

自动驾驶客车主动避障控制策略研究

吴德喜, 朱岩朋, 梁庆伟, 张刚, 明庆贺

(中通客车股份有限公司, 山东聊城 252000)

摘要:针对自动驾驶客车在主动避障过程中的路径规划与轨迹跟踪问题,本文提出一种基于纵横向解耦控制的优化算法。首先,建立车辆二自由度动力学模型,通过坐标变换将大地坐标系映射至Frenet坐标系,进而构建车辆路径跟踪误差模型。在此基础上,横向控制器基于线性二次型调节器(LQR)理论并结合前馈补偿进行设计;纵向控制器则采用改进的无模型自适应控制(MFAC)方法实现。此外,利用五次多项式进行平滑路径规划,生成满足运动边界条件的可行轨迹。实车试验结果表明,本文算法在路径跟踪精度、横向稳定性及速度控制平滑性方面均优于传统PID与模型预测控制(MPC)方法,且具有良好的实时性与鲁棒性。

关键词:路径规划; 误差模型; LQR; 前馈; MFAC; 五次多项式

中图分类号:U471.15

文献标志码:A

DOI:10.15917/j.cnki.1006-3331.2026.01.002

Research on Active Obstacle Avoidance Control Strategies for Autonomous Buses

WU Dexi, ZHU Yanpeng, LIANG Qingwei, ZHANG Gang, MING Qinghe

(Zhongtong Bus Holding Co., Ltd., Liaocheng 252000, China)

Abstract:To address the path planning and trajectory tracking problem for autonomous buses during active obstacle avoidance, this paper proposes an optimized algorithm based on decoupled longitudinal and lateral control. First, the algorithm establishes a two-degree-of-freedom vehicle dynamics model and transforms the global coordinate system into the Frenet coordinate system, thereby constructing a path tracking error model. On this basis, the lateral controller employs linear quadratic regulator (LQR) theory combined with feedforward compensation, while the longitudinal controller adopts an improved model-free adaptive control (MFAC) method. Furthermore, it utilizes a quintic polynomial for smooth path planning to generate feasible trajectories that satisfy motion boundary conditions. The real vehicle test results demonstrate that the proposed algorithm outperforms traditional PID and model predictive control (MPC) methods in path tracking accuracy, lateral stability, and speed control smoothness, while also exhibiting good real-time performance and robustness.

Key words:path planning; error model; LQR; feedforward; MFAC; quintic polynomial

近年来,自动驾驶成为汽车行业的焦点问题,一些先进驾驶辅助系统(如车道偏离预警系统^[1]、前碰撞预警系统^[2]及自适应前照明系统^[3])得到快速发展。在新兴技术蓬勃发展的基础上,开展高阶自动驾驶汽车主动避障系统的研究十分必要。汽车主动避障系统是一种可以在行驶过程中避免与其他车辆或行人碰撞的主动安全系统,具有高稳定性、可靠性、快速性及准确性优势。由于客车体型大、重心高、尺寸长,在避障时难以精准跟踪规划路径,因此,有必要研

究一种可靠的控制方法以解决自动驾驶客车主动避障的控制难题。客车主动避障系统的工作过程主要包括传感器感知、中央处理器决策、避障路径规划及路径跟踪4个部分^[4]。

目前,常用的局部路径规划算法主要分为4大类:人工势场法^[5]、基于图搜索的方法^[6]、基于采样的方法^[7]、基于离散优化的方法^[8-10]。人工势场法通过构造目标点吸引力和障碍物排斥力的虚拟力场,引导车辆沿合力方向移动,实现避障导航。该方法原理

收稿日期:2025-03-24。

第一作者:吴德喜(1993—),男,硕士,助理工程师,主要从事自动驾驶设计工作。E-mail:dexiwu@163.com。

直观,但易陷入局部极小点。基于图搜索的方法将环境或状态空间离散化为图结构,利用搜索算法在图中寻找最优路径,保证解的最优性,但其计算量随维度的增加而显著增长。基于采样的方法通过在状态空间中随机或启发式采样,并将采样点连接构建可行路径,适合高维空间且具有概率完备性,但路径可能非最优或不平滑。基于离散优化的方法通过数值积分和微分方式来描述车辆的运动,从而产生数量有限的候选路径,并通过设计代价函数,从候选路径中选择最优路径。该方法计算量小,实时性好。

现有的路径跟踪控制方法有很多,包括经典的比例积分微分(Proportional Integral Differential, PID)控制^[11]、模型预测控制(Model Predictive Control, MPC)^[12]、自适应控制^[13]等。PID 控制虽然简单方便,但在复杂路径条件下跟踪精度难以保证;而模型预测及其他复杂算法虽然在仿真中效果显著,但在实车应用中,由于计算量庞大,致使控制器产生较大延迟,同样难以保证控制精度。

为保证自动驾驶关键算法有效且充分利用域控制器的算力,本文基于经典的线性二次型调节器(Linear Quadratic Regulator, LQR)算法设计横向控制器,基于无模型自适应控制理论设计纵向控制器,并基于五次多项式设计路径规划控制器。

1 纵横向运动控制器的设计

1.1 车辆动力学模型

在实际应用中,复杂的车辆模型会显著增加控制器的计算负担,导致系统反应迟滞,即使算法再精确,其优势也难以发挥。因此,本文选用车辆二自由度模型作为控制模型来进行研究,假设车辆纵向和垂向稳定,仅考虑车辆横向及横摆运动。车辆二自由度模型如图 1 所示,图中 R 为转向半径, α 为后轮侧偏角, β 为质心侧偏角, δ 为前轮转角, a 为前轴距, b 为后轴距, v 为车辆质心的速度, F_{yr} 为后轴侧向力, F_{yf} 为前轴侧向力。

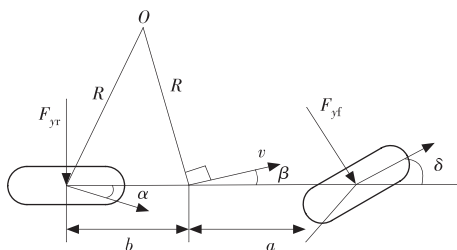


图 1 车辆二自由度模型

车辆在横向和横摆方向的力学平衡公式见式(1)和式(2)。

$$\begin{cases} ma_y = F_{yr} + F_{yf} \\ I_z \ddot{\varphi} = aF_{yf} - bF_{yr} \end{cases} \quad (1)$$

其中:

$$\begin{cases} F_{yf} = C_f \alpha_f \\ F_{yr} = C_r \alpha_r \end{cases} \quad (2)$$

式中: m 为整车质量; a_y 为侧向加速度; I_z 为车辆绕 z 轴的转动惯量; $\ddot{\varphi}$ 为车辆横摆角加速度; C_f 为前轴侧倾刚度; C_r 为后轴侧倾刚度; α_f 为前轴侧倾角; α_r 为后轴侧倾角。

结合式(1)和式(2),可得车辆二自由度系统的动力学方程,如式(3)所示。

$$\begin{bmatrix} \dot{v}_y \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \frac{C_f + C_r}{mv_x} & \frac{aC_f - bC_r}{mv_x} \\ \frac{aC_f - bC_r}{I_z v_x} & \frac{a^2 C_f + b^2 C_r}{I_z v_x} \end{bmatrix} \begin{bmatrix} v_y \\ \varphi \end{bmatrix} + \begin{bmatrix} -\frac{C_f}{mv_x} \\ -\frac{C_f}{I_z v_x} \end{bmatrix} \delta \quad (3)$$

式中: v_y 为侧向速度; \dot{v}_y 为侧向加速度; v_x 为纵向速度; φ 为横摆角速度。

1.2 横向控制器设计

为了更好地实现车辆纵横向运动的跟踪控制,本文采用 Frenet 坐标系^[14-15]对纵横向控制进行解耦。采用 LQR 算法设计横向控制器进行横向运动控制,计算出最优前轮转角。

图 2 为 Frenet 坐标系下车辆位置示意图,其中 v 为质心处的速度, s 为投影速度, θ 、 θ_q 分别为当前位置和投影点位置处的航向角大小, r 、 r_q 分别为当前位置和投影点位置的位置矢量, τ 、 n 分别为投影点处的法向和切向的单位向量, e_d 为横向距离误差,表示当前位置到其期望路径上投影点的距离。

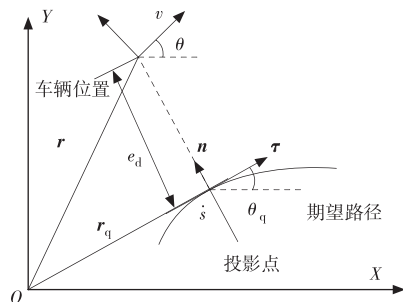


图 2 Frenet 坐标系下车辆位置示意图

根据图 2 的向量关系,可得到横向误差的表达式,具体见式(4)。

$$e_d = (\mathbf{r} - \mathbf{r}_q) \mathbf{n} \quad (4)$$

对横向误差表达式求导,可得到横向误差变化率的表达式,具体见式(5)。

$$\dot{e}_d = (\dot{\mathbf{r}} - \dot{\mathbf{r}}_q) \mathbf{n} + (\mathbf{r} - \mathbf{r}_q) \dot{\mathbf{n}} \quad (5)$$

式中: $\dot{\mathbf{r}} = \boldsymbol{\tau} v$; $\dot{\mathbf{r}}_q = \boldsymbol{\tau} s$ 。

根据弗莱纳公式^[16] $d\mathbf{n}/ds = -K\boldsymbol{\tau}$ (K 为投影点的曲率),将式(5)简化为 $\dot{e}_d = v\boldsymbol{\tau}\mathbf{n}$,其中 $\boldsymbol{\tau} = (\cos \theta, \sin \theta)$, $\mathbf{n} = (-\sin \theta_q, \cos \theta_q)$,则进一步简化为 $\dot{e}_d = v\sin(\theta - \theta_q)$ 。由于航向角是横摆角与质心侧偏角之和,定义 $e_\varphi = \varphi - \theta_q$,其中 e_φ 为航向角偏差, φ 为车辆横摆角。且纵向速度 $v_x = v\cos \beta$,横向速度 $v_y = v\sin \beta$,由于 β 值特别小,式(5)可写为 $\dot{e}_d = v_x e_\varphi + v_y$,则误差模型见式(6)。

$$\begin{cases} \dot{v}_y = \dot{e}_d - v_x e_\varphi \\ \dot{v}_x = \dot{e}_d - v_x e_\varphi \\ \dot{\varphi} = \dot{e}_\varphi + \dot{\theta}_q \\ \dot{\psi} = \dot{e}_\varphi \end{cases} \quad (6)$$

式中: $\dot{\varphi}$ 是横摆角速度; $\dot{\theta}_q$ 是投影点位置的航向角速度; \dot{e}_φ 是横摆角速度偏差。

将式(6)与动力学模型公式相结合,令横向误差为状态变量,写为 $T = [e_d \quad \dot{e}_d \quad e_\varphi \quad \dot{e}_\varphi]^T$,控制变量 u 代表前轮转角 ($u = \delta$),则可得到车辆动力学误差模型的状态空间方程,如式(7)所示。

$$\dot{T} = \mathbf{A}T + \mathbf{B}u + \mathbf{C}\dot{\theta}_q \quad (7)$$

式中: T 为状态变量; \mathbf{A} 、 \mathbf{B} 、 \mathbf{C} 为状态空间方程系数矩阵,表达式分别见式(8)、式(9)和式(10)。

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{C_f + C_r}{mv_x} & \frac{C_f + C_r}{m} & \frac{aC_f - bC_r}{mv_x} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{aC_f - bC_r}{I_z v_x} & \frac{aC_f - bC_r}{I_z} & \frac{a^2 C_f + b^2 C_r}{I_z v_x} \end{bmatrix} \quad (8)$$

$$\mathbf{B} = \begin{bmatrix} 0 & \frac{C_f}{m} & 0 & -\frac{aC_f}{I_z} \end{bmatrix}^T \quad (9)$$

$$\mathbf{C} = \begin{bmatrix} 0 & \frac{aC_f - bC_r}{mv_x} & 0 & \frac{a^2 C_f + b^2 C_r}{I_z v_x} \end{bmatrix}^T \quad (10)$$

为了使车辆更好地跟踪路线并减轻计算机的算力负担,本文基于车辆动力学误差模型,设计了横向LQR控制器。

由于动力学误差模型中 $\mathbf{C}\dot{\theta}_q$ 难以用LQR处理,本文暂时将其忽略,使公式成为常规的LQR状态方

程,后面再以前馈的方式进行补偿,据此得到的代价函数见式(11)。

$$\begin{cases} \min J = \sum_{k=0}^n (T_k^T \mathbf{Q} T_k + u^T \mathbf{R} u) \\ \text{s. t. } \dot{T} = \mathbf{A}T + \mathbf{B}u \end{cases} \quad (11)$$

式中: J 为代价函数; T_k 为状态方程变量; u 为控制输入; \mathbf{Q} 和 \mathbf{R} 分别为状态变量和控制输入的权重系数矩阵,通常取 \mathbf{Q} 为对角线矩阵来直观调节控制的权重, $\mathbf{Q} = \text{diag}[q_1 \quad q_2 \quad q_3 \quad q_4]$, q_1 、 q_2 、 q_3 、 q_4 分别为横向误差、横向误差变化率、航向误差、航向误差变化率的权重系数。

为了使方程离散化,使用MATLAB的c2d命令将线性状态方程转化为离散状态方程,见式(12)。

$$T(k+1) = \mathbf{A}_T T(k) + \mathbf{B}_T u(k) \quad (12)$$

其中:

$$\mathbf{A}_T = \frac{(\mathbf{I} + \mathbf{A}dt/2)}{(\mathbf{I} - \mathbf{A}dt/2)} \quad (13)$$

$$\mathbf{B}_T = \frac{\mathbf{B}dt}{(\mathbf{I} - \mathbf{A}dt/2)} \quad (14)$$

式中: \mathbf{I} 为单位矩阵; \mathbf{A}_T 为离散化的状态矩阵; \mathbf{B}_T 为离散化的控制输入矩阵。

相应地,成本函数转变为式(15)。

$$\begin{cases} J = \sum_{k=0}^{n-1} (T_k^T \mathbf{Q} T_k + u^T \mathbf{R} u) + T_n^T \mathbf{Q} T_n \\ \text{s. t. } T(k+1) = \mathbf{A}_T T(k) + \mathbf{B}_T u(k) \end{cases} \quad (15)$$

式中: T_k 是 k 时刻的状态变量; T_n 是最终 n 时刻的状态变量。

构建哈密顿方程,见式(16)。

$$H_k = \left[T_k^T \mathbf{Q} T_k + u^T \mathbf{R} u + \lambda_T^T (T(k) + \mathbf{B}_T u(k)) \right] \quad (16)$$

式中: λ_T 为协状态变量。

利用最优原理对式(17)求极值。

$$\begin{cases} \frac{\partial J}{\partial T_k} = \lambda_k = 2\mathbf{Q}T_k + \mathbf{A}_k \lambda_{k+1} \\ \frac{\partial J}{\partial u_k} = u_k = -\frac{1}{2} \mathbf{R}^{-1} \mathbf{B}_k \lambda_{k+1} \\ \frac{\partial J}{\partial \lambda_k} = T_{k+1} = \mathbf{A}_k T_k + \mathbf{B}_k u_k \\ \frac{\partial J}{\partial T_n} = \lambda_n = 2\mathbf{Q}T_n \end{cases} \quad (17)$$

式中: λ_k 为 k 时刻的伴随变量; u_k 为 k 时刻的控制变量; T_{k+1} 为 $k+1$ 时刻的状态变量; λ_n 为终端时刻的伴

随变量。

基于公式 (17), 通过消元法可推导出拉格朗日乘子与其他参数矩阵之间的关系, 见式 (18)。

$$\lambda_{n-1} = 2(\mathbf{Q} + \mathbf{A}_k^T \mathbf{Q} (\mathbf{I} + \mathbf{B}_k \mathbf{R} \mathbf{B}_k^T \mathbf{Q})^{-1} \mathbf{A}_k) T_k \quad (18)$$

令拉格朗日乘子的终端值 $\lambda_k = 2\mathbf{P}_k T_k$, 且 $\mathbf{P}_k = \mathbf{Q}$, 可以得到 Riccati 方程, 见式 (19)。

$$\mathbf{P}_{k-1} = \mathbf{Q} + \mathbf{A}_k^T \mathbf{P}_k (\mathbf{I} + \mathbf{B}_k \mathbf{R}^{-1}) \mathbf{A}_k \quad (19)$$

该方程的特性在于, 矩阵 \mathbf{P} 经过多次迭代后会收敛。利用这一收敛结果, 可以进一步求出最优控制解, 如式 (20) 所示。

$$u_k = -\frac{1}{2} \mathbf{R}^{-1} \mathbf{B}_k^T 2\mathbf{P}_k T_{k+1} \quad (20)$$

将式 (20) 代入状态方程得到式 (21)。

$$u_k = -(\mathbf{R} + \mathbf{B}_k^T \mathbf{P}_k \mathbf{B}_k) \mathbf{B}_k^T \mathbf{P}_k \mathbf{A}_k T_{k+1} = -k T_{k+1} \quad (21)$$

1.3 前馈控制

为了补偿 1.2 节中动力学误差模型中 $\mathbf{C}\dot{\theta}_q$ 带来的误差, 本节设计了前馈控制算法。因此最优的前轮转角计算公式见式 (22)。

$$u = -k T_k + \delta \quad (22)$$

将式 (22) 代入状态方程 (7), 可得到式 (23)。

$$\dot{T} = (\mathbf{A} - \mathbf{B}k) T + \mathbf{B}\delta + \mathbf{C}\dot{\theta}_q \quad (23)$$

令 \dot{T} 为 0, 得到关于 T 的公式, 如式 (24) 所示。

$$T = -(\mathbf{A}_k - \mathbf{B}k)^{-1} \mathbf{A}_k (\mathbf{B}\delta + \mathbf{C}\dot{\theta}_q) \quad (24)$$

将式 (8) ~ (10) 代入式 (24), 可得到稳态误差方程, 如式 (25) 所示。

$$D = \begin{bmatrix} \left\{ \delta - \frac{\dot{\theta}_q}{v_x} \left[a + b - bk_3 - \frac{mv_x^2}{a+b} \left(\frac{b}{C_f} + \frac{a}{C_r} k_3 - \frac{a}{C_r} \right) \right] \right\} \\ 0 \\ -\frac{\dot{\theta}_q}{v_x} \left(b + \frac{a}{a+b} \frac{mv_x^2}{C_r} \right) \\ 0 \end{bmatrix} \quad (25)$$

将式 (5) 变形为 $\mathbf{r}_q + e_d \mathbf{n}_q = \mathbf{r}$, 对两边求导可得到式 (26)。

$$s\boldsymbol{\tau}_q + v \sin(\theta - \theta_q) \mathbf{n}_q + e_d (-Ks\boldsymbol{\tau}) = v\boldsymbol{\tau} \quad (26)$$

消除式 (26) 的向量, 得到 Frenet 坐标系下的速度关系, 见式 (27)。

$$s = \frac{v_x \cos(\varphi - \theta_q) - v_y \sin(\varphi - \theta_q)}{1 - Ke_d} \quad (27)$$

一般认为 $s = v_x$, 根据曲率的定义可知, $K = \dot{\theta}_q / v_x$, 又因为 $a_y = v_x^2 / R$, 航向角误差可表示为式 (28)。

$$e_\varphi = \left(\frac{b}{R} + \frac{a}{a+b} m a_y \frac{1}{C_r} \right) \quad (28)$$

由于 $F_{yr} = \left(\frac{a}{a+b} \right) m a_y$, 可将式 (28) 改写为式 (29)。

$$e_\varphi = -\left(\frac{b}{R} + \alpha_r \right) \quad (29)$$

质心侧偏角的计算公式见式 (30)。

$$-\beta = -\left(\frac{b}{R} + \alpha_r \right) \quad (30)$$

最后, 得到了反馈转角的计算公式, 见式 (31)。

$$\delta = s \frac{K}{v_x} \left[a + b - bk_3 - \frac{mv_x^2}{a+b} \left(\frac{b}{C_f} + \frac{a}{C_r} k_3 - \frac{a}{C_r} \right) \right] \quad (31)$$

1.4 基于无模型自适应的纵向运动控制

为提高速度控制的舒适度, 采用无模型自适应的纵向运动控制方法控制车速 v 。为方便计算, 对 v 的描述采用离散化的表达形式, 令 $v(k)$ 表示当前时刻的车速, $U(k)$ 为当前时刻需要输入的加速度。鉴于该系统具有非线性, 故可将其描述为一种离散时间非线性系统, 即带外部输入的非线性自回归滑动平均模型, 如式 (32) 所示。

$$v(k+1) = f(v(k), \dots, v(k-k_v), U(k), \dots, U(k-k_u)) \quad (32)$$

其中 $f(\cdot)$ 表示非线性函数, $f(\cdot) : \mathbf{R}^{k_v + k_u + 2} \rightarrow \mathbf{R}$ 表示非线性速度更新规律, k_v 和 k_u 为未知参数, 以下称速度控制系统。在速度控制系统中, 要求取合适的 $U(k)$ 使得 $v(k)$ 能够尽快且无超调地跟踪到目标车速。

对于速度控制系统, 定义 $U_L(k) \in \mathbf{R}^L$ 为一个活动时间窗口 $[k-L+1]$, 所有加速度的输入信号见式 (33)。

$$U_L(k) = [U(k), \dots, U(k-L+1)]^T \quad (33)$$

下面介绍基于无模型自适应的纵向控制器设计步骤。

1.4.1 设计控制输入准则函数

在基于无模型自适应的纵向控制器设计中, 为了同时优化速度跟踪性能和控制输入平滑性, 设计如式 (34) 所示的控制输入准则函数。

$$J(U(k)) =$$

$$|v^*(k+1) - v(k+1)|^2 + \lambda |U(k) - U(k-1)|^2 \quad (34)$$

式中: $J(U(k))$ 为控制输入准则函数; $v^*(k+1)$ 为 $k+1$ 时刻的期望速度; λ 为权重因子, 其目的是使加速度

不产生突变; $v(k+1)$ 为 $k+1$ 时刻的预测速度, $v(k+1) = v(k) + \Phi_L^T(k)\Delta U(k)$, $\Phi_L^T(k)$ 为时变参数向量。

令准则函数为0并对 $U(k)$ 进行求导,为了更快地对参数进行调试,这里对求解过程进行了改进,对前一时刻的解进行乘系数 Δ ,最后得到式(35)。

$$U(k) = \Delta U(k-1) + \frac{\rho \Phi_1(k)(v^*(k+1) - v(k))}{\lambda + |\Phi_1(k)|^2} - \frac{\Phi_1(k) \sum_{i=2}^L \rho \Phi_i(k) \Delta U(k-i+1)}{\lambda + |\Phi_1(k)|^2} \quad (35)$$

其中 ρ 为补偿因子, $\rho \in (0, 1]$, λ 的引入则使得算法有更高的可控性。

1.4.2 设计 PG 估计准则函数

PG 估计准则函数如式(36)所示,其中 μ 为权重因子。

$$J(\Phi_L(k)) = |v(k) - v(k-1) - \Phi_L^T(k)\Delta U_L(k-1)|^2 + \mu \|\Phi_L(k) - \Phi_L(k-1)\|^2 \quad (36)$$

将 PG 估计准则函数对 $\Phi_L(k)$ 求导,则得到其计算公式,见式(37)。

$$\Phi_L(k) = \Phi_L(k-1) + \frac{\eta \Delta U_L(k-1)(v(k) - v(k-1))}{\mu + \|\Delta U_L(k-1)\|^2} - \frac{\eta \Delta U_L(k-1)(\Phi_L^T(k-1)\Delta U_L(k-1))}{\mu + \|\Delta U_L(k-1)\|^2} \quad (37)$$

式中: η 为步长因子; $\Phi_L(k)$ 为 PG 函数的估计值。

1.4.3 重置算法

为了使 PG 估算更好地适应车速控制的实际情况,设置重置算法如下:当 $\|\Phi_L(k)\| \leq \varepsilon$ 或 $\|\Delta U_L(k-1)\| \leq \varepsilon$ 或 $\text{sign}(\Phi_L(k)) \neq \text{sign}(\Phi_L(1))$ 时, $\Phi_L(k) = \Phi_L(1)$ 。其中, $\Phi_L(1)$ 为 $\Phi_L(k)$ 的初始值, ε 为较小的正数。

2 基于五次多项式的路径规划设计

五次多项式路径规划具有曲线平滑、曲率变化稳定的优势,只需给出位移的初始位置和末端位置即可得出规划路径,其计算公式见式(38)。

$$\begin{cases} x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \\ y(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5 \end{cases} \quad (38)$$

其中, $a_0, a_1, a_2, a_3, a_4, a_5, b_0, b_1, b_2, b_3, b_4, b_5$ 为

经验系数, $x(t)$ 、 $y(t)$ 代表车辆纵、横向位移,是关于时间 t 的函数,对其一阶二阶求导便得到速度、加速度状态方程,如式(39)和式(40)所示。

$$\begin{cases} \dot{x}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 \\ \dot{y}(t) = b_1 + 2b_2 t + 3b_3 t^2 + 4b_4 t^3 + 5b_5 t^4 \end{cases} \quad (39)$$

$$\begin{cases} \ddot{x}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3 \\ \ddot{y}(t) = 2b_2 + 6b_3 t + 12b_4 t^2 + 20b_5 t^3 \end{cases} \quad (40)$$

车辆的初始位置和末端位置状态如式(41)所示。

$$\begin{cases} S_{\text{start}} = [x_{\text{start}}, \dot{x}_{\text{start}}, \ddot{x}_{\text{start}}, y_{\text{start}}, \dot{y}_{\text{start}}, \ddot{y}_{\text{start}}]^T \\ S_{\text{end}} = [x_{\text{end}}, \dot{x}_{\text{end}}, \ddot{x}_{\text{end}}, y_{\text{end}}, \dot{y}_{\text{end}}, \ddot{y}_{\text{end}}]^T \end{cases} \quad (41)$$

通过式(42)获得所需要的 $[x_o, y_o, \theta_o, k_o]^T$,并将其代入到纵横向控制器中实现运动控制。

$$\begin{cases} x_o = x(t) \\ y_o = y(x) \\ \theta_o = \arctan\{\dot{y}[x(t)]\} \\ k_o = \frac{\ddot{y}[x(t)]}{\{1 + \dot{y}[x(t)]^2\}^{\frac{1}{2}}} \end{cases} \quad (42)$$

式中: x_o 为目标横向位置; y_o 为目标纵向位置; θ_o 为目标横摆角; k_o 为目标曲率。

3 试验验证

为了验证所提算法的有效性,本研究以中通 V60 为基础车型,搭建了自动驾驶客车试验平台。该平台的感知系统由 2 个 32 线机械激光雷达、1 个 128 线固态激光雷达、1 个毫米波雷达和 4 个摄像头组成。平台配备了 2 个算力为 21TOPS 的 NVIDIA 域控制器,负责处理感知和决策算法。试验中,利用车辆引导号牌模拟障碍物。将所设计的算法嵌入到域控制器中进行实车试验,试验情况如图 3 所示。

图 4 至图 6 为本文算法与 PID 算法、MPC 算法在仿真试验过程中记录的车辆数据曲线。其中,图 4 为路径跟踪曲线,从对比结果可知,本文算法与目标轨迹曲线基本一致,这表明路径规划合理且控制效果较好。图 5 和图 6 分别为速度和横摆角速度曲线,从对比结果可知,本文算法在控制车辆速度时更为稳定,波动较小;横摆角速度的变化幅度也较小,尤其是在初始阶段,车辆稳定性高。由于控制器在处理传感器信号后,其算力不足以支持复杂算法的运算,因此 MPC 算法难以充分发挥其控制优势。

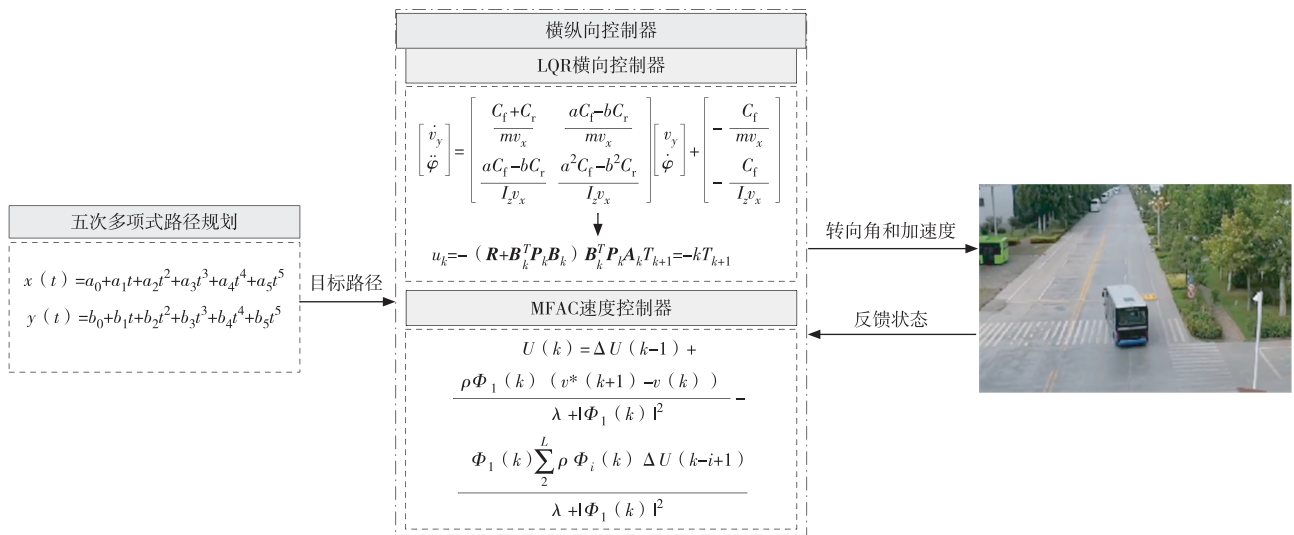


图 3 自动驾驶客车避障试验

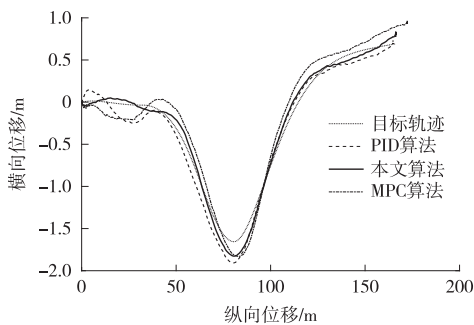


图 4 路径跟踪曲线

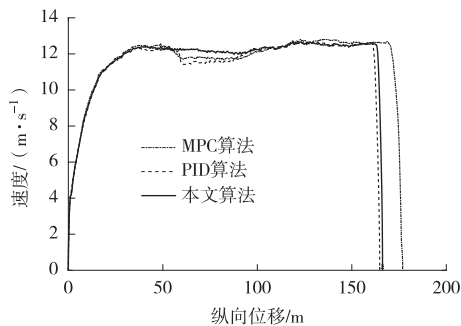


图 5 速度曲线

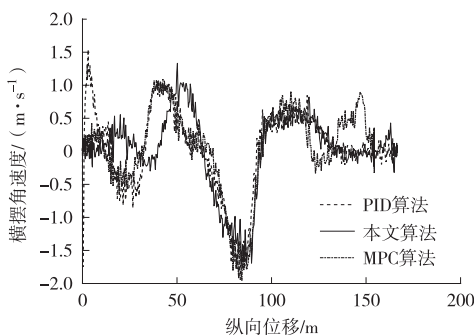


图 6 横摆角速度曲线

在横向位移和速度波动控制效果上的箱线图。从图 7 可知, 本文算法的横向位移波动范围为 $[-0.17, 0.18]$, 显著小于 PID 算法的 $[-0.14, 0.27]$ 和 MPC 算法的 $[-0.28, 0.27]$, 表明本文算法能有效抑制横向位移偏差, 从而提升系统稳定性(波动范围越小, 稳定性越好)。从图 8 可知, 本文算法速度波动范围为 $[-0.70, 0.08]$, 而 PID 算法为 $[-0.72, 0.68]$, MPC 算法为 $[-0.81, 0.37]$, 进一步验证了本文算法在速度控制方面具有更小的波动幅度和更强的鲁棒性。两图结果表明, 本文算法在位移与速度控制中均表现出更优的稳定性和精度。

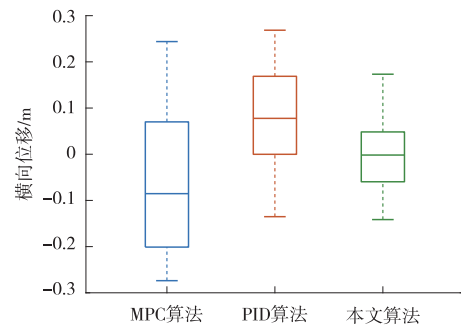


图 7 横向位移波动分布

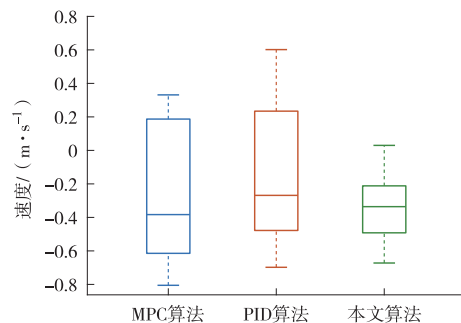


图 8 速度波动分布

图 7 和图 8 为本文算法与 PID 算法、MPC 算法

4 结束语

本文基于车辆紧急避障工况,进行了五次多项式路径规划和纵横向控制方法设计。首先建立了二自由度车辆模型,通过大地坐标向 Frenet 坐标转换,将车辆模型转换为误差模型,然后设计了基于 LQR 和无模型自适应理论的纵横向控制策略,最后设计了五次多项式的路径规划方法。通过搭建自动驾驶客车试验平台对所提算法进行验证,并与 PID、MPC 控制算法进行对比,结果显示本文所提控制算法在路径跟踪与稳定性方面都具有明显优势。

参考文献:

- [1] CHEN W W, WANG W X, WANG K, et al. Lane departure warning systems and lane line detection methods based on image processing and semantic segmentation: a review [J]. Journal of Traffic and Transportation Engineering (English Edition), 2020, 7(6): 748-774.
- [2] NIU C Y, SU Z L, WU K L, et al. Test and analysis on distance measurement accuracy of commercial vehicle forward collision warning system [M] // Proceedings of China SAE Congress 2019: Selected Papers. Berlin: Springer, 2020: 83-95.
- [3] RONG H, GONG J F, WU K L. Kinematics model and control strategy of adaptive front lighting system [J]. IEEE Computer, 2009(2): 70-74.
- [4] 马思群, 王兆强, 韩博, 等. 改进的 LQR 横向路径跟踪控制器 [J]. 机械科学与技术, 2024, 43(1): 130-140.
- [5] 孔慧芳, 夏露, 张倩. 基于改进人工势场法的智能车辆避障路径规划 [J]. 合肥工业大学学报 (自然科学版), 2023, 46(5): 583-589.
- [6] 周兵, 黄治坤, 柴天, 等. 基于图搜索与数值优化方法的分层轨迹规划方法 [J]. 湖南大学学报 (自然科学版), 2022, 49(12): 1-10.
- [7] 章微意. 基于神经网络的物体位姿识别与基于概率采样的机器人运动规划方法和仿真研究 [D]. 上海: 上海交通大学, 2020.
- [8] 张家旭, 施正堂, 赵健, 等. 基于 Radau 伪谱法的汽车高速紧急换道避障最优控制策略设计 [J]. 汽车工程, 2020, 42(8): 1040-1049.
- [9] 丛岩峰, 陈虹, SAWODNY O, 等. 车辆高速公路环境下基于微分平坦的紧急避障预测控制 [C] // 中国自动化学会控制理论专业委员会. 中国自动化学会控制理论专业委员会 D 卷, 2011: 1135-1140.
- [10] 任书宇, 吴钦木, 周还籍. 基于改进 A* 和 Lattice 算法的自动驾驶汽车路径规划研究 [J]. 计算机与数字工程, 2023, 51(2): 342-347.
- [11] PARRA-VEGA V, ARIMOTO S, LIU Y H, et al. Dynamic sliding PID control for tracking of robot manipulators: theory and experiments [J]. IEEE Transactions on Robotics and Automation, 2003, 19(6): 967-976.
- [12] SHEN C, GUO H Y, LIU F, et al. MPC-based path tracking controller design for autonomous ground vehicles [C] // 2017 36th Chinese Control Conference. New York: IEEE, 2017: 9584-9589.
- [13] 胡珉珉. 基于自适应模型预测控制的拖拉机路径跟踪控制研究 [D]. 镇江: 江苏大学, 2022.
- [14] 华航, 陈焕明, 李雪涵. 基于 Frenet 坐标系的换道轨迹规划及跟踪验证 [J]. 汽车实用技术, 2022, 47(16): 25-30.
- [15] WANG W, CHEN H Y, MA J H, et al. Path tracking for intelligent vehicles based on frenet coordinates and delayed control [J]. Acta Armamentarii, 2019, 40(11): 2336-2351.
- [16] 张洋瑞. 分布式驱动智能汽车路径跟踪与稳定性协同控制研究 [D]. 重庆: 重庆交通大学, 2024.