

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

5,800

Open access books available

142,000

International authors and editors

180M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Autonomous Vehicle Path Planning Using MPC and APF

Zahra Elmi and Soheila Elmi

Abstract

Autonomous vehicles have been at the forefront of academic and industrial research in recent decades. This study's aim is to reduce traffic congestion, improve safety, and accidents. Path planning algorithms are one of the main elements in autonomous vehicles that make critical decisions. Motion planning methods are required when transporting passengers from one point to another. These methods have incorporated several methods such as generating the best trajectory while considering the constraints of vehicle dynamics and obstacles, searching a path to follow, and avoiding obstacles that guarantee comfort, safety, and efficiency. We suggested an effective path planning algorithm based on Model Predictive Controller that determines the maneuvers mode such as lane-keeping and lane-changing automatically. We utilized two different artificial potential field functions for the road boundary, obstacles, and lane center to ensure safety. On the four scenarios, we examined the proposed path planning controller. The obtained results show that when a path planning controller is used, the vehicle avoids colliding with obstacles and follows the rules of the road by adjusting the vehicle's dynamics. An autonomous vehicle's safety is ensured by the path planning controller.

Keywords: motion planning, model predictive controller, potential field, autonomous vehicle, obstacle avoidance

1. Introduction

Today, autonomous robots can be utilized in a number of roles in our daily life. The autonomous robots without human intervention are able to move in the environment and perform their tasks safely and have a wide variety of applications. The main goals are to help humans with difficult, repetitive, and tedious tasks. Additionally, substituting the robots for humans in these tasks is an important dream of humans to reduce human-based errors. Therefore, many developments have done in term of software, hardware, computing, and control. One of the important techniques in robotic science is related to path planning that the goal is to plan a path with the movement of the robot from a start position to target while avoiding collisions with static and dynamic obstacles in the environment. Path planning is a challenging decision-making and control problem. This problem performs in two ways: first, global path planning that the knowledge of the environment is fully available for robot and robot is able to reach to target position safely. Second, local path planning is performed using only the sensed data by the robot, namely, the knowledge of the environment is unidentified or partially unidentified.

Many path planning approaches are presented, which can be divided into two categories: conventional and heuristic approaches. Common methods such as Roadmap [1], Potential Field [2], Cell Decomposition [3], and Mathematical Programming are examples of conventional approaches. These methods are used as hybridization in many applications. Heuristic approaches are presented to overcome the limitations of conventional methods. Probabilistic Roadmap [4], Simulated Annealing [5], Ant Colony Optimization [6], Particle Swarm Optimization [7], and Grasshopper optimization [8] are a few examples of heuristic methods. However, these algorithms have problems in static and dynamic environments. One of the simplest heuristic algorithms is the Dijkstra algorithm that is based on graph search and is able to find a minimum path between two different nodes on a graph by discretization of the environment. The other algorithm is A* which is similar to the Dijkstra algorithm but uses two cost functions to move from start position to target. These algorithms are applied only for environments with static obstacles. These algorithms guaranteed efficiency and optimality of obtaining path but the planned path depends on the resolution of the graph highly. Moreover, considering the dynamic constraints of robots is difficult during the planning process. In the enhanced version of A*, the authors [9] used the proposed method in a changeable environment, and the result is shown that the planned and tracked path is smoother than traditional methods. However, this method ignores the dynamic constraints of obstacles.

The planners of curve interpolation such as Clothoid, Polynomial, Spline, and Bezier curves are widely used for online path planning. These planners are similar to methods based on graph search and have low computational cost because the behavior of the curve is defined by a few control points or parameters. However, the optimality of obtained path is not guaranteed, and the dynamic constraints of a robot are not considered during the planning process and are additionally needed a smoothing process for the obtained path. In [10], a new method is presented by the Clothoid curve to reduce the length and curvature change of path. In this approach, two points are considered on the plane and the proposed algorithm generates a closed-form solution to connect two Clothoid sets for the position of a waypoint. This approach reduces sudden changes of curvature and sideslip by robot and improves the performance of the movement. To generate trajectory in [11], the authors used the polynomial parameterization that represents kinematic constraints and moving obstacles. Besides, the velocity of the robot is planned using this parameterization. To find the optimal solution, a guideline obtained by the Bezier curve is introduced. The result of the simulation has shown that the proposed method performs better than the traditional one. In [12], the authors used a combination of RRT* and Spline techniques to generate a smooth path. The proposed bidirectional Spline-RRT* algorithm is based on the cubic curve and satisfies direction constraints for both start and target positions. This algorithm is not similar to other path planning algorithms and the obtained result for the robot is sub-optimal yet feasible.

Some of the heuristic algorithms such as simulated annealing [13, 14] are used for path planning in environments with static and dynamic obstacles. This algorithm improves the obtained path for the robot and processing epoch. The obtained path is a near-optimal solution and is possible for online implementation, but it ignores the dimension of the robot and avoids only obstacles with circular shapes. Probabilistic Road Maps and Rapidly Random Tress are considered as the methods based on sampling. These methods are used for both holonomic and non-holonomic systems. In [15], a hybrid method for navigation of robots in dynamic and unknown environments is proposed. The proposed method is a combination of the proposed reactive planner and a global planner that Dynamic Rapidly exploring Random Tree

(DRRT) algorithm is used as a global planner. This method improves the speed of planning by reusing parts of the old tree when re-growing it. A probabilistic local planner is used to avoid obstacles. The result illustrates that the proposed method has a 77% reduction in the configured environments. The RRT* [16] guarantees the optimality of obtained path and can achieve convergence to global optimal solution by increasing the number of samples. These methods and their variants are widely used for autonomous robot research. But it is impossible to use in practical applications since they have high computational complexity.

In methods based on path optimization, the main idea is to formulate path planning as an optimization problem that the desired performance and constraints of the robot are considered. This approach is able to find a proper path between start and target positions. In [17], a novel method is presented to predict and avoid the collision of static and dynamic obstacles in an unknown environment. To predict the velocity of obstacles, they have used a decision-making process by using the information of the sensory system of the robot. Therefore, the robot is able to find the proper path, reach the target safely and without any collision. The result illustrates the efficient algorithm for complex and dynamic environments. In [18], an uncontrollable divergence metric is presented. A mechanism to switch between multiple predictive controllers is developed by using this metric to reduce the return time of the controller and maintain predictive accuracy. In [19], a nonlinear MPC for an autonomous underwater vehicle (AUV) is offered. The path planning problem is solved with a receding horizon optimization framework with a Spline template. A combination of the obtained result from path planning and MPC is used for tracking control. To determine the maneuvers mode for autonomous vehicles in dynamic environments, a path planning method with MPC is proposed [20]. To decide maneuvers of lane change and lane-keeping, the convex relaxation method is used. For ensuring the safety of vehicles, a collision-avoidance method is developed. Also, for having a comfortable and natural maneuver, the lane-associated potential field is presented.

The contribution of this paper is to develop a nonlinear MPC approach to solve the path planning problem of an autonomous vehicle. The rest of the paper is organized as follows. The overall framework of an autonomous vehicle and the artificial potential field functions for the road and obstacles, and the model predictive controller for path planning are introduced in Section 2. Section 3 evaluates and discusses the results of path planning for four scenarios. Finally, conclusion is provided in Section 4.

2. Problem description

This section describes the vehicle framework that was used for simulation and control design. **Figure 1** depicts the autonomous vehicle highway scenario used in this paper. The main goal of this paper is to transport a vehicle from a given origin to a given destination at a controlled speed while adhering to common traffic rules

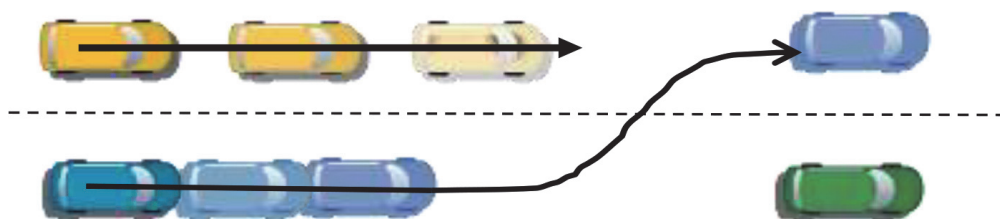


Figure 1.
 The structure of driving environment and the state of obstacle.

and conventions, as well as to avoid colliding with obstacles and to provide a pleasant driving experience. As mentioned earlier, mathematical optimization methods have recently been found to be interesting. These methods provide a symmetric and precise method for considering vehicle dynamics and safety constraints, and they generate the optimal control inputs as a result.

If the environment is fully pre-identified, a mathematical optimization method is used in open-loop form; if the environment is unidentified, a feedback controller is utilized to recognize it [21–22]. In most research, MPC is one of the mathematical optimization approaches applied for online path planning.

Since the driving environment is usually dynamic and stochastic, and cannot be fully predicted a priori, the Model Predictive Control (MPC) approach for path planning has become popular in recent years. MPC uses a recursive method to synthesize a sequence of control optimal inputs in a finite time. The state of the robot or vehicle is updated according to this sequence [23–26]. Two different levels of the controller are considered to solve the path planning problem in a dynamic environment, as shown in **Figure 2**. The first level is for path planning, which generates a reference path based on environmental and destination data. The other controller is for the tracking path, which tracks the reference path directly using control inputs. The path planner employs a kinematic model of the vehicle, while the path tracker employs a dynamic model. The main goal of this paper is to develop a nonlinear MPC approach to solve the path planning problem for autonomous vehicles. The vehicle can plan its path over a finite horizon by using MPC.

2.1 The framework of the vehicle

a bicycle model is used to model the vehicle dynamics. In this model, the vehicle’s two front wheels are combined into a single wheel in the front axle’s center, while the vehicle’s two rear wheels are in the rear axle’s center. The kinematic framework is used to model the ego vehicle as well as any obstacles or other vehicles in the area. In the meantime, **Figure 3** depicts the vehicle model.

The motion equations of the bicycle model are as follows:

$$\begin{aligned}\ddot{x} &= \dot{y}\theta + a_x \\ m\ddot{y} &= 2 \left[C_{af} \left(\delta_f - \frac{\dot{y} + l_f\theta}{\dot{x}} \right) + C_{ar} \frac{l_r\theta - \dot{y}}{\dot{x}} \right] - m\dot{x}\theta \\ \dot{\varphi} &= \omega\end{aligned}$$

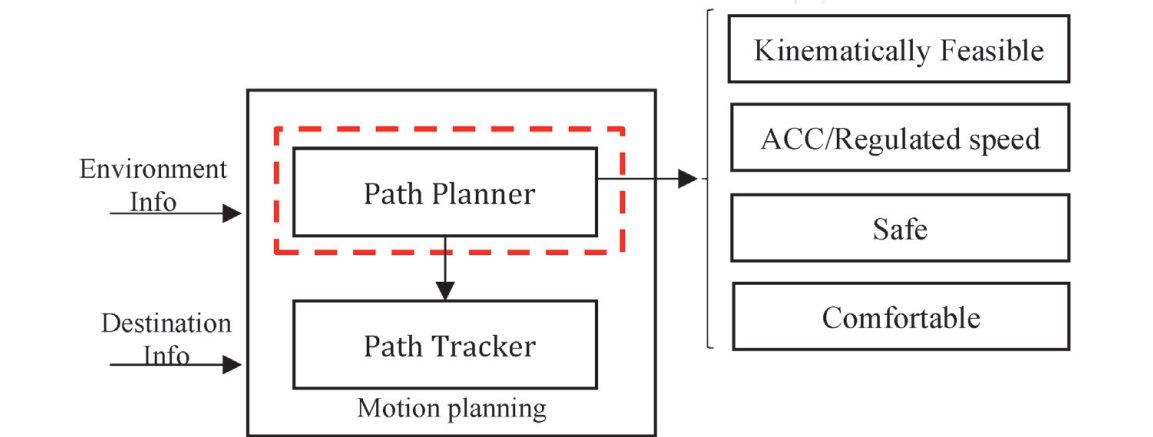


Figure 2.
The motion planning for the autonomous vehicle.

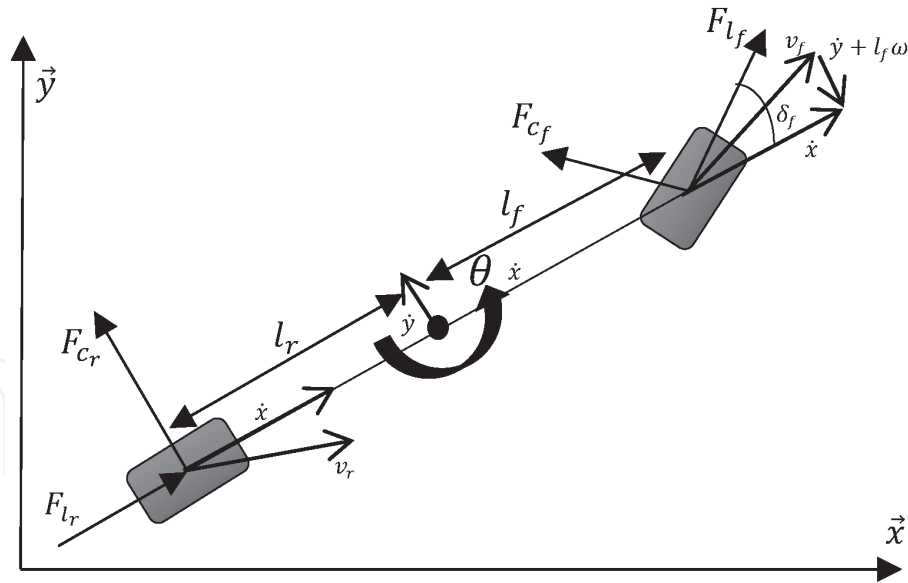


Figure 3.
 The model of bicycle vehicle.

$$I_z \dot{\theta} = 2 \left[l_f C_{af} \left(\delta_f - \frac{\dot{y} + l_f \dot{\theta}}{\dot{x}} \right) - l_r C_{ar} \frac{l_r \dot{\theta} - \dot{y}}{\dot{x}} \right]$$

$$\dot{X} = \dot{x} \cos \varphi - \dot{y} \sin \varphi$$

$$\dot{Y} = \dot{x} \sin \varphi + \dot{y} \cos \varphi \quad (1)$$

where the vehicle's longitudinal and lateral velocities are \dot{x} and \dot{y} . The yaw rate and yaw angle of the vehicle are $\dot{\theta}$ and φ . X and Y are the longitudinal and lateral positions. The front steering angle and the vehicle longitudinal acceleration are represented by δ_f and a_x . The distances between the front and rear axles and the vehicle center of gravity are denoted by l_f and l_r . The cornering stiffness of the front and rear tires are indicated by C_{af} and C_{ar} . The mass and inertia moment of the vehicle are m and I_z .

The model is linearized at an operational point and updated to the MPC internal prediction model at each control step in order to apply a non-linear model in linear MPC. Furthermore, the zero-order-hold technique is used to discretize the linearized model that can be obtained as follows:

$$\xi(t+1) = A(t)x(t) + B(t)u(t)$$

$$\xi = (\dot{x}, \dot{y}, \varphi, \theta, X, Y)u = (a_x, \delta_f) \quad (2)$$

where ξ and u are vectors of state and input, respectively. State and input matrices are represented by A and B .

2.2 Artificial potential field function (APF)

The attractive and repulsive functions in the potential field (PF) technique allow the vehicle to proceed towards the objective, while the repulsive function prevents the vehicle from colliding with obstacle vehicles. The target potential field attracts the vehicle since it has a minimum value in the target location however, the obstacle potential field function repulses the vehicle from the obstacle because it has a maximum value in the obstacle locations [27]. The major goal of this paper is the

navigation of a vehicle to a target point without colliding, which is accomplished by tracking an objective function. As a result, we just regard repulsive function as potential field. For this reason, the obstacle PF (U_O) and the road boundaries (U_R) are used to build the potential field function. At each prediction time, the total sum of potential field functions is obtained by reflecting the predicted surrounding environment. Obstacle vehicles are predicted as a model with constant velocity, and also the information of these vehicles is taken into account in real-time. The sum of the PFs is the potential field:

$$U_{tot} = \lambda_r U_R + \lambda_o U_O \quad (3)$$

where λ_r and λ_o are weights of PF for road and obstacle, respectively. To model road regulation and obstacles, Other functions can be also offered.

2.2.1 PF of lane marker

The lane marker PF is used to keep the vehicle from leaving the main route and driving too close to the boundaries of the road, which leads to raising the risk of a crash. As a result, the lane marker on the road borders should have a maximum value. Furthermore, the slope of achieving this peak point is maximum, so enables a restoring force of maximum value. Meantime, this peak point is operated at the position of the driving lane to prevent changing lanes. Hence, the vehicle tries to keep its current lane to avoid incurring further costs. For this reason, when the vehicle is not facing traffic or barriers, in the center of the lane, PF is zero and locally symmetric that is desired location. The vehicle can overcome this barrier when changing lanes is required. As a result, we utilize a 1D Gaussian function that approaches the left or right road border to get a larger potential value. The following is the PF for the lane marker (U_R):

$$U_R = A_r \exp \left(-\frac{(Y_h - Y_r)^2}{2\sigma_{rb}^2} \right) + A_r \exp \left(-\frac{(Y_h - Y_l)^2}{2\sigma_{rb}^2} \right) \quad (4)$$

where A_r is the maximum value of the potential field for the road boundary. Y_h denotes the lateral position of the ego vehicle in the local road frame, whereas Y_r , Y_l denote the lateral locations of the ego vehicle to the right and left of the center of the straight road, respectively. σ_{rb} is the variable for the road boundary's potential field. In this paper, we assume that the autonomous vehicle is driving on highway then the geometric shape of the road boundary is considered as a first order polynomial function. **Figure 4(a)** shows the 3D plot of the potential field of road boundary on the straight road with $Y_r = -3.8$, $Y_l = 3.8$, $A_r = 40$ and $\sigma_{rb} = 1$.

2.2.2 Obstacle potential field

The obstacle PF (U_O) framework is more complicated and essential than the road PF structure. According to the obstacle PF, the lane change movement is performed if the obstacle vehicle approaches the ego vehicle. This is based on highway driving's structure and protocol. In addition, the vehicle may shift to the left side to pass slower preceding vehicles. For accomplishing this, obstacle PF is modeled as a function of the measured position of the obstacle vehicle, relative and absolute velocity of the vehicle, road curvature, and obstacle vehicles. The available sensor readings from the obstacle are used to determine the location of the obstacle PF. The longitudinal and lateral distances between the ego vehicle and obstacle,

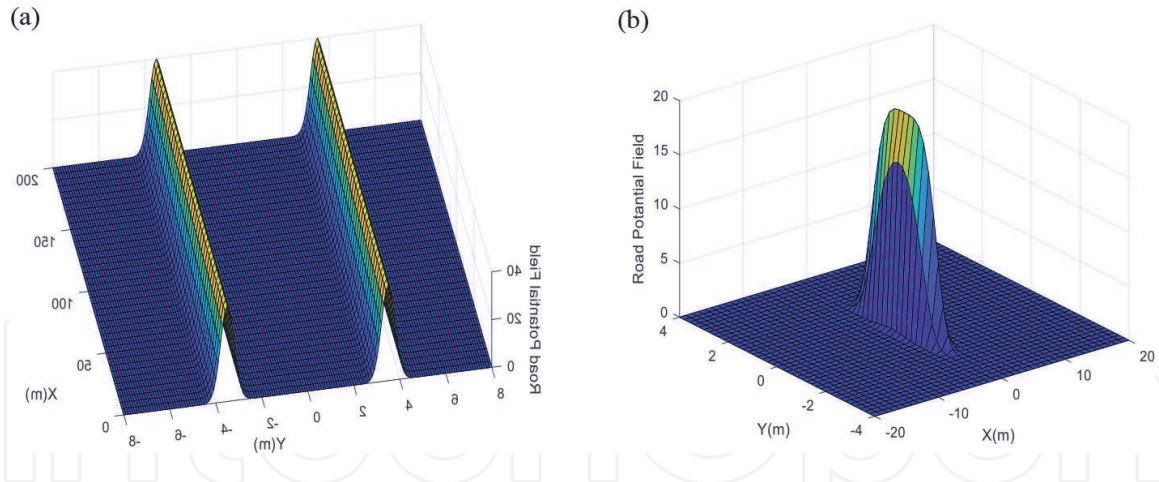


Figure 4.
 The potential field function (a) road boundary (b) obstacle or surrounding vehicle.

given by x_O and y_O , are the acquired information and do not contain the obstacle vehicle's heading angle.

Since it gives a better representation of the layout of an obstacle, the form of an obstacle vehicle is rectangular. To prevent slope discontinuities in PF, continuous functions, such as the hyperbolic function, must be used to describe the obstacle value. This function produces the required potential field by determining the distance between the ego and obstacle vehicles. The function's change rate is rigorously raised when the distance between the ego vehicle and the obstacle is too tiny, and its value approaches infinity, preventing the ego vehicle from colliding with the obstacle. The obstacle's repulsive potential function is as follows:

$$U_o = A_{obs} \exp \left(- \left(\frac{(x - x_{obs})^2}{2\sigma_x^2} + \frac{(y - y_{obs})^2}{2\sigma_y^2} \right)^c \right) \quad (5)$$

where A_{obs} shows the maximum potential field value of the obstacle. (x, y) is the current position of the vehicle and (x_{obs}, y_{obs}) represents the nearest point of ego vehicle from the obstacle. σ_x and σ_y are the convergence coefficient of the obstacle potential field that determines the spread of the horizontal influence of the potential field. In Eq. (5), c is a coefficient for adjusting the shape of the obstacle potential field's peak. The approaching velocity is equal to the difference of velocity between them, If the ego and obstacle are closing together in each direction otherwise, it is set to zero. $(x_{obs}, y_{obs}) = (0,0)$ is the location of the possible obstacle field, also $\sigma_x = \sigma_y = 1$ and $A_{obs} = 20$ are considered and obstacle potential field is shown in **Figure 4(b)**.

2.2.3 MPC framework

MPC is a hybrid method that combines optimum and adaptive control systems [28]. The approach employs a controller-based model that is used in the optimization stage of the model's anticipated states in order to provide the best control input. As a result, the MPC is comparable to an adaptive controller in that it can respond to changing conditions. It manages input and output restrictions at each control interval to solve the optimization problem. MPC is a viable alternative for path planning and tracking based on these characteristics' potential fields. Based on a dynamic model of the vehicle, road regulations and potential field functions, a model predictive controller is suggested. An optimization problem with conflicting needs can be defined using these objectives. The model predictive controller predicts the

response of the ego vehicle based on a horizon known as the prediction horizon (N) and optimizes the vehicle's response, obstacle avoidance, road regulation, and command following based on this value. The intended lane and speed are predefined. Hence, the desired lateral position (the center of the desired lane) and longitudinal velocity are the system outputs that should be tracked:

$$\begin{aligned} y &= [Y \ v_x]^T \\ y_{des} &= [Y_{des} v_{xdes}]^T \\ Y_{des} &= \left(l_{des} - \frac{1}{2} \right) L_w + \Delta Y_R \end{aligned} \quad (6)$$

where y is the output matrix tracking, y_{des} is the intended lateral position, v_{xdes} is the desired speed, l_{des} is the index number of the desired lane from the right, L_w is the lane width, and ΔY_R is the lateral offset of the road relative to a straight road. The path planning nonlinear optimization problem can be expressed in the form:

$$\begin{aligned} \min_{u, s} \quad & \sum_{i=1}^N \|y(t+i|t) - y_{des}(t+i|t)\|_Q^2 + \|u(t+i-1|t) - u(t+i-1|t)\|_R^2 \\ & + \|u(t+i-1|t)\|_S^2 + U_R(t+i|t) + U_O(t+i|t) + \|s_i\|_P^2 \end{aligned} \quad (7)$$

s.t.

$$x(t+i|t) = x(t+i-1|t) + u(t+i-1|t) \quad (8)$$

$$y(t+i|t) = x(t+i-1|t) + u(t+i-1|t) \quad (9)$$

$$v_{xmin} < v_x < v_{xmax} \quad (10)$$

$$u_{min} < u(t+i-1|t) < u_{max} \quad (11)$$

$$\Delta u_{min} < u(t+i-1|t) - u(t+i-2|t) < \Delta u_{max} \quad (12)$$

where $(t+i|t)$ index indicate the values at future time $t+i$ and predicted at current time t . N is the prediction horizon. The vector of slack variables at time t is denoted by s_i . The tracking quadratic term, changes in inputs, inputs, potential field functions, and slack variables compose the objective function. The predicted potential field, as well as quadratic terms of tracking, inputs, changes in inputs, and slack variables, are all included in the objective function, with weighting matrices Q , R , S , and P , respectively. The predicted states are obtained by (8). The tracking output is calculated by (9). The constraints of speed and octagon approximation are applied as soft constraints represented in Eqs. (10). To satisfy the limitations of actuator, the inputs of control and their changes are constrained in (11) and (12) where u_{min} and u_{max} are the lower and upper bounds matrices of control input, and Δu_{min} and Δu_{max} are the lower and upper bounds matrices of the control inputs changes.

3. Results and discussion

3.1 Test scenario

The most challenging problems in the field of autonomous vehicles are path planning and control design. In structured and dynamic environments such as roads, route planning consists of both global and local path planning, with global

path planning being utilized in conjunction with local path planning. Global path planning is a lengthy and deliberate procedure that is handled to develop long-distance routes to a destination. Local path planning, on the other hand, is a quicker procedure that is utilized for short-distance pathways and deals with duties like obstacle avoidance, comfort, safety, and vehicle stability. This planner is more responsive, as it operates in real-time.

Driving on organized roadways may be broken down into two fundamental vehicle maneuvers: lane keeping and lane switching. The primary goal of lane-keeping is to follow a vehicle and maintain its present position by changing its direction and distance from the lane center on a continual basis. Overtaking, obstacle avoidance and road departure are the most prevalent reasons for a vehicle to shift its present lane. The move may change depending on the route, lane, and obstacles on the road. In reality, there are several movements to be made. The performance of path planning systems may be assessed by watching these moves, as well as safety and road rules. If the vehicle's path is safe, the vehicle can stay in its lane. A lane change must be planned and implemented if this is not the case. This lane change occurs when the vehicle reaches the end of the road or encounters another barrier in its own route. If there are no obstacles or other vehicles on the targeted lane at the end of the road, lane switching is completed. Otherwise, the vehicle should slow down and perhaps stop before reaching the lane's end. When a vehicle encounters an obstacle in its path, it must predict the obstacle's path. The vehicle may pass the obstacle if there is adequate lateral distance; nevertheless, the vehicle is changing lanes to overtake. Otherwise, the vehicle should come to a complete stop in front of the obstacle or cross it. These are a few examples of movements that occur on the highway. Two scenarios are provided to evaluate the performance of autonomous vehicles: On both straight and curving roads, retain your lane. Maintaining a certain space between the ego vehicle and the vehicle in front of it.

3.2 Simulation

Since the provided potential field is a non-convex and nonlinear function, the optimization problem is non-convex and nonlinear, therefore solving it is expensive. Thus, the problem is converted into a quadratic and convex problem to reduce computing time. Convex functions are used to approximate PFs for this purpose. The obtained convex function is then approximated using the second-order Taylor series by a quadratic function. Around the nominal point, the resulting function is a near convex quadratic approximation of the original function. The resulting gradient is the same as the original function's gradient. The approximated function's Hessian matrix is the Frobenius norm's nearest positive definite matrix to the original function's Hessian matrix. Although the quadric approximation of the PFs increases the computation time, it is insignificant compared to the time required to solve a nonlinear optimization problem [29].

The problem of optimal control is a convex quadratic optimization problem when these PFs are used. This problem is related to a nonlinear problem that may be solved in one step using Sequential Quadratic Programming (SQP). Boggs et al. [30] calculate an upper bound for the optimization error of each SQP sequence, where this error is the difference between the sequence outcome and the local minimum of the nonlinear problem about the problem's initial value. According to this upper bound, if the problem's initial value is closer to a minimum, the optimization error will be reduced. The predicted vehicle position will be equal to the vehicle's location at a minimum point. Furthermore, in the Hessian matrix, the closer estimated PFs are to their minimum values, the lower the optimization error. As a

result, a PF with a lower convex quadratic approximation error and a lower variation of the Hessian matrix about the problem’s initial value has a lower optimization error.

In the following, the suggested MPC’s performance on an autonomous vehicle is simulated and tested in terms of maneuverability, road regulation, and obstacle avoidance. The YALMIP toolbox in MATLAB/Simulink and the fmincon solver SQP are used to solve the MPC formulation. The features of a controller for a dry road are shown in **Table 1**. The vehicle is moving at 20 m/s, and the controller time step is 50 milliseconds.

Figure 5(a) depicts the first scenario, which is relevant to path planning on a regular highway. Based on offline lane marking and mapping waypoints, a 4th order polynomial is used to approximate the road geometry. It’s a two-lane, one-way road. The ego vehicle is shown as a dashed blue circle traveling on lane 1, while the obstacle vehicle is shown as a red circle moving on the other lane. This scenario’s

Parameter	Value	Unit
m	1625	kg
I_z	2865.6	kg.m ²
l_f	1.108	m
l_r	1.502	m
N	10	—

Table 1.
The controller parameters.

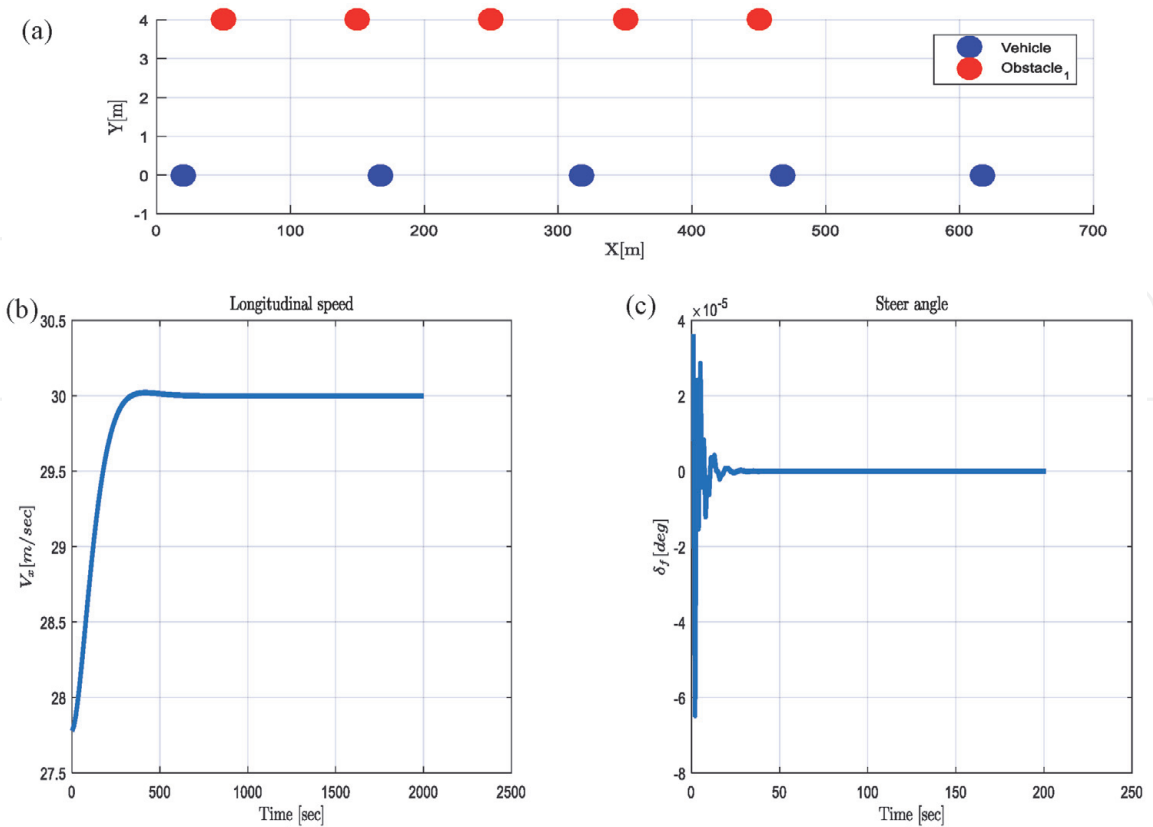


Figure 5.
(a) The path planning in the straight road with keeping lane, (b) longitudinal speed of the vehicle, (c) steering angle for the first scenario.

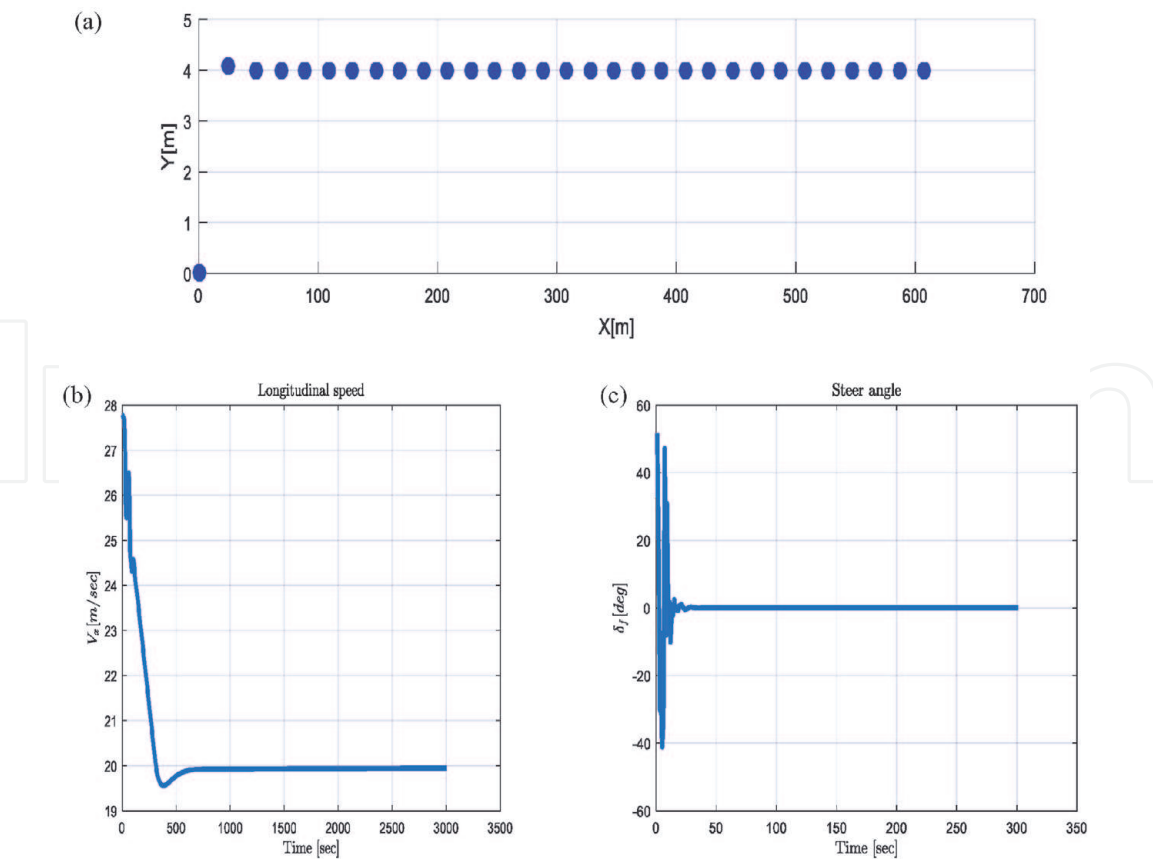


Figure 6.
(a) Changing lane the vehicle in the straight road and decrease decreases the speed from 27 m/sec to 20 m/sec, (b) longitudinal speed, (c) steering angle and lateral acceleration for the second scenario.

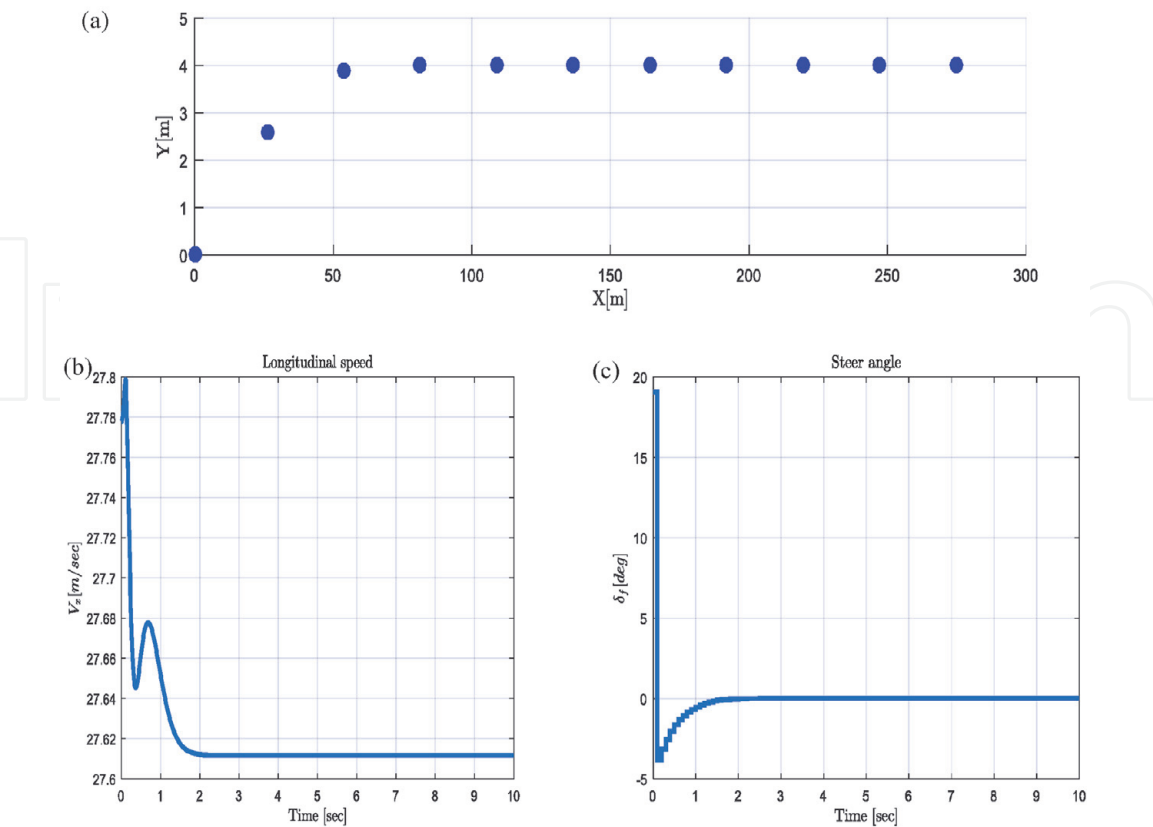


Figure 7.
(a) Changing lane the vehicle in the straight road, (b) longitudinal speed, (c) steering angle and lateral acceleration for the third scenario.

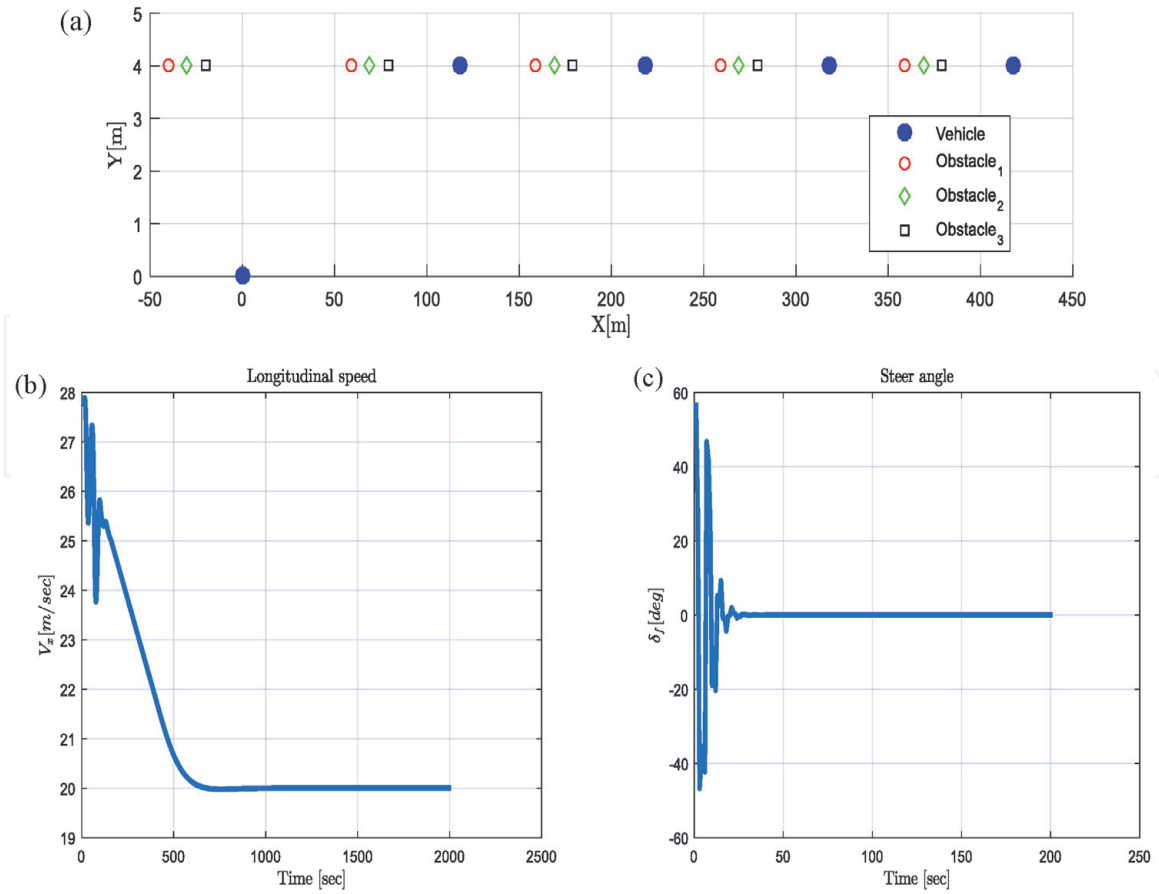


Figure 8. (a) Changing lane along with three obstacles in the straight road, (b) longitudinal speed, (c) steering angle for the last scenario.

main goal is to demonstrate lane-keeping ability on a straight road. The path of the ego and the obstacle vehicle is represented by a sequence of circles. The ego vehicle's desired speed is higher than the obstacle vehicle in the other lane. The obstacle is moving at a speed of 20 m/sec, vehicle tries to increase its speed to 30 m/sec without changing the lane. **Figure 5(b)** depicts the longitudinal speed of the vehicle. **Figure 5(c)** depicts the steering angle of the vehicle.

In the second scenario, the ego vehicle starts on lane 1. The vehicle just tries to change the lane and decreases the speed from 27 m/sec to 20 m/sec and changes its front wheel angle (**Figures 6**).

In the next scenario, the ego vehicle starts on lane 1 and tries to change the lane, however, it remains its speed at 27 m/sec and changes its front wheel angle (**Figure 7**).

Figure 8 shows a merging maneuver along with three obstacles which is moving on the lane 2 and the ego is moving on lane 1. It should change its lane from the lane 1 to the lane 2 while avoiding a possible collision to the obstacles. Due to the lack of lateral distance between the vehicle and obstacle, the vehicle cannot immediately and safely merge between them. In this scenario, the potential field used for obstacle keeps the vehicle away from the lane 2 when the obstacle is passing from another lane. Additionally, the potential field is used for a static obstacle to avoid collision. Obstacles are moving with speed 20 m/sec, vehicle tries to come in their lane with decreasing its speed and the front wheel angle. Therefore, the vehicle reduces its speed and sometimes stops before reaching to the vehicle and after passing obstacle, the vehicle changes safely its lane and then the road and lane centering potential fields keep the vehicle for going out from the road.

4. Conclusion

In this paper, we present a path planning method for autonomous vehicles in dynamic settings that is based on model predictive controllers. We offer two alternative possible field functions for the road, center of the lane, and barriers or surrounding vehicles to avoid collision and assure vehicle safety. By definition, the problem is a nonlinear optimal control problem. To formulate the problem of path planning using a quadratic objective function, the MPC framework is chosen. The computing load is considerably reduced when the problem is approximated as a convex quadratic problem. This function makes it easier to leverage the vehicle dynamics and system limitations inside the MPC framework to calculate lane keeping and lane shifting maneuvers. Simulations are used to compare the computing time and performance of nonlinear and quadratic issues. The results demonstrate that the quadratic formulation outperforms the nonlinear variant in terms of performance. Several simulations are run to analyze various possibilities. The findings show that the suggested path planning algorithm can generate safe and pleasant pathways for self-driving vehicles. Because the vehicle dynamics are utilized as the predicted model, the planned path is an ideal path based on the vehicle dynamics.

Author details


Zahra Elmi^{1*} and Soheila Elmi²

¹ Department of Computer Engineering, Istanbul Sabahattin Zaim University, Istanbul, Turkey

² Department of Electronics Engineering, Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran

*Address all correspondence to: zahra.elmi@izu.edu.tr

IntechOpen

© 2021 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Ravankar AA, Ravankar A, Emaru T, Kobayashi Y. Hpprm: Hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots. *IEEE Access*. 2020 Dec 8; 8:221743-221766.
- [2] Lin P, Choi WY, Chung CC. Local Path Planning Using Artificial Potential Field for Waypoint Tracking with Collision Avoidance. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC) 2020 Sep 20 (pp. 1-7)*. IEEE.
- [3] Kyaw PT, Paing A, Thu TT, Mohan RE, Le AV, Veerajagadheswar P. Coverage path planning for decomposition reconfigurable grid-maps using deep reinforcement learning based travelling salesman problem. *IEEE Access*. 2020 Dec 15;8: 225945-225956.
- [4] Chen G, Luo N, Liu D, Zhao Z, Liang C. Path planning for manipulators based on an improved probabilistic roadmap method. *Robotics and Computer-Integrated Manufacturing*. 2021 Dec 1;72:102196.
- [5] Huo L, Zhu J, Wu G, Li Z. A novel simulated annealing based strategy for balanced UAV task assignment and path planning. *Sensors*. 2020 Jan;20(17): 4769.
- [6] Wang L, Kan J, Guo J, Wang C. 3D path planning for the ground robot with improved ant colony optimization. *Sensors*. 2019 Jan;19(4):815.
- [7] Shao S, Peng Y, He C, Du Y. Efficient path planning for UAV formation via comprehensively improved particle swarm optimization. *ISA transactions*. 2020 Feb 1;97:415-430.
- [8] Elmi Z, Efe MÖ. Online path planning of mobile robot using grasshopper algorithm in a dynamic and unknown environment. *Journal of Experimental & Theoretical Artificial Intelligence*. 2021 May 4;33(3):467-485.
- [9] Sipahioglu A, Yazici A, Parlaktuna O, Gurel U. Real-time tour construction for a mobile robot in a dynamic environment. *Robotics and Autonomous Systems*. 2008 Apr 30;56(4):289-295.
- [10] Kim YH, Park JB, Son WS, Yoon TS. Modified turn algorithm for motion planning based on clothoid curve. *Electronics Letters*. 2017 Dec 7;53(24): 1574-1576.
- [11] Yang S, Wang Z, Zhang H. Kinematic model based real-time path planning method with guide line for autonomous vehicle. In *2017 36th Chinese Control Conference (CCC) 2017 Jul 26 (pp. 990-994)*. IEEE.
- [12] Sudhakara P, Ganapathy V, Sundaran K. Optimal trajectory planning based on bidirectional spline-RRT* for wheeled mobile robot. In *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS) 2017 May 4 (pp. 65-68)*. IEEE.
- [13] Miao H, Tian YC. Dynamic robot path planning using an enhanced simulated annealing approach. *Applied Mathematics and Computation*. 2013 Oct 1;222:420-437.
- [14] Chaudhary KL, Ghose D. Path planning in dynamic environments with deforming obstacles using collision cones. In *2017 Indian Control Conference (ICC) 2017 Jan 4 (pp. 87-92)*. IEEE.
- [15] D'Arcy M, Fazli P, Simon D. Safe navigation in dynamic, unknown, continuous, and cluttered environments. In *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR) 2017 Oct 11 (pp. 238-244)*. IEEE.

- [16] Karaman S, Frazzoli E. Incremental sampling-based algorithms for optimal motion planning. *Robotics Science and Systems VI*. 2010 May 5;104(2).
- [17] Kamil F, Hong TS, Khaksar W, Moghrabiah MY, Zulkifli N, Ahmad SA. New robot navigation algorithm for arbitrary unknown dynamic environments based on future prediction and priority behavior. *Expert Systems with Applications*. 2017 Nov 15; 86:274-291.
- [18] Zhang K, Sprinkle J, Sanfelice RG. A hybrid model predictive controller for path planning and path following. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems* 2015 Apr 14 (pp. 139-148).
- [19] Shen C, Shi Y, Buckham B. Model predictive control for an AUV with dynamic path planning. In *2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)* 2015 Jul 28 (pp. 475-480). IEEE.
- [20] Liu C, Lee S, Varnhagen S, Tseng HE. Path planning for autonomous vehicles using model predictive control. In *2017 IEEE Intelligent Vehicles Symposium (IV)* 2017 Jun 11 (pp. 174-179). IEEE.
- [21] Gray A, Gao Y, Lin T, Hedrick JK, Tseng HE, Borrelli F. Predictive control for agile semi-autonomous ground vehicles using motion primitives. In *2012 American Control Conference (ACC)* 2012 Jun 27 (pp. 4239-4244). IEEE.
- [22] Carvalho A, Gao Y, Gray A, Tseng HE, Borrelli F. Predictive control of an autonomous ground vehicle using an iterative linearization approach. In *16th International IEEE conference on intelligent transportation systems (ITSC 2013)* 2013 Oct 6 (pp. 2335-2340). IEEE.
- [23] Ji J, Khajepour A, Melek WW, Huang Y. Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints. *IEEE Transactions on Vehicular Technology*. 2016 Apr 22;66 (2):952-964.
- [24] Worthmann K, Mehrez MW, Zanon M, Mann GK, Gosine RG, Diehl M. Model predictive control of nonholonomic mobile robots without stabilizing constraints and costs. *IEEE transactions on control systems technology*. 2015 Oct 29;24(4):1394-1406.
- [25] Gao Y, Gray A, Tseng HE, Borrelli F. A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles. *Vehicle System Dynamics*. 2014 Jun 3;52 (6):802-823.
- [26] Brown M, Funke J, Erlien S, Gerdes JC. Safe driving envelopes for path tracking in autonomous vehicles. *Control Engineering Practice*. 2017 Apr 1;61:307-316.
- [27] Silva-Ortigoza R, Márquez-Sánchez C, Carrizosa-Corral F, Hernández-Guzmán VM, García-Sánchez JR, Taud H, Marciano-Melchor M, Alvarez-Cedillo JA. Obstacle Avoidance Task for a Wheeled Mobile Robot: A Matlab-Simulink-Based Didactic Application. *MATLAB: Applications for the Practical Engineer*. 2014 Sep 8:79-102.
- [28] Elmi Z, Efe MÖ. Path planning using model predictive controller based on potential field for autonomous vehicles. In *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society* 2018 Oct 21 (pp. 2613-2618). IEEE.
- [29] Tanaka M, Nakata K. Positive definite matrix approximation with condition number constraint. *Optimization Letters*. 2014 Mar;8(3):939-947.
- [30] Boggs PT, Tolle JW. Sequential quadratic programming. *Acta numerica*. 1995 Jan;4:1-51.