Model Predictive Control in Adaptive Cruise Control

Shuhao Bian

April 9, 2023

Abstract

This paper presents a method of using MPC into Adaptive Cruise Control (ACC). The MPC is a powerful tool for solving the optimal control problem. We present a linear model of the system.

1 Problem Discreption

Adaptive Cruise Control (ACC) is an advanced driver assistance system designed to automatically regulate a vehicle's speed in order to maintain a safe following distance from the vehicle ahead. By utilizing a combination of sensors, such as radar or LiDAR, and cameras, the ACC system continuously monitors the relative velocity and distance between the two vehicles, as illustrated in the accompanying figure. When the system detects that the vehicle ahead is slowing down or if the following distance becomes too short, it automatically adjusts the vehicle's speed by applying the brakes or reducing throttle input. Conversely, when the traffic flow resumes or the distance increases, the system will accelerate the vehicle back to its pre-set speed. This not only enhances driving comfort and convenience but also contributes to improved road safety and traffic flow efficiency.

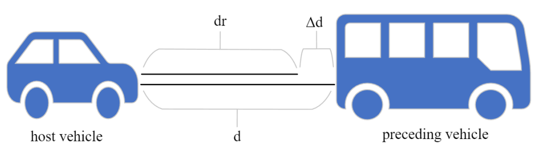

Figure 1: The structure of the MPC controller

Although Model Predictive Control (MPC) [1] is a popular method for solving the ACC problem, it still presents challenges due to the computational load involved in solving large-scale optimization problems. Therefore, this paper proposes a method to linearize the ACC system model and use the powerful CasADi toolbox[2] for solving the optimal control problem.

2 Formulation

2.1 ACC system

Adaptive Cruise Control (ACC) is a driver

In this paper, there are three parameters that can be measured as below by the ACC system:

- Δd: Inter-vehicle distance following error, it represents the error between the desired inter-vehicle distance d_r and actual inter-vehicle distance d. st. $\Delta d = d - d_r$. It's supposed to converge to zero if the ACC system works well.
- Δv: The speed following error, it represents the speed error between the host vehicle speed v_h and target vehicle speed v_r. st. $\Delta v = v_h - v_r$. It's supposed to converge to zero.
- $\dot{v_h}$: the acceleration of the host vehicle. It's supposed to converge to be zero if target vehicle speed v_r is constant. Because it's a simple linear model, we suppose the velocity of the target vehicle is constant.

2.2 Vehicle Dynamics

A vehicle dynamics model[3] will be used to describe the vehicle dynamics and to build the MPC. The logitudinal dynamics of the host vehicle is given by (2.1):

$$m\dot{v_h} = F_f - r_{travel} \tag{2.1}$$

where m is the mass of the host vehicle, F_f is the total traction force of the host vehicle, r_{travel} is the rolling resistance of the host vehicle. The input/output relationship of the vehicle dynamics model is given by the differential equation (2.2):

$$\begin{cases} \dot{x}_f = f_{act}\left(x_f, u\right) \\ a_f = h_{act}\left(x_f\right) \end{cases} \tag{2.2}$$

where $x_f = \mathbb{R}^{n_f}$ is the state of the vehicle dynamics model, u is the input of the vehicle system (2.2) which is often a control input calculated by embeded micro cruise controller. r_{travel} is travel resistance which can be calculated by the following equation (2.3):

$$r_{travel} = r_{air}v_h^2 + r_{roll}(v_h) + r_{accel}\dot{v_h} + r_{grad}(\theta) \tag{2.3}$$

where r_{air} is the air resistance coefficient, r_{roll} is the rolling resistance, the r_{accel} is the acceleration resistance coefficient, r_{grad} is the gradient resistance, θ is the slope angle.

2.3 State-Space Model for ACC System

To construct a plant model of the Adaptive Cruise Control (ACC) system, it is necessary to consider both the desired inter-vehicle distance d_r and the minimum inter-vehicle distance d_0.

$$d_r = T_{hw}v_h + d_0 \tag{2.4}$$

where T_{hw} is the headway time, d_0 is the minimum inter-vehicle distance.

In this paper, the state variables of the ACC system are as $x = [\Delta d, \Delta v, x_f^T]^T$, $x \in \mathbb{R}^{2+n_f}$, the state-space model is formulated as (2.5):

$$\begin{cases} \dot{x} = f_{act}\left(x, u\right) + Gv + Hw \\ y = Cx + Jv \end{cases} \tag{2.5}$$

where $f_{act}(x,u) = \begin{bmatrix} \Delta v - T_{hw}x_f \\ -h_{act}\left(x_f\right) \\ f_{act}\left(x_f, u\right) \end{bmatrix}$, $G = \begin{bmatrix} T_{hw}/m \\ 1/m \\ 0 \end{bmatrix}$, $H = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, $C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $J = \begin{bmatrix} 0 \\ 0 \\ -1/m \end{bmatrix}$.

where $u \in \mathbb{R}$ is the input of the plant and $y = [\Delta d, \Delta v, \dot{v_h}]^T \in \mathbb{R}^3$ is the output of the plant.

3 Optimal-based Control Method

3.1 Controller Model Design

In order to reduce the computational cost, the ACC system model is linearized as (3.1):

$$\begin{cases} \dot{a_f} = A_f(t)a_f + B_f(t)u \\ a_f = C_f x_f \end{cases} \tag{3.1}$$

where the acceleration of the linearized model is $a_f \in \mathbb{R}$, the input of the linearized model is u, the output of the linearized model is a_f and $A_f(t)$, $B_f(t)$, C_f are shown as (3.2):

$$
\begin{aligned}
A_f(t) &= \begin{cases} -\frac{1}{T_{eng}}, & \text{if } u(t) \geq a_{thr\_off} \\ -\frac{1}{T_{brk}}, & \text{if } u(t) < a_{thr\_off} \end{cases} \\
B_f(t) &= \begin{cases} \frac{K_{eng}(t)}{T_{eng}}, & \text{if } u(t) \geq a_{thr\_off} \\ \frac{K_{brk}(t)}{T_{brk}}, & \text{if } u(t) < a_{thr\_off} \end{cases} \\
C_f &= 1.
\end{aligned} \tag{3.2}
$$

where T_{eng} is the constant of acceleration of acceleration engine, T_{brk} is time constant of deceleration using brake, a_{thr_off} is the threshold of acceleration, $K_{eng}(t)$ and $K_{brk}(t)$ are the gain of acceleration engine and brake respectively.

In the system (3.1), the dynamics of the vehicle is simplified as a first-order delay system. The system has two modes:a acceleration mode and a deceleration mode.

So the prediction model (2.5) can be simplified as (3.3).

$$
\begin{cases} \dot{x} = A(t)x + B(t)u + Gv + Hw \\ y = Cx + Jv \end{cases} \tag{3.3}
$$

where $x \in \mathbb{R}^3$ and $A(t) = \begin{bmatrix} 0 & 1 & -T_{hw} \\ 0 & 0 & -1 \\ 0 & 0 & A_f(t) \end{bmatrix}, \quad B(t) = \begin{bmatrix} 0 \\ 0 \\ B_f(t) \end{bmatrix}$.

3.2 Optimization Problem

3.3 MPC Controller Design

4 Preliminary Results

5 Conclusion

In this paper, we proposed a MPC controller for ACC system. The simulation results show that the proposed controller can achieve the desired performance.

What's more, we can adjusts

References

[1] J Rawlings and D Mayne. Postface to model predictive control: Theory and design. *Nob Hill Pub*, 5:155–158, 2012.

[2] Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, In Press, 2018.

[3] Taku Takahama and Daisuke Akasaka. Model predictive control approach to design practical adaptive cruise control for traffic jam. *International Journal of Automotive Engineering*, 9(3):99–104, 2018.