Class Project: Nonlinear MPC with reference Bezier Curve in Autonomous Driving

Seoyeon Choi
Mechanical Engineering in Yonsei University

I. Introduction

In this report, a methodology for the class project, using a nonlinear model predictive control (NMPC) with a reference Bezier curve is presented. The environment of the class project is set up as a random created tracks in a single-car environment. The objective of the class project was to minimize the lap time of any arbitrary curve with given information of the track's centerline and the vehicle's model. The presented methodology using Bezier curves and NMPC showed the best results among all the other class project presented methods.

II. Problem Formulation

A. Vehicle Model

A bicycle model was given for the vehicle model, where the vehicle is modeled as a rigid body with a mass m and an inertia I_z, and the symmetry of the car is used to reduce it to a bicycle. Since the objective is to control a vehicle in a 2D plane, the pitch and roll dynamics are neglected. The model is described as the resulting differential equations (1).

$$\begin{aligned}
\dot{X} &= v_x cos\phi - v_y sin\phi \\
\dot{Y} &= v_x sin\phi + v_y cos\phi \\
\dot{\phi} &= \omega \\
\dot{v_x} &= \frac{1}{m}\left(F_{r,x} - F_{f,y} sin\delta + m v_y \omega\right) \\
\dot{v_y} &= \frac{1}{m}\left(F_{r,y} + F_{f,y} cos\delta - m v_x \omega\right) \\
\dot{\omega} &= \frac{1}{I_z}\left(F_{f,y} l_f cos\delta - F_{r,y} l_r\right)
\end{aligned} \tag{1}$$

In the equation of motion, the states are the position X, Y of the center of gravity in the inertial frame and ϕ as the angle of the car relative to the inertial frame. Also, longitudinal and lateral velocity of the car v_x, v_y and the yaw rate ω is used to define the states. The control inputs are the throttle of the vehicle denoted as d and the steering angle δ. The subscripts x, y are the longitudinal and lateral direction, while f, r refer to the front and rear tires. l_f and l_r denotes the distance from the center of gravity to the front and rear wheel. The tire forces F are modeled using a simplified Pacejka Tire Model.

III. Methods

To control the vehicle, it is crucial to take the vehicle model into account. One of the best ways that is proven in many papers is to use a model predictive control. To use a MPC, one must create a reference to minimize with the MPC. For this problem Bezier curve is used to show a more optimal path than the track's centerline.

A. Construction of reference Bezier curves

For the full formation of the MPC, a reference position at each timestep is needed for the MPC horizon. Two 3rd order Bezier curves are needed for reference trajectory: (1) position profile $p(s)$; (2) Curve parameter profile given normalize time $s(t)$.

For Bezier curve for the position profile, 4 nodes are needed to create a 3rd order curve. The first node P_0 would be the current position of the vehicle, P_1 being the point along the line where the vehicle is headed. P_2 is a point along the tangent line of the centerline at P_3 and the last P_4 is set as a loookahead point in the centerline. The resulting position profile curve is shown in Figure 1, and each position x, y of the nodes is obtained by equation (2) where the centerline of

$$\begin{aligned}
\theta_N &= \theta_0 + \theta_H \\
[x_N, y_N] &= c(\theta_N) \\
\psi &= \tan^{-1} c'(\theta_N) - \phi \\
\boldsymbol{P_3} &= \begin{bmatrix} cos\phi & sin\phi \\ -sin\phi & cos\phi \end{bmatrix} [x_N - x_0 \quad y_N - y_0] \\
\boldsymbol{P_1} &= [l_{01}, 0] \\
\boldsymbol{P_2} &= \boldsymbol{P_3} - l_{23}[cos\psi, sin\psi]
\end{aligned} \tag{2}$$

the track is expressed as $[x, y] = c(\theta)$ as θ indicates the arc length parameter when the track is expressed in path parametric form and $c'(\theta)$ indicates the tangent angle. Each l_{01} and l_{23} indicates the length between each node, which are parameters to be turned. Also θ_H is another parameter to tune, which indicates the lookahead distance. θ_N can also be seen as the horizon length of the MPC. By getting these nodes profile curve $p(s)$ is constructed as equation (3).

$$\boldsymbol{p(s)} = (1-s)^3 \boldsymbol{P_0} + 3(1-s)^2 \boldsymbol{P_1} + 3(1-s)s^2 \boldsymbol{P_2} + s^3 \boldsymbol{P_3} \; (0 \le s \le 1) \tag{3}$$

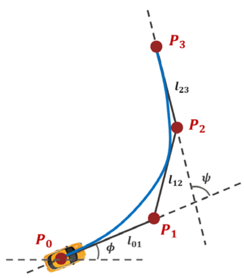


Figure 1. A schematic drawing of the Bezier curve for the position profile.

The profile of the curve parameter s is needed since we need to pinpoint where the position will be in a specific timestep. Therefore, for the curve parameter profile $s(t)$, the nodes are constructed as in equation (4).

$$\begin{aligned}
\boldsymbol{s_0} &= 0, \boldsymbol{s_3} = 1 \\
\boldsymbol{s_1} &= \frac{1}{3} \frac{vx(t=0)}{\left\| \frac{dp}{ds}(s=0) \right\|_2} \cdot N \cdot dt \\
\boldsymbol{s_2} &= \tau
\end{aligned} \tag{4}$$

τ is a parameter to be tuned bounded by [0, 1]. The final curve parameter profile can be constructed as $\boldsymbol{s(t)} = (1-\alpha)^3 \boldsymbol{s_0} + 3(1-\alpha)^2 \boldsymbol{s_1} + 3(1-\alpha)\alpha^2 \boldsymbol{s_2} + \alpha^3 \boldsymbol{s_3} \; (0 \le \alpha \le 1)$.

Identify applicable funding agency here. If none, delete this text box.

B. MPC

Sequential quadratic programming (SQP) based nonlinear model predictive control is used. Since the controller has to compute in real-time, online method real-time iteration (RTI) scheme [1] is used. The states for the MPC are set up as $x_t = [x(t), y(t), \phi(t), vx(t), vy(t), \omega(t), \delta(t), D(t)]$ and the controls intended to solved by the MPC are denoted as $u_t = [\dot{\delta}, \dot{D}]$.

To solve the control objective, we solve an OCP arising from a multiple shooting formulation with the Gauss-Newton Hessian approximation with sampling time $dt = 0.05s$ and $N = 30$ shooting intervals. The vehicle model functions are denoted as f and the reference positions are denoted as y. The OCP then reads as

$$\begin{aligned} \underset{u_0,\dots,u_{N-1}}{\text{minimize}} \quad & \sum_{k=0}^{N-1} \left\| f(x_k, u_k) - y_{f,k} \right\|_W^2 + \left\| f_N(x_N) - y_{f,N} \right\|_{W_N}^2 \qquad (5) \\ \text{subject to} \quad & x_0 = \bar{x}_0 \\ & \underline{x} \le x_k \le \overline{x} \qquad k = 0, \dots, N-1 \\ & \underline{u} \le u_k \le \overline{u} \qquad k = 0, \dots, N-1 \end{aligned}$$

The weight (cost) matrix are designed as a linear LS where W is set as a diagonal matrix with [1e0, 1e0, 1e-2, 1e-2, 1e-8, 1e-8, 1e-3, 5e-3, 1e-3, 5e-3] as diagonal elements and W_N is also a diagonal matrix with [1e0, 1e0, 1e-2, 1e-2, 1e-8, 1e-8, 1e-3, 5e-3] as diagonal elements.

The MPC repeatedly solve the OCP (5) approximately by performing real time iterations. As an underlying QP solver, used partial condensing HPIPM. More detailed information about the MPC can be seen in reference [2].

C. Full methodology

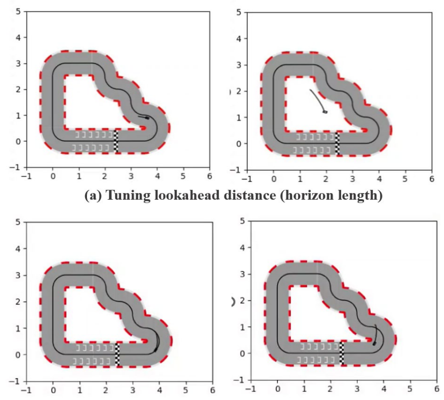
At a timestep, a reference Bezier curve is generated for a horizon length θ_H. Then MPC reads this Bezier curve as a reference and computes the control needed for N steps with a time dt in a horizon. The first control value (the first control value in N step is selected as the control value for this time step. In the continued time step the same procedure is repeated.

IV. Experiments and Results

A. Parmeter Tuning

As described in section III.A., there were a few parameters to be tuned. Tuning the lookahead distance, or the horizon length had effects on the converging behavior of the QP solver. If set to a small number such as 0.2, the QP solver will converge. The simulation result can be seen in the left figure of Figure 2.(a). However, the vehicle will drive close to the centerline which will not result in a short lap time. If set too large such as 0.6, the QP solver diverges and could not obtain any reasonable control outputs. The vehicle will just fly out of the track as seen in the right figure of Figure 2.(a).

Tuning the length between the nodes of the Bezier curve had effects on the manner the vehicle drives. If l_{01} is set larger than l_{23}, the vehicle will drive in a smoother manner, driving far from the track's border at the track's corners. The left figure of Figure 2.(b) shows that the vehicle will have a smooth turn. If l_{01} is set smaller than l_{23}, the vehicle will drive in a more aggressive manner, making abrupt turns at the corners as shown in the right figure of Figure 2.(b).



(a) Tuning lookahead distance (horizon length)

(b) Tuning length between the nodes of the Bezier curve

Figure 2. A figure showing the difference in how the vehicles drive when set the parameters differently.

The tuned parameters of the algorithm are described below.

- Parameter for the time profile curve : 0.75
- Lookahead distance (horizon length) : 0.48
- Length between nodes of the Bezier curve for the time profile : $l_{01} = \frac{1}{3}(horizon\ length)$
  $l_{23} = \frac{1}{3}(horizon\ length)$

B. Comparing with Other Algorithms

Table 1. The result of the class project

	RL based methods	**Other method using MPC**	**MPC with reference Bezier curve**
Taken timesteps to finish 1 lap	Minimum 514 Average 842.86 Maximum 1524	514	**384**

The results of the lap time of all the algorithms can be seen in Table 1. Most of the students participating in the class project used reinforcement learning. However, due to not enough training and not effective reward design, most of the results showed oscillation and slow racing performance. There was a method that used a MPC as a controller but failed to reach my algorithm's result since it didn't use any reference trajectory with the MPC.

MPC with reference Bezier Curve showed the best lap time while almost no oscillations occurred. Also, the designed controller could make the vehicle drive safely without colliding in to the track's boundary for any arbitrary tracks. It can be said that MPC with reference Bezier Curve is the best algorithm among the class participants.

V. Conclusion

In this report, an algorithm was created for the class project which objective is to design a controller that can make the vehicle drive in the smallest lap time for any arbitrary tracks. The procedure of creating the MPC with reference Bezier curve is shown in the report. MPC was used to take the vehicle's model into account and reference Bezier curve was used to lessen the oscillation and to make the lap time shorter. This algorithm showed the best results among the other algorithms created for the class project.

REFERENCES

[1] Diehl, M., Bock, H.G., Schl¨oder, J.P., Findeisen, R., Nagy, Z., Allg¨ower, F.: Realtime optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. Journal of Process Control 12(4), 577–585 (2002)

[2] Verschueren, R., Frison, G., Kouzoupis, D. et al. acados—a modular open-source framework for fast embedded optimal control. Math. Prog. Comp. 14, 147–183 (2022).