrete curvature

The fact of being in $\mathcal{C}_{free}$ does not ensure that the planned path is feasible to be followed. Hence, it is important to know the curvature along the path length s to be tracked by the vehicle. It makes the representation of the path in **space-domain** more suitable for this purpose. Making $v(t)dt = ds$ and assuming $v(t) \neq 0$ as a continuous function, from (2.7):

$$\begin{aligned} dx(s)/ds &= \cos(\theta(s)) \\ dy(s)/ds &= \sin(\theta(s)) \\ d\theta(s)/ds &= \kappa(s) \\ d\kappa(s)/ds &= \sigma(s), \end{aligned} \tag{2.13}$$

where $\sigma(s)$ is the variation of the curvature with respect to the distance traveled along the path.

The curvature of discrete planar curves can be estimated considering the curvature at each point P_i , with i being the point index as in Figure 2.5 and using the circum-circle form with two adjacent points of the curve (MULLINEUX; ROBINSON, 2007 apud MCCRAE; SINGH, 2009). Then,

$$\kappa(P_i) = \frac{2 \sin(\Delta\theta/2)}{\sqrt{\|\nu_1\| \cdot \|\nu_2\|}}, \tag{2.14}$$

where $\nu_1 = P_i - P_{i-1}$, $\nu_2 = P_{i+1} - P_i$ and $\Delta\theta = \arccos\left(\frac{\nu_1}{\|\nu_1\|} \cdot \frac{\nu_2}{\|\nu_2\|}\right)$. The distance traveled along the path is the sum of the Euclidean norm of each vector ν_i .

Figure 2.5: Representation of a sampled curve depicting the inputs used to compute the discrete curvature.

Source: Prepared by the author.

2.4 Comfort level

The comfort level experienced in vehicles is associated with lateral and longitudinal jerks, which are the derivatives of lateral and longitudinal accelerations with respect to time, respectively. The lateral acceleration experienced in a turn is given by

$$a_{lat}(t) = \kappa(t) \cdot v^2(t), \quad (2.15)$$

which means that a_{lat} is null when the vehicle is traveling on a straight line. If v is assumed to be constant, a_{lat} is maximum when $|\kappa| = \kappa_{max}$ (the maximum curvature the vehicle can perform). The lateral jerk ς_{lat} is computed as

$$\varsigma_{lat}(t) = \tau(t) \cdot v^2(t). \quad (2.16)$$

From (2.7) it should be noted that $\tau(s) = \sigma(s)v(s)$. Therefore, ς_{lat} in space-domain becomes

$$\varsigma_{lat}(s) = \sigma(s) \cdot v^3(s), \quad (2.17)$$

meaning that, once a constant longitudinal velocity has been set, the comfort level can be controlled by $\sigma(s)$. Aashto (2001) bounded within 0,9 m/s³ the lateral jerk for comfort assurance in railways.

Curvature discontinuities significantly decrease the level of comfort since $\sigma(s) \rightarrow \infty$. The presence of curvature discontinuities depends on the type of interpolation used in the planning. Consider the two configurations depicted in Figure 2.6. The aim is to plan a path connecting those configurations. When only (x_1, y_1, θ_1) and (x_2, y_2, θ_2) are matched at the given configurations, the path is geometrically continuous, also known as a **G^1 Hermite interpolation**. If κ_1 and κ_2 are also matched, then the path is said to be a two point **G^2 Hermite interpolation** (WALTON; MEEK, 2009). G^2 interpolation is

Figure 2.6: Configuration change from q_1 to q_2 .

Source: Prepared by the author.

suitable for car-like robots path planning since it provides a continuous-curvature profile, which is required for feasible tracking (SHIN; SINGH, 1990).

2.5 Final considerations

Some requirements must be fulfilled in order to plan feasible and comfortable paths for autonomous vehicles. The comfort level is directly associated to curvature derivative, meaning that curvature discontinuities must be avoided in the planning. Therefore, G^2 interpolation is desirable in path planning for autonomous vehicles. Also, the vehicle can perform turns with curvature values within a range given by the vehicle's physical limits.

In this work, it is assumed that there is no obstacle in the vehicles's surrounding area, meaning that $\mathcal{C}_{free}$ is given by the width and length of lanes. Moreover, the steering angle and its rate change are bounded, and the vehicle is allowed to move only forward.

3 RELATED WORK

Many studies have been developed in the field of path planning for autonomous vehicles. Relevant work applied to road geometry parametrization and to path planning considering configuration changes are discussed in this chapter.

3.1 Interpolating Curve Planners

A sparse representation of the roadway can significantly reduce the amount of data storage. On the other hand, reducing the number of points can decrease the smoothness of the path. Works in literature has aimed to develop strategies to sparsely represent the roadway without loss of information of its geometry and to compute smooth configuration transitions from interpolating curves. Planners that use this approach are known as Interpolating Curve Planners, which attempt to extract a smooth path given a set of way-points using some technique of interpolation (GONZÁLEZ et al., 2016):

- **Lines and Circles:** two configurations can be interpolated using only straight lines, and circular arcs. It is the case of Dubin's minimal path length aforementioned. The constraint on the maximum curvature that the vehicle can drive is handled using circular arc segments with a minimal radius. However, the resulting path is jerky because of curvature discontinuities at the link point of two segments.
- **Clothoid Curves:** curves whose curvature varies linearly with respect to its arc length, providing a constant curvature rate change, which make them suitable for railways and highways design (BAASS, 1982; WALTON, 1989; AASHTO, 2001). Clothoids can be parameterized from their curvature, taking into account non-holonomic constraints and comfort level. They are evaluated in function of the Fresnel's integrals (ABRAMOWITZ; STEGUN, 1964), which do not have analytical solution, requiring numerical techniques to be computed.
- **Polynomial Curves:** the geometry of a polynomial curve can be determined by its coefficients, and the degree of continuity (G^1 , G^2 etc.) by the polynomial degree. Nevertheless, the coefficients computation can become difficult since higher polynomial degree are required for smooth configuration transitions (4th degree or higher).
- **Bézier Curves:** these curves are parameterized by control points. Their geometry is determined by intuitively placing the control points to fit the road shape. Also, Bézier curves can be fast computed since they have a closed-form solution. However,

like polynomial curves, the computational time increases with respect to the curve degree and malleability is lost.

- **Spline Curves:** a spline is a piecewise parametric curve that can be divided in sub-intervals. The pieces of the polynomial curve tie together to automatically produce smooth overall curves. The join between two segments is named knots. The curvature continuity only can be achieved with a properly concatenation between the parts, which can decrease the malleability to fit the road constraints.

Next sections present relevant works in literature that has applied Interpolating Curves Planners to road network modeling and to configuration transition planning.

3.2 Road network modeling

Bétaille and Toledo-Moreo (2010) used an Extended Kalman Filter (EKF) to model roadways with a 5 cm maximum fitting error. The lane geometry is assumed to be a stochastic process and GPS coordinates are used as observations to estimate the parameters of straight lines, circular arcs, and clothoids, using the second order Taylor's expansion of the Fresnel's integrals as model equations. The process is restarted when the maximum error is reached. Connections between clothoid segments are automatically evaluated considering traffic flow, but scenarios with intersections and roundabout are not discussed. Moreover, the extraction process does not guarantee path continuity since the end configuration of a segment might be different from the start configuration of the following one. A lane-level road network is also proposed by Zhang et al. (2016). Roundabouts and intersections are also considered. Lanes are represented by Cubic Hermite splines (third-degree polynomials) that ensures G^2 continuity between adjacent points belonging to the same lane. Those splines are also used to generate paths at intersections and roundabouts, as well as to compute lane change maneuvers with, at least, G^1 continuity.

An automatic semantic way-point mapping was developed by Zoccoler et al. (2015). Roundabouts and lanes are automatically recognized once a GPS predefined route is given. The amount of data is reduced by eliminating points in the straight lanes segments using filtering techniques. Roundabouts are defined by their center and radius. The connection points between lanes and roundabouts are stored in order to reconstruct the path. However, the amount of data can still be large since the path was not parameterized. Moreover, there is no a curvature analysis of the path after the filtering process has been performed.

Roadway modeling is also addressed by Jo and Sunwoo (2014). The problem of finding a parametric path is treated using B-splines, which are splines constructed by basic functions, knots and control points. A set of acquired GPS data is processed in such a way to produce an optimal smoothing road modeling regarded to the number of control points. The resulting path has a 0.1 m maximum fitting error and curvature continuity.

Clothoids also can be used to fit the road shape. McCrae and Singh (2009) described a dynamic programming algorithm that searches for connectivity between points in the curvature, attempting to find the lower cost (defined as the sum of the segment lengths that connect the points) starting from the initial point to the last one, therefore producing G^2 continuity curves. The number of segments is minimized by defining a penalty cost for each linear piece. Although less penalty costs produces more accurate fitting, the maximum fitting error can be determined before hand. This drawback was addressed by Lima et al. (2015a) by making a piecewise linearization of the curvature while maintaining the fitting error within a prescribed value. The curvature is linearized by minimizing its second order derivative subject to the equations that impose the minimum fitting error.

3.3 Dynamic paths

For all aforementioned works the computational time is not a critical factor. All of them can be computed off-line by considering that the roadway is static. However, a class of maneuvers might be computed on-line because they depend on the chosen route. The most common ones performed in urban scenarios are lanes changes, and intersection and roundabout traversing.

Rastelli, Lattarulo and Nashashibi (2014) described a dynamic Bézier interpolation method that can be used to generate paths using road and obstacle information. Considering environmental characteristics it is possible automatically set the control points used in the Bézier interpolation. An optimization function is applied to select feasible paths based on the maximum curvature that the vehicle is constrained to drive and also to reduce the curvature continuity error when different segments are connected. In this way, a global G^1 path is achieved, but with low continuity error in curvature ensuring low lateral acceleration changes. Optimization techniques are also used by Montemerlo et al. (2008) and Werling et al. (2010). A quintic polynomial optimal path is computed by minimizing a cost function related to lateral jerk. Among many possible paths, the algorithm selects the one that provides the maximum lateral comfort. At each iteration of the optimization process, a new path must be generated until the optimal solution is reached.

In those three works, an optimization stage is required to prevent curvature discontinuities and to decrease the lateral jerk. However, optimization is time consuming and must be performed at each planned maneuver (GONZÁLEZ et al., 2016). Some authors prefer using clothoids curves due to their computation can be done from their curvature, not requiring a post processing strategy to handle curvature constraints and comfort limits. The use of clothoids in G^1 interpolation has been extensively studied (WALTON; MEEK, 2008; WALTON; MEEK, 2009; BERTOLAZZI; FREGO, 2012; CONNOR; KRIVODONOVA, 2014; BERTOLAZZI; FREGO, 2015). Commonly, clothoid parameters are evaluated solving either non-linear equations or non-linear systems by numeric methods employing

as few iteration steps as possible. Although fast time convergence algorithms have been developed so far, the resulting curvature at the assigned points can not be previously known or determined. Joining such curves might produce discontinuity in curvature enforcing the vehicle to stop in order to reorient its front wheels at each prescribed configuration.

A G^2 fitting approach using clothoids, circular arcs, and straight lines was given by Walton and Meek (2005). The resulting curves have always initial and final curvature equal to zero, allowing many of them to be connected with G^2 continuity. Control points are allocated in order to obtain the desired curve. Nevertheless, there is no control over the overall curvature (it is only known that it is zero at the link points). The maximum curvature in the path may exceed the one the vehicle can drive, turning the maneuver unfeasible. This problem was addressed by Fraichard and Scheuer (2004), who also considered the kinematic model of car-like robots. They proposed an improvement for continuous paths consisting of circular arcs and straight lines applied to forward and backward motions (REEDS; SHEPP, 1990 apud FRAICHARD; SCHEUER, 2004). Their approach uses clothoids with bounded curvature and sharpness to solve the problem of curvature discontinuity, turning the path feasible to be followed by car-like robots. Such paths are called *Continuous Curvature Paths*, or just *CC Paths*, made up of *CC Turns*, which are performed with bounded curvature, $|\kappa| \leq \kappa_{max}$, and sharpness, $|\sigma| = \sigma_{max}$, defined as inputs. Because of that, in many situations there is an excessive use of steering, which might decrease the passengers comfort and increase the use of steering wheel actuators. Wilde (2009) proposed that at each turn the maximum curvature and sharpness are computed based on what he named *clothoid trigonometry*. A circular arc is added whenever the curvature reaches κ_{max} . This approach ensures a minimal steering usage and it is exemplified by a lane change maneuver using four clothoids. However, the approach feasibility is only ensured in situations of symmetry between the initial and final configurations, as it is discussed in next chapter.

Lane changes can also be performed using two CC Turns as used by Girbés, Armesto and Tornero (2011) to create Double Continuous Curvature paths (DCC). Two configurations with start and end null curvature, respectively, can be connected. These two curves present positive and negative profiles of curvature and the final curve has a S-shape. Girbés, Armesto and Tornero (2014) extended DCC paths to the case of non-null initial and final curvatures. That increases the number of configurations that can be joined using only straight lines, circular arc, and clothoids. Both approaches require an optimization stage to find minimal length clothoids.

Not only regular intersections may be present on the vehicle roadway. Urban and rural scenarios may contain circular intersections with features that enforce speed reduction, the so-called roundabouts. Modern roundabouts are mainly characterized by yield-at-entry, one way traffic flow, path deflection and entry flare (FOULADVAND;

SADJADI; SHAEBANI, 2004). They can be seen as an alternative to common intersection regarding safety effectiveness (PERSAUD et al., 2001; GROSS et al., 2013). Besides that, the United States Department of Transportation¹ describes other benefits such as operation performance, traffic calm, access management and pedestrian safety. Therefore, planning feasible paths at roundabouts is an expected attribute for autonomous vehicles.

A path planning approach at roundabouts is given in (PÉREZ et al., 2011). Straight lines and circumferences are used to represent the navigable path. However, the path is no longer geometrical continuous at the point of the link between the straight lines and the circumference due to the abrupt change of the orientation angle, which is not allowed in car-like robots since non-holonomic constraints might be present (KATRAKAZAS et al., 2015). Smoother paths using Bézier curves were described in (PÉREZ et al., 2013) and practical results presented in (RASTELLI; PEÑAS, 2015). The approach ensures G^2 continuity, i.e. geometrical and curvature continuity, and the path can be parameterized by control points. The authors detail where to manually allocate them in order to achieve the desired path, but they do not mention the final curvature profile. Instead of using only one Bézier curve to perform the whole path, in (GONZÁLEZ; PÉREZ; MILANÉS, 2017) the path is divided into three maneuvers: a) entrance; b) circulatory roadway and c) exit. The control points of each maneuver are automatically generated taking into account roundabout constructive aspects. The final path is the coupling of the three maneuvers. An optimization stage ensures bounded curvature and smooth driving.

3.4 Final considerations

The purpose of this Master's dissertation is to compute feasible paths to be tracked by autonomous vehicles with non-holonomic constraints considering an urban scenario. It was not proposed in the works aforementioned how a structured environment can be represented in such a way to provide a continuous-curvature path towards the goal point. Furthermore, approaches to compute paths at intersections and roundabouts, and for lane changes rely on optimization to improve the comfort level, which increases the complexity of the problem since these paths are generally computed on-line. Thereby, this work proposes methods that can efficiently be applied to path generation considering an structured environment.

The first task is to obtain an efficient representation of the navigable environment regarded to data storage, accuracy, bounded curvature, and comfort. For the sparse lane representation, the method described by Lima et al. (2015a) showed to be a good option since clothoids are used for highways design, being a natural choice for representing the roadway geometry. This method will be used together with the road network model proposed by Zhang et al. (2016) with some adaptations to include roundabout parameters.

¹ <https://safety.fhwa.dot.gov/>

Paths computed after the route has been given must yield overall curvature continuity. Also, curvature rate change has to be smooth in order to improve comfort level, since it influences on jerk computation (see Section 2.4). This can be achieved using the clothoid trigonometry proposed by Wilde (2009) without any extra step to improve the comfort, such as optimization.

The major drawback of clothoids is that the Fresnel's integrals do not have analytical solution. The accuracy of approximations such as second order Taylor's expansion (BÉTAILLE; TOLEDO-MOREO, 2010) are very dependent on the integration step. Even small integration errors can be suffice to cause curvature discontinuities when two clothoids are linked. On the other hand, the use of small integration steps can increase the computation accuracy, but the computational burden increases as well, making clothoids unfeasible for real-time path planning applications. Nevertheless, approximations of clothoids that differ from the usual numerical integration methods and do not depend on the integration step have been proposed (HEALD, 1985; BREZAK; PETROVIC, 2014; VÁZQUEZ-MÉNDEZ; CASAL, 2016). In this work, the real-time bounded approximation of clothoids using lookup table described by Brezak and Petrovic (2014) is used. General clothoid coordinates are fast computed by reorienting and rescaling a basic clothoid stored in the lookup table. The basic clothoid is dependent on the algorithm bounded error input and it is controlled according to the desired accuracy.

4 INTERSECTIONS, LANE CHANGES AND ROUNDABOUTS

In order to reach its destination, the vehicle must deal with some elements along the way, such as intersection and roundabouts. Also, lane changes might be performed regarding traffic rules. This chapter presents path planners based on piecewise linear continuous-curvature curves at intersections, roundabouts and lane changes. Firstly, a formal definition of clothoids is introduced, as well as the requirements of their use in G^2 interpolation. After, the computation of clothoids applied to configuration changes with a minimal amount of steering is presented, named clothoid trigonometry (WILDE, 2009). Since this method only guarantees continuity when the two configurations are symmetrical regarding the straight line connecting them, this work proposes an extension to the case of non-symmetrical configuration changes. All three path planners presented here provide continuous and bounded curvature, according to the vehicle's physical limits and can be computed in real-time.

4.1 Clothoids

Clothoids are defined in the form of the Fresnel's integrals as (MCCRAE; SINGH, 2009)

$$x(\xi) = \pi B \cdot C(\xi) \quad (4.1a)$$

$$y(\xi) = \pi B \cdot S(\xi), \quad (4.1b)$$

where $B = 1/\sqrt{\pi\sigma}$ is the scaling factor with σ constant because of the curvature linearity, ξ is the non-negative arc length parameter and

$$C(\xi) = \int_0^\xi \cos\left(\frac{\pi}{2}\zeta^2\right) d\zeta \quad (4.1c)$$

$$S(\xi) = \int_0^\xi \sin\left(\frac{\pi}{2}\zeta^2\right) d\zeta \quad (4.1d)$$

are the Fresnel's integrals. σ is commonly called **sharpness** when dealing with clothoids (FRAICHARD; SCHEUER, 2004; BREZAK; PETROVIC, 2014). The curvature $\kappa(\xi)$ is given by

$$\kappa(\xi) = \frac{\xi}{B}. \quad (4.1e)$$

The initial tangent angle, curvature, and point coordinates are equal to zero. In order to change the initial coordinates and tangent angle, the clothoid must be translated and rotated, respectively. Also, $C(-\xi) = -C(\xi)$ and $S(-\xi) = -S(\xi)$. Nevertheless, clothoid coordinates x and y can be directly computed from initial parameters, arc length deviation

and sharpness by (BREZAK; PETROVIC, 2014)

$$x(s) = x_0 + \int_0^s \cos \left(\theta_0 + \kappa_0 \zeta + \frac{1}{2} \sigma \zeta^2 \right) d\zeta \quad (4.2a)$$

$$y(s) = y_0 + \int_0^s \sin \left(\theta_0 + \kappa_0 \zeta + \frac{1}{2} \sigma \zeta^2 \right) d\zeta \quad (4.2b)$$

with

$$\kappa(s) = \kappa_0 + \sigma s, \quad (4.2c)$$

$$\theta(s) = \theta_0 + \kappa_0 s + \frac{1}{2} \sigma s^2. \quad (4.2d)$$

where x_0 and y_0 are the initial coordinates and θ_0 and κ_0 are the initial tangent angle and curvature, respectively. Therefore, there is no need of translating and rotating each curve so that to obtain the desired initial parameters.

Figure 4.1 shows three clothoids with different σ . The clothoids have the same initial parameters (all of them equal to zero). As σ increases, higher values of curvature are achieved for the same length. If $\sigma = 0$ the curve is no longer a clothoid: when $\kappa_0 = 0$ it is a straight line; otherwise it is a circular arc segment. Hence, clothoids are used together with straight lines, and circular arcs to create curves whose curvature is piecewise linear.

Figure 4.1: Clothoids with different σ . Higher κ are reached for the same s as σ increases.

Source: Prepared by the author.

Equations (4.2) show that clothoids can be totally determined by six parameters or degrees of freedom: x_0 , y_0 , θ_0 , κ_0 , σ and s . However, considering a four-degree configuration change, the states sum up to eight. Therefore, the six parameters are insufficient to execute the configuration change. If κ_1 and κ_2 are neglected, there would be only six states to be satisfied, equaling the degrees of freedom of a unique clothoid. Thus, a single clothoid can be used in G^1 interpolation. G^2 interpolation can be accomplished joining clothoid segments, since only one clothoid does not provide the degrees of freedom required for G^2

interpolation (WALTON; MEEK, 2005). Figure 4.2 and 4.3 show examples of G^1 and G^2 interpolations, respectively, using clothoids.

Figure 4.2: Example of G^1 Hermite interpolation.

Source: Prepared by the author.

Figure 4.3: Example of G^2 Hermite interpolation.

Source: Prepared by the author.

4.1.1 Real-time approximation of clothoids

The Fresnel's integrals do not have analytical solution, therefore a numerical approach that provides acceptable accuracy and low computational burden must be used to evaluate clothoids.

The approach described by Brezak and Petrovic (2014) is used in this work. A lookup table generated from an accurate clothoid computation, which the authors call

“basic clothoid”, is used to perform the calculus of general clothoids. This can be achieved by rescaling, translating and rotating the basic clothoid. The number of points stored in the lookup table depends on the inputs that encompasses the typical scaling factor $C_{typ} = 1/\sqrt{\sigma_{typ}}$, the typical approximation error at typical values of scaling (e_{typ}), the maximum approximation error (e_{max}), the maximum length (s_{max}) and the maximum orientation change ($\Delta\theta_{max}$) of a single clothoid. The lookup table can be computed off-line, thus the computational cost at this stage is not a critical factor. To attenuate the problem of sampling error, the authors proposed an interpolation made of one circular arc segment between adjacent points.

Figure 4.4 shows a clothoid with $\sigma = -0.01$, and all initial values equal to zero. The clothoid computation is performed using both machine accuracy (10^{-15} m) and the lookup table method, which is computed from the inputs in Table 4.1. The clothoid length is 20

Table 4.1: Inputs used to compute the lookup table for the path in Figure 4.4.

$C_{typ}(\text{m})$	$e_{typ}(\text{m})$	$e_{max}(\text{m})$	$s_{max}(\text{m})$	$\Delta\theta_{max}(\text{rad})$
316.23	10^{-7}	10^{-4}	80	1.05

Source: Prepared by the author.

m and the arc length step size Δs is chosen as 0.5 m. It should be noted from Figure 4.4(c) that both accurate computation, and the lookup table method deviate from the assigned clothoid, as the Fresnel’s integrals do not have analytical solution. However, the error can be considered too small to change the linear behavior of the curvature and to cause discontinuities in the path.

4.2 Turning

All approaches proposed here require that the vehicle is able to plan right and left turns. Two turns are necessary to perform lane changes and, at roundabouts, up to three turns might be executed.

Clothoid segments developed by Wilde (2009) use only a minimal amount of steering to generate paths. This method takes as inputs the clothoid length along x axis x_{ct} and the deflection angle of the turn δ , as illustrated in Figure 4.5. With these parameters it is possible to compute the clothoid total length, its curvature and sharpness based on what the author called *clothoid trigonometry* defined by the “sine” and “cosine” functions

Figure 4.4: Clothoid computation using the lookup table method.

(a) Path.

(b) Curvature and sharpness.

(c) Sharpness error computation using machine accuracy, and the lookup table method.

Source: Prepared by the author.

bellow:

$$\sin_c \delta = \frac{y_{ct}}{L} = \begin{cases} (-\sin \delta \cdot C(\eta) + \cos \delta \cdot S(\eta))/\eta, & \delta > 0 \\ 0, & \delta = 0 \\ -\sin_c(-\delta), & \delta < 0 \end{cases} \quad (4.3)$$

$$\cos_c \delta = \frac{x_{ct}}{L} = \begin{cases} (\cos \delta \cdot C(\eta) + \sin \delta \cdot S(\eta))/\eta, & \delta > 0 \\ 1, & \delta = 0 \\ \cos_c(-\delta), & \delta < 0, \end{cases}$$

where C and S are the Fresnel's integrals described by (4.1c) and (4.1d), respectively, $\eta = \sqrt{2|\delta|\pi}$ and $|\delta| < \pi/2$. From (4.3),

$$L = \frac{x_{ct}}{\cos_c \delta} = \frac{y_{ct}}{\sin_c \delta}, \quad \kappa = \frac{2\delta}{L}, \quad \sigma = \frac{\kappa}{L}. \quad (4.4)$$

It may occur that κ exceeds the maximum curvature κ_{max} that the vehicle can drive.

Figure 4.5: Clothoid trigonometry.

Source: Prepared by the author and adapted from Wilde (2009).

Whenever that occurs, a circular arc whose radius is κ_{max}^{-1} is added to the curve. The deflection of the circular arc ϑ is computed solving $f(\vartheta) = 0$, where

$$f(\vartheta) = 2(\delta - \vartheta)(\cos_c(\delta - \vartheta) \cos \vartheta + \sin_c(\delta - \vartheta) \sin \vartheta) + \sin \vartheta - x_{ct}\kappa_{max}. \quad (4.5)$$

Equation (4.5) does not have analytical solution and ϑ is numerically evaluated by the Newton's recurrence as

$$\vartheta_{k+1} = \vartheta_k - \frac{f(\vartheta_k)}{f'(\vartheta_k)} = -\vartheta_k + 2\delta - \frac{x_{ct}\kappa_{max} - \sin \vartheta_k}{\cos_c(\delta - \vartheta_k) \cos \vartheta_k + \sin_c(\delta - \vartheta_k) \sin \vartheta_k}. \quad (4.6)$$

Hence,

$$L_{circ} = \frac{\vartheta}{\kappa_{max}}, \quad L_c = \frac{2(\delta - \vartheta)}{\kappa_{max}}, \quad \sigma = \frac{\kappa_{max}}{L_c}. \quad (4.7)$$

where L_{circ} is the circular arc length and L_c is the clothoid arc length. Although numerically evaluated, (4.6) quickly converges because $f(\vartheta_k)$ is nearly linear in the range 0 to δ (WILDE, 2009).

Suppose a curve described by $\delta = \pi/4$ rad and $x_{ct} = 10$ m, where $\mathbf{q}_s = [0 \quad 0 \quad \pi/4 \quad 0]^T$. Equation (4.3) and (4.4) yield $\kappa = 0.1321$, $\sigma = 0.0111$ and $L = 11.89$ m, as depicted in Figure 4.6.

G^2 continuity can be achieved when two symmetrical clothoids are used, as in Figure 4.7. The result is a curve with null curvature at the assigned configurations. In order to get a continuous curve, the clothoids must have the same longitudinal length and deflection. This equality is a constraint when using this method to generate continuous-curvature paths. In order to have null curvature at the given configurations, the computation of a minimal steering is only possible when there is symmetry between $\mathbf{q}_s$ and $\mathbf{q}_g$ regarding the straight line that passes through them. Considering that, this work proposes an approach to extend those turns to the cases of non-symmetrical configuration changes.

The deflection angles δ_1 and δ_2 of the first and second clothoids, respectively, are defined by $\mathbf{q}_s$ and $\mathbf{q}_g$ because they depend on the straight line passing through the two configurations. Analyzing Figure 4.8, it can be seen a discontinuity between the two

Figure 4.6: Clothoid path using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta = \pi/4$ rad.

Source: Prepared by the author.

Figure 4.7: Path made up of two symmetrical clothoids using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta = -\pi/4$ rad.

Source: Prepared by the author.

clothoids when $\delta_1 \neq \delta_2$. x_{ct} is the same for both clothoids, but they have different deflections in order to have the same tangent angle at the link point, yielding a discontinuous path.

Here, an approach to produce G^2 continuity curves even when $\delta_1 \neq \delta_2$ is proposed. x_{ct} is computed based on $\mathbf{q}_s$ and $\mathbf{q}_g$ as depicted in Figure 4.9, from which the following trigonometric relations can be established:

$$z = \sin(|\alpha_1| - \delta/2) \cdot h, \quad n = \frac{z}{\tan(\delta/2)}, \quad x_{ct} = \frac{z - n \cdot \tan(|\alpha_1| - \delta/2)}{2 \cdot \tan(|\alpha_1| - \delta/2)}. \quad (4.8)$$

In (4.8) it is assumed that $|\alpha_1| > |\alpha_2|$. When $|\alpha_2| > |\alpha_1|$, the argument $|\alpha_1| - \delta/2$ must be replaced by $|\alpha_2| - \delta/2$.

Consider the desired path described in Figure 4.8, where $\alpha_1 = -\pi/4$ rad and

Figure 4.8: Discontinuous path constituted of two clothoids using a minimal amount of steering. Inputs: $x_{ct} = 10$ m and $\delta_1 = -\pi/4$ rad (blue clothoid); $x_{ct} = 10$ m and $-\delta_2 = \pi/6$ rad (red clothoid).

Source: Prepared by the author.

Figure 4.9: Scheme depicting the trigonometric relation in order to compute x_{ct} .

Source: Prepared by the author.

$\alpha_2 = -\pi/6$. Equation (4.4) and (4.8) yield $x_{ct} = 8.21$ m, $\kappa = 0.14$ and $\sigma = 0.015$ and the planned path is shown in Figure 4.10.

4.3 Intersections

Consider the intersection shown in Figure 4.11. As it should be noted, three maneuvers are allowed for each entrance, totalizing twelve possible maneuvers. From 4.11(b) it can be seen that up to five curve segments are needed to build all the three

Figure 4.10: Path constituted of two linked clothoids using a minimal amount of steering with $\delta_1 \neq \delta_2$.

Source: Prepared by the author.

Figure 4.11: Three examples of possible maneuvers at a intersection. Each lane is 4 m wide and the maximum curvature allowed is $|0.2| m^{-1}$.

Source: Prepared by the author.

paths, which are computed based on the required configuration changes.

4.4 Lane change

Left and right turns can be connected to produce lane change paths. These curves take place when the configuration can not be done using only one turn. One of the contributions of this work is a non-parallel configuration change used in lane change paths composed by two turns with a minimal amount of steering. The join point, P_j , connecting the two opposite turns is computed according to Figure 4.12. The two turns have a minimal

Figure 4.12: Scheme depicting the computation of the lane change parameters.

Source: Prepared by the author.

amount of steering, which implies symmetrical configuration changes at each turn. For simplicity, P_j is assumed to lie on a vertex of a isosceles triangle, where q_s and q_g are the other vertexes. In order to accomplish the imposed requirements, the following linear system must be satisfied:

$$\begin{cases} 2\varphi_1 - 2\varphi_2 - |\alpha_1| + |\alpha_2| = 0 \\ \varphi_1 - |\alpha_1| = |\alpha_2| - \varphi_2. \end{cases} \quad (4.9)$$

The first equation is related to the sum of the triangle interior angles, which must be equal to π rad. The second one is the restriction to construct an isosceles triangle. Solving (4.9) yields

$$\varphi_1 = \frac{3|\alpha_1| + |\alpha_2|}{4}, \quad \varphi_2 = \frac{|\alpha_2| - |\alpha_1| + 2\varphi_1}{2}. \quad (4.10)$$

From the law of cosine it is possible to get the following system:

$$\begin{cases} b^2 = h^2 + b^2 - 2bh \cdot \cos(\varphi_1 - |\alpha_1|) \\ b^2 = h^2 + b^2 - 2bh \cdot \cos(|\alpha_2| - \varphi_2) \end{cases} \quad (4.11)$$

which yields

$$b = \frac{h}{\cos(\varphi_1 - |\alpha_1|) \cdot \cos(|\alpha_2| - \varphi_2)}. \quad (4.12)$$

φ_1 and b are used to determine P_j . φ_2 gives the tangent angle of the second turn.

Figure 4.13 shows a lane change for $\alpha_1 = 0.47$ rad, $\alpha_2 = -0.93$ rad and $h = 22.2$ m, whose curvature is depicted in Figure 4.13(b). These inputs produces $\varphi_1 = 0.58$ rad, $\varphi_2 = 0.81$ rad and $b = 11.2$ m.

Figure 4.13: Lane change. Inputs: $\alpha_1 = 0.47$ rad, $\alpha_2 = -0.93$ rad and $h = 22.2$ m.

Source: Prepared by the author.

Wilde (2009) gave an example of a lane change where start and goal configurations are parallel, as illustrated in Figure 4.14. The difference of the method proposed here is that different tangent angles for start and goal configurations are allowed (Figure 4.15).

4.5 Roundabout path planner

One of the contributions of this work is roundabout path planner using clothoids, circular arcs, and straight lines. Some requirements have to be accomplished in order to apply the proposed path planner:

- The curvature values at the start and goal configurations are null.
- The entrance, rotatory and exit paths are assumed to be collision-free.
- The traffic flow inside roundabouts is counter-clockwise.

Figure 4.14: Example of lane change with parallel initial and final positions.

Source: Prepared by the author.

Figure 4.15: Example of a lane change with non parallel initial and final positions.

Source: Prepared by the author.

Paths at roundabouts can be described by three maneuvers (GONZÁLEZ; PÉREZ; MILANÉS, 2017):

- **Entrance:** right turn used to make the vehicle merge into the roundabout. The curvature is bounded according to vehicle's kinematic constraints.
- **Rotatory:** circulatory roadway inside the roundabout. The maximum curvature considered is the inverse of the roundabout radius.
- **Exit:** right turn that leads the vehicle out of the roundabout. Also, the maximum curvature is the one that the vehicle can perform.

Here, the first exit is handled as a special case and it is discussed at the end of this section. The number and the kind of maneuvers might vary depending on the desired configurations and the roundabout parameters.

Figure 4.16 depicts the parameters used in the path planning. $\mathbf{q}_s$ and $\mathbf{q}_g$ are the

Figure 4.16: Scheme depicting the roundabout path planning approach.

Source: Prepared by the author and adapted from the Roundabouts Technical Summary of the United States Department of Transportation available on <https://safety.fhwa.dot.gov/>. Published in Silva and Grassi Jr. (2017).

given configurations at the roundabout entrance and exit, respectively. The aim is to find the configuration just before entering the circulatory roadway $\mathbf{q}_i = [x_i \ y_i \ \theta_i \ \kappa_i]^T$ and the one just after leaving it $\mathbf{q}_o = [x_o \ y_o \ \theta_o \ \kappa_o]^T$, where κ_i and κ_o are assumed to be null, as well as to compute minimal curvature derivatives used in the left turn calculation. Hence, the right turns ($\mathbf{q}_s$ to $\mathbf{q}_i$ and $\mathbf{q}_o$ to $\mathbf{q}_g$) can be computed using the approach described in Section 4.2, and the left turn can be calculated from the roundabout radius and the curvature derivatives.

The Cartesian coordinates x_i and y_i are computed using the informations about the roundabout center C_{rbt} , the roundabout radius r_{rbt} , the entrance radius offset value r_{offi} and the entrance angle λ_i . The use of r_{offi} moves $\mathbf{q}_i$ out of the circulatory roadway. It avoids that the radius of curvature varies from 0 to $1/r_{rbt}$ instantaneously, which would require infinite curvature derivative. Both r_{offi} and λ_i has an influence on entrance turns. If they are not properly chosen, the path planned may require a large amount of steering. These roundabout parameters are supposed to be known *a priori* and are passed as inputs to the path planner. The computation of the coordinates x_o and y_o follows the same

approach as for x_i and y_i , but using r_{off_o} and λ_o instead.

With those parameters defined, x_i and y_i are calculated as

$$\begin{aligned} d_i &= \cos \lambda_i (y_C - y_s) - \sin \lambda_i (x_C - x_s) \\ \rho_i &= \sin^{-1}(d_i / r_{rbt}) \\ x_i &= (r_{rbt} + r_{off_i}) \cdot \cos(\theta_s + \lambda_i + \rho_i) + x_C \\ y_i &= (r_{rbt} + r_{off_i}) \cdot \sin(\theta_s + \lambda_i + \rho_i) + y_C, \end{aligned} \quad (4.13)$$

where d_i is the distance between the two parallel straight lines oriented by θ_s passing through $\mathbf{q}_s$ and through C_{rbt} ; ρ_i is the angle that defines the triangle given by d_i and r_{rbt} ; and x_C and y_C are the coordinates of C_{rbt} .

The other state to be determined is θ_i . The aim here is to compute θ_i in such a way that the curvature rate change of the clothoid just before entering the circulatory roadway is the minimal one. To do so, θ_i is firstly approached by $\hat{\theta}_i$, which is the orientation angle of the straight line passing through $\mathbf{q}_i$ and tangent to the circulatory roadway at point P_i . This straight line can be computed as

$$\begin{cases} (x_{P_i} - x_C)^2 + (y_{P_i} - y_C)^2 = r_{rbt}^2 \\ y_{P_i} - y_i = m(x_{P_i} - x_i) \end{cases} \quad (4.14)$$

where x_{P_i} and y_{P_i} are the coordinates of P_i and m is the angular coefficient of the straight line. After some algebraic manipulation, (4.14) yields

$$\begin{aligned} m &= \frac{2\bar{x}_i\bar{y}_i \pm \sqrt{(2\bar{x}_i\bar{y}_i)^2 - 4(\bar{x}_i^2 - r_{rbt}^2)(\bar{y}_i^2 - r_{rbt}^2)}}{2(\bar{x}_i^2 - r_{rbt}^2)} \\ x_{P_i} &= \frac{-(2m\bar{y}_i - 2m^2\bar{x}_i)}{2(m^2 + 1)} + x_C, \quad y_{P_i} = m(x_{P_i} - \bar{x}_i) + y_i, \end{aligned} \quad (4.15)$$

where $\bar{x}_i = x_i - x_C$ and $\bar{y}_i = y_i - y_C$. There are two straight lines that are solutions for (4.14). As can be seen in Figure 4.16, the right choice for m is the one that results in a negative deflection (right turn) from $\mathbf{q}_s$ to $\mathbf{q}_i$. Thus, $\hat{\theta}_i = \tan^{-1} m$.

The curvature derivative used in the left turn can be computed by the clothoid trigonometry considering the scheme depicted in Figure 4.17. The forward distance traveled $x_{ct} = r_{rbt} + r_{off_i}$ is chosen so that one of the vertexes of the “clothoid triangle” defined by x_{ct} and r_{rbt} lies on the roundabout center C_{rbt} . In this way, δ_i can be computed in a straightforward manner from C_{rbt} , x_i , y_i and $\hat{\theta}_i$. The curvature derivative σ_i is determined considering $\kappa_{max} = 1/r_{rbt}$ in (4.6), (4.7) and (4.8). A new circumference centered at point A_{rbt} arises from σ_i and κ_{max} , deviating from the circular roadway centered at C_{rbt} . The point B_{rbt} , which is the end of the computed curve, is calculated from σ_i and κ_{max} using (4.1) and the circular arc deflection angle. Now, it is possible to compute the center point A_{rbt} of the new circumference considering the orientation of vector $C_{rbt} - B_{rbt}$ and the

Figure 4.17: Computation of β_i in order to reduce the error relative to the circulatory roadway.

Source: Prepared by the author and published in Silva and Grassi Jr. (2017).

magnitude $|B_{rbt} - A_{rbt}| = r_{rbt}$. Therefore, the new circumference is rotated around $\mathbf{q}_i$ so that the straight line passing through $\mathbf{q}_i$ and A_{rbt} , and the one passing through $\mathbf{q}_i$ and C_{rbt} get aligned. In this way, the rotation angle β_i , and θ_i are given by

$$\beta_i = \sin^{-1}(|A_{rbt} - C_{rbt}|/x_{ct}), \quad \theta_i = \hat{\theta}_i - \beta_i. \quad (4.16)$$

The rotation angle β_i does not ensure A_{rbt} and C_{rbt} to match, as the distance from $\mathbf{q}_i$ to A_{rbt} may be slightly different from the distance from $\mathbf{q}_i$ to C_{rbt} . Despite that, simulations showed errors less than 10 cm in the circulatory roadway even when long clothoids ($r_{off} \leq 4$ m) with great curvature ($\kappa_{max} \leq 0.15 \text{ m}^{-1}$) are computed, which can be considered good.

Determining σ_i is just one of the requirements to perform the left turn, since a circular arc with radius r_{rbt} may be necessary. Its deflection angle ϑ_{arc} , which is the total deflection from $\mathbf{q}_i$ to $\mathbf{q}_o$ minus the sum of the deflection of the two clothoids, is given by

$$\vartheta_{arc} = (\theta_o - \theta_i) - ((1/r_{rbt})^2/2\sigma_i + (1/r_{rbt})^2/2\sigma_o), \quad (4.17)$$

where θ_o and σ_o are computed using the same approach as for θ_i and σ_i and noting, by Figure 4.16, that $\theta_o = \hat{\theta}_o + \beta_o$. This approach does not guarantee that the end of the left turn coincides with $\mathbf{q}_o$ because of the error previously discussed. Therefore, instead of using x_o and y_o to compute the exit maneuver, approximate values $\hat{x}_o$ and $\hat{y}_o$ are calculated from the parameters of the left turn. If $\vartheta_{arc} \leq 0$, the whole maneuver can be done using a turn with a minimal amount of steering (Figure 4.9), as a circular arc is not required.

The first exit may require different approaches to compute the path. If $\theta_o > \theta_i$ the second maneuver is executed using a left turn with a minimal amount of steering. If this condition is not met, the maneuver is computed considering Figure 4.18, where r_{turn} is

Figure 4.18: Evaluation of the radius of curvature when the first exit is taken.

Source: Prepared by the author and published in Silva and Grassi Jr. (2017).

the curvature radius of the turn given by

$$r_{turn} = \frac{\sin \omega}{1 - \sin \omega} |IF|, \quad (4.18)$$

ω is the angle between IF and IH , $|IF|$ is the distance between the point belonging to the roundabout lane $F = [x_F \ y_F]^T$,

$$\begin{aligned} x_F &= r_{rbt} \cos \psi + x_C \\ y_F &= r_{rbt} \sin \psi + y_C, \end{aligned} \quad (4.19)$$

and the intersection point I between the straight lines passing through the two given configurations $\mathbf{q}_s$ and $\mathbf{q}_g$, where $\psi = (\theta_s + \theta_g + \lambda_i + \lambda_o + \rho_i + \rho_o)/2$. With $\kappa_{max} = 1/r_{turn}$, the right turn may be performed using two clothoids with different curvature derivative σ_1 and σ_2 computed by the clothoid trigonometry considering $\mathbf{q}_s$ and $\mathbf{q}_i$ in σ_1 computation, and $\mathbf{q}_o$ and $\mathbf{q}_g$ for σ_2 . Consider L_1 and L_2 as the lengths of the entrance and exit clothoids, respectively. Assuming small deflections for both clothoids and $\omega \approx \pi/4$, the forward distance traveled by both clothoids can be approached by L_1 and L_2 , according to (4.7). If $r_{turn} < |MI| - L_1$ and $r_{turn} < |IH| - L_2$ the right turn can be performed using σ_1 , σ_2 and $1/r_{turn}$, where

$$|MI| = |IH| = (|IF| + r_{turn}) \cos \omega. \quad (4.20)$$

Otherwise, the turn can not be done using only one right turn because there would not be enough space to be traveled by the clothoids until they reach κ_{max} . In this case, the unique right turn is replaced by two ones. The first turn is made considering $\mathbf{q}_s$ and

$\mathbf{q}_m = [x_m \ y_m \ \theta_m \ \kappa_m]^T$, where

$$\begin{aligned} x_m &= (r_{rbt} + (r_{off_i} + r_{off_o})/2) \cos \psi + x_C \\ y_m &= (r_{rbt} + (r_{off_i} + r_{off_o})/2) \sin \psi + y_C \\ \theta_m &= (\theta_i + \theta_o)/2, \quad \kappa_m = 0. \end{aligned} \quad (4.21)$$

The second turn takes $\mathbf{q}_m$ and $\mathbf{q}_g$ as inputs.

Simulations are performed in order to test the roundabout planner. Two double-lane roundabouts with different characteristics are considered. The entrance lanes in all of them are the ones at bottom right. All paths are plotted in thick black lines for all the four exits. Only the curvature and sharpness profiles of three paths are plotted, since they can properly represent the profiles of all seven paths.

The first roundabout is showed in Figure 4.19. In this scenario, each lane is 3 m wide with a 11 m radius for the external lane. All entrances/exits are double-lane, making

Figure 4.19: Simulations considering a 11 m radius for the external lane. All lanes are 3 m wide.

Source: Prepared by the author and published in Silva and Grassi Jr. (2017).

the roundabout geometrically uniform. The chosen parameters are $r_{off_i} = r_{off_o} = 0.3$ m and $\lambda_i = \lambda_o = 0.37$ rad for external lanes, and $r_{off_i} = r_{off_o} = 1$ m and $\lambda_i = \lambda_o = 0.43$ rad for internal lanes. Since all entrances/exits are assumed to have similar shapes, r_{off} and λ are the same in all paths.

The second roundabout is depicted in Figure 4.20. A 25 m external lane radius

Figure 4.20: Simulations considering a 25 m radius for the external lane. All lanes are 4 m wide.

Source: Prepared by the author and published in Silva and Grassi Jr. (2017).

is considered and the lanes are 4 m wide. The chosen parameters are $r_{off_i} = r_{off_o} = 0.6$ m, $\lambda_i = \lambda_o = 0.26$ rad (second and fourth exits), $r_{off_o} = 0.1$ m, $\lambda_o = 0.40$ rad (first and third exits) for external lanes. For internal lanes, $r_{off_i} = r_{off_o} = 1.1$ m, $\lambda_i = \lambda_o = 0.29$ rad (second and fourth exits), $r_{off_o} = 0.8$ m, $\lambda_o = 0.40$ rad (first and third exits). As it should be noted, the roundabout entrance is double-lane whereas the first and third exits are not. The paths do not have similar curvature and sharpness profiles considering entrance and exit maneuvers when the single-lane exits are taken. This can be seen in Figure 4.20(b) and 4.20(c). Also, the first exit is performed using two different right turns.

4.6 Final considerations

Paths with bounded curvature and low curvature rate change can be properly planned using clothoids, circular arc segments, and straight lines. Paths at intersections and lane changes require only the start and initial configurations for planning, while parameters describing entry/exit lanes, center and radius are required on the roundabout path computation. For all three approaches, the path is assumed to have null curvature at the start and end configurations.

5 GLOBAL PATH PLANNING

This chapter presents the lane-level road network modeling approach using clothoids. Roads, lanes, roundabouts and intersections are defined in function of parameters, making the model sparse. The road network is used to compute the best route from the vehicle's current position to a destination point regarded the shortest travel distance. As a result, a global continuous curvature path planning is achieved due to the connection between lanes by either intersections or roundabouts present in the computed route. Also, lane changes are performed to accomplish traffic rules. The construction of the model is made off-line, while the continuous-curvature global path planning is performed on-line considering the information in the road network model.

5.1 Piecewise linearization of the curvature

The number of points required to represent lanes geometry can be significantly reduced when parametric curve are used. In this work, clothoids are chosen for this purpose. The algorithm developed by Lima et al. (2015a) is used here to make the piecewise linearization of the curvature by optimization. The function to be minimized and the inequality constraints are

$$\begin{aligned} \min_{\boldsymbol{\kappa}} \quad & \|\mathbf{W} \odot \mathbf{D}\boldsymbol{\kappa}\|_1 \\ \text{s. t.} \quad & |\mathbf{x}(\boldsymbol{\kappa}) - \mathbf{x}_{raw}| \leq \boldsymbol{\epsilon} \\ & |\mathbf{y}(\boldsymbol{\kappa}) - \mathbf{y}_{raw}| \leq \boldsymbol{\epsilon}, \end{aligned} \tag{5.1}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the coordinate vectors of all points of the reconstructed curve and $\boldsymbol{\kappa}$ is the curvature vector. $\mathbf{x}_{raw}$ and $\mathbf{y}_{raw}$ are the coordinate vectors of all points of the original curve, $\mathbf{D}$ is the second order difference operator matrix, $\boldsymbol{\epsilon}$ is the bounded error vector, $\odot$ is the Hadamard product and $\mathbf{W}$ is the vector that contains weights that are initially set to 1 and updated at each iteration as

$$\mathbf{W}_{i+1} = \frac{1}{\mathbf{W}_i \odot \mathbf{D}\boldsymbol{\kappa}}, \tag{5.2}$$

performing a normalization so that the weights sum up to the number of samples. Thereby, the values of $\mathbf{D}\boldsymbol{\kappa}$ that are close to zero receive higher weights enforcing the second difference of the curvature to be zero in many points, performing a ℓ_0 -norm computation. A new clothoid starts at those points where $\mathbf{D}\boldsymbol{\kappa} \neq 0$, because of the discontinuity in the second order derivative of the curvature. In the end of the process, the curvature is constituted of linear pieces with respect to the arc length of the curve and the error between the original and new curves is bounded according to ϵ .

Roundabout and intersection paths always have null curvature at the start and end configurations. Hence, the minimization problem given in Lima et al. (2015a) has to be modified in order to make the curvature at the start and end of all lanes equal to 0 to achieve overall G^2 continuity. Therefore, two constraints are added to the minimization problem and (5.1) becomes

$$\begin{aligned}
& \min_{\boldsymbol{\kappa}} \quad \|\mathbf{W} \odot \mathbf{D}\boldsymbol{\kappa}\|_1 \\
& \text{s. t.} \quad |\mathbf{x}(\boldsymbol{\kappa}) - \mathbf{x}_{raw}| \leq \epsilon \\
& \quad \quad |\mathbf{y}(\boldsymbol{\kappa}) - \mathbf{y}_{raw}| \leq \epsilon \\
& \quad \quad \kappa_s = 0 \\
& \quad \quad \kappa_e = 0.
\end{aligned} \tag{5.3}$$

where κ_s and κ_e are the start and end curvatures, respectively.

Consider the path depicted in Figure 5.1. The goal is to make the curvature piecewise linear applying the method previously described. A 0.1 m maximum deviation error is chosen for this experiment. Figure 5.2 shows the result of the optimization process. The curvature has become piecewise linear, which is plotted in Figure 5.2(b).

Figure 5.1: Original path to be reconstructed by the optimization process.

Source: Prepared by the author.

5.2 Lane parametrization

The piecewise linearization of the curvature turns the path suitable to be parameterized. It is proposed in this work that the entire path can be reconstructed from its initial point coordinates (x_0, y_0) , tangent angle θ_0 , curvature κ_0 and from the subsequent sharpness σ and length L of each clothoid. The parametrization requires knowing the points in the curvature where a new clothoid starts. It can be accomplished analyzing $\mathbf{W}$. It is implicit in (5.2) that $\mathbf{W}$ has values smaller than one at those points where there is a

Figure 5.2: Reconstructed path resulting from the optimization process.

Source: Prepared by the author.

sharpness transition. The indexes of the samples where it occurs can be used to determine where a new clothoid initiates.

Figure 5.3 illustrates two clothoid segments and their parameters. As discussed in Section 4.1, there must be six parameters in order to compute a clothoid. However, it should be noted that there are no explicit initial parameters for the second clothoid in Figure 5.3 because they are computed at the beginning of each new segment. Hence, considering the path in Figure 5.3 the subsequent parameters are computed using (4.2) as

$$\begin{aligned}
 x &= x_0 + \int_0^{L_1} \cos \left(\theta_0 + \kappa_0 s + \frac{1}{2} \sigma_1 s^2 \right) ds, \\
 y &= y_0 + \int_0^{L_1} \sin \left(\theta_0 + \kappa_0 s + \frac{1}{2} \sigma_1 s^2 \right) ds, \\
 \kappa &= \kappa_0 + \sigma_1 L_1, \quad \theta = \theta_0 + \kappa_0 L_1 + \frac{1}{2} \sigma_1 L_1^2.
 \end{aligned} \tag{5.4}$$

One issue arises when defining θ_0 , as illustrated in Figure 5.4. The distance between adjacent points, Δs , influences on the accuracy of θ_0 . The approach used here is to compute the clothoid-composed curve by the Fresnel's integrals assuming a null initial tangent

Figure 5.3: Parameters used to reconstruct the clothoids.

Source: Prepared by the author.

angle. This curve is rotate by an amount of θ_o so that to match the curve resulting from the optimization stage. The parameters of the path plotted in Figure 5.2 can be seen in Table 5.1 and 5.2.

Figure 5.4: Error in θ_o caused by the distance Δs between the samples.

Source: Prepared by the author.

Table 5.1: Initial parameters of the path depicted in Figure 5.2.

$x_0(\text{m})$	$y_0(\text{m})$	$\theta_0(\text{rad})$	$\kappa_0(1/\text{m})$
30.0249	30.0911	1.6666	0

Source: Prepared by the author.

Table 5.2: Clothoids sharpness and length of the path depicted in Figure 5.2.

Clothoid	$\sigma(1/m^2)$	$L(m)$
1	-0.0035	5.00
2	-0.0171	10.00
3	0.0186	9.50
4	0.0017	10.50
5	0.0017	7.00
6	0.0094	12.00
7	-0.0096	10.00
8	-0.0025	13.50

Source: Prepared by the author.

5.3 Lane-level road network model

A road network model aims to satisfactorily represent the structured environment the vehicle is inserted in. The model should provide information about roads connectivity, lengths and geometry; roundabout positions; lanes belonging to the same road, etc. A representative example of a road network is depicted in Figure 5.5. Three elements are

Figure 5.5: Representative example of a road network representation. Roads are separated by solid lines, whereas dashed lines separate lanes.

Source: Prepared by the author.

considered in the topological representation:

- **Lanes:** are strips in the roads which can accommodate a single line of vehicles.
- **Intersections:** are junctions where two or more lanes meet or cross.
- **Roundabouts:** like intersections, are used to join lanes and are characterized by a circular roadway that enforces speed reduction.

In this Master Dissertation it is proposed a clothoid-based model used to calculate piecewise continuous-curvature curves, in such a way to guarantee G^2 continuity in the global path planning. The physical road network $\mathbf{Z}$ is given by

$$\mathbf{Z} = (\mathbf{C}, \mathbf{R}),$$

where $\mathbf{C}$ is the set of all roundabouts $\{\mathbf{c}_i\}_{i=1}^{N_c}$ and $\mathbf{R}$ is the set of roads $\{\mathbf{r}_i\}_{i=1}^{N_r}$. The road $\mathbf{r}$ is the set of lanes belonging to same carriageway

$$\mathbf{r} = (\mathbf{T}, u),$$

where $\mathbf{T}$ is the set of lanes $\{\gamma_i\}_{i=1}^{N_\gamma}$ and u indicates which roundabout the road ends at. If the road ends at an intersection, u is set as 0. The lane γ is defined as

$$\gamma = (\mathbf{S}_\gamma, w, \lambda_i, \lambda_o, r_{off_i}, r_{off_o}, \mathbf{q}_s, \mathbf{q}_e),$$

where $\mathbf{S}_\gamma$ is the set of sharpnesses and lengths describing all clothoids that compose the lane (see Section 5.2),

$$\mathbf{S}_\gamma = \begin{bmatrix} \sigma_1 & L_1 \\ \sigma_2 & L_2 \\ \vdots & \vdots \\ \sigma_k & L_k \end{bmatrix},$$

where k is the number of segments in the lane. w is the relative position of the lane inside the road (the number 1 is assigned to leftmost lanes by convention (BÉTAILLE; TOLEDO-MOREO, 2010)), and λ_i , λ_o , r_{off_i} and r_{off_o} are parameters used in the roundabout path planning (see Section 4.5). These parameters are set to 0 when the lane ends at an intersection. $\mathbf{q}_s$ and $\mathbf{q}_e$ are the path configurations at the start and end of the lane, respectively.

It is considered that roads can be connected by roundabouts and intersections. In order to compute a route between start and destination points, a connection matrix is defined. Each road must end either at a roundabout or at an intersection, which are considered as connection entities.

5.4 Route computation

The route is determined between a start point P_s and a goal point P_g on the map. The first task is to determine which lanes these points belong. This is simply done considering the distance between the assigned points to all lanes on the map. The closest lane to P_s is chosen as the start lane. The same follows for P_g .

After that, the points must be localized inside the lanes. Figure 5.6 depicts that P_s and P_g are localized with respect to s inside the initial and destination lane, respectively. The planned path starts at s_s and ends s_e , considering the start and destination lane, respectively (see Figure 5.6(a) and 5.6(b)).

Figure 5.6: Curvature profiles of initial and destination lanes.

Source: Prepared by the author.

It is possible to compute a route from P_s to P_g making use of the connection between roads and considering each road as a node inside a graph. This can be achieved using Dijkstra's algorithm (DIJKSTRA, 1959) and considering that there is a cost associated to each road to be traveled in order to compute the shortest cost path. In this work, the cost is related to the road length. The graph in Figure 5.7 shows an example of a road network with four roads. As it can be seen, the cost to move from $\mathbf{r}_B$ to $\mathbf{r}_C$ is the same to move

Figure 5.7: Example of a graph representing connections between roads.

Source: Prepared by the author.

from $\mathbf{r}_B$ to $\mathbf{r}_D$ because the length of $\mathbf{r}_B$ remains the same regardless the destination road. When P_s is not at the start point of the lane, which is the general situation, the cost is simply changed considering the distance from P_s to the end point of the lane.

Two situations arise when the start and destination roads are the same:

- $s_s < s_e$: the Dijkstra's algorithm returns the assigned road, since the cost of moving from a road to itself is null.
- $s_s > s_e$: the assigned road is divided into two different nodes, which are not connected between them. This allows the search of a path that connects P_s to P_g taking into account the other roads on the road network.

5.5 Lane change path

The vehicle might perform lane changes in order to reach its destination, as it is exemplified in Figure 5.8. In this work, it is assumed that lane changes are required in the following scenarios:

- The vehicle must be on the rightmost lane when performing right turns at intersections.
- The vehicle must be on the leftmost lane when performing left turns at intersections.
- The vehicle must be on the rightmost lane when taking the first and second exits at roundabouts.
- The vehicle must be on the leftmost lane when taking the third and fourth exits at roundabouts.

The approach used to compute lane change paths is constrained to have null curvature at the start and end points (see Section 4.4). A good choice to initiate the lane change maneuver is at the start point of the departure lane, since the curvature is null at that point. The next step is to compute a lane change with an assigned longitudinal distance. Finally, it must be connected to the destination lane without curvature discontinuity.

Figure 5.9 depicts how the complete lane change is performed. After the lane change path has been computed, it must be connected to the destination lane. The point where the curvature is null after the longitudinal distance s_{root} is chosen to be the point of link between the calculated path and the destination lane. Since s_{root} is ahead the lane change, a connection path is added in order to ensure the curvature continuity. The whole maneuver could be computed using only a non-parallel lane change path. However, in that way, the maneuver longitudinal distance could not be determined *a priori* and would be strictly dependent on the chosen s_{root} .

Figure 5.8: Required lane changes.

Source: Prepared by the author.

Figure 5.9: Scheme depicting the lane change path planning.

Source: Prepared by the author.

6 RESULTS

CaRINA II (FERNANDES et al., 2014), a probe-vehicle equipped with a GPS-RTK (Real Time Kinematic) with a 4 cm maximum deviation, was used to acquire data information on USP - São Carlos, Campus 2, as the driver tried to keep the vehicle in the lane center. CaRINA II is equipped with other sensors such as LIDARS (laser sensor), monocular stereo-spherical-panoramic cameras, Compass, IMU, etc. and actuators that enable the vehicle navigates in autonomous mode and it has been used as an outdoor vehicle platform (Figure 6.1).

Figure 6.1: CaRINAII: Intelligent Robotic Car for Autonomous Navigation.

Source: Laboratory of Mobile Robotics of USP - São Carlos. Available on <http://lrm.icmc.usp.br>.

6.1 Road network model

Since straight and curved lanes, intersections, roundabouts, and double-lane roads are present, most of the attributes of the global planner can be verified. The resulting map can be seen in Figure 6.2. The beginning and end of each lane were manually defined based on the positions of roundabouts and intersections. The resulting road network model is composed by 15 roads, 21 lanes, 1 intersection and 5 roundabouts, where 6 roads are double-lane. 16435 GPS points were acquired in the mapping process, so summing up to 32870 points coordinates. The amount of data was reduced to 863 values after the parametrization of lanes and roundabouts, where:

- **Lane geometry S_γ :** 282 clothoids were required to represent all lanes with a 0.1 m tolerance, summing up to 564 values (σ and s).

Figure 6.2: Road network considering a real map. The solid lines represent the center of the lanes. Two adjacent green dots define a clothoid segment.

Source: Prepared by the author.

- $\mathbf{q}_s$ and $\mathbf{q}_e$: The start and end configurations of each lane require a total of 8 parameters to be stored, since each configuration has four states. Since there are 21 lanes, the states sum up to 168.
- w : 21 values (one for each lane).
- u : 15 values (one for each road).
- λ_i , λ_o , r_{off_i} and r_{off_o} : 4 values for each lane, summing up to 84 values.
- **Roundabout parameters**: 5 roundabout centers, summing up to 10 values (x and y coordinates), and 10 roundabout radii, since all roundabouts are double-lane.

In this way, there is a reduction of 38 times in the storage data. Despite that, the path can be computed in such a way to ensure smoothness and continuous curvature.

6.2 Intersections

The three lanes in Figure 6.3 can be connected by intersection paths, as it is described in Section 4.3. The results are depicted in Figure 6.4, where a 0.2 m^{-1} maximum curvature is assumed. The path connecting lane 1 to lane 3 can not be computed using a single straight line because the configurations considered in the planning are not aligned. The curves proposed in Section 4.4 are used instead.

Figure 6.3: Intersection connecting three lanes in the road network model.

Source: Prepared by the author.

6.3 Lane change path

A lane change maneuver is planned considering the two lanes in Figure 6.5. The

Figure 6.4: Intersection path planning.

Source: Prepared by the author.

Figure 6.5: Two lanes belonging to the same road in the road network model.

Source: Prepared by the author.

approaches presented in Section 4.4 and 5.5 are applied here to perform the path planning. The resulting paths are depicted in Figure 6.6, where three longitudinal distances are used: 30 m, 65 m and 100 m.

Figure 6.6: Lane change path planning considering tree longitudinal distances: 30 m, 65 m and 100 m.

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

6.4 Roundabouts

Five roundabouts are present on the chosen scenario. Roundabouts centers and radii were evaluated considering the trajectory performed by the driver. The parameters λ and r_{off} were manually taken according to Section 4.5, while attempting to produce small values of curvature derivative. Roundabout 4 is chosen to show resulting paths at real roundabouts, since it is the one that has the greater number of entrance/exits.

Figure 6.7 depicts the resulting paths when the first, second and third exits are taken traveling along the external lane, where the top-left road is the roundabout entrance

for this case. The internal lane paths of the same road are showed in Figure 6.8, where the second, third and fourth exits are taken.

Figure 6.7: Roundabout path planning. The entrance road is double-lane and the external lane inside the roundabout is taken.

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

Figure 6.8: Roundabout path planning. The entrance road is double-lane and the internal lane inside the roundabout is taken.

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

The previous case considered a double-lane departure road. A single-lane departure road is chosen as the roundabout entrance in Figure 6.9 and 6.10, where the paths are

planned considering the external lane, and internal lane, respectively. The parameters λ

Figure 6.9: Roundabout path planning. The entrance road is single-lane and the external lane inside the roundabout is taken.

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

Figure 6.10: Roundabout path planning. The entrance road is single-lane and the internal lane inside the roundabout is taken.

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

and r_{off} may be readjusted to improve the comfort level of the maneuver when a single-lane road precedes a double-lane roundabout. That is what occurs in the previous plotted path: λ and r_{off} were adjusted considering the roundabout external lane and not the

internal one. To handle this situation, r_{off} is increased by an amount of 0.8 m, empirically determined, as λ remains the same. This approach is also used when a single-lane road is taken as exit in a double-lane roundabout. The resulting paths can be seen in Figure 6.11. It should be noted a reduction in the curvature derivatives compared to the paths in

Figure 6.11: Roundabout path planning. Increase of r_{off} by an amount of 0.8 m in the roundabout planning when the entrance road is single-lane and the internal lane inside the roundabout is taken.

Source: Prepared by the author.

Figure 6.10.

6.5 Global path planning

Three different examples of routes are considered in the path planning, as shown in Figure 6.12. It is considered a 0.2 m^{-1} maximum absolute curvature for the vehicle and lane changes are performed considering a 30 m longitudinal distance. The same current location of the vehicle is assumed in all planned routes.

The road network and the path planning approach were implemented using *Python* language in a common personal computer (Intel Core I7 3770 3.4GHz). The elapsed times,

in seconds, after running the algorithm for different Δs in the path interpolation are shown

Figure 6.12: Global path planning. Three paths are planned from the current position to a destination point: 1 (path 1), 2 (path 2) and 3 (path 3).

(a) Path.

(b) Curvature.

(c) Curvature derivative.

Source: Prepared by the author.

in Table 6.1. The longer step in the path computation is due to the search to find where

Table 6.1: Elapsed times assuming different Δs in the path interpolation.

	$\Delta s = 0.1$ m	$\Delta s = 0.2$ m	$\Delta s = 0.5$ m	$\Delta s = 1$ m	$\Delta s = 2$ m
Path 1	1.94 sec	1.30 sec	0.914 sec	0.782 sec	0.718 sec
Path 2	1.59 sec	1.10 sec	0.807 sec	0.699 sec	0.630 sec
Path 3	1.76 sec	1.17 sec	0.822 sec	0.706 sec	0.647 sec

Source: Prepared by the author.

the current and destination points are on the map (about 0.605 sec). The assigned lanes must be found as well as the distance traveled from the start of the lanes to the prescribed points (see Section 5.4). Beside that, the total elapsed time increases in function of the distance between points required for interpolation and path total length. Since the path

total length is determined by the chosen route, shorter time computation can be achieved from longer arc length step size. Therefore, there is a trade-off between computation time and smoothness of the final path. The results can be considered good, since the route planning is made before the vehicle initiates the travel ($v = 0$), characterizing a non-risky situation. Moreover, the path can be computed as the vehicle progresses towards the goal point, not necessarily requiring that the whole path be calculated at once. In fact, assuming more complex scenarios like big cities, the time spent to compute the route may become too long if the global planning approach was applied directly as in Figure 6.12.

The curvature profile depicted in Figure 6.13(b) is bounded to the assigned maximum value that the vehicle can drive. The low rate curvature change is associated with passenger's comfort. Since there is no abrupt change in the curvature, it is expected that the vehicle steering wheel angle will smoothly vary during the path tracking.

7 CONCLUSION

In this work, a global path planning for autonomous vehicles considering an urban scenario was presented. Experimental results showed that the proposed approach was able to accomplish the objectives of this work. The following remarks discuss relevant topics covered in this work as well as gives directions for future work:

- **Road network model:** The amount of data required to represent lanes was significantly reduced with the sparse representation. Clothoids replaced GPS points while providing continuous curvature curves within a prescribed deviation error. A road network model gives information about lanes, roundabouts and roads that is posteriorly used in the path planning. The process to extract the road network model from acquired GPS points was manually done. This task can take too long time if more complex scenarios are considered. Hence, an approach to automatically determine GPS points belonging either to intersections (XIE et al., 2016), lanes or roundabouts (ZOCOLER et al., 2015) has to be developed, as well as the automatic generation of the connection matrix considering the traffic flow.
- **Intersections:** Turns at intersections were made using a minimal amount of steering when performing the path with only symmetrical clothoids, arc segments, and straight lines. The advantage is that the planner needs only the initial and goal configurations in order to generate the path without requiring other parameters.
- **Lane changes:** Lanes belonging to the same road were automatically connected by a lane change maneuver. The planner uses the information about the two lanes and the longitudinal traveled distance of the whole maneuver. The drawback of this approach is that only configurations with null curvature can be connected with G^2 continuity. However, as this work considered that lanes are free from obstacles, lane changes performed at the beginning of the lane suffices to lead the vehicle to its destination. For scenarios with other participants such as pedestrians, bicyclists, and other vehicles, approaches that considered non-null curvature at the start and end of the path must be considered (GIRBÉS; ARMESTO; TORNERO, 2014).
- **Roundabouts:** The parametric-based roundabout planner was able to generate paths at symmetrical and asymmetrical roundabouts connecting two assigned configurations. The parameters required in the planning are manually extracted considering the geometry of entrance/exit lanes as attempting to increase comfort. The task of manually evaluating these parameters may be tedious and very time consuming on scenario with a large number of roundabouts. Therefore, an intelligent method of automatically determining these parameters could turn this task easier and fast.

- **Global path planning:** Two assigned points could be connected using piecewise continuous-curvature curves. The planner uses a road network model to compute lanes, intersections, roundabouts and lane change paths. The Dijkstra's algorithm fast generated a minimal length route from a directed graph. Nevertheless, the Dijkstra's algorithm can become unfeasible if the graph is constituted of too many nodes, requiring the use of better approaches (PADEN et al., 2016). Furthermore, processing time can be improved by an efficient approach to localize the start and destination point inside the road network model, which was showed to be the longer step in the path computation.

The path tracking at comfort limit can be performed choosing proper speeds for navigation (VILLAGRA et al., 2012; LIMA et al., 2015b), considering the curvature derivative to produce bounded lateral jerk. Also, a controller that satisfactorily deal with the tracking problem has to be implemented (LIMA et al., 2015a). This work can encounter great applicability in door-to-door assistance provided by autonomous vehicle, like delivering and taxi services.

BIBLIOGRAPHY

- AASHTO, A. Policy on geometric design of highways and streets. **American Association of State Highway and Transportation Officials**, Washington, DC, 2001. v. 1, n. 990, p. 158, 2001.
- ABRAMOWITZ, M.; STEGUN, I. A. **Handbook of mathematical functions: with formulas, graphs, and mathematical tables.** : Courier Corporation, 1964.
- ALIA, C.; GILLES, T.; REINE, T.; ALI, C. Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method. In: IEEE. **Intelligent Vehicles Symposium (IV)**, 2015 IEEE. 2015. p. 674–679.
- BAASS, K. **Use of clothoid templates in highway design.** 1982.
- BERTOLAZZI, E.; FREGO, M. Fast and accurate clothoid fitting. **arXiv preprint arXiv:1209.0910**, 2012. 2012.
- _____. G1 fitting with clothoids. **Mathematical Methods in the Applied Sciences**, 2015. Wiley Online Library, v. 38, n. 5, p. 881–897, 2015.
- BÉTAILLE, D.; TOLEDO-MOREO, R. Creating enhanced maps for lane-level vehicle navigation. **IEEE Transactions on Intelligent Transportation Systems**, 2010. IEEE, v. 11, n. 4, p. 786–798, 2010.
- BREZAK, M.; PETROVIC, I. Real-time approximation of clothoids with bounded error for path planning applications. **IEEE Transactions on Robotics**, 2014. IEEE, v. 30, n. 2, p. 507–515, 2014.
- BUEHLER, M.; IAGNEMMA, K.; SINGH, S. **The 2005 DARPA grand challenge: the great robot race.** : Springer Science & Business Media, 2007.
- CONNOR, D.; KRIVODONOVA, L. Interpolation of two-dimensional curves with euler spirals. **Journal of Computational and Applied Mathematics**, 2014. Elsevier, v. 261, p. 320–332, 2014.
- DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische mathematik**, 1959. Springer, v. 1, n. 1, p. 269–271, 1959.
- DUBINS, L. E. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. **American Journal of mathematics**, 1957. JSTOR, v. 79, n. 3, p. 497–516, 1957.
- DUDEK, G.; JENKIN, M. **Computational Principles of Mobile Robotics.** 2nd. ed. New York, NY, USA: Cambridge University Press, 2010. ISBN 0521871573, 9780521871570.
- FERNANDES, L. C.; SOUZA, J. R.; PESSIN, G.; SHINZATO, P. Y.; SALES, D.; MENDES, C.; PRADO, M.; KLASER, R.; MAGALHÃES, A. C.; HATA, A.; PIGATTO, D.; BRANCO, K. C.; GRASSI Jr., V.; OSORIO, F. S.; WOLF, D. F. Carina intelligent robotic car: architectural design and applications. **Journal of Systems Architecture**, 2014. Elsevier, v. 60, n. 4, p. 372–392, 2014.

- FERNÁNDEZ, C.; DOMÍNGUEZ, R.; FERNÁNDEZ-LLORCA, D.; ALONSO, J.; SOTELO, M. A. Autonomous navigation and obstacle avoidance of a micro-bus. **International Journal of Advanced Robotic Systems**, 2013. SAGE Publications Sage UK: London, England, v. 10, n. 4, p. 212, 2013.
- FOULADVAND, M. E.; SADJADI, Z.; SHAEBANI, M. R. Characteristics of vehicular traffic flow at a roundabout. **Physical Review E**, 2004. APS, v. 70, n. 4, p. 046132, 2004.
- FRAICHARD, T.; SCHEUER, A. From reeds and shepp's to continuous-curvature paths. **IEEE Transactions on Robotics**, 2004. IEEE, v. 20, n. 6, p. 1025–1035, 2004.
- GIRBÉS, V.; ARMESTO, L.; TORNERO, J. On generating continuous-curvature paths for line following problem with curvature and sharpness constraints. In: IEEE. **Robotics and Automation (ICRA), 2011 IEEE International Conference on**. 2011. p. 6156–6161.
- _____. Path following hybrid control for vehicle stability applied to industrial forklifts. **Robotics and Autonomous Systems**, 2014. Elsevier, v. 62, n. 6, p. 910–922, 2014.
- GONZÁLEZ, D.; PÉREZ, J.; MILANÉS, V. Parametric-based path generation for automated vehicles at roundabouts. **Expert Systems with Applications**, 2017. Elsevier, v. 71, p. 332–341, 2017.
- GONZÁLEZ, D.; PÉREZ, J.; MILANÉS, V.; NASHASHIBI, F. A review of motion planning techniques for automated vehicles. **IEEE Transactions on Intelligent Transportation Systems**, 2016. IEEE, v. 17, n. 4, p. 1135–1145, 2016.
- GROSS, F.; LYON, C.; PERSAUD, B.; SRINIVASAN, R. Safety effectiveness of converting signalized intersections to roundabouts. **Accident Analysis & Prevention**, 2013. Elsevier, v. 50, p. 234–241, 2013.
- HEALD, M. A. Rational approximations for the fresnel integrals. **Mathematics of Computation**, 1985. v. 44, n. 170, p. 459–461, 1985.
- JO, K.; SUNWOO, M. Generation of a precise roadway map for autonomous cars. **IEEE Transactions on Intelligent Transportation Systems**, 2014. IEEE, v. 15, n. 3, p. 925–937, 2014.
- KATRAKAZAS, C.; QUDDUS, M.; CHEN, W.-H.; DEKA, L. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. **Transportation Research Part C: Emerging Technologies**, 2015. Elsevier, v. 60, p. 416–442, 2015.
- LI, C.; LI, X.; LI, J.; ZHU, Q.; DAI, B. Trajectory planning for autonomous ground vehicles driving in structured environments. In: IEEE. **Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2017 9th International Conference on**. 2017. v. 2, p. 41–46.
- LIMA, P. F.; TRINCAVELLI, M.; MÅRTENSSON, J.; WAHLBERG, B. Clothoid-based model predictive control for autonomous driving. In: IEEE. **Control Conference (ECC), 2015 European**. 2015. p. 2983–2990.

_____. Clothoid-based speed profiler and control for autonomous driving. In: IEEE. **Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on**. 2015. p. 2194–2199.

LITMAN, T. Autonomous vehicle implementation predictions. **Victoria Transport Policy Institute**, 2014. v. 28, 2014.

MCCRAE, J.; SINGH, K. Sketching piecewise clothoid curves. **Computers & Graphics**, 2009. Elsevier, v. 33, n. 4, p. 452–461, 2009.

MONTEMERLO, M.; BECKER, J.; BHAT, S.; DAHLKAMP, H.; DOLGOV, D.; ETTINGER, S.; HAEHNEL, D.; HILDEN, T.; HOFFMANN, G.; HUHNKE, B. et al. Junior: The stanford entry in the urban challenge. **Journal of field Robotics**, 2008. Wiley Online Library, v. 25, n. 9, p. 569–597, 2008.

MULLINEUX, G.; ROBINSON, S. T. Fairing point sets using curvature. **Computer-Aided Design**, 2007. Elsevier, v. 39, n. 1, p. 27–34, 2007.

OBAYASHI, M.; UTO, K.; TAKANO, G. Appropriate overtaking motion generating method using predictive control with suitable car dynamics. In: IEEE. **Decision and Control (CDC), 2016 IEEE 55th Conference on**. 2016. p. 4992–4997.

PADEN, B.; ČÁP, M.; YONG, S. Z.; YERSHOV, D.; FRAZZOLI, E. A survey of motion planning and control techniques for self-driving urban vehicles. **IEEE Transactions on Intelligent Vehicles**, 2016. IEEE, v. 1, n. 1, p. 33–55, 2016.

PARENT, M. Advanced urban transport: Automation is on the way. **IEEE Intelligent Systems**, 2007. v. 22, n. 2, p. 9–11, March 2007. ISSN 1541-1672.

PÉREZ, J.; GODOY, J.; VILLAGRÁ, J.; ONIEVA, E. Trajectory generator for autonomous vehicles in urban environments. In: IEEE. **Robotics and Automation (ICRA), 2013 IEEE International Conference on**. 2013. p. 409–414.

PÉREZ, J.; MILANÉS, V.; PEDRO, T. D.; VLACIC, L. Autonomous driving manoeuvres in urban road traffic environment: a study on roundabouts. **IFAC Proceedings Volumes**, 2011. Elsevier, v. 44, n. 1, p. 13795–13800, 2011.

PERSAUD, B.; RETTING, R.; GARDER, P.; LORD, D. Safety effect of roundabout conversions in the united states: Empirical bayes observational before-after study. **Transportation Research Record: Journal of the Transportation Research Board**, 2001. Transportation Research Board of the National Academies, n. 1751, p. 1–8, 2001.

RASTELLI, J. P.; LATTARULO, R.; NASHASHIBI, F. Dynamic trajectory generation using continuous-curvature algorithms for door to door assistance vehicles. In: IEEE. **Intelligent Vehicles Symposium Proceedings, 2014 IEEE**. 2014. p. 510–515.

RASTELLI, J. P.; PEÑAS, M. S. Fuzzy logic steering control of autonomous vehicles inside roundabouts. **Applied Soft Computing**, 2015. Elsevier, v. 35, p. 662–669, 2015.

REEDS, J.; SHEPP, L. Optimal paths for a car that goes both forwards and backwards. **Pacific journal of mathematics**, 1990. Mathematical Sciences Publishers, v. 145, n. 2, p. 367–393, 1990.

SHIN, D. H.; SINGH, S. **Path generation for robot vehicles using composite clothoid segments**. 1990.

SILVA, J. A.; GRASSI Jr., V. Path planning at roundabouts using piecewise linear continuous curvature curves. In: IEEE. **Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR), 2017 Latin American**. 2017. p. 1–6.

URMSON JOSHUA ANHALT, D. B. C. B. a. M. B. K. I. S. S. e. C. **The DARPA Urban Challenge: Autonomous Vehicles in City Traffic**. 1. ed. : Springer-Verlag Berlin Heidelberg, 2009. (Springer Tracts in Advanced Robotics 56). ISBN 3642039901,9783642039904.

VÁZQUEZ-MÉNDEZ, M. E.; CASAL, G. The clothoid computation: a simple and efficient numerical algorithm. **Journal of Surveying Engineering**, 2016. American Society of Civil Engineers, v. 142, n. 3, p. 04016005, 2016.

VILLAGRA, J.; MILANÉS, V.; PÉREZ, J.; GODOY, J. Smooth path and speed planning for an automated public transport vehicle. **Robotics and Autonomous Systems**, 2012. Elsevier, v. 60, n. 2, p. 252–265, 2012.

WALTON, D. Spiral spline curves for highway design. **Computer-Aided Civil and Infrastructure Engineering**, 1989. Wiley Online Library, v. 4, n. 2, p. 99–106, 1989.

WALTON, D. J.; MEEK, D. S. A controlled clothoid spline. **Computers & Graphics**, 2005. Elsevier, v. 29, n. 3, p. 353–363, 2005.

_____. An improved euler spiral algorithm for shape completion. In: IEEE. **Computer and Robot Vision, 2008. CRV'08. Canadian Conference on**. 2008. p. 237–244.

_____. G1 interpolation with a single cornu spiral segment. **Journal of Computational and Applied Mathematics**, 2009. Elsevier, v. 223, n. 1, p. 86–96, 2009.

WERLING, M.; ZIEGLER, J.; KAMMEL, S.; THRUN, S. Optimal trajectory generation for dynamic street scenarios in a frenet frame. In: IEEE. **Robotics and Automation (ICRA), 2010 IEEE International Conference on**. 2010. p. 987–993.

WILDE, D. K. Computing clothoid segments for trajectory generation. In: IEEE. **2009 IEEE/RSJ International Conference on Intelligent Robots and Systems**. 2009. p. 2440–2445.

XIE, X.; LIAO, W.; AGHAJAN, H.; VEELAERT, P.; PHILIPS, W. A novel approach for detecting intersections from gps traces. In: **2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)**. 2016. p. 1816–1819.

ZHANG, T.; ARRIGONI, S.; GAROZZO, M.; YANG, D.-g.; CHELI, F. A lane-level road network model with global continuity. **Transportation Research Part C: Emerging Technologies**, 2016. Elsevier, v. 71, p. 32–50, 2016.

ZOCCOLER, M.; SHINZATO, P. Y.; HATA, A. Y.; WOLF, D. F. Automatic semantic waypoint mapping applied to autonomous vehicles. In: **Robotics**. : Springer, 2015. p. 96–113.