International Journal of Innovative Technology and Interdisciplinary Sciences
www.IJITIS.org
Volume 2, Issue 3, pp. 200-211, 2019
DOI: https://doi.org/10.15157/IJITIS.2019.2.3.200-211
Received November 20, 2018; Accepted January 11, 2019

Trajectory Generation and Control of Autonomous Vehicles

Thibaud Poulain

National Institute of Applied Sciences of Lyon, France
thibaud.poulain@insa-lyon.fr

ABSTRACT

The objective of this paper is to find feasible path planning algorithms for non-holonomic vehicles subject to their real dynamical constraints. Symmetric polynomial trajectory generation is proposed as reference. Then a trajectory tracking controller for a nonlinear vehicle model is developed, linearizing and discretizing the model, using a linear-quadratic regulator (LQR) control algorithm. Results of numerical simulations are shown. At the end, other controllers are presented in order to continue this work and compare their performances.
Keywords: Path planning, path tracking, nonlinear systems

1. INTRODUCTION

1.1. General Overview

The promises of the autonomous car are great: accident reduction thanks to a better reaction time of detection systems, reduction of traffic jams thanks to more homogeneous speeds and circulation, access to driving for people with limited mobility, reduction of the width of the lanes, the increase of the speed limits, the removal of the constraints related to the long journeys, the dynamic changes of routes thanks to a communication between the cars which would indicate a problem on the lane. In the long run, it seems logical to imagine a disappearance of police checks and even motor insurance, because of the absence of accidents. The arrival of the autonomous car will, however, face significant challenges: skepticism and anxiety of a large part of the chilly population to the idea of giving way to an automated system. [1] What we can be sure, it will change our society and our way of moving.

1.2. Problem Statement

In this paper, we have to solve how to control a autonomous vehicle tracking exactly on a trajectory. This trajectory is generated by a symmetric polynomial method. A LQR controller is proposed and analyzed. To conclude, other controllers are proposed in order to compare the results and the performances.

2. KINETIC MODEL

We consider a four wheel vehicle driving without sliding on a horizontal plane. The steering angle is simplified by one wheel in the middle of the front axle. The kinematic model of a forward rear-wheel driving vehicle can be written as :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ \tan\frac{(\varphi)}{l} \\ 0 \end{bmatrix} rv_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2 \qquad (1)$$

where $X = [x, y, \theta, \varphi]^t$ is the system state variables,
(x , y) are the Cartesian coordinates of the middle point of the rear wheel axis,
θ is the angle of the vehicle body to the x-axis, φ is the steering angle,
l is the vehicle wheel base, r is the wheel radius,
v_1 is the angular velocity of the rear wheel,
v_2 is the angular steering velocity.
We can see these variables on the following figure:

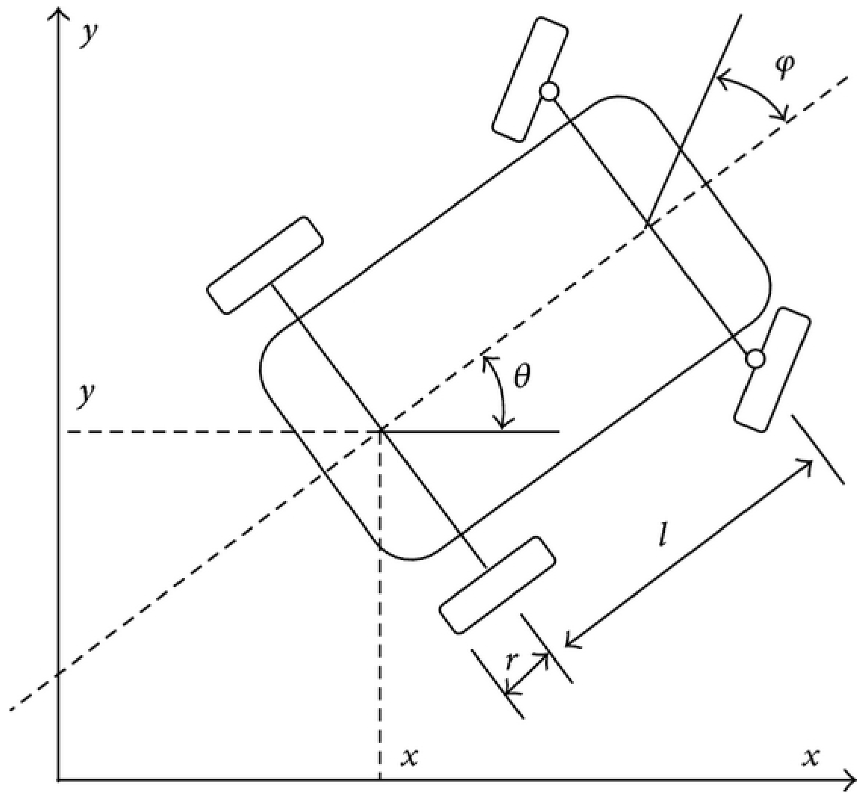

Figure 1 : A simplified vehicle model [2]

We consider the initial state $X(0) = [x_0, y_0, \theta_0, \varphi_0]^t$ at time t = 0 and the final state $X(T) = [x_T, y_T, \theta_T, \varphi_T]^t$ at time $t = T$.

3. VEHICLE TRAJECTORY GENERATION

3.1. Symmetric Polynomial Trajectory Generation

In this paper a symmetric polynomial trajectory generation is used. It is developed and analyzed in [1] with two other methods. With the following equation:

$$v_1 = \frac{\sqrt{\dot{x}^2 + \dot{y}^2}}{r}$$

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

$$\varphi = \tan^{-1}\left(\frac{l\cos^3\theta\ddot{y}}{\dot{x}^2}\right)$$

$$\dot{\theta} = \frac{\tan\varphi}{l} r v_1$$

$$\dot{\varphi} = \frac{\partial(\tan^{-1}(l((\ddot{y}\dot{x} - \ddot{x}\dot{y})/r(\dot{x}^2 + \dot{y}^2)^{3/2})))}{\partial t} \quad (2)$$

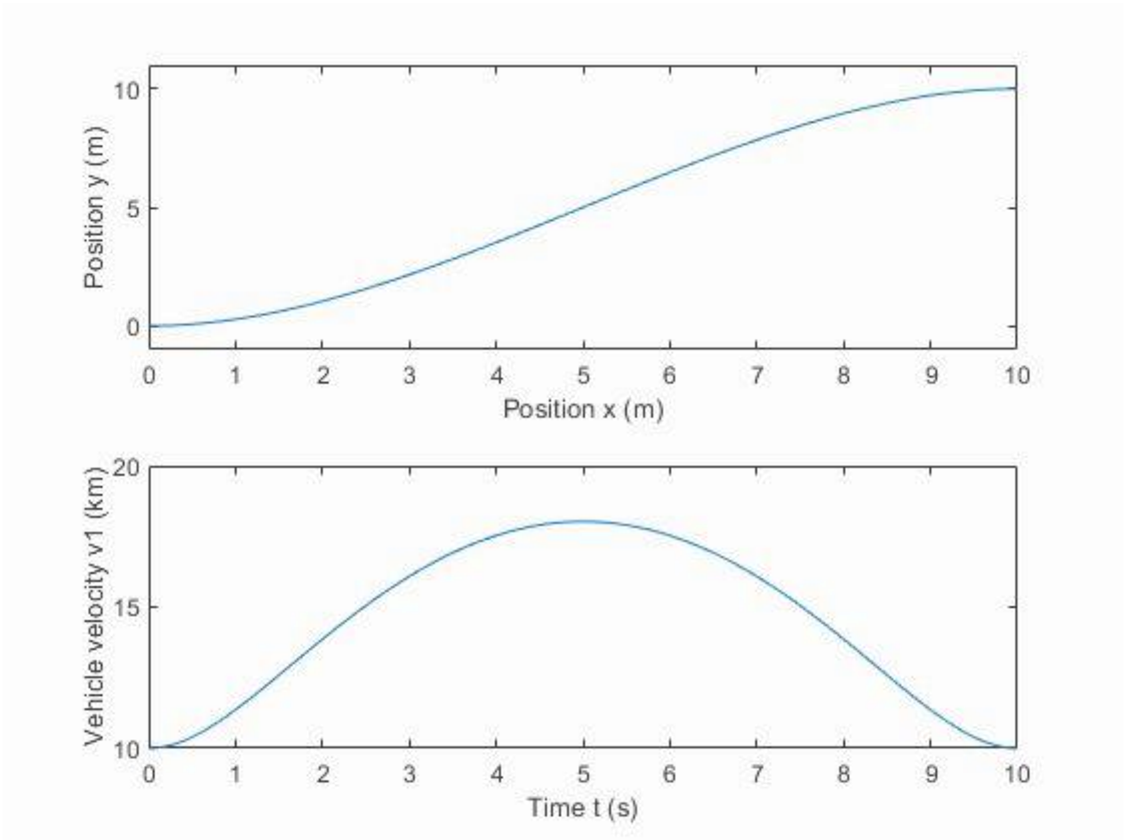

Figure 2 : Trajectory and velocity

We get:

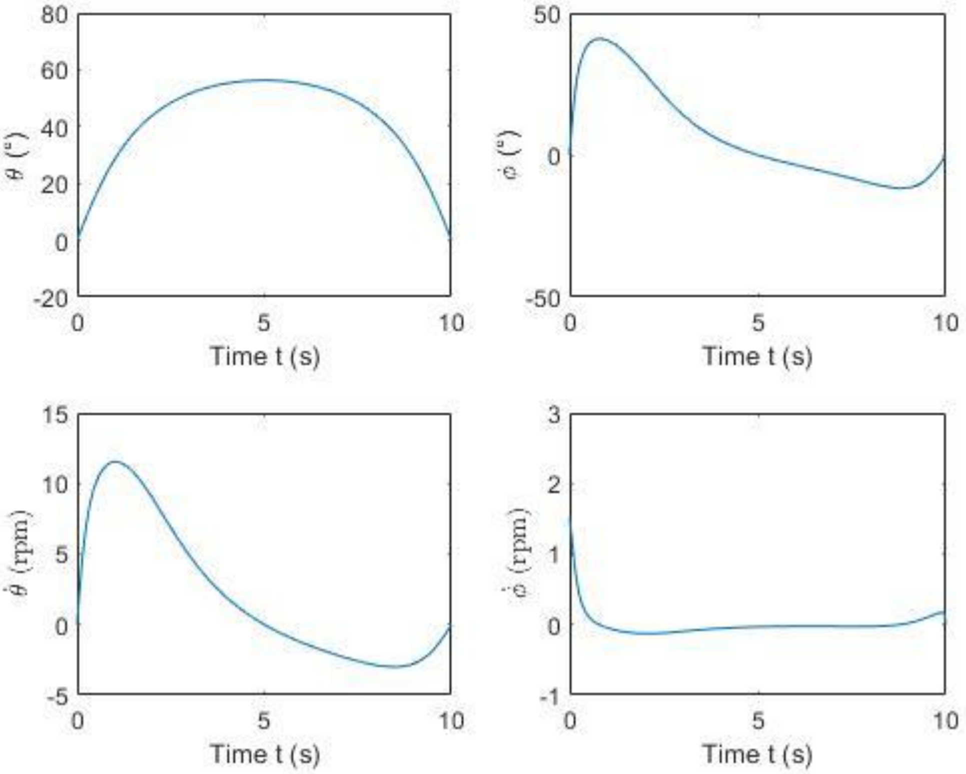

Figure 3 : Body and steering angle and velocity

In figure 3, the constraint of the steering angle is respected $\varphi_{max} = 41° < 45°$. In figure 2, the vehicle velocity is increases at the beginning and decreases at the end of the trajectory. The performance of this method is good to develop a feedback controller.

4. CONTROL PART

An autonomous vehicle has to be able to plan a trajectory, then to track it with a feedback controller. The aim is to reach the destination point and to control the vehicle on the trajectory. However, automated control of vehicles is a complicated task due to the no-linearity of the dynamics, the Multiple-Input Multiple-Output (MIMO) system and because it's a non-holonomic system. Some controllers are easier to design but they can have robustness issues. In this paper, we select a linear-quadratic regulator (LQR) controller. To develop this controller, the kinetic model (1) is linearized in the next part.

4.1.Linearistion

From the kinetic model (1) we can find an approximate linear system[3]. The first order derivative form of the system is:

$$\dot{X} = f(x, u) \tag{3}$$

where the state variables are$X = [x, y, \theta, \varphi]^t$and the imputs are $u = [u_{1,} u_2]^t$, the nonlinear equation in (3) can be expanded in Taylor series around the reference set points (x_r, u_r)at $\dot{X}_r = f(x_r, u_r)$:

$$\dot{X} = f(x_r, u_r) + f_{x,r}(x - x_r) + f_{u,r}(u - u_r) \tag{4}$$

where $f_{x.r}$ and $f_{r.x}$ are the Jacobean of f corresponding to x and u, evaluated around the reference set points(x_r, u_r).

With$\tilde{X}(t) = X(t) - X_r(t)and\tilde{u}(t) = u(t) - u_r(t)$:

$$\dot{\tilde{X}}(t) = A(t)\tilde{X}(t) + B(t)\tilde{u}(t) \tag{5}$$

$$A(t) = \begin{bmatrix} 0 & 0 & -u_{r1}(t)\sin\theta_r(t) & 0 \\ 0 & 0 & u_{r1}(t)\cos\theta_r(t) & 0 \\ 0 & 0 & 0 & \dfrac{u_{r1}(t)}{l\cos^2\varphi_r(t)} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B(t) = \begin{bmatrix} \cos\theta_r(t) & 0 \\ \sin\theta_r(t) & 0 \\ \dfrac{\tan\varphi_r(t)}{l} & 0 \\ 0 & 1 \end{bmatrix}$$

The continuous time system in (5) can be transformed to a discrete-time with a sampling interval,$k + 1 = k + \Delta t$and, Δt is the length of the sampling interval. The inputs *u(k)* do not vary during this time interval.

$$\dot{\tilde{X}}(k+1) = A(k)\tilde{X}(k) + B(k)\tilde{u}(k) \tag{6}$$

$$\tilde{Y}(k) = C(k)\tilde{X}(k)$$

With$A(t) = \begin{bmatrix} 1 & 0 & -u_{r1}(k)\sin\theta_r(k)\Delta t & 0 \\ 0 & 1 & u_{r1}(k)\cos\theta_r(k)\Delta t & 0 \\ 0 & 1 & 0 & \frac{u_{r1}(k)}{l\cos^2\varphi_r(k)}\Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}$,$B(t) = \begin{bmatrix} \cos\theta_r(k)\Delta t & 0 \\ \sin\theta_r(k)\Delta t & 0 \\ \frac{\tan\varphi_r(k)}{l}\Delta t & 0 \\ 0 & \Delta t \end{bmatrix}$,$C(k) = [1]$,$\tilde{X}(k) = X(k) - X_r(k)$and, $\tilde{u}(k) = u(k) - u_r(k)$.

Linearized equations (6) are used to develop LQR controller in the next part.

4.2. Controller

From the discretized linearized model (6):

$$\dot{\tilde{X}}(k+1) = A(k)\tilde{X}(k) + B(k)\tilde{u}(k)$$

We can create an algorithm on Matlab using the command 'dlqr' [4] : the linear-quadratic (LQ) state-feedback regulator for discrete-time state-space system. [K,S] = dlqr(A,B,Q,R) calculates the optimal gain matrix *K* such that the state-feedback law :

$$\tilde{u}[k] = -K\tilde{X}[k] \tag{7}$$

Minimize the quadratic cost function:

$$J(\tilde{u}) = \tilde{X}(k)^t Q\tilde{X}(k) + \tilde{u}(k)^t R\tilde{u}(k) \tag{8}$$

For the discrete-time state model (48). In addition to the state-feedback gain *K*, 'dlqr' returns the infinite horizon solution *S* of the associated discrete-time Riccati equation :

$$A^tSA - S - (A^tSB)(B^tSB + R)^{-1}(B^tSA) + Q = 0 \tag{9}$$

K is derived from S by :

$$K = (B^tSB + R)^{-1}(B^tSA) \tag{10}$$

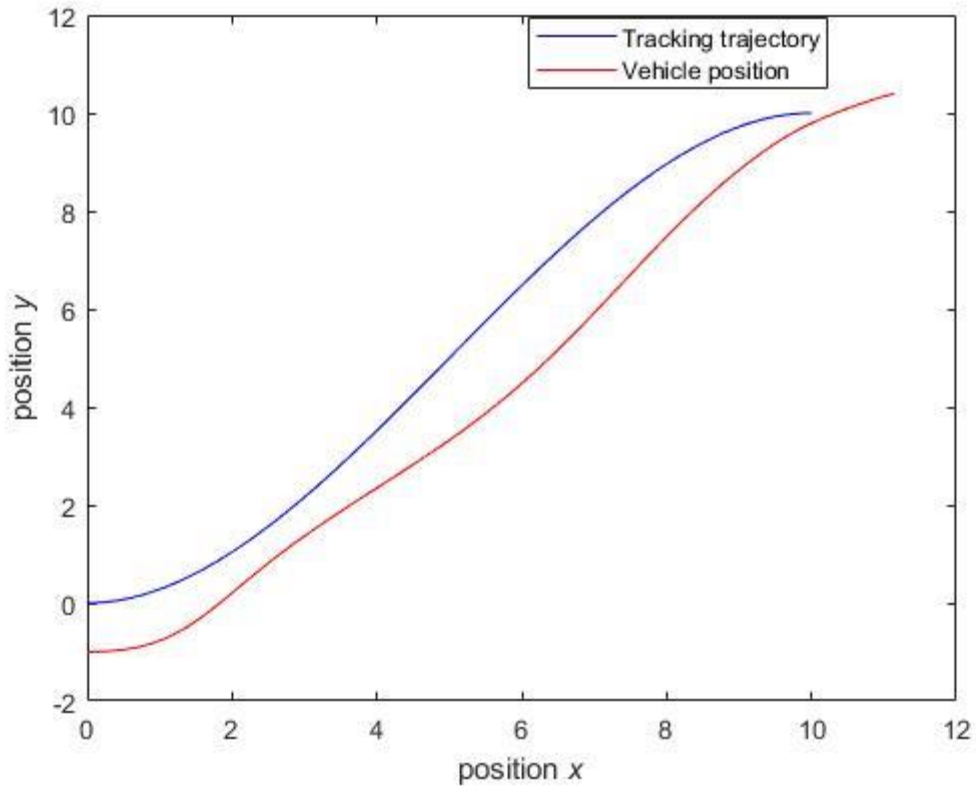

Figure 4 : LRQ controller for tracking polynomial trajectory

4.3. Simulation

For the simulation, we use the symmetric polynomial trajectory from the initial position$[x_0, y_0] = [0,0]$,to the final position$[x_T, y_T] = [10,10]$. We set the time *T=10* s. The initial position of the vehicle is set at $X_0 = [0, -1,0,0]^t$. Penalty matrices are set at$Q = diag(1,1,1,1)$and$R = diag(1,1)$. In the figure 4, the final position is not reached. The tracking errors are visible. The values of vehicle velocity in the figure 5 are good but it is not as smooth as expected .

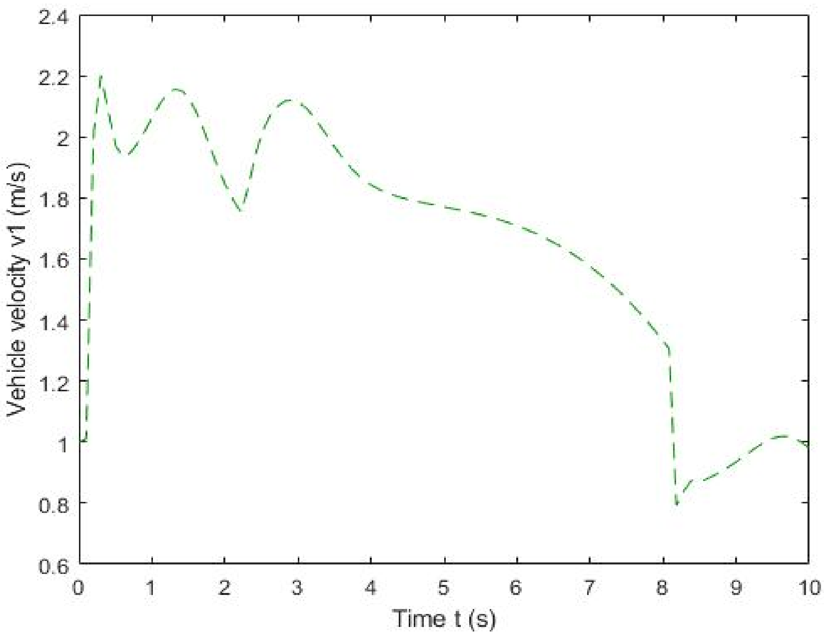

Figure 5 : Vehicle velocity

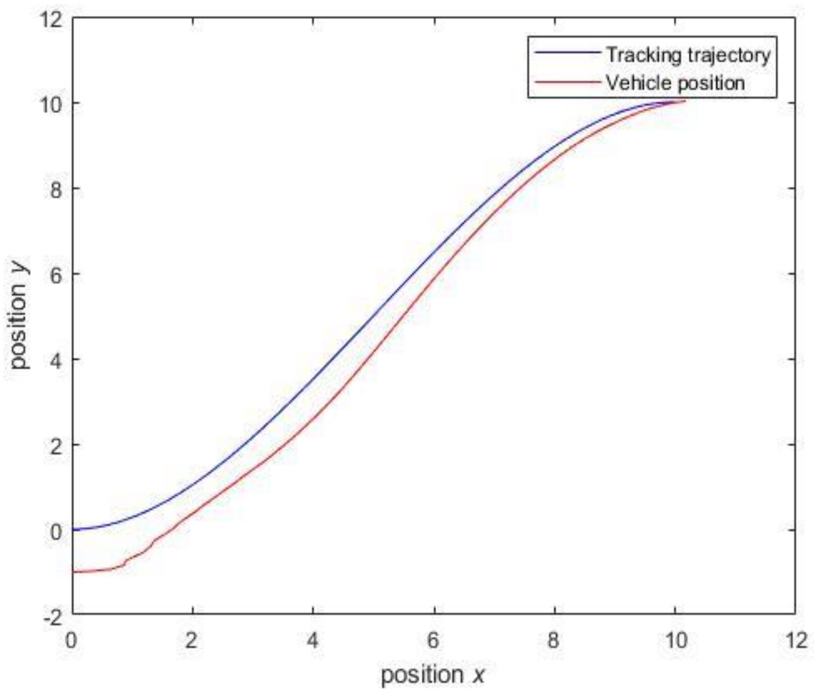

Figure 6 : LRQ controller for tracking polynomial trajectory$Q = 10^2 diag(1,1,1,1)$

We change the penalty matrices to$Q = 10^2 diag(1,1,1,1)$and$R = diag(1,1)$. As we can see in figure 6, the controller is better with these parameters : the final point is reached and there are fewer tracking errors. However, we can observe in figure 7, there are 2 weird values for the vehicle velocity, at *T1*=2,1s and at *T2*=8s. It may be due to the imperfection of the controller.

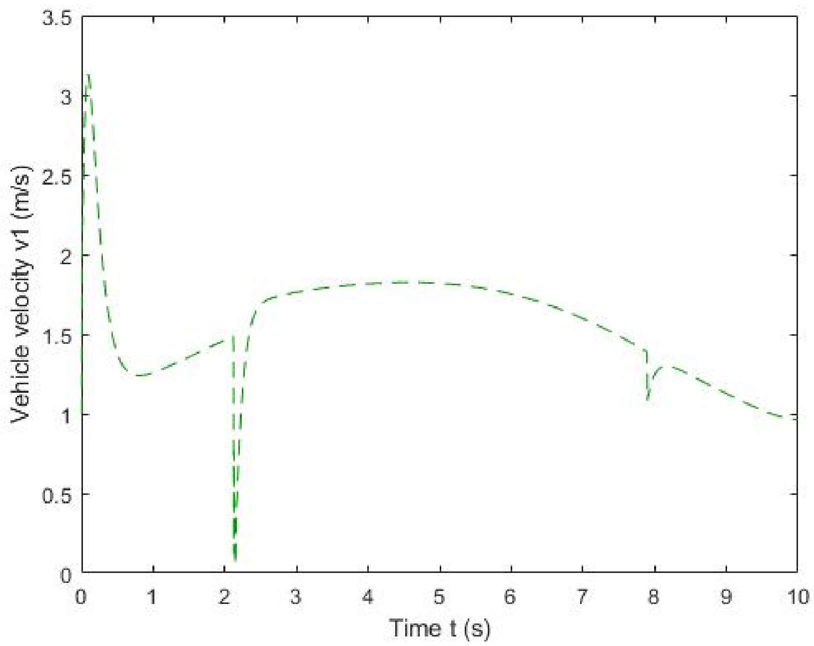

Figure 7 : Vehicle velocity $Q = 10^2 diag(1,1,1,1)$

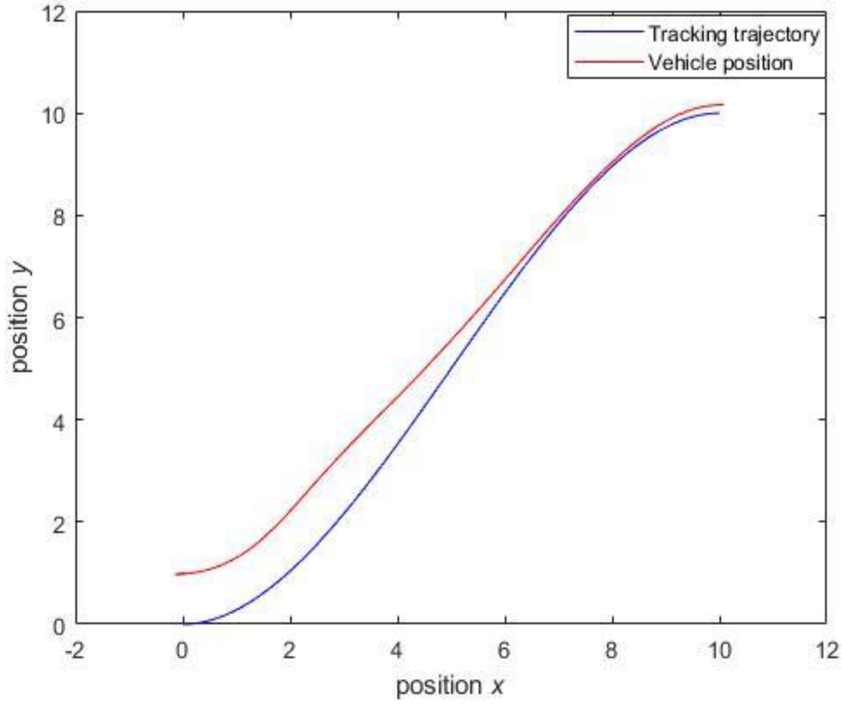

Figure 8 : LRQ controller for tracking polynomial trajectory$X_0 = [0,1,0,0]^t$

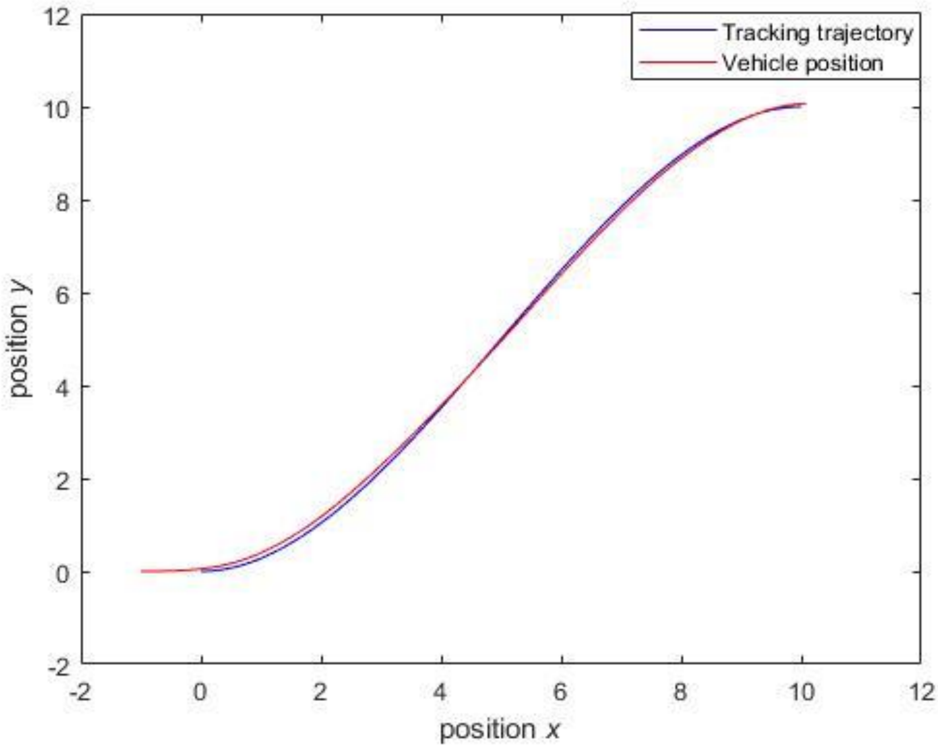

Figure 9 : LRQ controller for tracking polynomial trajectory $X_0 = [-1,0,0,0]^t$

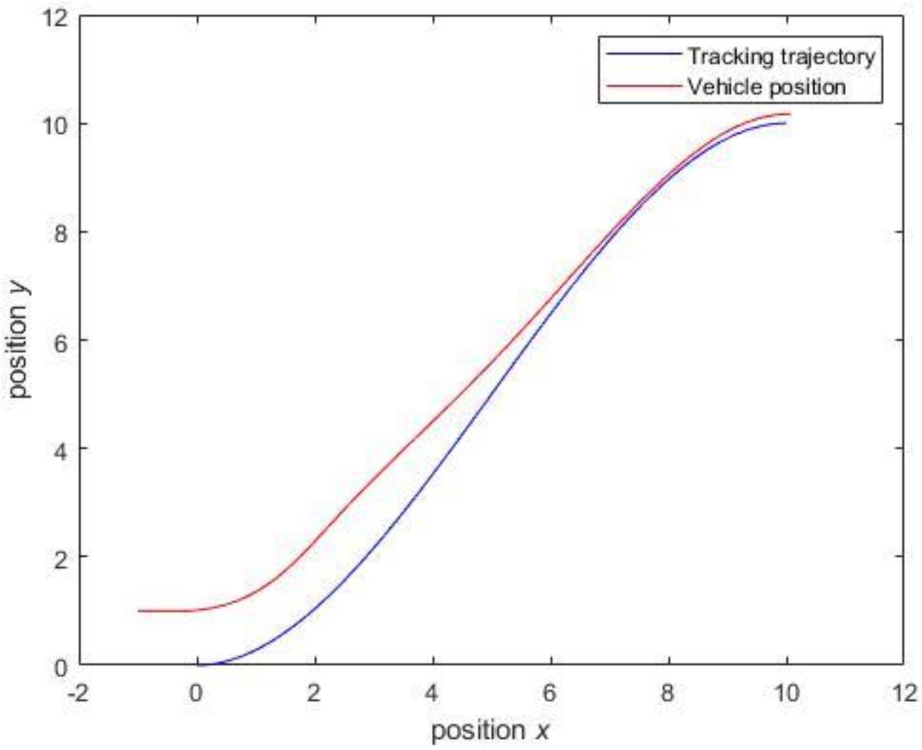

Figure 10 : LRQ controller for tracking polynomial trajectory$X_0 = [1,1,0,0]^t$

For the next simulations, we set$Q = 10^4 diag(1,1,1,1)$and$R = diag(60,60)$and we change the initial position. The results are shown in figure 8, 9 and 10. The performance of the controller is influenced. The best one is reached for $X_0 = [-1,0,0,0]^t$,

For the next simulations, we improve the generation path algorithm. We can select several coordinates in order to create a longer trajectory, more complicated with more curves. In figure 11, the path-tracking in good, even if the direction is changing, $[x_{T1}, y_{T1}] = [10,10]$ at t=$T1$=10 s , $[x_{T2}, y_{T2}] = [18,0]$ at t=$T2$=20s and $[x_{T3}, y_{T3}] = [25,5]$ at t=$T3$=30s. The trajectory is longer, so the influence of the initial position is less visible.

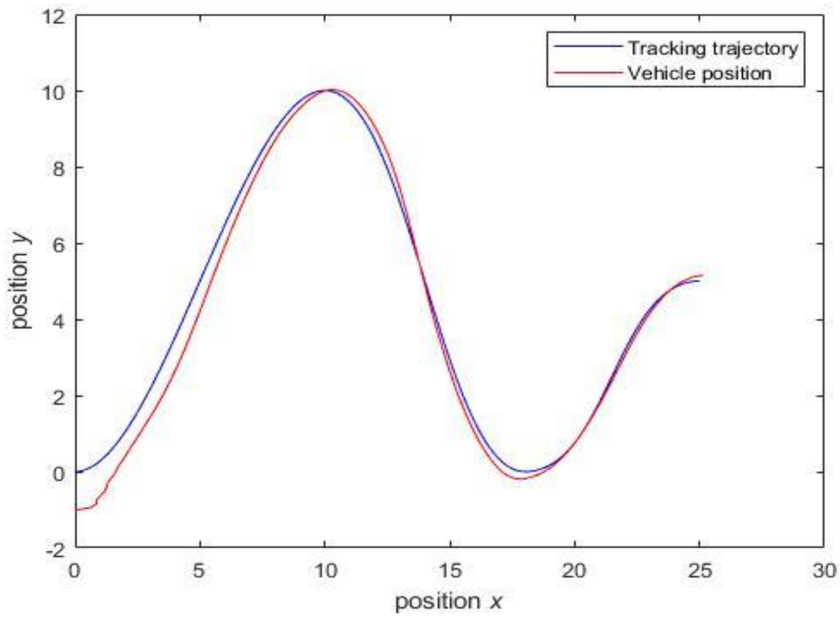

Figure 11: LRQ controller for tracking polynomial trajectory with 2 intermediate points

For the figure 12, $[x_{T1}, y_{T1}] = [10,10]$ at t=$T1$=10s , $[x_{T2}, y_{T2}] = [20,0]$ at t=$T2$=20s, $[x_{T3}, y_{T3}] = [35,15]$ at t=$T3$=30s, and $[x_{T4}, y_{T4}] = [50,0]$ at t=$T4$=40s. We get the same results on the controller performances. The above results show that the vehicle can follow different trajectories, from different initial positions.

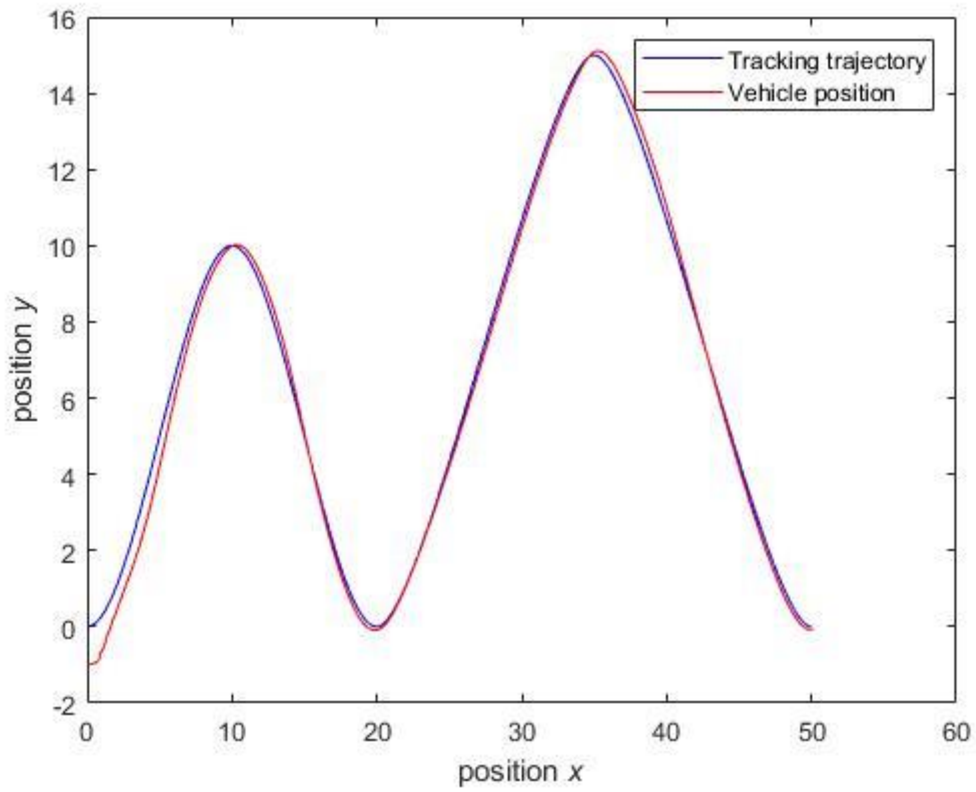

Figure 12 : LRQ controller for tracking polynomial trajectory with 3 intermediate points

5. CONCLUSION

In this paper, a third-order symmetric polynomial trajectory was used to develop a feedback controller. The control is efficient. Many other controllers exist but they must be adapted to our system. A way to improve this work is to design several controllers and to compare them. Because of the difficulty of the system and the short time to do this work, only research on different studies had been done. On the search of a Sliding Mode Control (SMC) design, we can found SMC for discrete time [5], for MIMO systems [6], for MIMO nonlinear systems [7][8], for discrete MIMO uncertain linear system[9][10][11]. In a last study [12], the authors propose comparison between a wide variety of different control schemes for autonomous vehicles. It is not a good choice to use a Proportional-Integral-Derivative controller for this MIMO state-space. It has to be convert in transfer function system, but in our case, the system is nonlinear, so we may not have the transfer function solution. I tried to adapt those methods to this system but without convincing results.

ACKNOWLEDGEMENT

The author would like to thank National Institute of Applied Sciences of Lyon and Tallinn University of Technology (TTU) for supporting this research project and Professor Vu Minh Trieu for his support, his answers and his commitment throughout the time of this thesis.

CONFLICT OF INTERESTS

The author declare that the there is no conflict of interests regarding the publication of this paper.

REFERENCES

[1] Aymar Tissèdre. Voiture Autonome [Online]. Voiture Autonome, 2015, 2018, [6/12/2018] Availaible:https://www.voiture-autonome.net/

[2] Vu Trieu Minh and John Pumwa, “Feasible Path Planning for Autonomous Vehicles,” Mathematical Problems in Engineering, vol. 2014, Article ID 317494, 12 pages, 2014.

[3] Vu Trieu Minh. (2016). Nonlinear Model Predictive Controller and Feasible Path Planning for Autonomous Robots. Open Computer Science., 2016, DOI: https://doi.org/10.1515/comp-2016-0015

[4] Mathworks. (2018). *Global Optimization Toolbox: User's Guide* (r2018a). from https://fr.mathworks.com/help/control/ref/dlqr.html

[5] K. Abidi and J.-X. Xu, Advanced Discrete-Time Control, Studies in Systems, Decision and Control Chapter 2, p9-61 (2015), DOI 10.1007/978-981-287-478-8_2

[6] Andrey Polyakov, Denis Efimov, Wilfrid Perruquetti. Sliding Mode Control Design for MIMO Systems: Implicit Lyapunov Function Approach. European Control Conference, Jun 2014, Strasbourg, France,. 2014.

[7] Krokidis, P.I & Dounis, Anastasios & Syrkos, George & Tseles, Dimitris. Robust sliding mode control applied to nonlinear systems for tracking and regulation in terms of uncertainty. 2005.

[8] Benamor, A., Chrifi-Alaoui, L., Ouriagli, M., Chaabane, M., Massaoud, H. : “Lyapunov Based Second Order Sliding Mode Control for MIMO Nonlinear Systems ”, Proc. International Journal on Sciences and Techniques of Automatic control, IJ-STA, 5(1), 1486-1499, 2011

[9] M. R. Amini, M. Shahbakhti, S. Pan and J. K. Hedrick, "Discrete adaptive second order sliding mode controller design with application to automotive control systems with model uncertainties," *2017 American Control Conference (ACC)*, Seattle, WA, 2017, pp.4766-4771.doi: 10.23919/ACC.2017.7963692

[10] Vu Trieu Minh *et al*, Development of a Wireless Sensor Network Combining MATLAB and Embedded Microcontrollers, *Sensor Letter*, 13(12), 1091-1096, 2015

[11] R. Moezzi, V. T. Minh, and M. Tamre, “Fuzzy Logic Control for a Ball and Beam System”, Int. J. Innov. Technol. Interdisc. Sci., vol. 1, no. 1, pp. 39-48, 2018

[12] D. Calzolari, B. Schürmann and M. Althoff, "Comparison of trajectory tracking controllers for autonomous vehicles," *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, 2017, pp. 1-8. doi: 10.1109/ITSC.2017.8317800.