

# 四轮阿克曼模型移动机器人运动轨迹控制系统设计

李厚霖 520020910007

## 摘要

在前半学期的学习中，我们已经完成了对阿克曼模型移动机器人的建模及其运动控制系统设计。但移动机器人在执行任务时，往往需要在复杂和动态的环境中运动，这就对它的路径规划和控制提出了极高的要求。所以我们将在在此基础上，我们要构建其动力学模型，将进一步探讨移动机器人的路径规划和控制问题。

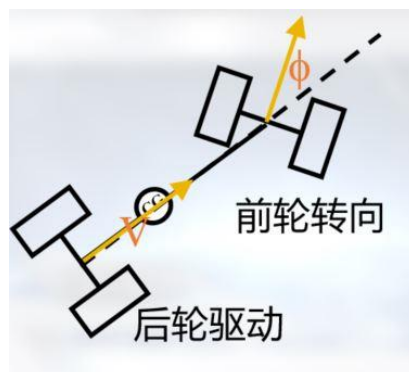
本论文以 B 样条为基础，研究了小车行驶路径规划，并使用阿克曼模型对移动机器人进行动力学简化模型分析。在此基础上，提出了小车曲线跟踪的线性化模型系统结构，并对跟踪误差进行了分析。通过阿克曼模型小车曲线跟踪的线性化模型，设计了相应的控制器，并对控制系统性能进行了分析。本研究对于小车路径规划和曲线跟踪控制的优化具有重要意义。

**关键词：B 样条、路径规划、阿克曼模型、线性化模型**

## 1.引言

阿克曼模型四轮小车，由直流电机通过减速齿轮带动后轮旋转驱动小车以速度  $V$  前进；由角度伺服电机驱动前轮转向，转向角为  $\phi$ 。

在前人的研究中，对于移动两轮机器人的运动控制的研究主要集中于运动学和动力学的分析，在理论上做了许多开创性的工作<sup>[1][2]</sup>。但



这样的算法依赖于精确的模型和完整的信息，实际应用中并不多见。大部分实际应用的移动机器人左右轮的运动控制都是基于双闭环的电机控制，内环是电流环，外环是速度环，每个电机单独控制。

机器人在地面的运动可归结为平面刚体运动。而其运动可分解为直线运动、旋转运动。这两种运动完全靠两个驱动轮速度来实现。在闭环控制之前，先对线速度  $V$ ，角速度  $\omega$  的信息进行融合，分别求出实际加在闭环驱动器上期望的左右轮的各自速度。求出左右轮速度后，再对两轮的速度进行闭环控制。这样的控制方案在实践中证明是可行的，还可以达到相当好的控制效果。

对于平面刚体运动的移动机器人，可以首先规划出符合运动学与动力学约束的路径，其后的运动控制问题即归于行驶的循线控制。

## 2.以 B 样条为基础规划小车行驶路径<sup>[1][3]</sup>

在其中的设计报告当中我已经设计好了阿克曼模型四轮小车，其车轮的直径为  $D = 20cm$ 、车轮轴距为置为  $L = 90cm$ 、车轮间距为  $H = 40cm$ 。由于要求为移库，故机器人初始姿态为车头朝向  $y$  轴正方向，结束姿态为车头也为朝向  $y$  轴正方向。

轨迹生成方式选择 3 阶 B 样条 (B-Spline) 法，其中四个样条函数为：

$$f_1(s) = \frac{1}{6}(1-s)^3$$

$$f_2(s) = \frac{1}{6}(-3s^3 + 3s^2 + 3s + 1)$$

$$f_3(s) = \frac{1}{6}(3s^3 - 3s^2 + 4)$$

$$f_4(s) = \frac{1}{6}s^3$$

选取 11 个控制点：

$$c_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, c_2 = \begin{bmatrix} 0.3 \\ 1.25 \end{bmatrix}, c_3 = \begin{bmatrix} 1 \\ 1.75 \end{bmatrix}, c_4 = \begin{bmatrix} 2 \\ 1.75 \end{bmatrix}, c_5 = \begin{bmatrix} 2.7 \\ 1 \end{bmatrix}, c_6 = \begin{bmatrix} 3 \\ 0 \end{bmatrix}, c_7 = \begin{bmatrix} 3.3 \\ -1 \end{bmatrix}, c_8 = \begin{bmatrix} 4 \\ -1.75 \end{bmatrix},$$

$$c_9 = \begin{bmatrix} 5 \\ -1.75 \end{bmatrix}, c_{10} = \begin{bmatrix} 5.7 \\ -1.25 \end{bmatrix}, c_{11} = \begin{bmatrix} 6 \\ 0 \end{bmatrix}$$

又由要求可知，该轨迹要经过起点 (0, 0) 和终点 (6, 0)。因此，需要增加额外控制点才能符合要求。故增加额外控制点 (设经过起点和终点的速度均为 1m/s)：

$$c_1 + \begin{bmatrix} 0 \\ 1 \end{bmatrix}L \quad c_1 - \begin{bmatrix} 0 \\ 1 \end{bmatrix}L \quad c_7 + \begin{bmatrix} 0 \\ 1 \end{bmatrix}L \quad c_7 - \begin{bmatrix} 0 \\ 1 \end{bmatrix}L$$

其中  $L$  为车长的一半即  $0.9/2 = 0.45m$ 。故增加的额外控制点为：

$$c_1 = \begin{bmatrix} 0 \\ -0.45 \end{bmatrix}, c_2 = \begin{bmatrix} 0 \\ 0.45 \end{bmatrix}, c_3 = \begin{bmatrix} 6 \\ -0.45 \end{bmatrix}, c_4 = \begin{bmatrix} 6 \\ 0.45 \end{bmatrix}$$

编写 MATLAB 程序实现，得到如图 1 所示的过起点和终点的 B 样条曲线图：

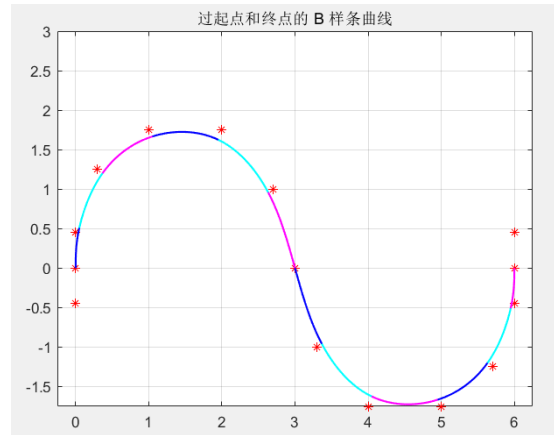


图 1：B 样条曲线

将设计好的阿克曼移动机器人简化为  $L = 0.9m$ ， $H = 0.4m$  的长方形，在 MATLAB 中画出其位姿变化情况如图 2 所示：

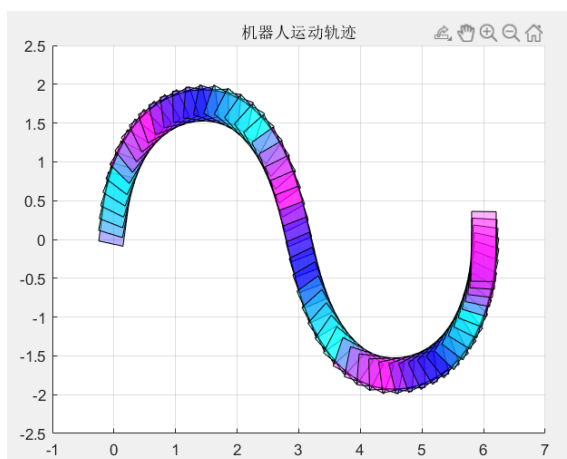


图 2：阿克曼模型位姿变换图

曲线曲率  $K$  是曲线弯曲程度一种度量, 曲率  $K$  越大曲线越弯曲, 小车跟踪更为困难, 曲率计算公式为:

$$K = \frac{1}{R} = \frac{\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)}{(\sqrt{\dot{x}(t)^2 + \dot{y}(t)^2})^3} = \frac{\dot{x}(s)\ddot{y}(s) - \ddot{x}(s)\dot{y}(s)}{(\sqrt{\dot{x}(s)^2 + \dot{y}(s)^2})^3}$$

应用上式于 MATLAB, 得到如图 3 所示的曲率变化情况:

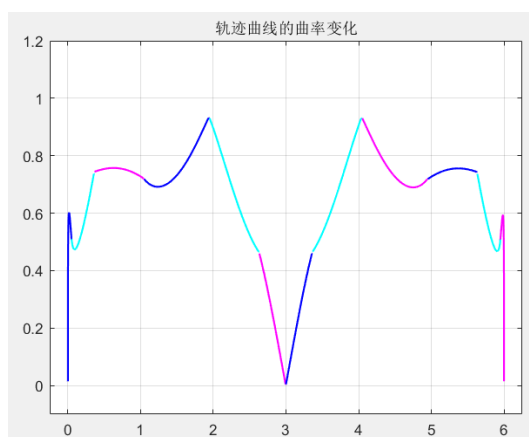


图 3：轨迹曲线的曲率变化

我们可知：以车长为  $L$  的阿克曼模型小车为例, 其最大转向曲率  $K$  的范围为:

$$K \leq \frac{\tan \phi_{\max}}{L}$$

其中,  $\phi_{\max}$  为前轮的最大转向角。因此, 我们可以用曲线曲率  $K$  评价轨迹曲线的好坏: 当曲线曲率  $K$  超过此上界时, 小车无法完全地跟踪此曲线, 导致行驶轨迹无法满足我们的期望, 因此我们不能采用这条曲线作为小车的轨迹曲线。若曲线曲率  $K$  在可接受范围内, 则能实现该曲线轨迹的运动。

经查阅资料和文献<sup>[2][6]</sup>可知, 常用的阿克曼模型实验小车的前轮最大转向角

为  $45^\circ$  左右，带入公式（3）由此计算得到最大转向曲率为 1.11。而从 MATLAB 画的该曲线的曲率图可知，曲率最大值约为 0.93。故在可接受的转向曲率范围内。因此，机器人物理上可以沿该路径行驶。

### 3.阿克曼模型移动机器人动力学简化模型

在期中课程设计中，我们已经对阿克曼移动机器人的运动学模型进行了分析，在此部分，我们将对其动力学模型进行分析。

动力学(dynamics)方程：基于小车受力分析，建立小车驱动变量与系统输出变量间关系。首先基于小车受力分析，建立起小车驱动变量与系统输出变量间关系也就是小车的动力学方程。由于在不考虑前轮之间以及后轮之间的速度插值的情况下，阿克曼模型小车可以认为是对称的，那么我们可以等效为自行车模型进行推导。设小车质量为  $M_u$ ，绕质心转动惯量为  $J_u$ 。小车推力为  $F_D$  总是沿着小车中轴线的，前轮侧向受力为  $F_F$ ，后轮侧向受力为  $F_R$ ，小车受力示意图如下所示：

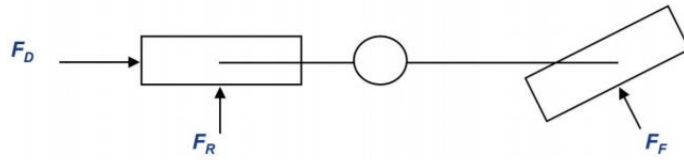


图 4：阿克曼模型移动机器人动力学简化模型

当转向角  $\phi \neq 0$ ， $\dot{\phi} \neq 0$  时，对小车模型进行受力分析：

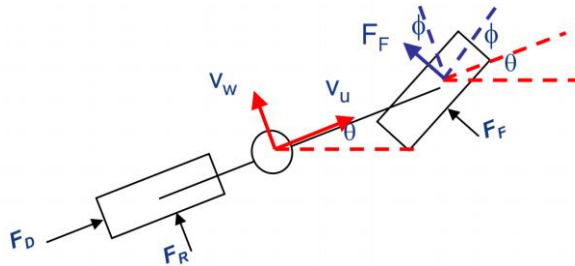


图 5：阿克曼模型移动机器人轴向、法向、转动受力示意图

对小车轴向受力分析：

$$M_u * (\dot{v}_u) = F_D - b_u v_u - F_F \sin \phi$$

对小车法向受力分析：

$$M_u * (\dot{v}_w) = F_R + F_F \cos \phi$$

对小车转动受力分析：

$$J_u * (\ddot{\theta}) = -F_R * b + F_F \cos \phi * a$$

由此，我们得到了阿克曼模型小车三个方向上的受力方程，但是这三个受力方程形成的是非完整约束，我们只能进行部分建模：

根据这三个受力方程，我们可以合并得到：

$$\begin{cases} b * M_u * (\dot{v}_w) + J_u * (\ddot{\theta}) = F_F \cos \phi * L \\ [b * M_u * (\dot{v}_w) + J_u * (\ddot{\theta})] \frac{\tan \phi}{L} = F_F \sin \phi \end{cases}$$

消去未知的侧向力 $F_F$ 和 $F_R$ ，得：

$$M_u * (\dot{v}_u) + [b * M_u * (\dot{v}_w) + J_u * (\ddot{\theta})] \frac{\tan \phi}{L} = F_D - b_u v_u$$

利用已经得到的运动学公式：

$$\begin{aligned} v_w &= b \dot{\theta}, \quad \dot{\theta} = \frac{v_u}{L} \tan \phi \\ \dot{v}_w &= b \ddot{\theta}, \quad \ddot{\theta} = \frac{\dot{v}_u}{L} \tan \phi + \frac{v_u}{L} \frac{1}{\cos^2 \phi} \dot{\phi} \end{aligned}$$

我们可以得到小车的前向非线性动力学方程：

$$[M_u + \frac{\tan^2 \phi}{L^2} (J_u + M_u b^2)] \dot{v}_u + \frac{\tan \phi}{L^2 \cos^2 \phi} (J_u + M_u b^2) v_u \dot{\phi} = F_D - b_u v_u$$

其中， $M_u + \frac{\tan^2 \phi}{L^2} (J_u + M_u b^2)$ 是小车的等效质量， $\frac{\tan \phi}{L^2 \cos^2 \phi} (J_u + M_u b^2) v_u \dot{\phi}$ 为离心力项。

当转向角 $\phi = 0$ ， $\dot{\phi} = 0$ 时，直线动力学方程简化为：

$$v_u(s) = \frac{1}{M_u s + b_u} F_D(s)$$

然而实际上，考虑到电机减速驱动等诸多现实因素，精确得到小车解析动力学方程非常困难。在实际中结合实验测量、系统辨识等方法可以得到简化动力学方程，即控制变量 $(V_c, \phi_c)$ 到系统输出 $(V, \phi)$ 满足一阶惯性环节关系。

$$\begin{cases} V(s) = \frac{1}{T_v s + 1} V_c(s) \\ \phi(s) = \frac{1}{T_\phi s + 1} \phi_c(s) \end{cases}$$

示意图如下所示：

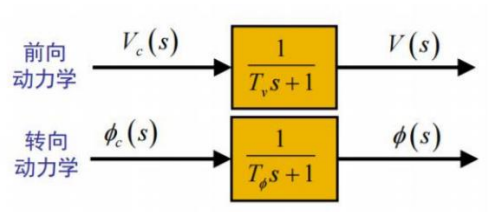


图 6: 阿克曼小车动力学传递函数模型

## 4. 小车曲线跟踪线性化模型系统结构

### 4.1 跟踪误差分析

曲线跟踪误差主要有三种：

I. 方位角误差：  $\theta_p = \theta - \theta_d$ ，控制目标为  $\theta_p \rightarrow 0$ ：

具体分析图如图 7 所示。对于方向角，若定义规划路径曲线在  $(x_d, y_d, \theta_d)$  位置处与 X 轴的夹角为  $\theta_d$ ，阿克曼小车朝向与 X 轴的夹角为  $\theta$ ，那么方向角误差  $\theta_p = \theta - \theta_d$ 。使得小车运行轨迹与规划轨迹时刻同向。

:

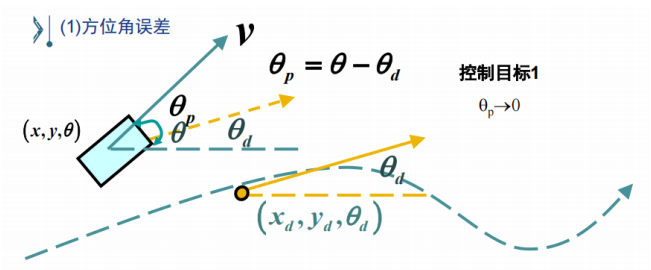


图 7: 方向角误差控制分析图

II. 前向速度：  $v_s = v * \cos \theta_p$ ，控制目标为  $v_s \rightarrow v_d$

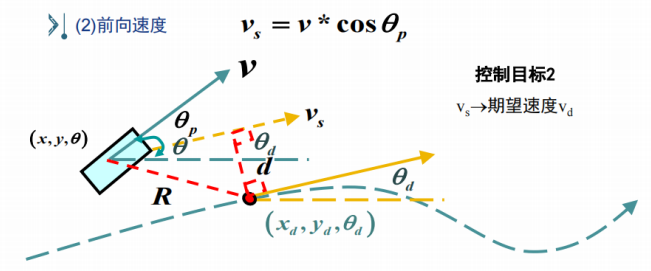


图 8: 前向速度控制分析图

III. 控制目标：法向位置误差  $d \rightarrow 0$

在小车沿轨迹运行过程中，要尽可能不使小车偏离轨迹，或者说偏离轨迹后应该迅速调整回到轨迹上。法向误差的控制分析图如图 9 所示。由位置误差  $R = \sqrt{(x - x_d)^2 + (y - y_d)^2}$ （其中，小车位置在  $(x, y, \theta)$ ），可以得出法向误差  $d$  的表达式，并且可以得到法向偏离轨迹的速度  $\dot{d}$  的表达式分别如下：

$$\dot{d} = v_s \tan \theta_p$$

$$d = R \cos\left(\tan^{-1}\left(\frac{y - y_d}{x - x_d}\right) - \frac{\pi}{2} - \theta_d\right)$$

(3) 法向位置误差

$$d = R * \cos\left(\tan^{-1}\left(\frac{y - y_d}{x - x_d}\right) - \frac{\pi}{2} - \theta_d\right)$$

$$\dot{d} = v * \sin \theta_p = v_s \tan \theta_p$$

控制目标3  
 $d \rightarrow 0$

图 8：法向误差控制分析图

在得到曲线跟踪误差后，我们便可根据跟踪误差  $e(t)$  建立从  $(x_d, y_d, \theta_d)$  预期轨迹到速度与转向角的控制模型，如下图所示：

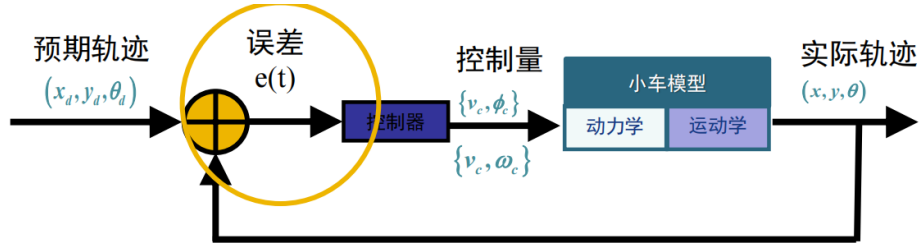


图 9：轨迹跟踪系统模型

## 4.2 阿克曼模型小车曲线跟踪线性化模型

根据图 9 可知，在得到跟踪误差  $e(t)$  后，我们可以通过控制器得到控制量即速度与转向角  $(v_c, \phi_c)$ ，根据  $(v_c, \phi_c)$  我们便可根据之前计算得到的动力学模型和运动学模型控制小车沿着预期轨迹行驶并实时计算跟踪误差对路径进行校正。

在期中的报告中，我们得到了阿克曼模型的运动学方程。在上面的分析中，我们又得到了阿克曼模型的动力学方程和跟踪误差方程，由此我们有：



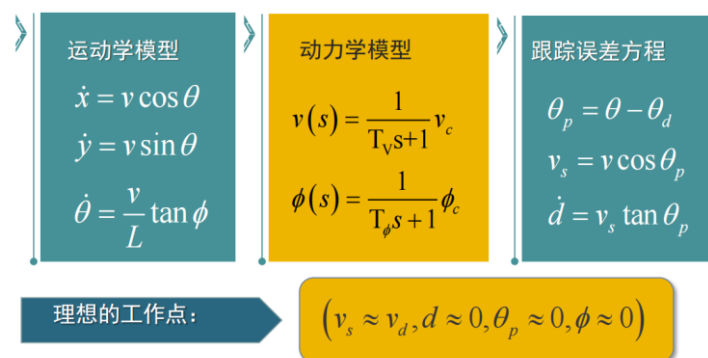


图 10 阿克曼模型小车曲线跟踪算法流程

接下来我们分别论述小车的法向控制模型以及前向控制模型，讨论控制方法并且描述系统的动态特性以及静态特性。

### I. 法向控制模型

由上述建立的模型，我们可以得到控制量 $\phi_c$ 与方向误差  $d$  之间的关系，具体信号流图<sup>[4]</sup>见图 11，系统框图见图 12。我们可以通过对输入 $\phi_c$ 的控制使得法向误差趋近于 0。

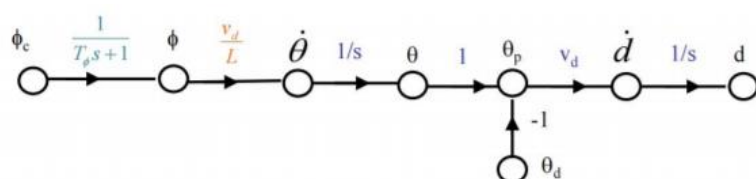


图 11 法向控制模型信号流图

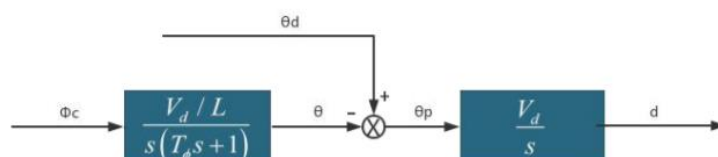


图 12 法向控制模型系统框图

### II. 前向控制模型

在前向控制中， $V_c$ 为控制输入，具体信号流图如图 13 所示，系统框图见图 14。通过控制输入使得前进速度达到期望的前进速度 $V_c$ 。

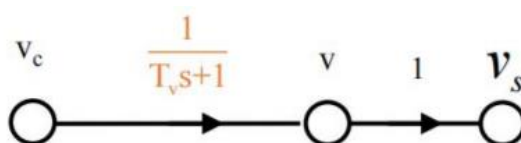


图 13 前向控制模型信号流图

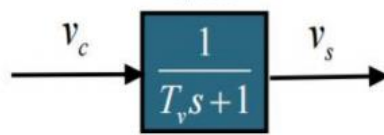


图 14 前向控制模型系统框图

### 4.3 控制器设计<sup>[7]</sup>

在所得图 12、图 14 的线性化模型基础上，我们需要针对这两个系统进行控制器的设计：

首先是前向速度控制器的设计。由于机器人的目标是跟踪曲线行驶，对于速度的要求并不高，保持匀速行驶也是可实行的方案，因此使用简单的 P 控制即可（如图 15 所示）。

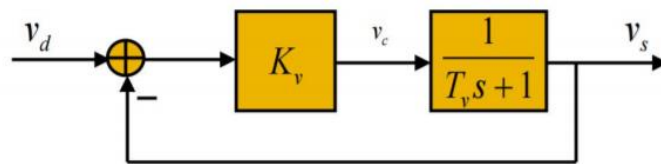


图 15 前向速度控制器

而对于法向控制，考虑到法向控制的精度要求较高，且由于前向速度控制器保证了机器人在转弯过程中维持较低的速度，所以在一定程度上，法向控制系统的响应速度可以适当降低。因此综合考虑，法向控制器采用 PID 控制（如图 16 所示）。

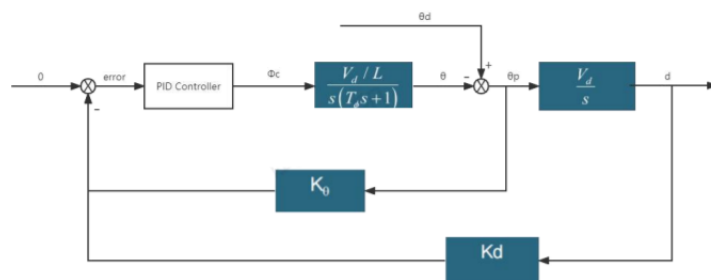


图 16: 法向控制器设计

### 4.4 控制系统性能分析

对于前向速度控制系统，比例控制足以使系统的反应足够迅速，但其仍存

在一些问题。由于设定值为转向角的倒数，则当转向角接近 0 的时候，设定值将变得非常大，此时控制器的输出也将会非常大，驱动电机的转速将会骤升，容易引起机器人的失控。为防止意外情况发生，可以对控制方案进行改进。对于输入控制器的误差来说，对其进行判断：当误差大于某一个阈值 $e_{max}$ 时，输入比例控制器的误差保持为 $e_{max}$ ，其余情况正常处理。在增加了该环节后，系统的鲁棒性进一步得到提升，保证了速度变化的平稳，进而保证了移动机器人在执行任务过程中的稳定行驶。

对于法向控制系统，该系统响应为  $d$  与  $\theta p$ ，可以看到这两个目标都可以通过系统中的状态进行表示。系统目标为使得这两者趋近于 0，那么便可以通过调节反馈  $K$  的值，改变系统传递函数的极点位置，极点的实部为负且距离虚轴越远，则系统达到预期目标（0）的速度越快，由于系统本身存在积分环节，所以四轮阿克曼模型移动机器人运动控制系统设计误差一定可以趋于 0，自然，系统响应的越快，小车调整时间越短，轨迹跟踪的精度就越高。对于参数整定，我们可以使用 临界比例法，Ziegler-Nichols 等工程方法。

除此之外，我们也可以通过 LQR 最优控制的方法设计输入 $\phi_c$ 与  $V_c$ 。对于位置误差  $d$ ，我们可以更为详细地将其表示为  $x$  方向上面的误差以及  $y$  方向上面地误差。

## 5.总结

本论文在期中建立好的小车运动模型的基础上，针对小车行驶路径规划和曲线跟踪控制进行了研究。首先，以 B 样条为基础，通过合理的路径规划算法，实现了小车在复杂环境下的高效行驶。其次，利用阿克曼模型对移动机器人进行了动力学简化模型分析，为后续曲线跟踪控制提供了理论基础。在小车曲线跟踪的线性化模型系统结构中，通过对跟踪误差的分析，确定了关键性能指标，为控制器设计提供了依据。基于阿克曼模型小车曲线跟踪的线性化模型，本论文设计了相应的控制器，并对控制系统的性能进行了分析。实验结果表明，所设计的控制系统在小车曲线跟踪任务中取得了良好的效果。本研究对于提升小车路径规划和曲线跟踪控制的性能具有一定的理论和实践意义，为相关领域的

研究和应用提供了有益的参考。

## 参考文献

- [1] Belta C , Isler V , Pappas G J . IEEE TRANSACTIONS ON ROBOTICS 1 Discrete Abstractions for Robot Motion Planning and Control in Polygonal Environments[J]. IEEE Transactions on Robotics, 2008, 21(5):864-874.
- [2] Zotov Y K . Controllability and stabilization of the programmed motions of a transport robot[J]. Journal of Applied Mathematics and Mechanics, 2000, 64( 6):871-890.
- [3] R·西格沃特, I·R·诺巴克什, D·斯卡拉穆扎, 等. 自主移动机器人导论[M]. 西安交通大学出版社, 2013.
- [4] 胡寿松. 自动控制原理(第四版)[M]. 科学出版社, 2001.
- [5] 寇发荣, 杨慧杰, 张新乾等. 采用状态反馈的无人车路径跟踪横向控制 [J]. 机械科学与技术, 2022, 41(01):143-150. DOI:10.13433/j.cnki.1003-8728.20200330.
- [6] 陈旭. 四驱四转智能车转向轨迹规划及稳定性研究 [D]. 安徽工程大学, 2022. DOI:10.27763/d.cnki.gahgc.2022.000118.
- [7] 徐保来, 管贻生, 苏泽荣等. 改进动态窗口法的阿克曼移动机器人局部路径规划器[J]. 机电工程技术, 2016, 45(09):21-26.

## 附录 (MATLAB 代码)

```
1  C=[0 0.3 1 2 2.7 3 3.3 4 5 5.7 6
2      0 1.25 1.75 1.75 1 0 -1 -1.75 -1.75 0 ]; %控制点坐标
3  V1=[0;1]; %起点 C1 出发速度方向 V1,
4  V2=[0;1]; %到终点 Cn 速度方向 V2,
5  L=0.9/2; %L 为小车长度
6  NC=length(C); %原始控制点数
7  RC=[C(:,1)-V1*L,C(:,1),C(:,1)+V1*L,C(:,2:NC-1), ...
8      C(:,NC)-V2*L,C(:,NC),C(:,NC)+V2*L];
9  N=length(RC); %控制点扩充后的控制点数目
10 s=0:0.01:1; %归一化路程
11 f1s=(1-s).^3/6; %四个样条函数 f1(s)
12 f2s=(3*s.^3-6*s.^2+4)/6; %f2(s)
13 f3s=(-3*s.^3+3*s.^2+3*s+1)/6; %f3(s)
14 f4s=s.^3/6; %f4(s)
15 %绘制 B 样条曲线
16 figure(1); mycolor='mbc';
17 plot(RC(1,:),RC(2,:), 'r*');
18 hold on;
19 for i=1:N-3
20     P = RC(:,1)*f1s+RC(:,i+1)*f2s+RC(:,i+2)*f3s+RC(:,i+3)*f4s;
21     plot(P(1,:),P(2,:), mycolor(mod(i,3)+1), LineWidth=1.25);
22     title('过起点和终点的 B 样条曲线');
23     xlim([-0.25 6.25]);
24     ylim([-1 3]);
25     hold on;
26     grid on;
27 end
```

```

30 — figure(2)
31 — for i=1:N-3
32 —     P = RC(:,i)*f1s+RC(:,i+1)*f2s+RC(:,i+2)*f3s+RC(:,i+3)*f4s;
33 —     v_x = diff(P(1,:));
34 —     a_x = diff(v_x);
35 —     v_y = diff(P(2,:));
36 —     a_y = diff(v_y);
37 —     a_x(length(v_x)) = a_x(end); % 使数组维度一致
38 —     a_y(length(v_y)) = a_y(end);
39 —     K = abs(v_x.*a_y-a_x.*v_y) ./ (v_x.^2+v_y.^2).^(3/2);
40 —     K(length(s)) = K(end);
41 —     plot(P(1,:),K,mycolor(mod(i,3)+1),LineWidth=1.25);
42 —     title(' 轨迹曲线的曲率变化' );
43 —     xlim([-0.25 6.25]);
44 —     ylim([-0.1 1.2]);
45 —     hold on;
46 —     grid on;
47 — end
48 — hold off;
49 — %绘制机器人运动轨迹
49 — %绘制机器人运动轨迹
50 — figure(3)
51 — for i=1:N-3
52 —     P = RC(:,i)*f1s+RC(:,i+1)*f2s+RC(:,i+2)*f3s+RC(:,i+3)*f4s;
53 —     v_x = diff(P(1,:));
54 —     v_y = diff(P(2,:));
55 —     title(' 机器人运动轨迹' );
56 —     phi=v_y./v_x;
57 —     for k = 1:20:length(s)-1
58 —         car = polyshape([P(1,k)-0.9/2 P(1,k)-0.9/2 P(1,k)+0.9/2 P(1,k)+0.9/2],
59 —             [P(2,k)-0.4/2 P(2,k)+0.4/2 P(2,k)+0.4/2 P(2,k)-0.4/2]);
60 —         plot(rotate(car,atan(phi(k))*180/3.1415,[P(1,k),P(2,k)]), 'FaceColor',mycolor(mod(i,3)+1),...
61 —             'FaceAlpha',0.3);
62 —     end
63 —     hold on;
64 —     grid on;
65 — end
66 — hold off;

```