Advanced Methods for Autonomous Driving based on Adaptive Model Predictive Control

Alberto Franco

Department of Information Engineering
Università degli Studi di Padova
alberto.franco.3@studenti.unipd.it

Abstract—**In this paper we present two different strategies for a self-driving car. The aim of this project is to study and implement a path tracking framework in order to prevent ATLASCAR2 collision with moving obstacle vehicles. The proposed algorithms, based on the Model Predictive Control paradigm, solve optimal problems formulated in terms of cost minimization under constraints.**

I. INTRODUCTION

In robotic research, the problem of navigation is among the most important. Basically all autonomous mobile robots need some kind of navigation to fulfill the mobile term. We understand navigation as a process of planning a path of a mobile robot from its current position to a desired goal location, following the planned path, and avoiding any discovered obstacles along the way. The desired paths have to fulfill several conditions to ensure safety and feasibility of the navigation. Moreover, the paths can be also compared in terms of desirability for example short or smooth paths are usually more desirable than long and curved ones. Such paths should therefore be preferred in the navigation process. Beyond the path planning, the navigation problem also involves reacting to changes of the environment model. Robots are required to move towards target in a short time and avoid either static or dynamic obstacles observed by their sensors, which involves efficient path planning and valid obstacle avoidance. Though these two topics have been well researched, currently, there is no ideal solution to handling the navigation problem within cluttered dynamic environments.

II. CONTEXT OF THE PROBLEM AND PROPOSED SOLUTION

The ATLACAR2 has a sensor, such as lidar, that measures the distance to an obstacle in front of the vehicle and in the same lane. The obstacle can be static, such as a large pothole, or moving, such as a slow-moving vehicle. The most common maneuver from the driver is to temporarily move to another lane, drive past the obstacle, and move back to the original lane afterward In this case, we want to design an obstacle avoidance system that moves the ATLASCAR2 around a moving obstacle in the lane using throttle and steering angle. This system uses an adaptive Model Predictive Controller that updates both the predictive model and the mixed input/output constraints at each control interval. Moreover this vehicle is also equipped with a lane-keeping assist system: it has a sensor, such as camera, that measures the lateral deviation and relative yaw angle between the centerline of a lane and the ATLASCAR2. The sensor also measures the current lane curvature and curvature derivative. Depending on the curve length that the sensor can view, the curvature in front of the vehicle can be calculated from the current curvature and curvature derivative. This system keeps the autonomous car travelling along the centerline of the lanes on the road by adjusting the front steering angle. The goal for lane keeping control is to drive both lateral deviation and relative yaw angle close to zero.

TO DO THEORETICAL BACKGROUND ADAPTIVE MPC

III. OBSTACLE AVOIDANCE

A. Problem Formulation

The collision avoidance problem is very dependent on the vehicle modeling since it is a requirement for adaptive MPC law design.

The model used in this paper should take into account the kinematic and dynamic aspects of the vehicle. Here, we present a non linear mathematical model of a vehi-
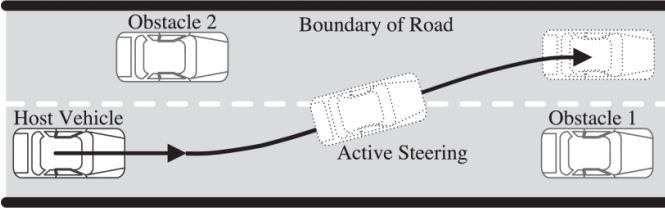
Fig. 1. Problem description of collision avoidance on road.

cle used for the development of a collision avoidance system. The model has four states and two inputs:

$$\mathbf{x} = \begin{bmatrix} x & y & \theta & v \end{bmatrix}^T \qquad \mathbf{u} = \begin{bmatrix} T & \delta \end{bmatrix}^T$$

where (x, y) are the global coordinates of the center point of the car, θ is the heading angle of the car body with respect to the x-axis and v is the speed of the car (positive). The manipulated variables are T the throttle (positive when accelerating/negative when decelerating) and δ the steering angle (0 when aligned with car, counterclockwise positive).

The simple non-linear model to describe the dynamics of the ATLASCAR2 is the following

$$\begin{cases} \dot{x} = -v\sin(\theta) \cdot \theta + \cos(\theta) \cdot v \\ \dot{y} = v\cos(\theta) \cdot \theta + \sin(\theta) \cdot v \\ \dot{\theta} = (\tan(\delta)/C_L) \cdot v + \left(v\left(\tan(\delta)^2 + 1 \right)/C_L \right) \cdot \delta \\ \dot{v} = 0.5 \cdot T \end{cases}$$

where C_L is the car length. The state equations can be re-written in more compact form as follows:

$$\begin{aligned} \dot{\mathbf{x}} &= A_c\mathbf{x} + B_c\mathbf{u} \\ \mathbf{y} &= C_c\mathbf{x} + D_c\mathbf{u} \end{aligned} \qquad (1)$$

where

$$A_c = \begin{bmatrix} 0 & 0 & -v\sin(\theta) & \cos(\theta) \\ 0 & 0 & v\cos(\theta) & \sin(\theta) \\ 0 & 0 & 0 & \tan(\delta)/C_L \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B_c = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & v\left(\tan(\delta)^2 + 1 \right)/C_L \\ 0.5 & 0 \end{bmatrix},$$

$$C_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \mathbf{I}_4, \qquad D_c = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \mathbf{0}_{4\times 2}.$$

The system to be controlled is usually modeled by a discrete state-space model in the MPC literature. Therefore, (1) is transformed into a discrete state-space model to be used by the model predictive controller:

$$\begin{aligned} \mathbf{x}(k+1) &= A_d\mathbf{x}(k) + B_d\mathbf{u}(k) \\ \mathbf{y}(k) &= C_d\mathbf{x}(k) + D_d\mathbf{u}(k) \end{aligned} \qquad (2)$$

where A_d and Bd are the state and control matrices for the discrete state-space equation, respectively, which can be calculated with the Euler method as

$$A_d = e^{A_c T_s}, \qquad B_d = \int_{kT_s}^{(k+1)T_s} e^{A_c[(k+1)T_s - \eta]} B_c d\eta$$

where T_s is the sampling interval for the discrete state-space model. The matrices C_d and D_d are equivalent to those in the continuous case. For simplicity, we assume that all the states are measurable and the ATLASCAR2 drives east with a constant speed at the nominal operating point. In the scenario we are going to consider, the road is straight and our vehicle stays in the middle of the center lane when not passing. Without losing generality, the ATLASCAR2 passes an obstacle both to the right and to the left lane depending on where it is placed on the road. We create also a safe zone around the obstacles so that the vehicle does not get too close the osbtacle when passing it.

B. Design of Adaptive Model Predictive Control

We designed a model predictive controller that can make the ATLASCAR2 mantain a desired velocity and stay in the middle of center lane. We used an Adaptive MPC controller because it handles the nonlinear vehicle dynamics more effectively than a traditional MPC controller in fact the latter uses a constant plant model, however, the former allows us to provide a new plant model at each control interval. Because the new model describes the plant dynamics more accurately at the new operating condition, an adaptive MPC controller performs better than a traditional MPC controller. In practice at each control interval, the adaptive MPC controller updates the plant model and the nominal conditions. Once updated, the model and the conditions remain constant over the prediction horizon. At the beginning, we specified the constraints for the manipulated variables: to prevent the ATLASCAR2 from accelerating or decelerating too quickly, we added an hard constraint on the throttle rate of change and another one on the steering angle rate of change. We used an approach that takes advantage of the ability of MPC to handle constraint explicitly. When an obstacle is detected, it

defines an area on the road (in terms of constraints) that the ATLASCAR2 must not enter during the prediction horizon. At the next control interval the area is redefined based on the new positions of the vehicle and the obstacle until passing is completed. To define the area to avoid, we used the following mixed Input/Output constraints:

$$E\mathbf{u} + F\mathbf{y} \leq G$$

where $\mathbf{u}$ and $\mathbf{y}$ are respectively the manipulated variable vector and the output variable vector, while E, F, G are the matrices that can be updated when the controller is running:

- the first constraint is an upper bound on the y-coordinate (right boundary of the road);
- the second constraint is a lower bound on the y-coordinate (left boundary of the road);
- the third constraint is for the obstacle avoidance; even though no obstacle is detected at the nominal condition, we must add this fake constraint here because we cannot change the dimensions of the constraint matrices at run time (it has the same form as the second one).ù

The matrices for the above inequality are the following:

$$E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ cS & -1 & 0 & 0 \end{bmatrix}, \quad G = \begin{bmatrix} W/2 \\ W/2 \\ -cI \end{bmatrix}$$

where W is the width of the road and cI and cS are the required parameters such that the ATLASCAR2 must be above the line formed from the vehicle to safe zone corner for left/right passing.

C. Simulation Results

TO DO

IV. LANE FOLLOWING

A. Problem Formulation

A lane-following system is a control system that keeps the vehicle traveling along the centerline of a highway lane, while maintaining a user-set velocity. The lane-following scenario is depicted in the following figure:

In a classic lane keeping assist, it is assumed that the longitudinal velocity is constant. This restriction is relaxed in this model because the longitudinal acceleration varies in this MIMO control system. This lane-following system manipulates both the longitudinal acceleration and the front steering angle of the vehicle to keep the
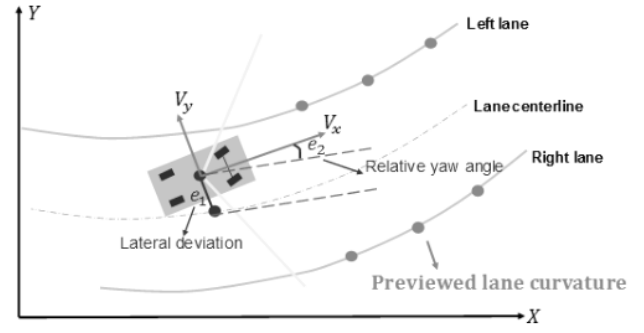

Fig. 2. Problem description of a lane following system.

lateral deviation and the relative yaw angle small and the longitudinal velocity close to a driver set velocity. If these two goals cannot be met at the same moment, the system tries to balance them.

The model that we are considering, contains a lot of parameters. The first fundamental block describes the vehicle dynamics: we have applied the bicycle model of lateral vehicle dynamics and approximate the longitudinal dynamics using a time constant:

1) Longitudinal dynamics:

$$G(s) = \frac{1}{\tau s^2 + s} \quad \text{where } \tau \text{ is a time constant} \quad (3)$$

2) Lateral dynamics: Local function: we have a continuous vehicle lateral model from parameters

$$\begin{aligned} \dot{\mathbf{x}}_{\text{lat}} &= A_g \mathbf{x}_{\text{lat}} + B_g \mathbf{u}_{\text{lat}} \\ \mathbf{y}_{\text{lat}} &= C_g \mathbf{x}_{\text{lat}} + D_g \mathbf{u}_{\text{lat}} \end{aligned} \quad (4)$$

where the input is the steering angle in radians, and the outputs are the lateral velocity in meters per second and yaw angle rate in radians per second.

$$\mathbf{x}_{\text{lat}} = \begin{bmatrix} V_y & \dot{\psi} \end{bmatrix}^T \qquad \mathbf{u}_{\text{lat}} = \delta$$

and

$$A_g = \begin{bmatrix} -\frac{2C_F + 2C_R}{mV_x} & -\frac{2C_F l_F - 2C_R l_R}{mV_x} - V_x \\ -\frac{2C_F l_F - 2C_R l_R}{I_Z V_x} & -\frac{2C_F l_F^2 + 2C_R l_R^2}{I_Z V_x} \end{bmatrix},$$

$$B_g = \begin{bmatrix} 2C_F/m \\ 2C_F l_F/I_Z \end{bmatrix}, \quad C_g = \mathbf{I}_2, \quad D_g = \mathbf{0}_{2 \times 1}.$$

The parameters that we are considering are the following ones:

- V_x is the longitudinal velocity of the car;
- m is the total mass parameter;
- I_Z is the yaw moment of inertia parameter;
- l_F and l_R are the longitudinal distances from center of gravity to front and rear tires parameters;

- C_F and C_R are the cornering stiffnesses of front and rear tires parameters.

The goal for the driver steering model is to keep the vehicle in its lane and follow the curved road by controlling the front steering angle . This goal is achieved by driving the yaw angle error $e_2 = \psi - \psi_{\text{des}}$ and lateral displacement error e_1 to zero ($\dot{e}_1 = V_x e_2 + V_y$). We can incorporate these two paramenters in the augmented model:

$$\begin{aligned} \dot{\mathbf{x}}_{\text{aug}} &= A_a \mathbf{x}_{\text{aug}} + B_a \mathbf{u}_{\text{aug}} \\ \mathbf{y}_{\text{aug}} &= C_a \mathbf{x}_{\text{aug}} + D_a \mathbf{u}_{\text{aug}} \end{aligned} \tag{5}$$

where

$$\mathbf{x}_{\text{aug}} = \begin{bmatrix} V_y & \dot{\psi} & e_1 & e_2 \end{bmatrix}^T, \qquad \mathbf{u}_{\text{aug}} = \begin{bmatrix} \delta & \dot{\psi}_{\text{des}} \end{bmatrix}^T$$

and

$$A_a = \begin{bmatrix} A_g & \mathbf{0}_{2\times2} \\ \begin{matrix} \mathbf{I}_2 \end{matrix} & \begin{matrix} 0 & V_x \\ 0 & 0 \end{matrix} \end{bmatrix}, \qquad B_a = \begin{bmatrix} B_g & \mathbf{0}_{2\times1} \\ 0 & 0 \\ 0 & -1 \end{bmatrix},$$

$$C_a = \begin{bmatrix} \mathbf{0}_{2\times2} & \mathbf{I}_2 \end{bmatrix}, \quad D_a = \mathbf{0}_{2\times2}.$$

Combining (3) with (5) yields the state-space model that characterizes the model predictive controller:

$$\begin{aligned} \dot{\mathbf{x}}_{\text{tot}} &= A_f \mathbf{x}_{\text{tot}} + B_f \mathbf{u}_{\text{tot}} \\ \mathbf{y}_{\text{tot}} &= C_f \mathbf{x}_{\text{tot}} + D_f \mathbf{u}_{\text{tot}} \end{aligned} \tag{6}$$

where

$$\mathbf{x}_{\text{tot}} = \begin{bmatrix} V_x \\ \dot{V}_x \\ V_y \\ \dot{\psi} \\ e_1 \\ e_2 \end{bmatrix}, \qquad \mathbf{u}_{\text{tot}} = \begin{bmatrix} a \\ \delta \\ \dot{\psi}_{\text{des}} \end{bmatrix}$$

and

$$A_f = \begin{bmatrix} \begin{matrix} -2 & 0 \\ 1 & 0 \end{matrix} & \mathbf{0}_{2\times4} \\ \mathbf{0}_{4\times2} & A_a \end{bmatrix}, \qquad B_f = \begin{bmatrix} \begin{matrix} 2 \\ 0 \end{matrix} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{4\times1} & B_a \end{bmatrix},$$

$$C_f = \begin{bmatrix} \begin{matrix} 0 & 1 \end{matrix} & \mathbf{0}_{1\times4} \\ \mathbf{0}_{2\times2} & C_a \end{bmatrix}, \qquad D_f = \mathbf{0}_{3\times3}.$$

However the system to be controlled is usually modeled by a discrete state-space model:

$$\begin{aligned} \mathbf{x}_{\text{tot}}(k+1) &= A\mathbf{x}_{\text{tot}}(k) + B\mathbf{u}_{\text{tot}}(k) \\ \mathbf{y}_{\text{tot}}(k) &= C\mathbf{x}_{\text{tot}}(k) + D\mathbf{u}_{\text{tot}}(k) \end{aligned} \tag{7}$$

where A and B are the state and control matrices for the discrete state-space equation, respectively, which can be calculated, also in this case, with the Euler method as:

$$A = e^{A_f T_s}, \qquad B = \int_{kT_s}^{(k+1)T_s} e^{A_f[(k+1)T_s - \eta]} B_f d\eta$$

where T_s is the sampling interval for the discrete state-space model. The matrices C and D are equivalent to those in the continuous case.

B. Design of Adaptive Model Predictive Control

We created an Adaptive MPC controller with a prediction model that has six states, three outputs (longitudinal velocity, lateral deviation, relative yaw angle), and two inputs. The model has two manipulated signals: acceleration and steering. The product of the road curvature and the longitudinal velocity is modeled as a measured disturbance. We have set the constraints for manipulated variables and the scale factors. Moreover we have specified the weights in the standard MPC cost function. The third output, yaw angle, is allowed to float because there are only two manipulated variables to make it a square system. In this controller, there is no steady-state error in the yaw angle as long as the second output, lateral deviation, reaches 0 at steady state. Finally we have also penalized acceleration change more for smooth driving experience. This controller uses a linear model for the vehicle dynamics and updates the model online as the longitudinal velocity varies.

C. Simulation Results

TO DO

At the start of the simulation, this velocity is equal to the Initial condition for longitudinal velocity parameter. At run time, this velocity is equal to the Longitudinal velocity input signal.

V. CONCLUSIONS AND FUTURE WORK

In this work we proposed two advanced methods for the ATLASCAR2 based on adaptive Model Predictive Control. In the first case we analyzed an obstacle avoidance system that moves the vehicle around different moving obstacles in the lane using throttle and steering angle. This system updates both the predictive model and the mixed input/output constraints at each control interval. Instead, in the second scenario we have developed a lane following system that keeps the ATLASCAR2 travelling along the centerline of the lanes on the road by adjusting the front steering angle of the car. The flexibility of the concepts used in these methods allows a multitude of refinements and extensions to this project. For example the next possible work cuold be to combine the two previous control strategies in a way that they can operate simultaneously. Moreover the final method could be tested in a real autonomous driving scenario.

REFERENCES

[1] J. Ji, A. Khajepour, W. W. Melek and Y. Huang, "Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control With Multiconstraints," in IEEE Transactions on Vehicular Technology, vol. 66, no. 2, pp. 952-964, Feb. 2017.

[2] M. Werling and D. Liccardo, "Automatic collision avoidance using model-predictive online optimization," 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), Maui, HI, 2012, pp. 6309-6314.

[3] Wei Xi and J. S. Baras, "MPC based motion control of car-like vehicle swarms," 2007 Mediterranean Conference on Control & Automation, Athens, 2007, pp. 1-6.

[4] J. V. Frasch et al., "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," 2013 European Control Conference (ECC), Zurich, 2013, pp. 4136-4141.

[5] T. Xu and H. Yuan, "Autonomous vehicle active safety system based on path planning and predictive control," 2016 35th Chinese Control Conference (CCC), Chengdu, 2016, pp. 8889-8895.

[6] N. Wada and T. Matsumoto, "Driver assistance for collision avoidance by constrained MPC," 2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Kanazawa, 2017, pp. 90-93.

[7] Y. Nishio, K. Nonaka and K. Sekiguchi, "Moving obstacle avoidance control by fuzzy potential method and model predictive control," 2017 11th Asian Control Conference (ASCC), Gold Coast, QLD, 2017, pp. 1298-1303.

[8] Yu, S., Li, X., Chen, H. and Allgöwer, F. (2015), "Nonlinear model predictive control for path following problems". Int. J. Robust Nonlinear Control, 25: 1168–1182.