Tracking Control of Intelligent Vehicle Lane Change Based on RLMP

Quanshan Hou^{1,2*}, Yanan Zhang^{1,2}, Shuai Zhao^{1,2}, Yunhao Hu^{1,2} and Yongwang Shen^{1,2}

¹China Automotive Technology Research Center Co.,Ltd, 30093 Tianjin, China

²Automotive Data of China (Tianjin) Co.,Ltd, 30093 Tianjin, China

Abstract. Autonomous lane changing, as a key module to realize high-level automatic driving, has important practical significance for improving the driving safety, comfort and commuting efficiency of vehicles. Traditional controllers have disadvantages such as weak scene adaptability and difficulty in balancing multi-objective optimization. In this paper, combined with the excellent self-learning ability of reinforcement learning, an interactive model predictive control algorithm is designed to realize the tracking control of the lane change trajectory. At the same time, two typical scenarios are verified by PreScan and Simulink, and the results show that the control algorithm can significantly improve the tracking accuracy and stability of the lane change trajectory.

1 Preface

Intelligent vehicles perceive the environment around the vehicle through advanced on-board sensors, and at the same time rely on network communication technology and actuators for vehicle decision-making control [1], so that related driving behaviors can be completed by the vehicle autonomously.

On highways, any complicated driving behavior can be decomposed into two types: lane changing and lane keeping. The accuracy, stability and conversion efficiency of vehicle lane change have important practical significance for vehicle driving safety and solving traffic jams. Nowadays, with the development of artificial intelligence and the proposal of intelligent transportation, the unpredictable human factors are excluded. How to make the vehicle perceive the surrounding environment through advanced sensors, and at the same time design a trajectory tracking algorithm that meets the requirements of high robustness and high real-time to control the vehicle, is one of the current focuses of ICV research.

Due to the influence of many factors during the movement of intelligent vehicles, when designing the control model, most researchers generally constrain the target through approximation and hypothesis, and carry out relevant research on this basis. Hatipoglu et al^[2] assumed that the longitudinal speed change during the vehicle movement is small, and set it as a fixed value. Through the analysis of the vehicle yaw rate change, the lateral control and the longitudinal control during the lane change process are decoupled and designed. A variable trend controller based on changes in yaw rate is presented. Emirler et al^[3] designed an improved parameter self-tuning lateral controller based on PID

algorithm through the control of lateral state variables, and verified it in a simulation environment. The results showed that the controller has good performance on curves. Tracking effect. Ren Dianbo et al^[4] assumed that the lateral acceleration of the vehicle during the lane change satisfies the trapezoidal constraint, based on the vehicle's horizontal and longitudinal coupled control model, solved the given target trajectory and calculated the expected yaw rate and actual yaw rate during the lane change. Angular velocity deviation, while relying on the on-board GPS to obtain the status information of the main vehicle, and using the finite time synovial membrane approaching law to design the tracking control of the lane-changing trajectory. Qiu Shaolin et al^[5] added a "vehicle-road" model on the basis of preview control theory. At the same time, a trajectory tracking controller is designed based on the linear quadratic regulator theory, which improves the tracking accuracy. Petrov et al^[6] built a nonlinear adaptive controller based on the vehicle dynamics model for the situation of overtaking and changing lanes, but the state update of the controller depends too much on the surrounding environment.

With the popularization of Model Predictive Control (MPC) theory, with its estimation of the future state and real-time update of the current state, as well as the natural advantages of dealing with multi-objective constraints, more and more scholars have begun to try to use it to go to the tracking control of the trajectory. In order to improve the tracking accuracy of lane change trajectory, Yoon et al^[7] designed a prediction module by using a non-linear tire model and a vehicle model, and built a non-linear controller based on MPC to track the trajectory in real time. Zhang Wei simplified the vehicle dynamics model in order to reduce computing resources

* Corresponding author: houquanshan@catarc.ac.cn

and improve the real-time performance of trajectory tracking, and built a trajectory tracking controller that actively changes lanes to avoid collisions, which enables the vehicle to track the target path stably and quickly.

In order to improve the scene adaptability of the control model under the premise of ensuring control accuracy and real-time performance, this paper combines the reinforcement learning theory to study an interactive model predictive control algorithm (Reinforcement Learning Model Predictive Control, RLMPC), which can make full use of RL's free The learning ability and MPC's multi-constraint processing ability ensure the safety and stability of the vehicle lane change.

2 RLMPC trajectory tracking control algorithm design

Tracking control of trajectory is the key to realize autonomous lane change. Compared with most controllers, MPC has natural advantages in dealing with multiple constraints, but it has the disadvantage of weak scene adaptability. Therefore, this section is based on MPC, combined with reinforcement learning to improve its algorithm.

2.1. Overall structure

In the control process of MPC, its internal predictive model $\phi[\cdot]$ plays a decisive role in its control performance. The predictive model realizes model predictive control by predicting the future control sequence, but it is extremely susceptible to external interference factors. Therefore, traditional predictive models such as ARIMA model and BP neural network model cannot meet the actual control requirements, and the algorithm complexity is relatively high. Reinforcement learning has the ability of interactive learning with the external environment, which makes the MPC prediction model based on reinforcement learning more accurate prediction effects and has the ability to reflect the external objective environment in real time.

Aiming at this feature, this article will study an interactive model predictive control algorithm based on reinforcement learning, and its internal basic structure is shown in Figure 1. In the RLMPC controller, the reinforcement learning structure is mainly composed of three parts: control decision, agent and reward function.

Fig. 1. RLMPC control principle.

2.2 Algorithm design

RLMPC includes four parts: reference trajectory module, rolling optimization module, prediction module based on reinforcement learning, and feedback correction module.

2.2.1 Reference trajectory design

This article is based on the fifth-order polynomial model to plan the lane change trajectory. In the vehicle body coordinate system, the horizontal and vertical lane change trajectories can be expressed as:

$$\begin{cases} x(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0, t \in [0, t_f] \\ y(t) = b_5 t^5 + b_4 t^4 + b_3 t^3 + b_2 t^2 + b_1 t + b_0, t \in [0, t_f] \end{cases} \quad (1)$$

Define the time parameter matrix as shown in formula (2):

$$T_{6 \times 6} = \begin{bmatrix} t_0^5 & t_0^4 & t_0^3 & t_0^2 & t_0 & 1 \\ 5t_0^4 & 4t_0^3 & 3t_0^2 & 2t_0 & 1 & 0 \\ 20t_0^3 & 12t_0^2 & 6t_0 & 2 & 0 & 0 \\ t_f^5 & t_f^4 & t_f^3 & t_f^2 & t_f & 1 \\ 5t_f^4 & 4t_f^3 & 3t_f^2 & 2t_f & 1 & 0 \\ 20t_f^3 & 12t_f^2 & 6t_f & 2 & 0 & 0 \end{bmatrix} \quad (2)$$

Then the coefficient matrix sum satisfies the boundary condition of formula (2):

$$\begin{cases} [x_0 \ x_0' \ x_0'' \ x_f \ x_f' \ x_f'']^T = T_{6 \times 6} A^T \\ [y_0 \ y_0' \ y_0'' \ y_f \ y_f' \ y_f'']^T = T_{6 \times 6} B^T \end{cases} \quad (3)$$

The vehicle should drive straight along the current lane before performing lane change operations (Figure 2). Assuming that the initial position of the lane change is 0, according to the kinematics of the vehicle, the lateral displacement, lateral velocity, lateral acceleration, and longitudinal acceleration of the main vehicle are all 0 at this time. At the end of the lane change, the vehicle is driving in a straight line in the target lane. At this time, the host vehicle should satisfy that the lateral velocity, acceleration, and longitudinal acceleration are all 0.

Fig. 2. Schematic diagram of lane change track.

The longitudinal speed of the main vehicle does not change during the lane change, so the conditional parameters of the starting and ending points are:

$$\begin{cases} [x_0 \ x_0' \ x_0'' \ x_f \ x_f' \ x_f''] = [0 \ v_{x0} \ 0 \ L \ v_{x0} \ 0] \\ [y_0 \ y_0' \ y_0'' \ y_f \ y_f' \ y_f''] = [0 \ 0 \ 0 \ W \ 0 \ 0] \end{cases} \quad (4)$$

Bring it into the original matrix to solve, you can get the predefined polynomial lane change trajectory:

$$\begin{cases} x(t) = v_0 t, t \in [0, t_f] \\ y(t) = \frac{6Wt^5}{t_f^5} - \frac{15Wt^4}{t_f^4} + \frac{10Wt^3}{t_f^3}, t \in [0, t_f] \end{cases} \quad (5)$$

2.2.2 Rolling optimization module design

As shown in formula (6), the design of the rolling optimization module mainly uses optimization algorithms to calculate its minimum. Traditional optimization algorithms require continuous iterative optimization to get the optimal solution, and their computational efficiency is low. Therefore, this article will adopt the MPC online fast optimization strategy based on the effective set method.

$$\min J = \sum_{i=1}^P \|y_m(k+i) - y_r(k+i)\| + \sum_{i=1}^N \|u(k+i|k)\|_R^2 + \sum_{i=1}^N \|\Delta u(k+i|k)\|_S^2 \quad (6)$$

In order to obtain the optimal control sequence of the MPC controller, formula (6) is used to optimize the minimum value corresponding to the objective function calculator and to prevent sudden changes in the control quantity. Therefore, the following constraints are added to formula (7):

$$u_{\min}(k+i) \leq u(k+i|k) \leq u_{\max}(k+i) \quad (7)$$

$$\Delta u_{\min}(k+i) \leq \Delta u(k+i|k) \leq \Delta u_{\max}(k+i) \quad (8)$$

$$y_{m,\min}(k+i) \leq y_m(k+i) \leq y_{m,\max}(k+i) \quad (9)$$

Also, because the quadratic objective function given by equation (6) is a nonlinear function, it needs to be converted into a QP problem based on a numerical method for optimal value calculation.

The MPC rapid optimization method based on the effective set method has the following steps:

1. Parameter initialization. After the given initial feasible point, and given the initial working set.

2. Iterative point update calculation. Assuming that a disturbance variable is used as the direction of the feasible point in the next iteration, the corresponding Lagrange multiplier satisfies the formula. $G u_k + g = -\sum_{i \in W_k} a_i \lambda_i^k$

3. If the disturbance variable is 0, then the feasible point can be calculated by constructing the least square expression corresponding to the effective set method:

$$\min \frac{1}{2} (u_k)^T G (u_k) + g^T (u_k) \quad (10)$$

Finally, the control law of rolling optimized output can be obtained:

$$u_{k+1} = u_k + \min \left\{ 1, \min_{\substack{i \in W_k \\ a_i^T p < 0}} \left(\frac{a_i^T u_k - b_i}{a_i^T p} \right) \right\} \times p \quad (11)$$

2.2.3 Predictive model design

Suppose that at time t , the agent of the reinforcement learning module receives the feedback output from the control object, that is, the environment state. Then the Agent makes the corresponding secondary control decision as the predictive control output of P steps in the future.

The specific process is as follows:

1. The agent gets the reward value. The feedback output of the control object obtains the reward value $r(t)$ through the following reward function.

$$r(t) = \exp \left[- \left(\left(\frac{x_1 - x_{E1}}{\lambda_1} \right)^2 + \left(\frac{x_2 - x_{E2}}{\lambda_2} \right)^2 \right) \right] \quad (12)$$

In the actual vehicle lane-change trajectory tracking process, in order to ensure that the vehicle can realize the lane change safely and stably, the controller must meet the low overshoot and the fast control convergence speed. If the vehicle is showing stable and fast trajectory tracking, the variables are closer to the expected overshoot and convergence time, and the reward value will be close to the maximum value of 1. If the vehicle has poor stability during the lane change and the lane change speed is slow, then the variables are far away from the expected overshoot and convergence time, and the reward value will be close to the minimum value of 0 at this time. Therefore, the reward value can reflect the control performance of the actual control system.

2. The agent calculates the decision sequence of the future trajectory according to the reward value and the environment state (in this article, the environment state is the center of mass slip angle, yaw angle and control output). The reinforcement learning algorithm used by the agent is the PPO (Proximal Policy Optimization) algorithm, and the strategy trajectory sequence formed is:

$$D_k = \{[s(1), a(1)], \dots, [s(T), a(T)]\} \quad (13)$$

The operation decision parameters at the next moment can be expressed as:

$$\theta_{k+1} = \arg \max_{\theta} \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^T \min \left(\frac{p_{\theta}(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t), g(\varepsilon, A^{\theta^k}(s_t, a_t)) \right) \quad (14)$$

The control decision for predicting P moments in the future can be expressed as:

$$\begin{cases} \theta_{k+1} = \arg \max_{\theta} \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^T \min \left(\frac{p_{\theta}(a_t|s_t)}{p_{\theta^k}(a_t|s_t)} A^{\theta^k}(s_t, a_t), g(\varepsilon, A^{\theta^k}(s_t, a_t)) \right) \\ \dots\dots\dots \\ \theta_{k+j} = \arg \max_{\theta} \frac{1}{|D_{k+j-1}|T} \sum_{\tau \in D_{k+j-1}} \sum_{t=0}^T \min \left(\frac{p_{\theta}(a_t|s_t)}{p_{\theta^{k+j-1}}(a_t|s_t)} A^{\theta^{k+j-1}}(s_t, a_t), g(\varepsilon, A^{\theta^{k+j-1}}(s_t, a_t)) \right) \\ \dots\dots\dots \\ \theta_{k+P} = \arg \max_{\theta} \frac{1}{|D_{k+P-1}|T} \sum_{\tau \in D_{k+P-1}} \sum_{t=0}^T \min \left(\frac{p_{\theta}(a_t|s_t)}{p_{\theta^{k+P-1}}(a_t|s_t)} A^{\theta^{k+P-1}}(s_t, a_t), g(\varepsilon, A^{\theta^{k+P-1}}(s_t, a_t)) \right) \end{cases} \quad (15)$$

Finally, input the decision parameters made by formula (15) into the feedback correction module for adjustment.

2.2.4 Feedback correction module design

After receiving the prediction model based on reinforcement learning, the feedback correction module will make the following adjustments:

$$y_{m1}(k+j) = y_m(k+j) + \theta_{k+j} \times [y(k) - y_m(k)] \quad (16)$$

In formula (16), when the adjustment decision value is small, the feedback correction adjustment amount is small, so the adjustment time of RLMPCC is longer, but the adjustment process is relatively stable; when the adjustment decision value is large, the feedback correction adjustment amount is larger, then The adjustment time is short, but it is prone to unstable adjustment; when the adjustment decision value is 0, the controller does not make adjustments, and its function is equivalent to the traditional MPC controller. Through this step, the accuracy of the prediction model can be guaranteed, so that the predicted value is close to the actual value.

3 Simulation verification

3.1. Verification platform and parameter setting

In order to verify the reliability of the trajectory tracking control algorithm designed in this paper, after the lane change model is built based on MATLAB / Simulink, the automatic driving simulation software prescan is is used for joint verification.

As shown in Figure 3, the software mainly includes three parts: GUI interface for scene editing, 3D visserver interface for virtual animation display and MATLAB / Simulink interface. Visink / server is used to define the main parameters of the traffic flow simulation, including the input and output of the traffic flow model, and the main input and output interface of the traffic flow simulation, including the input and output of the traffic flow model and the main traffic flow control interface.

Fig. 3. Prescan scene modeling process.

In the construction of the test scenario, the main vehicle selects Audi A8 as the test prototype. The sensor includes two millimeter wave radars, front and rear, and a monocular camera for target detection.

3.2. Verification platform and parameter setting

According to the traffic environment around the main vehicle, the simulation test scenarios are divided into two types: the scene with traffic vehicles in front and the scene with traffic vehicles in front and behind the side.

(1) There is a traffic scene ahead

According to China's highway standards, the lane width $W = 3.75\text{m}$; the main vehicle is located at the starting position of the middle lane, driving at the target speed of 80km/h ; the traffic vehicle is located at 100m in front of the current lane, driving at the target speed of 60km/h .

It can be seen from Fig. 4 that the traditional MPC controller has good tracking control effect in the process of lane change close to 10s . In terms of lateral parameters, the peak lateral velocity is less than 1m/s in this scenario, which ensures the stability of vehicles in the lane changing process. The lateral acceleration is overshoot in the initial stage, but it is stable in a reasonable range after 1s feedback regulation, and the control effect of rlmpc is more reliable.

Fig. 4. Lane change trajectory and error comparison.

Fig. 5. Comparison of lateral speed and error of lane change.

Fig. 6. Comparison of lateral acceleration and error.

(2) There is a traffic scene ahead

In this scenario, the main vehicle is located at the beginning of the middle lane and is driving at a constant speed of 90km/h ; traffic vehicle 1 is located 80m in front

of the current lane and is driving at a constant speed of 70km/h; traffic vehicle 2 is on the left Drive at a constant speed at a target speed of 100km/h at 50m behind the lane.

Through comparison, it can be seen that MPC, as a relatively mature control theory, has good results in high-speed trajectory tracking with multiple target constraints, but its robustness is still slightly lacking. In contrast, the RLMPC used in this article can achieve better tracking effects.

In terms of lateral parameter comparison, both the RLMPC controller proposed in this paper and the traditional MPC controller have excellent control effects at the initial stage of lane change, but as time goes by, when the lateral acceleration reaches the peak value, the traditional MPC control The device began to show obvious fluctuations, and quickly converged in the next few seconds, and the error accumulation became larger and larger. In addition, in terms of lateral acceleration comparison, the traditional MPC controller has experienced two surges. In comparison, the RLMPC controller used in this paper has a small cumulative error, can ensure rapid convergence, and has good robustness and stability.

Fig. 7. Lane change trajectory and error comparison.

Fig. 8. Comparison of lateral speed and error of lane change.

Fig. 9. Comparison of lateral acceleration and error.

4 Conclusion

In this paper, combined with the better self-learning ability of reinforcement learning, an interactive model predictive control algorithm is designed to realize the tracking control of the lane change trajectory. In order to verify its effectiveness and rationality, firstly, the fitting curve of the lane change trajectory was designed based on multiple scenes as a reference trajectory, and it was verified by PreScan and Simulink. The results show that the control algorithm can significantly improve the tracking accuracy and stability of the lane change trajectory.

References

1. Chen, J.Y., Li, R.B., Xin, X.Y.(2019)Review of research on intelligent evaluation of autonomous vehicles. Journal of Tongji University(Natural Science)., 12: 1785–1790+1824.
2. Hatipoglu, C., Ozguner, U., Redmill, K.(2003) Automated lane changing controller. IEEE Transactions on Intelligent Transportation Systems., 01: 13–22.
3. Emirler, M.T., İsmail, M.C.U., Güvenç, B.A.(2014) Robust PID steering control in parameter space for highly automated driving. International Journal of Vehicular Technology., 02: 1–8.
4. Ren, D.B., Zhang, J.M., Cui, S.M.(2007)Stability analysis of time-delay vehicle following system based on vector Liapunov function. J Traff Transp Eng., 04: 89–92.
5. Cui, S.L., Qian, L.J., Lu, J.H.(2019)Intelligent vehicle lane change control based on optimal preview. China Mechanical Engineering., 23: 2778–2783.
6. Petrov, P., Nashasbi, F.(2014)Modeling and nonlinear adaptive control for autonomous vehicle. IEEE Transactions on Intelligent Transportation Systems., 15:1643–1656.
7. Yoon, Y., Shin, J., Kim, H.J.(2009)Model-predictive active steering and obstacle avoidance for autonomous ground vehicles.Control Engineering Practice., 17: 741–750.