

---

队伍编号	MC2205063
题号	C

---

## 基于 A\*模型的自动泊车问题研究

### 摘 要

自动泊车系统可以通过车辆周身搭载的传感器测量车身与周围环境之间的距离和角度，收集传感器数据计算出操作流程，同时调整方向盘的转动实现停车入位。因此，自动泊车技术是比较实用的一项功能，自动泊车技术有助于解决人口密集城区的一些停车和交通问题，驾驶者不用再担心因技术不过关而泊不好车。

本文针对自动泊车问题，主要通过建立 A\*模型，应用拟合算法及微分方程参数估计实现三种不同泊车位置下小车行驶路径轨迹的拟合，及对最小的转弯半径和曲率变化率的相关求解。

针对问题一，我们根据题目中的方向盘最大转角，最大转速等数据来确定控制点位置，计算出前轮的最大转角；通过相关公式可求出最小转弯半径，再通过假设无人车做匀加速直线运动列不等式关系组可求最短路径；最后，通过拟合算法分三部分来讨论曲率相对路径长度的变化率。

针对问题二，以车库入口为无人车初始位置，基于 A\*算法，可以获得无人车从起点到终点的几何路径轨迹；再通过曲线差值法来求出轨迹规划进而求解出位置，速度，加速度及加加速度；最后，再求出角速度及角加速度。

针对问题三，我们建立了基于蚁群算法的泊车模型。根据停车位的空闲状况，首先规划了合适的泊车轨迹，建立出相应的泊车模型，得到泊车过程中所需的路程及时间。我们以汽车在倒车过程中的旋转角度来辅助我们判断 3 种不同的倒车情况，并以此计算三类情况下所需要的倒车时长，并通过系统的将无人车在直线弯道倒车入库各环节的时间汇总，在已知空余车位的情况下寻求最优解，从而得到整体的停车规划路线，然后对模型进行优化。

针对问题四，我们依据前三问所求得的数据，建立基于蚁群算法的动态泊车模型。由于车辆的进入和离开，导致停车位会被随机占用或释放，在问题一和问题二中我们可以知道最小转弯半径，曲率变化，各个时刻位置速度等相关参数，根据无人车选择不同的停车位，当可用停车位数量不变情况下，建立数学模型，建立无人车泊车轨迹规划模型，得出无人车泊车策略，并进行仿真实验。

**关键词：**最小转弯半径；A\*算法；仿真实验；运动轨迹规划；MATLAB；蚁群算法

# 目录

基于 A*模型的自动泊车问题研究.....	1
摘 要.....	1
一、问题重述.....	1
1.1 问题背景.....	1
1.2 目标任务.....	1
二、模型假设及符号说明.....	2
2.1 模型假设.....	2
2.2 主要的符号说明.....	2
三、问题分析.....	3
3.1 针对问题一.....	3
3.2 针对问题二.....	3
3.3 针对问题三.....	3
3.4 针对问题四.....	3
四、问题一模型的建立与求解.....	4
4.1 最小转弯半径的求解.....	4
4.2 最短距离的求解.....	5
4.3 曲率对相对路径长度的变化率的影响.....	5
4.3.1 曲率变化率.....	5
4.3.2 曲率变化率对交通安全的影响.....	6
4.3.3 曲率变化率的安全评价标准.....	7
五、问题二模型的建立与求解.....	8
5.1 平行泊车路径规划.....	8
5.1.1 基于圆弧-直线的平行泊车路径规划.....	8
5.1.2 平行泊车的起始区域计算.....	9
5.1.3 平行泊车路径的仿真规划.....	12
5.2 垂直泊车路径规划.....	13
5.2.1 一步式垂直泊车路径规划.....	13
5.2.2 垂直泊车起始区域计算.....	14
5.2.3 垂直泊车路径规划仿真.....	15
5.3 斜式泊车路径规划.....	17
5.3.1 斜式泊车路线分析.....	17
5.3.2 斜式车位起始泊车区域计算.....	17
5.3.3 斜式泊车路径规划仿真.....	19
六、问题三模型的建立与求解.....	21
6.1 问题三具体分析与 A*算法.....	21
6.1.1 问题三的详细分析.....	21
6.1.2 A*算法.....	22
6.2 准备停车位置与目的地的距离.....	22
6.3 模型的建立及仿真研究.....	23
七、问题四模型的建立与求解.....	27

7.1 自动泊车的轨迹规划.....	27
7.1.1 模型的选取与假设.....	27
7.1.2 模型的建立与求解.....	28
八, 模型的评价与改进.....	30
8.1 模型的优点.....	30
8.2 模型的缺点.....	30
8.3 模型的改进.....	30
8.3.1 改进蚁群算法求解连续性空间优化问题.....	31
8.3.2 基于蚁群算法的二维路径规划.....	31
8.3.3 蚁群算法求解 TSP 问题.....	32
九, 参考文献.....	34
附录.....	35

# 一、问题重述

## 1.1 问题背景

自动驾驶技术是近年来十分热门的智能技术，自动泊车是自动驾驶技术中落地最多的场景之一，随着汽车数量的不断增加，公路上，街道上，停车场里，居民小区中等地拥挤不堪，可以用来利用的泊车空间越来越少；此外，每年学习驾车的人不断增多，由于行车不熟练而出现的在泊车时的各种问题。以无人成用车为例，实现在停车场中进行自动泊车的功能，因此，如何去控制车辆的行驶过程中出现的转向、制动来实现较为安全的自动驾驶，并且通过自动寻找空余的停车位成为了较受大家关注的问题；不仅要实现自动泊车，而且同时防止潜在的碰撞问题隐患的功能是关键。

## 1.2 目标任务

本研究以无人成用车为例，实现在停车场中进行自动泊车的功能，假设控制点位于后轴中心上，给定各个约束条件，请你们的参赛队，研究解决以下问题：

**问题一：**请根据给出的无人车模型的参数，计算车辆最小转弯半径。少距离能加速到最大限制速度 20km/h？当车速为 20km/h 时，无人车如果如果限制车辆最大加加速度为  $20\text{m/s}^2$ ，无人车沿直线行驶时，最短需要多需要转弯，从沿直线行驶状态开始转弯，路径上的曲率相对路径长度的变化率大小有何限制？

**问题二：**无人车初始位置为车库入口，请建立无人车泊车的数学模型，并给出从初始位置到指定停车位的泊车轨迹，轨迹应包括停车位都被占用，泊车过程中无人车不能与其发生冲突或碰撞。分别速度、角加速度等，并给出可视化轨迹图。在这个过程中标注红色禁停的考虑三种不同的车位情况，10 号垂直停车位、82 号平行停车位、31 号倾每时刻无人车的行驶路径长度、车辆朝向、速度、加速度、加加速度、角斜停车位（倾斜角为  $45^\circ$ ）。

**问题三：**请你们根据当前停车位轨迹；标注红色禁停的停车位都被占用，泊车过程中无人车不能与其发生冲突或碰撞。在这个过程中，试建立通用模型，并考虑算法怎样设计的状况，建立泊车模型，计算出最优停车位，给出从当前位置到停车位的能适应车库中任意停车位被占用的状况，并考虑这个过程算法复杂性。

**问题四：**假设在当前状态下每小时内从入口进入和从出口离开停车场的车辆均为 30 辆，因车辆的进入和离开，导致停车位会被随机占用或释放。请你们为无人车建立泊车模型，并给出从当前位置到最优停车位的行驶轨迹的仿真结果。

## 二、模型假设及符号说明

### 2.1 模型假设

为了便于考虑问题，我们在不影响模型的准确性的前提下，作出下面的几项假设：

- (1) 假设在加速阶段，无人车做匀加速运动，在减速阶段，无人车做匀减速运动；
- (2) 假设所有转弯过程车做匀速运动且运动轨迹均为圆弧；
- (3) 假设轮胎与地面的摩擦消耗的时间忽略不计；并且忽略无人车在垂直地面的方向上的运动；
- (4) 假设无人车不受转向系统相关的影响，直接用前轮转角作为输入；
- (5) 忽略车辆受到的空气阻力；车辆与地面保持良好的滚动摩擦；
- (6) 假设蚁群算法中只考虑信息素浓度对蚂蚁选择路径的影响。

### 2.2 主要的符号说明

注：此为本文的主要符号说明，其他符号解释详见正文部分。

表 2.1 主要符号说明

符号	含义	单位
$\alpha$	方向盘最大转角	°
$v_s$	无人车的方向盘的最大转速	°/s
$L$	车身长度	m
$w$	车身宽度	m
$L_1$	无人车的轴距	m
$R_{min}$	无人车最小转弯半径	m
$L_{min}$	最小车位长度	m
$\theta$	车身导航角	°
$L_d$	路面宽度	m
$A$	车位占用率	%

## 三、问题分析

### 3.1 针对问题一

本题主要研究根据给出的无人车模型的数据，计算车辆最小转弯半径，加速到最大限制速度的行驶距离，以及转弯时路径上曲率相对路径长度的变化率大小限制条件。首先，我们根据题目中无人车轴距，轮间距，方向盘最大转角，最大转速等已知数据来确定控制点位置，进一步计算出前轮的最大转角；通过相关公式可求出最小转弯半径，再通过假设无人车做匀加速直线运动列不等式关系组可求最短路径；最后，通过拟合算法分三部分讨论曲率相对路径长度的变化率的影响。

### 3.2 针对问题二

本题主要研究了平行泊车系统的路径规划方法，首先根据车辆参数计算出理论可行泊车起始区域；然后分析对比了多种平行泊车路径规划方法的优缺点，最终选取了曲率连续且易于车辆跟踪的等速偏移和正弦曲线的叠加曲线作为此次泊车路径，以车库入口为无人车初始位置，针对车位尺寸较小情况规划了泊车姿态调整，讨论了泊车过程出现障碍物时的路径选取；最后进行了路径仿真分析，验证了路径的可行性。基于 A\* 算法，可以获得无人车从起点到终点的几何路径；再通过曲线差值法轨迹规划和 Lingo 软件编写程序来求解位置，速度，加速度和加加速度；最后再求角速度和角加速度。

### 3.3 针对问题三

本题主要研究了题目需要需要根据停车位现有的状况，建立相应的泊车模型，基于蚁群算法，计算出较优停车位，并得出从当前位置到停车位的轨迹。得到泊车过程中所需时间，首先需要得到整个行驶过程中所需要经过的路段及路段对应的时间，同时需要汽车在倒车过程中的倒车时长，我们以汽车在倒车过程中的旋转角度来确定在不同的倒车情形下，所需要的倒车时长，并通过系统的将无人车在直线弯道倒车入库各环节的时间汇总，在已知空余车位的情况下寻求最优解，从而得到整体的停车规划路线。

### 3.4 针对问题四

本题主要研究了每小时内从入口进入和出口离开停车场的车辆均为 30 辆情况下无人车到最优停车位的行驶轨迹，由于车辆的进入和离开，导致停车位会被随机占用或释放，在问题一和问题二中我们可以知道最小转弯半径，曲率变化，各个时刻位置速度等相关参数，根据无人车选择不同的停车位，当可用停车位数量不变情况下，建立数学模型，建立无人车泊车轨迹规划模型，得出无人车泊车策略。现有的基于对路径规划问题的相关的算法研究主要包含 Dijkstra 算法、遗传算法、D\* 算法、A\* 算法以及粒子群算法等。我们针对实际的停车场环境，进行相关的道路规划，首先对实际的停车场的基本环境进行建模，根据无人车车辆的运动学模型，基于蚁群算法进行路径规划，采用垂直泊车、平行泊车、倾斜泊车三种路径规划方法进行泊车，基本做出车辆安全无碰的泊

车入位。

## 四、问题一模型的建立与求解

### 4.1 最小转弯半径的求解

最小转弯半径：当方向盘转到极限的位置时，汽车以最低的较稳定车速转向行驶，外侧的车轮中心行走的圆轨迹半径。它表示汽车为了通过狭窄地带或弯曲地带时，绕过障碍物的能力。越小的转弯半径，标志着汽车的机动性越好。

控制点指的是车身中某一特定点的轨迹；控制点一般是事先选定的，它通常位于无人车车辆的对称轴（无人车车辆可以近似地认为是左右轴对称）上一点，在行驶时，控制点的位置会与轨迹点相重合，控制点所处的速度方向一般与轨迹点的方向角相一致，题目中假设控制点位于后轴中心上。

首先，由题目给出的指定条件长  $L=4.9\text{m}$ ，宽  $W=1.8\text{m}$ ；无人车轴距  $L_1=2.8\text{m}$ ，轮间距为  $W_1=1.7\text{m}$ ，方向盘的最大转角为  $470^\circ$ ，方向盘与前轮转角的传动比是  $16:1$ ，即方向盘转动  $16^\circ$  时，前轮转动  $1^\circ$ ，方向盘的最大转速为  $V_s=400^\circ/\text{s}$ 。我们由此可以计算出前轮的最大转角为  $\alpha=470^\circ/16=29.375^\circ$ 。

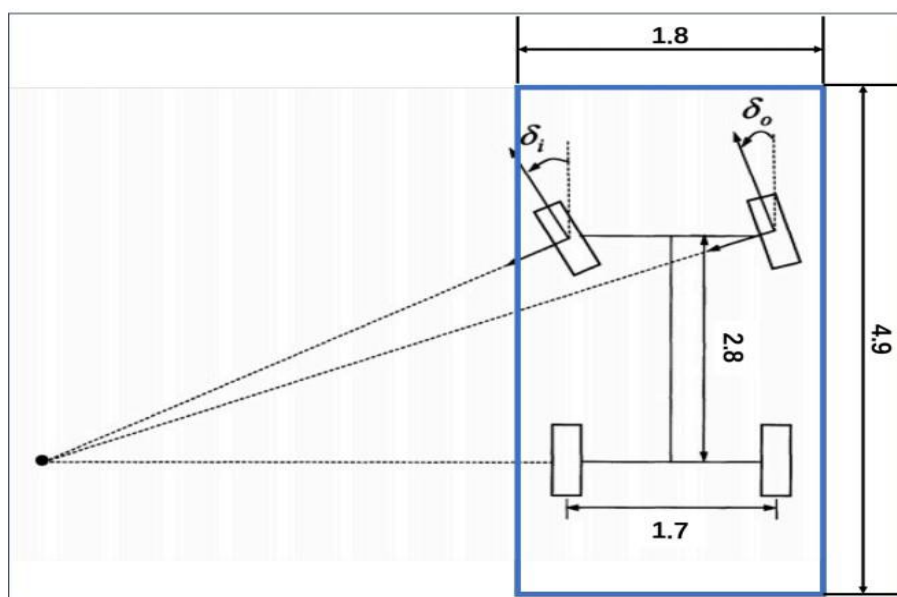


图 4-1 阿克曼车辆模型示意图

由于轴距  $L_1$  为 2.8 米，由调查研究可以知，无人车的最小转弯半径为：

$$R_{min} = \frac{L_1}{\sin \alpha} \quad (4.1)$$

式中， $R_{min}$  是最小转弯半径， $L$  是无人车的轴距， $\sin \alpha$  是最大转弯角度的正弦值。可求得最小转弯

半径 $R_{min} = 5.708m$ 。控制点到转弯半径圆心的距离为 $R_c = 4.858m$ 。需要注意的是，此时最小转弯半径是前外轮到其轨迹圆心的距离，并非控制点到圆心的距离。计算可得控制点轨迹的曲率为 $\rho = \frac{1}{R_c} = 0.206$ ，高于轨迹线的最大曲率 0.21，低于 0.205，所以该半径下的轨迹线不合理。同时假设一般车辆轮胎宽度为  $d1=0.25m$  以及无人车轮胎中心与车身对称轴的距离为  $l=0.85m$ ，因此，控制点到圆心的距离为

$$R_c = R_{min} - \left(l - \frac{d_1}{2}\right) = 4.983, \text{ 曲率 } \rho = \frac{1}{R_c} = 0.201, \text{ 低于 } 0.205, \text{ 所以该半径下的轨迹较为合理, 即 } R_{min} = 5.833m。$$

## 4.2 最短距离的求解

问题假设：无人车做匀加速直线运动，加速度  $a = 3m/s^2$  且初速度为 0 根据经典运动学可列出如下不等式关系组：

$$\begin{cases} 0 < \frac{da}{dt} < 20 \\ 0 < a \leq 3 \\ 0 < v \leq 5.56 \\ v \leq at \\ x_{min} \leq \frac{1}{2}at^2 \end{cases} \quad (4.2)$$

式中 $x_{min}$ 为无人车加速到 20km/h 所达到的最短距离， $v$  为无人车的行进速度。根据以上不等式关系组求得 $x_{min} = 5.134m$

## 4.3 曲率对相对路径长度的变化率的影响

### 4.3.1 曲率变化率

曲率变化率 CCR：平曲线路段的单位长度与角度的变化之和，它是通过与半径、转角、曲线长度的有关弯曲过程来描述曲线的特征，表示出平曲线曲率变化的连续性。查阅相关资料可知，用来描述道路特性中，最具有实际意义的参数即为带有缓和曲线的单圆曲线曲率变化率  $CCR_s$ ：

$$CCR_s = \frac{200}{\pi} \left( \frac{L_{s1}}{2R} + \frac{L_c}{R} + \frac{L_{s2}}{2R} \right) \cdot L^{-1} \quad (4.3)$$

该式中， $CCR_s$ 指有缓和曲线的单个平曲线的曲率变化率，单位为  $gon \cdot km^{-1}$ ； $L$  指曲线的总长度，单位 km； $R$  指圆曲线半径，单位 m。 $L_{s1}$ 、 $L_{s2}$ 指前后的缓和曲线长度，其单位为 m； $L_c$ 指圆曲线的长度，单位 m；从该公式中我们不难看出， $CCR_s$ 的计算与平曲线的半径、缓和曲线长度、曲线长度、转角等因素有关，该式将各种因素组合得到了道路特征的综合表现。



### 4.3.2 曲率变化率对交通安全的影响

查阅相关文献我们了解到, kerbs 的研究中发现, 交通事故的发生率随着  $CCR_s$  的数值的增加也会增大, 但其增大的幅度相对来说较小,  $CCR_s$  值比较高的路段交通事故发生的概率一般是  $CCR_s$  值较低的路段的 2 到 3 倍, 然而, 对于  $CCR_s$  值较高的路段 ( $CCR_s > 500 \text{gon} \cdot \text{km}^{-1}$ ), 交通事故发生的概率会很高, 如图 4-2 所示:

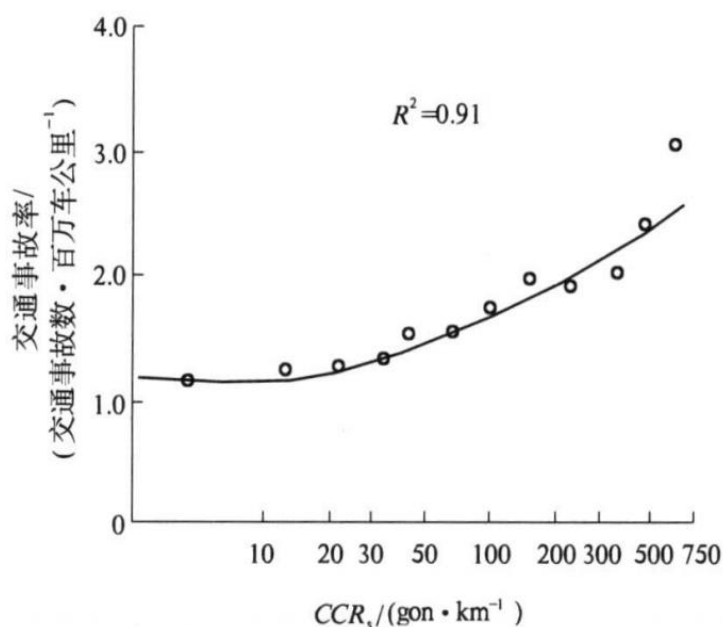


图 4-2 交通事故率与  $CCR_s$  关系图

交通事故的发生过程可以看做是一个能量的转化过程, 在比较困难的行车条件下, 发生交通事故的汽车的动能被转化成热能以及车辆变形时的机械能, 由此可见  $CCR_s$  的值越高, 事故发生过程中的能量的转化也就越多, 其后果也就越严重, 如图 4-3 所示:

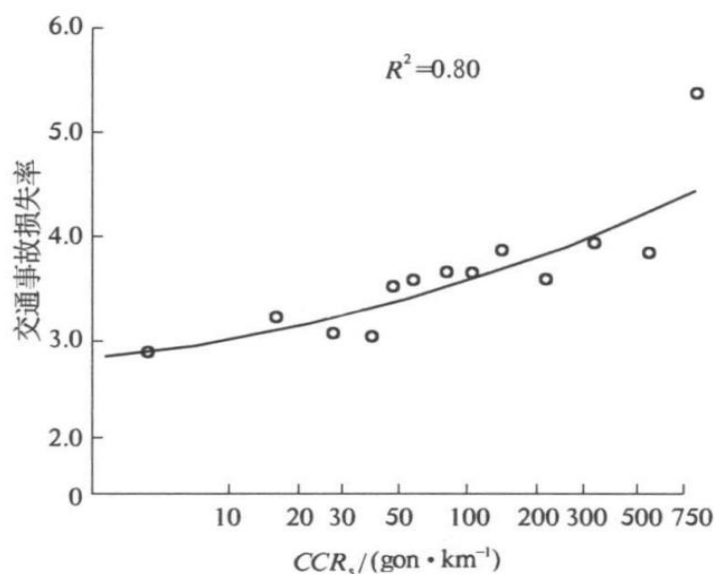


图 4-3 交通事故损失率与  $CCR_s$  关系图

### 4.3.3 曲率变化率的安全评价标准

关于安全评价标准，我们基于三个方向考虑：设计均衡性、运行速度连续性、行驶稳定性这三个方面具体细节如下表：

表 4-1 设计均衡性的安全标准

成果评价	评价指标	评价结论
优秀	$ \overline{CCR_s} - CCR_s  \leq 180 \text{ gon} \cdot \text{km}^{-1}$	无需修改设计参数,可以保证设计一致性和安全性.
一般	$180 \text{ gon} \cdot \text{km}^{-1} <  \overline{CCR_s} - CCR_s  < 360 \text{ gon} \cdot \text{km}^{-1}$	无需重新设计,可通过修改相关设计参数调整路段的一致性.无法修改时,应采用交通安全设施.
较差	$ \overline{CCR_s} - CCR_s  \geq 360 \text{ gon} \cdot \text{km}^{-1}$	①不能满足设计一致性,是交通事故频发路段; ②宜重新设计,若由于地形等不能修改设计参数时,必须采用交通安全设施以控制车速.

表 4-2 运行速度连续性的安全标准

成果评价	评价指标	评价结论
优秀	$ \overline{CCR_{s_i}} - CCR_{s_{i+1}}  \leq 180 \text{ gon} \cdot \text{km}^{-1}$	线形连续可以保证运行速度连续性和运行安全.
一般	$180 \text{ gon} \cdot \text{km}^{-1} <  \overline{CCR_{s_i}} - CCR_{s_{i+1}}  < 360 \text{ gon} \cdot \text{km}^{-1}$	无需重新设计,可通过修改相关设计参数调整路段的一致性.无法修改时,应采用交通安全设施.
较差	$ \overline{CCR_{s_i}} - CCR_{s_{i+1}}  > 360 \text{ gon} \cdot \text{km}^{-1}$	①不能满足设计一致性,是交通事故频发路段; ②宜重新设计,若由于地形等不能修改设计参数时,必须采用交通安全设施以控制车速.

表 4-3 行驶稳定性的安全标准

成果评价	评价指标	评价结论
优秀	$CCR_{s_i} \leq 180 \text{ gon} \cdot \text{km}^{-1}$	弯道处最大有效横向力系数数值可以保证汽车安全舒适的通过.
一般	$180 \text{ gon} \cdot \text{km}^{-1} < CCR_{s_i} < 360 \text{ gon} \cdot \text{km}^{-1}$	车乘人员能感觉到弯道的存在,但无需重新设计,可通过修改相关设计参数调整路段.
较差	$CCR_{s_i} > 360 \text{ gon} \cdot \text{km}^{-1}$	①车辆经过弯道受到的离心作用越明显,车辆通过越不安全,容易出现翻车、冲出公路等事故; ②宜重新设计,若由于地形等不能修改设计参数时,必须采用交通安全设施以控制车速.

表中  $CCR_{s_i}$  是平曲线路段的曲率变化率;  $\overline{CCR_s}$  表示平曲线路段的曲线的平均变化率。其计算公式如下：

$$\overline{CCR_s} = \sum_{i=1}^{i=n} (CCR_{s_i} L_i) \cdot \left( \sum_{i=1}^{i=n} L_i \right)^{-1} \quad (4.4)$$

式中,  $CCR_s$  为平曲线曲率变率, 单位为  $\text{gon} \cdot \text{km}^{-1}$ ;  $L_i$  为第  $i$  段的平曲线的长度, 单位为  $\text{m}$ ;  $i$  是平曲线段内曲线个数,  $i=1, 2, 3, \dots, n$ 。

## 五，问题二模型的建立与求解

### 5.1 平行泊车路径规划

#### 5.1.1 基于圆弧-直线的平行泊车路径规划

对于平行泊车本文采用基本的圆弧-直线-圆弧的路线进行路线规划。我们首先假设无人车在进入车位前平行与车位，无人车从C点到达F点的轨迹具体如图5-1所示，我们将其分为三个阶段：CD段是以半径为 $R_1$ 的圆弧运动；DE段是直线运动；EF段是以半径为 $R_2$ 的圆弧运动。

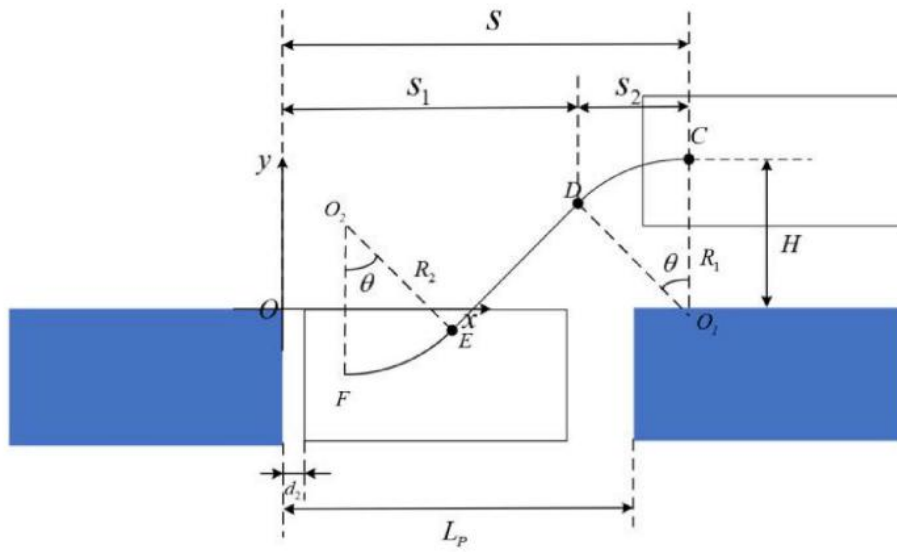


图 5-1 平行泊车路径规划

基于最短路径的优化考虑，本文将 $R_2$ 取为由问题一得到的最小转弯半径 $R_{min}$ ，与此同时为了不受泊车起始点的限制， $R_1$ 的值将随着无人车的初始位置改变，我们可以得到如下表达式：

$$R_1 = \frac{s_2}{\sin \theta} \quad (5.1)$$

$$s_2 = s - s_1 \quad (5.2)$$

式中， $\theta$ 为车身导航角； $s$ 为无人车起始点在 $xOy$ 中的横坐标值； $s_1$ 为D点的横坐标值。

我们采取逆向行驶的方法来确定无人车起始点的区域，根据避障条件来确定F, E, D点的坐标，即确定 $s_1$ 的值，之后只需要根据泊车起始位置调整第一段圆弧路径的转弯半径即可。再根据车辆的最小转弯半径以及道路宽度确定泊车起始点区域的上下限值，从而得到泊车起始区域。

### 5.1.2 平行泊车的起始区域计算

为了满足最短路径，将最后部分的圆弧设为最小转弯半径和直线路段的起点尽可能靠近坐标系原点，具体情况如图 5-2 所示

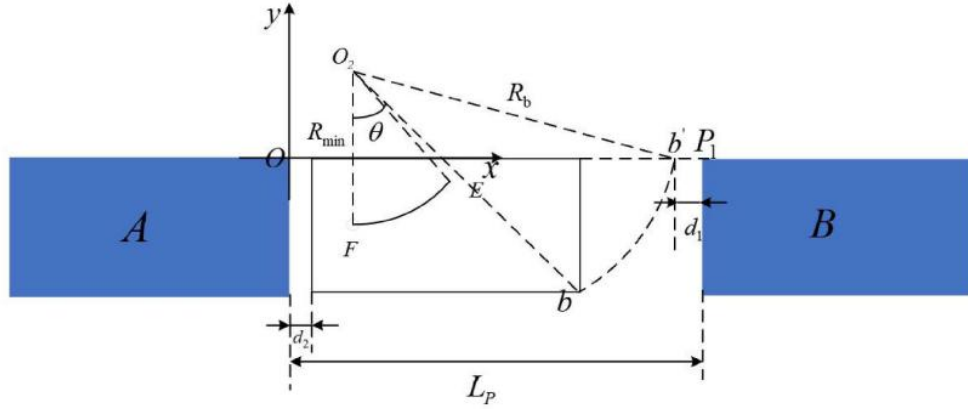


图 5-2 最小平行车位示意图

根据几何关系可以得出最小车位长度 $L_{min}$ 为：

$$L_{min} = d_2 + l_r + \sqrt{R^2 - \left(R_{min} - \frac{w}{2}\right)^2} + d_1 \quad (5.3)$$

5.3 式中， $d_1$ 、 $d_2$  分别是无人车进入泊车位后的前后安全距离，值得注意的是  $d_1$  是随着车位长度增加而增加； $R_b$  为无人车  $b$  点的转弯半径； $w$  为车辆宽度； $l_r$  为后悬距离。我们可以跟几何知识继续得出  $F$ 、 $E$  点的坐标：

$$F\left(d_2 + l_r, -\frac{w}{2}\right) \quad (5.4)$$

$$E\left(x_F + R_{min} \sin \theta, -\frac{w}{2} + R_{min}(1 - \cos \theta)\right) \quad (5.5)$$

式中， $x_F$  为  $F$  点横坐标值。通过对图形的分析得出  $E$  点车辆的航向角为  $\theta$ ，直线路段  $EF$  的斜率为  $\tan \theta$ ，那么无人车起始点  $C$  点的选取范围与直线路段  $EF$  平行，其斜率也为  $\tan \theta$ 。在求到  $E$ 、 $F$  点坐标后，根据最小平行车位得到  $b'$  点的位置来获得  $D$  点位置，具体情况如图 5-3 所示；

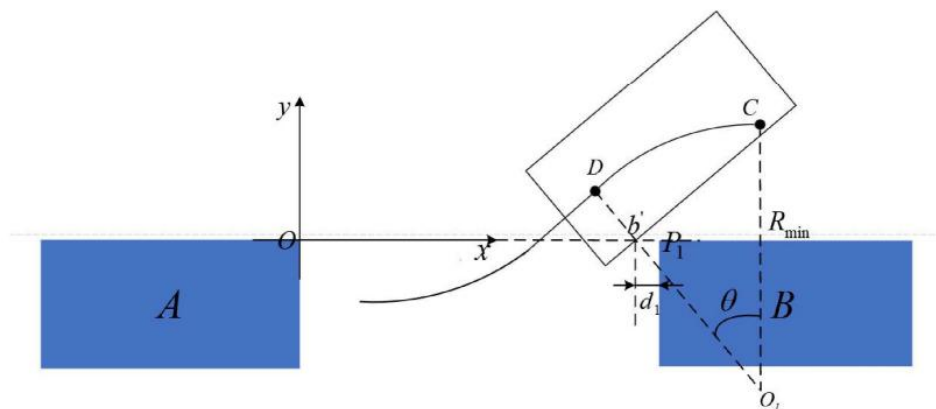


图 5-3 p1 点避障分析

我们此时可以得到 D 点坐标，

$$\begin{cases} x_D = L_p - d_1 - w \sin \frac{\theta}{2} \\ y_D = \left(\frac{w}{2}\right) \cos \theta \end{cases} \quad (5.6)$$

然后根据此时的 D 点计算 C 点范围区域。

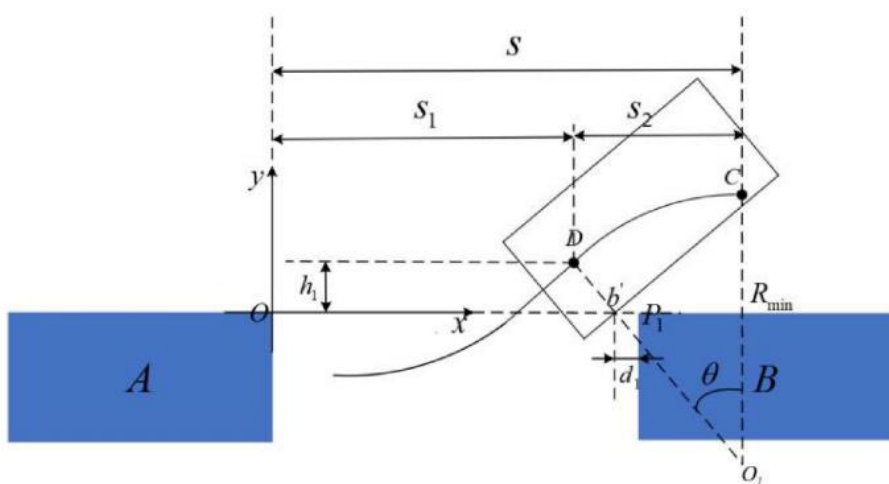


图 5-4 C 点纵坐标下极限值计算

如图 5-4 所示，当无人车转弯时为最小转弯半径  $R_{min}$  时，这时 C 点位置为起始位置区域的下限值，其坐标为：

$$\begin{cases} x_1 = s_1 + R_{min} \sin \theta \\ y_1 = h_1 + R_{min} (1 - \cos \theta) \end{cases} \quad (5.7)$$

式中， $x_1$ ， $y_1$  分别是 C 点下极限位置的横纵坐标； $h_1$  为 D 点的纵坐标； $s_1$  为 D 点的横坐标。C 点上极限位置与无人车进行第一次圆弧倒退行驶时，无人车左上顶点 a 不与道路边缘碰撞，具体如图 5-5 所示：

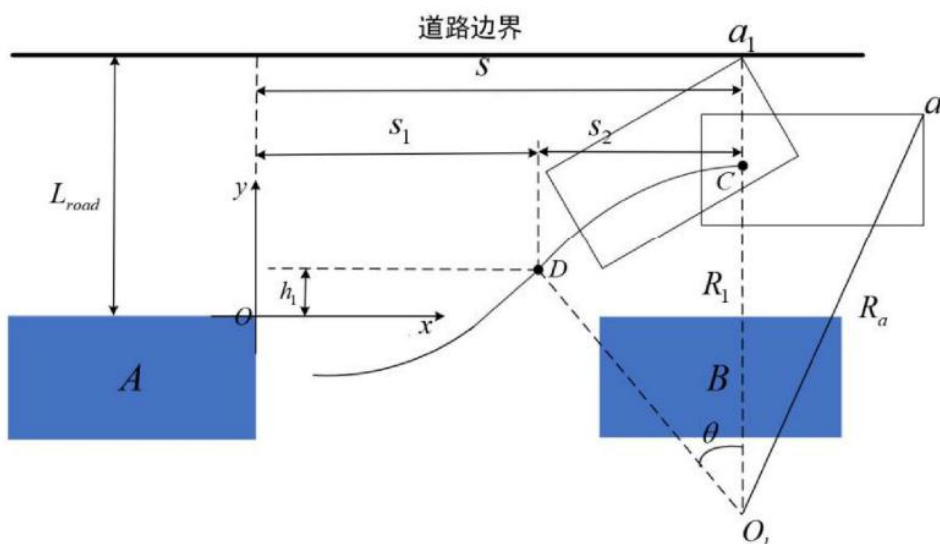


图 5-5 道路边界避障分析

当无人车不在下极限位置时，它的转弯半径随起始点我位置变化而增大，值得注意的是无人车起始点需满足车辆碰撞分析。根据图 5-5 我们得出 C 点坐标：

$$\begin{cases} x_2 = s_1 + R_1 \sin \theta \\ y_2 = h_1 + R_1 (1 - \cos \theta) \end{cases} \quad (5.8)$$

式中， $x_2$ ， $y_2$  分别为 C 点的上极限横纵坐标， $R_1$  为无人车此时的转弯半径。这时  $R_1$  要满足：

$$R_a + h_1 - R_1 \cos \theta \leq L_d \quad (5.9)$$

式中， $L_d$  为路面宽度； $R_a$  为无人车左上顶点 a 的旋转半径。因此我们得到了 C 点的上极限值以及下极限值，则 C 点的选取范围如图 5-6 所示：

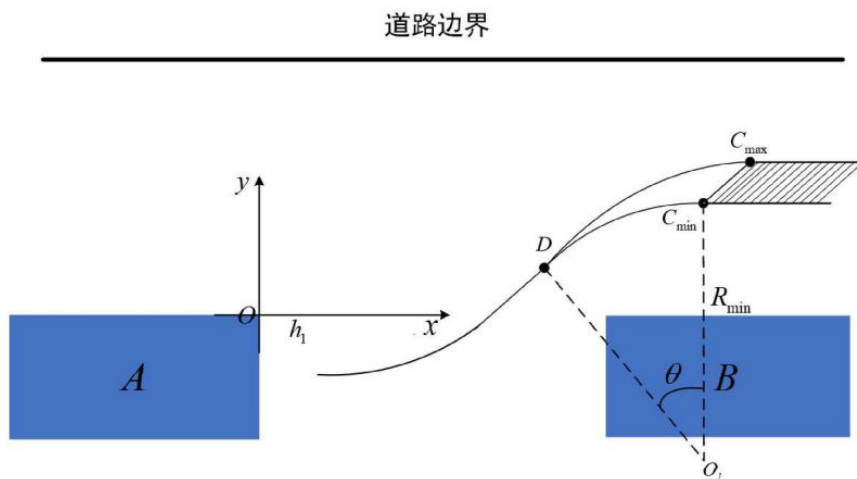


图 5-6 平行泊车起始区域示意图

通过此图我们可以得知，只有当无人车位于阴影部分的时候才可以按照我们所规划的路径进入车位。

### 5.1.3 平行泊车路径的仿真规划

我们借助 MATLAB 软件中的 simulink 中建立约束模型，设置参数，得到如下图仿真轨迹：

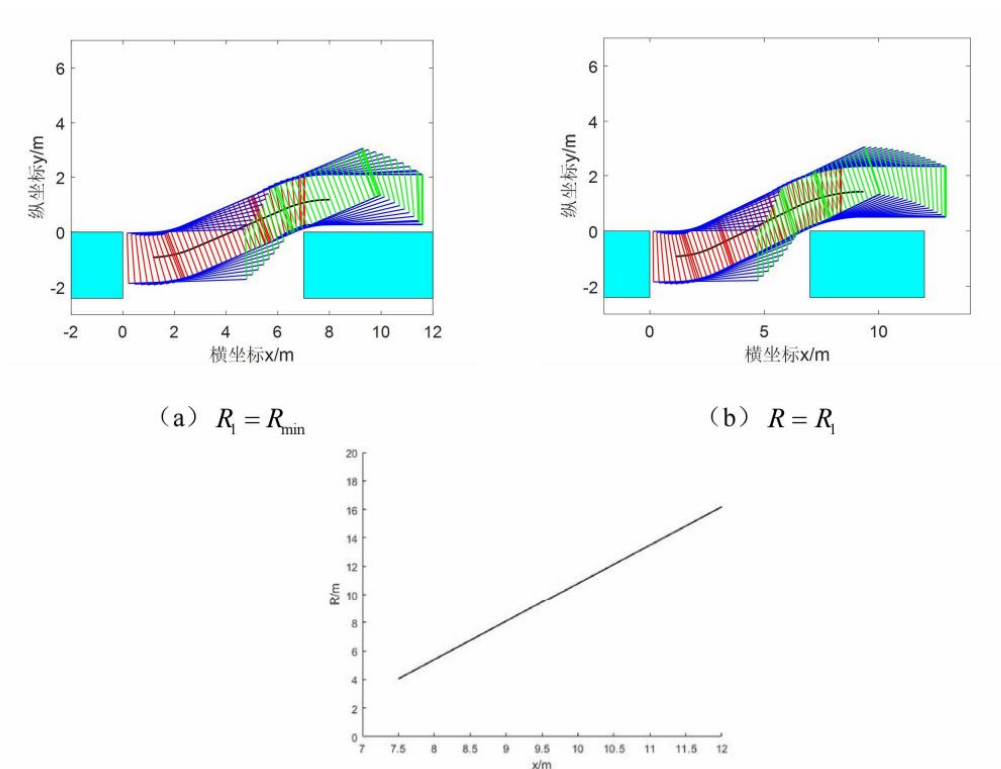
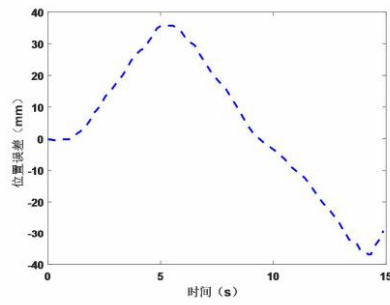
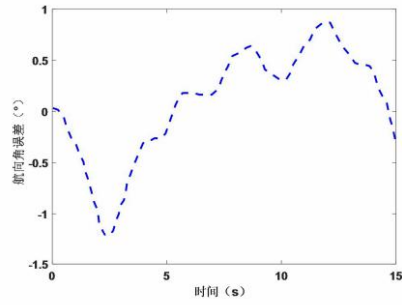


图 5-7 平行泊车路径规划仿真图

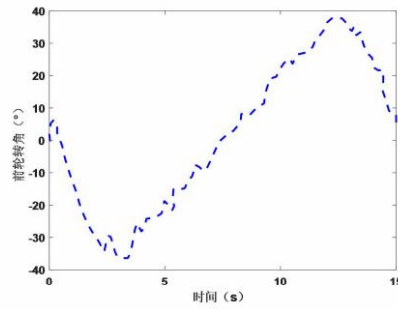
我们在根据路径仿真跟踪得到了无人车航向角，前轮转向角，位置误差等参数，具体如图 5-7 所示：



(a) 位置误差



(b) 航向角误差



(c) 前轮转向角变化

图 5-8 平行泊车路径跟踪图

## 5.2 垂直泊车路径规划

### 5.2.1 一步式垂直泊车路径规划

垂直泊车我们根据无人车所在的初始位置来进行规划，如图十所示，我们可得出无人车的转弯半径 $R_1$ ：

$$s_2 = s - s_1 \quad (5.10)$$

$$s_2 = R_1 \quad (5.11)$$

式中， $s$  是无人车初始坐标值， $s_1$  为车位宽度一半，则 $s_1$ ：

$$s_1 = \frac{w_p}{2} \quad (5.12)$$

进而可得：

$$R_1 = s_1 - \frac{w_p}{2} \quad (5.13)$$



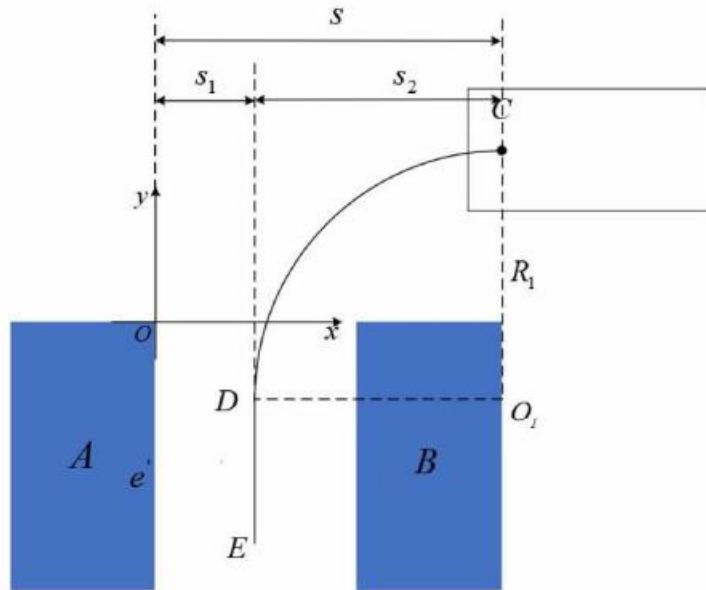


图 5-9 垂直泊车路径规划

### 5.2.2 垂直泊车起始区域计算

首先假设无人车始终位于车位中轴线上, 并且设置安全距离  $d_3$  保证不与右侧发生碰撞, 具体如图 5-10 所示:

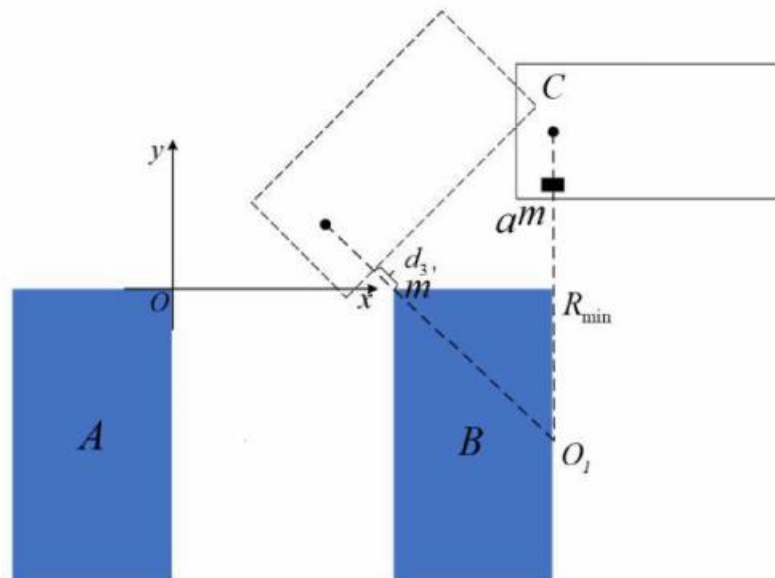


图 5-10 起始位置下极限值计算分析图

根据几何学知识, 我们可得:

$$\sqrt{(x_0 - w_p)^2 + (R_{min} - y_c)^2} + d_3 = R_{min} - \frac{w}{2} \quad (5.14)$$

其中

$$x_0 = R_{min} + \frac{w}{2} \quad (5.15)$$

式中， $y_c$ 即为我们所求下极限值，联立两式得

$$y_c = R_{min} - \sqrt{\left(R_{min} - \frac{w}{2} - d_3\right)^2 - \left(R_{min} - \frac{w_p}{2}\right)^2} \quad (5.16)$$

可以得到 D 点的坐标为

$$\begin{cases} x_D = \frac{w}{2} \\ y_D = -\sqrt{\left(R_{min} - \frac{w}{2} - d_3\right)^2 - \left(R_{min} - \frac{w_p}{2}\right)^2} \end{cases} \quad (5.17)$$

再根据无人车与道路边界的分析可以得到 C 点的上极限值：

$$\begin{cases} x_{Cmax} = R_1 + \frac{P}{2} \\ y_{Cmax} = R_1 + y_D \end{cases} \quad (5.18)$$

其中：

$$y_D = R_1 - y_c \quad (5.19)$$

最终通过计算我们得到图 5-11

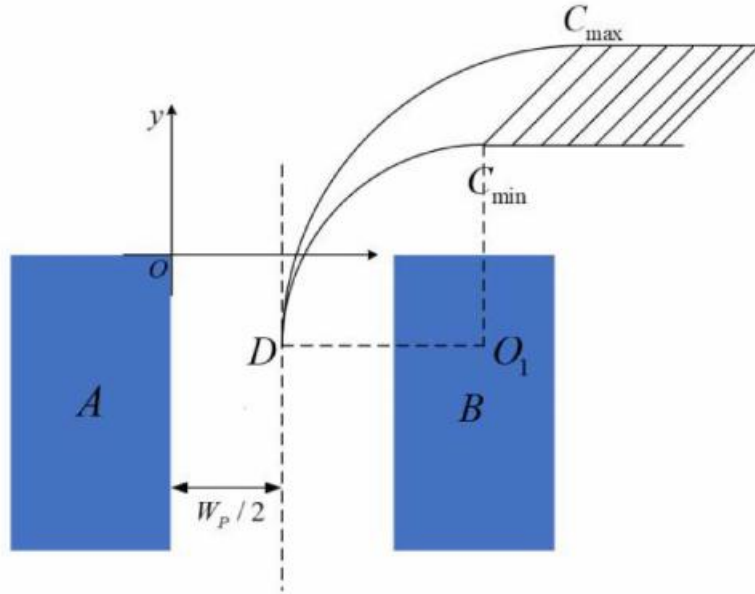


图 5-11 垂直泊车起始区域

### 5.2.3 垂直泊车路径规划仿真

我们依旧采用 MATLAB 软件对其进行仿真模拟，得到如下结果：

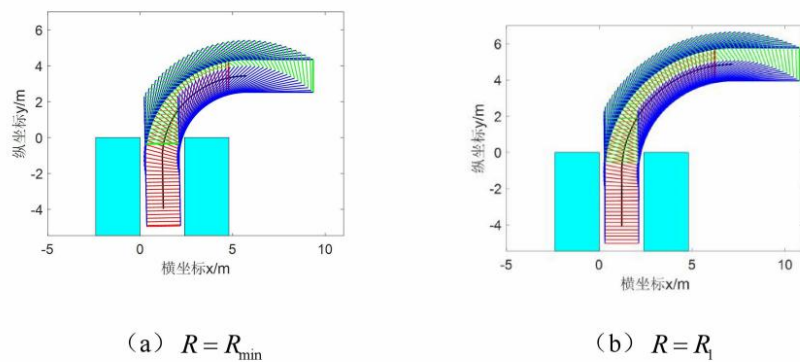


图 5-12 垂直泊车路径规划仿真图

我们接着根据路径仿真跟踪得到了无人车航向角，前轮转向角，位置误差等参数，具体如图 5-13 所示：

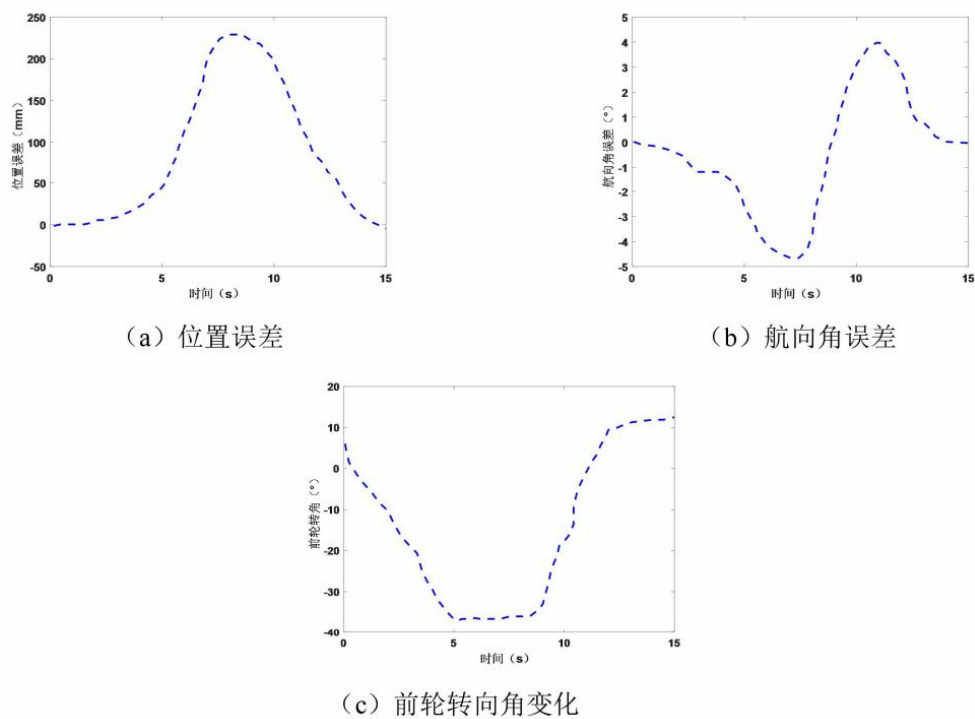


图 5-13 垂直泊车路径跟踪图

## 5.3 斜式泊车路径规划

### 5.3.1 斜式泊车路线分析

根据题目中所要求，我们选取停车角度为  $45^\circ$  建立我们的模型，无人车先经过圆弧轨迹，在进行直线运动进入车库具体路线如图 5-14 所示：

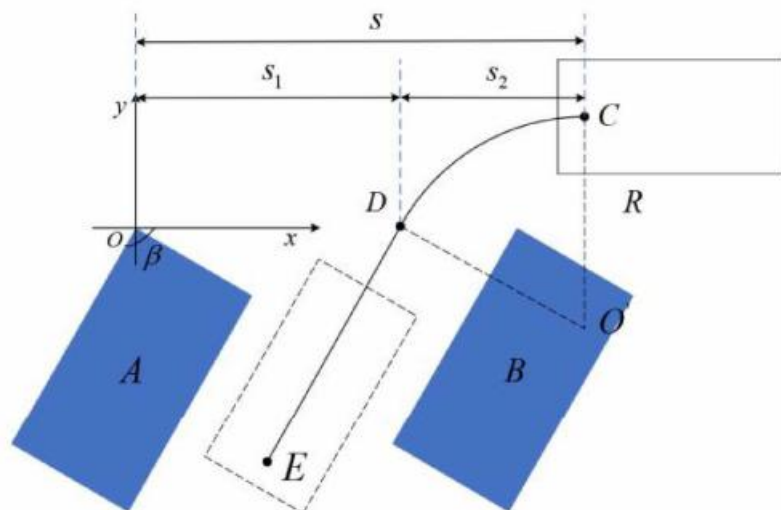


图 5-14 斜式（ $45^\circ$ ）车位路径规划

我们分析此图，无人车首先在起始点 C 以 O 点作为圆心，转弯半径为 R 开始倒车。当车辆航向角为  $45^\circ$  时开始沿直线后退到停车点 E，最终完成停车。

转弯半径为：

$$R = \frac{(s-s_1)}{\sin \alpha} \quad (5.20)$$

式中， $\alpha$  为无人车运动过程中的转弯角度， $\alpha = \pi - \beta$ 。

### 5.3.2 斜式车位起始泊车区域计算

通过对前面建立的模型我们可以先确定泊车过程的终点 E，之后再根据转弯半径再来计算起始区域。

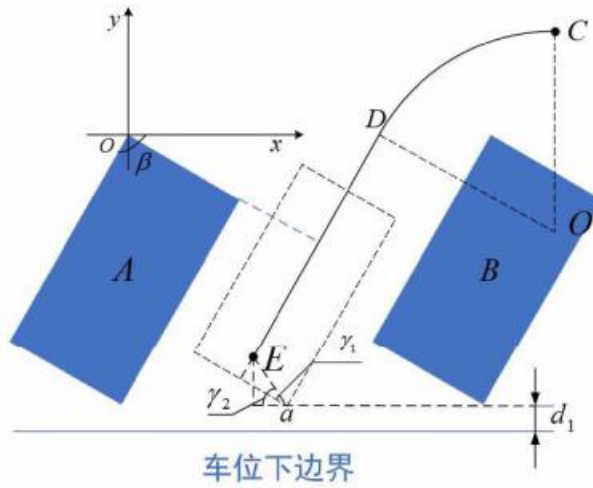


图 5-15 泊车终点位置计算

如图 5-15 所示，使无人车在停进车位后，其右后顶点 a 距离车位边界的距离为  $d_1$ ，根据几何关系，我们可以得到 E 点的纵坐标值：

$$y_E = L - d_1 - l_{Ea} \sin \gamma_2 \quad (5.21)$$

式中， $L$  为车位深度； $l_{Ea}$  为点 E 与 a 之间的直线距离，其为：

$$l_{Ea} = \sqrt{\left(\frac{w}{2}\right)^2 + l_r^2} \quad (5.22)$$

$\gamma_2$  为  $l_{Ea}$  与水平线的夹角：

$$\gamma_2 = \arctan\left(\frac{l_r}{w/2}\right) + \beta - \frac{\pi}{2} \quad (5.23)$$

通过 E 点的横坐标我们可以计算 DE 的表达式，后面代入 E 点的纵坐标的到横坐标。我们已知直线斜率为  $\tan(\pi - \beta)$ ，再求一点坐标  $D'$  即可知道 DE 的表达式。具体如图 5-16 所示：

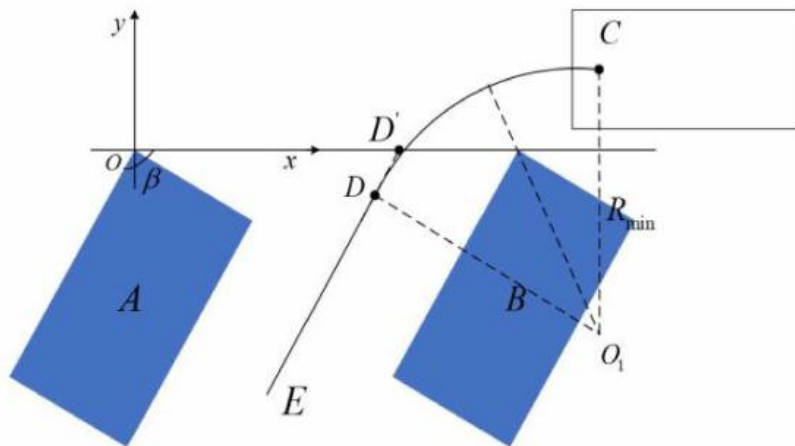


图 5-16  $D'$  点坐标求取

我们根据几何关系可得：

$$\begin{cases} x_{D'} = \left(w + \frac{w'}{2}\right) / \cos\left(\beta - \frac{\pi}{2}\right) \\ y_{D'} = 0 \end{cases} \quad (5.24)$$

式中， $w'$ 为车位在 $\pi - \beta$ 方向上的长度。之后我们可以得到 DE 表达式：

$$y = [\tan(\pi - \beta)x - \tan(\pi - \beta) \times \left(w + \frac{w'}{2}\right)] / \cos\left(\beta - \frac{\pi}{2}\right) \quad (5.25)$$

则 E 点的横坐标为：

$$x_E = y_E / \tan(\pi - \beta) + \left(w + \frac{w'}{2}\right) / \cos\left(\beta - \frac{\pi}{2}\right) \quad (5.26)$$

我们确定 E 点的坐标后，继而确定 D 点的坐标，我们还是用最小半径来计算 D 点的最小值，设 D 点的坐标为 $(x_D, y_D)$ 。根据几何关系我们可以得到 D 点和 C 点之间的关系：

$$\begin{cases} x_C = x_D + R_{min} \sin(\pi - \beta) \\ y_C = y_D + R_{min} [1 - \cos(\pi - \beta)] \end{cases} \quad (5.27)$$

其中；

$$y_D = [\tan(\pi - \beta)x_D - \tan(\pi - \beta) \times \left(w + \frac{w'}{2}\right)] / \cos\left(\beta - \frac{\pi}{2}\right) \quad (5.28)$$

再通过分析无人车对右侧障碍发生碰撞的临界条件可得：

$$R_{min} - \frac{w}{2} = \sqrt{(X_c - X_p)^2 + (R_{min} - y_c)^2} \quad (5.29)$$

对上面两式联立可得到 D 点坐标最小值。

在确定 D、E 点坐标后，斜式泊车起始区域计算与平行泊车计算同理，在此就不在赘述，我们可以得到最终结果如图 5-17：

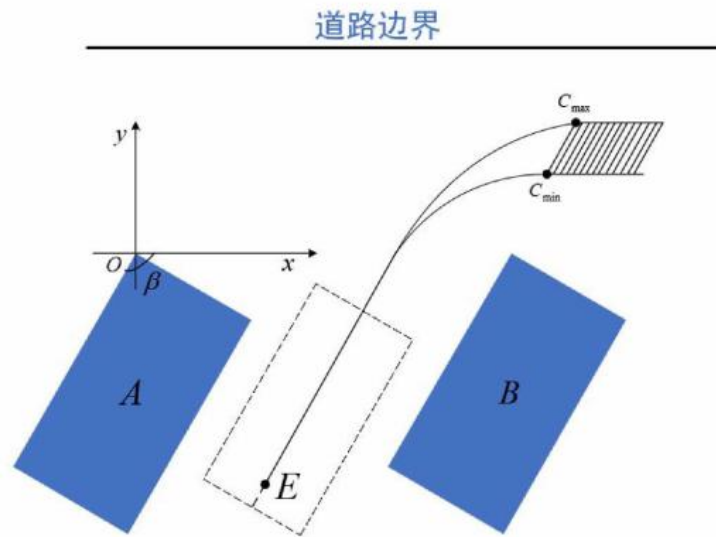


图 5-17 斜式泊车路径起始区域

### 5.3.3 斜式泊车路径规划仿真

我们依旧使用 MATLAB 软件仿真得到以下结果：

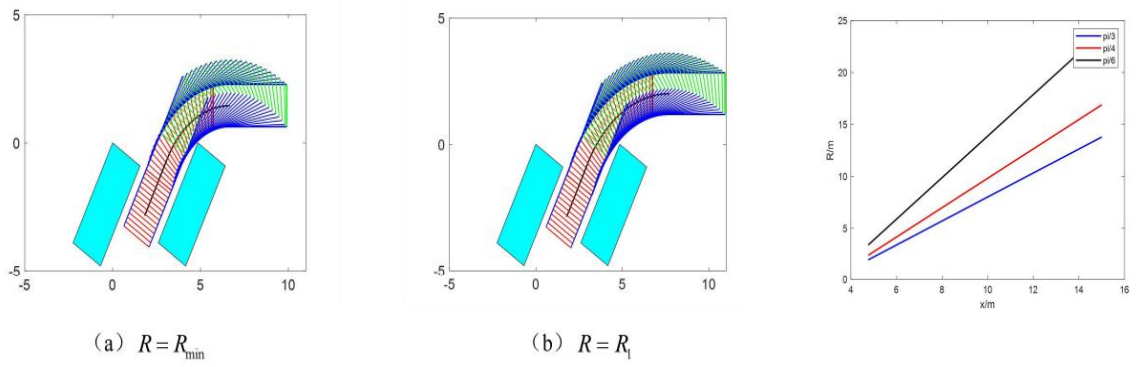


图 5-18 斜式泊车路径规划仿真

我们接着根据路径仿真跟踪得到了无人车航向角，前轮转向角，位置误差等参数得到图 5-19：

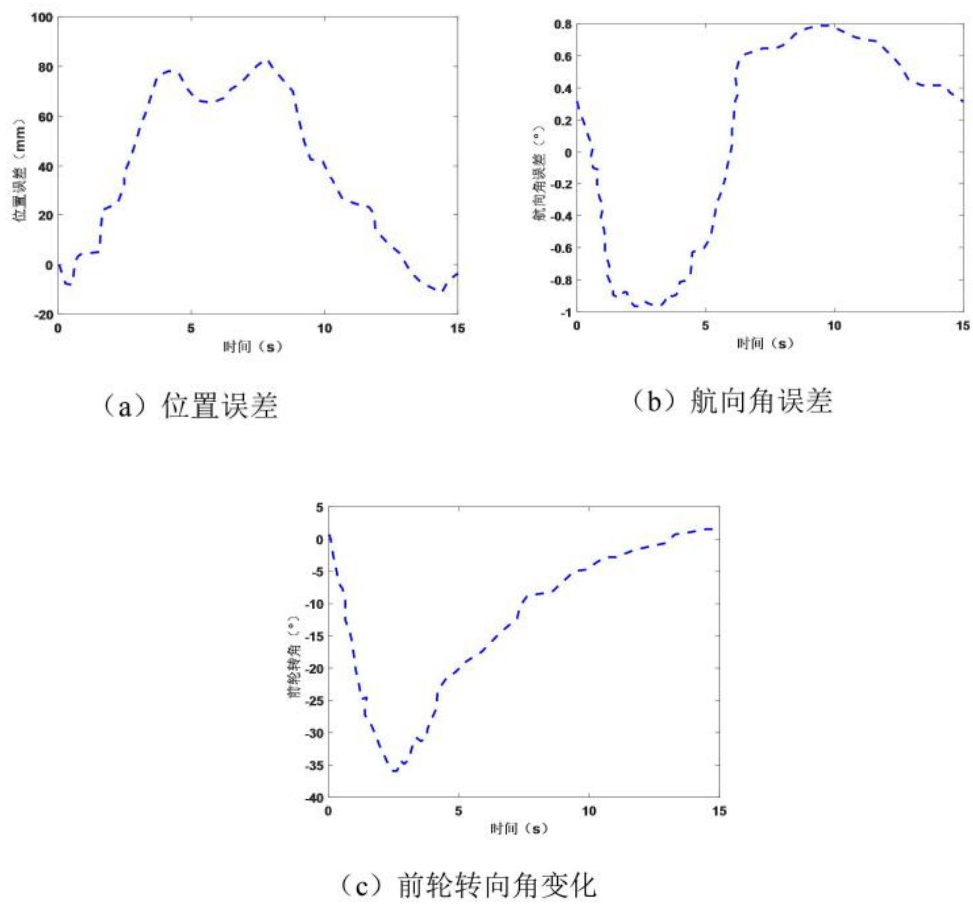


图 5-19 斜式泊车路径跟踪图

## 六，问题三模型的建立与求解

### 6.1 问题三具体分析与 A\* 算法

#### 6.1.1 问题三的详细分析

题目需要根据停车位现有的状况，建立相应的泊车模型，计算出较优停车位，并得出从当前位置到停车位的轨迹。得到泊车过程中所需时间，首先需要得到整个行驶过程中所需要经过的路段及路段对应的时间，同时需要汽车在倒车过程中的倒车时长，我们以汽车在倒车过程中的旋转角度来确定在不同的倒车情形下，所需要的倒车时长，并通过系统的将无人车在直线弯道倒车入库各环节的时间汇总，在已知空余车位的情况下寻求最优解，从而得到整体的停车规划路线。

通过查找相关论文，一篇来自于波士顿大学的物理学家 Paul Krapivsky 和来自圣塔菲研究所 Sidney Redner 的《Simple parking strategies》的研究论文中提出来了在停车场中，最好的停车点的位置的问题。在该研究中，研究者们首先将停车场初步简化为较为理想的单排停车场，如图 6-1 所示，汽车首先从右侧进入到停车场，在找到空余停车位之后，司机下车，需要步行到左侧的出口。如果司机在刚进入停车场时立即找最近的停车位停车，那么它下车以后需要走较长的一段路程才会到达出口，这可能耽误一段时间；反之，如果司机在进入停车场后继续开，寻找靠近该停车场出口的停车位，便会缩短步行距离。但通常，离出口进的停车场比较抢手，司机可能会找不到车位而往返，浪费时间。

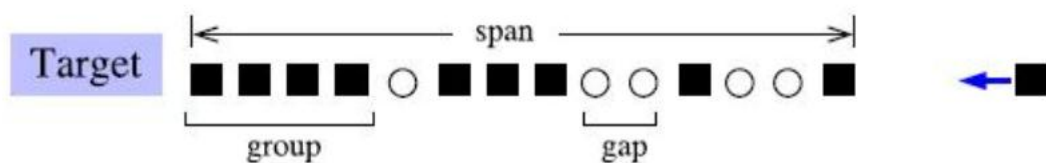


图 6-1 停车位置策略

研究者对温和、谨慎、乐观，三种停车位置策略分别进行了建模。

<1>温和策略：在进入停车场后，司机将看到的第一个空余停车位作为目标。汽车将停在第一个空余停车位。

谨慎策略：当司机看到第一个空余停车位后，如果该空余停车位左边还有空余停车位，就将车辆停在该空余停车位的左边；如果该空余停车位旁边没有空余停车位，就继续往前开到头，然后回到最靠近出口的空余停车位。

乐观策略：司机一直将车开到出口处，然后回到离出口最近的空余停车位。如果后退到入口处还没有发现空余停车位，就重复巡逻，直至找到最靠近出口的停车位。

在上述的这三种策略中，温和策略花费的时间最多，谨慎策略花费的时间是最少的。乐观策略花费的时间在二者之间，温和策略的司机往往会在离出口较远的地方停车，导致司机步行到出口的时间过长。



## 6.1.2 A\*算法

A\*算法，作为一种启发性的算法，通过设定评估函数，全面性地对网格上的各个节点进行评估。而每个节点就是无人车所到达的位置，对每个位置点都进行智能化评估，找到最好的位置，从而最终找到目标位置。A\*算法评估函数如下：

$$F(n)=G(n)+H(n)$$

其中， $F(n)$  是一个对总过程的节点的相关评估函数，表示从起始的节点到目标的节点的总的估计代价， $g(n)$  是表示在特定状态空间下，从起始节点到达下一个节点的实际代价； $H(n)$ ，是预测的从当前节点到达目标节点，大致需要多少代价的最佳路径的估计值，最短路径的关键在于选取代价函数。对于预估启发方法，通常选用曼哈顿预估方法，该方法在 A\*算法中较为常见使用，比较容易入手。

$$H(n)=D*ab(n.x-goal.x)+ab(n.y-goal.y))$$

其中  $D$  为曼哈顿距离， $ab$  为下所属的矩形面积。

曼哈顿距离：可以大概地描述为在一个特定的矩形区域内，某一点到达其他点距离，可以近似地等于  $y$  轴方向的垂直距离与  $x$  轴上垂直距离之和，但是曼哈顿距离不是处于固定值，随着节点的变化，曼哈顿值也会随之进行相对应的变化。

## 6.2 准备停车位置与目的地的距离

设  $x$  为一个车位被占用的概率，那么显然  $1 - x$  则为空闲率。停车位坐标如下：

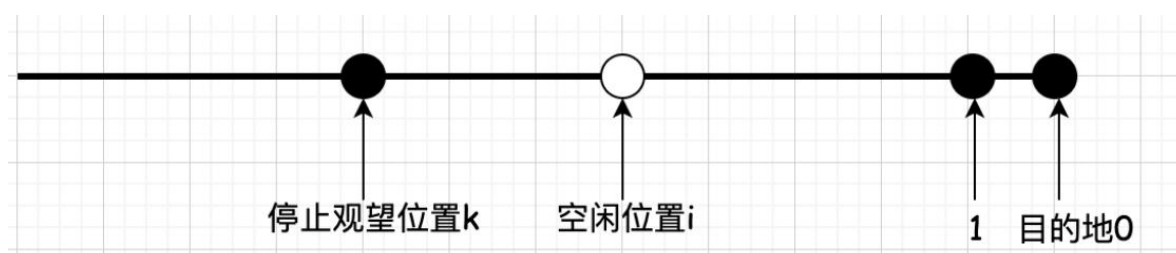


图 6-2 准备停车位置与目的地的距离

设  $k$  为司机开始考虑停车的位置，那么实际可以停车的位置  $i$  肯定满足在  $i < k$ ，还有两个约束：司机在  $i$  点停车，说明前面没有停车位；若  $i$  是最优的停车位，则意味着在  $i$  之后没有停车位。可以求出成功停车的概率：

$$P(k) = \sum_{i=1}^k a^{k-i} \times (1-a) \times a^i = \sum_{i=1}^k a^k \times (1-a) = (1-a)ka^k \quad (6.1)$$

对该公式求导可得

$$P'(k) = (1-a)a^k + (1-a)ka^k \ln a = a^k \times (1-a + k \ln a) \quad (6.2)$$

所以，在  $k = \frac{a-1}{\ln a}$  时，效果最好。这说明，当已知车位占用率为  $a$  时，离目的地  $k = \frac{a-1}{\ln a}$  开始，找到一个车位就停进去，是最优的选择。

表 6-1 占用率  $a$  与准备停车距离  $k$

占用 率%(a)	准备停车位置与目的地的距离(k)
0	0
50	1
75	3
80	4
85	5
90	7
95	14
96	17
97	23

### 6.3 模型的建立及仿真研究

表 6-2 车辆部分参数

名称	符号	数值
整车长度 (mm)	$L_a$	4900
整车宽度 (mm)	$W_a$	1800
整车轴距 (mm)	$L_b$	2800
后轮轮距 (mm)	$W_b$	1700
车辆前悬长度 (mm)	$L_f$	722
车辆后悬长度 (mm)	$L_r$	378
车辆最小小转弯半径 (mm)	$R_{\min}$	5708

在进行车辆的泊车路径规划研究之前，首先要对车辆进行建模，将问题转化为数学问题进行分析，以保证在进行仿真研究时，车辆的运动可满足实际环境约束。

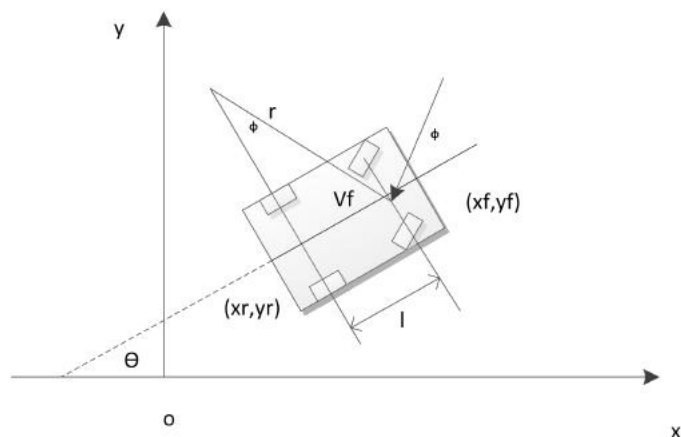


图 6-3 汽车阿克曼转向模型

我们在设计平行泊车路径时，车速  $v \leq 10 \text{ Km/h}$ ，且不考虑车辆的侧向移动，以及塑性变形等因素的影响。将车辆看成是刚体运动，车辆运动学模型如图 3.1 所示。车辆各类参数为：长 4.8m，宽 1.9m。根据阿克曼转向模型，无人车车辆以后轴延长线上一点为圆心作圆周运动。我们选定前轴中心点为基准点，坐标  $F(x, y)$ ，则后轴中心点坐标为：

$$R(x - l \cos \theta, y - l \sin \theta) \quad (6.3)$$

根据问题分析并结合问题一问题二的求解结果，令无人车在停车场停车过程中无论是转弯，直线行驶，还是倒车入库的过程中，其车速  $V_{\text{车}}$  均为该路段限速最高值。对于所有停车位进行统计，取  $n$  个可用于停车的位置，分别为  $x_i, y_i, a_i, b_i$ 。a, b 以逆时针为正方向。

表 5.2 符号说明

符号	说明
$x_i$	无人车在 x 方向上的坐标
$y_i$	无人车在 y 方向上的坐标
$a_i$	无人车在倒车入库前一刻的方向
$b_i$	无人车在倒车过程中需要旋转的角度

车辆由入口驶入，到达泊车起始区域，然后对这个过程进行泊车路径规划。题中所给停车场模型采用垂直泊车方式进行泊车，根据上文中机动车停车场设计参数表所示，停车场模型的行车道宽度为 5.5m。车辆寸为长 4.9m，宽 1.8m。

根据上述分析，首先对实际停车场建立二维数据模型，在此基础上对泊车路径问题进行研究。

以题中所给的图中车辆进入停车场的位置为原点，以小车入口直线为 x 轴，建立如图所示的直角坐标系，车辆的位置信息为  $(x, y, a)$  根据题目图中所给数据，停车位长 5.3m 宽 2.4m，道路宽 5.5m，可以得到表 3 中的空余停车位的位置坐标。设车辆进入停车场时位置为  $(0, 0, 0)$ 。

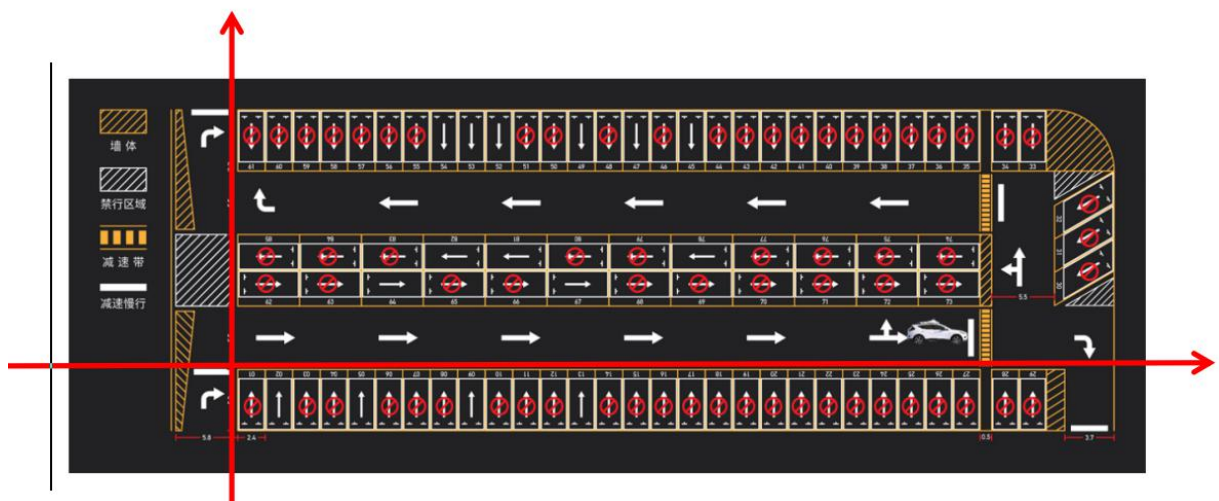


图 6-4 停车场位置坐标

表 6-3 空余车位号车头的位置坐标

空余车	空余车位号	X(m)	Y(m)	A(rad)
1	02	3.6	-5.3	0
2	05	10.8	-5.3	0
3	09	20.4	-5.3	0
4	13	30	-5.3	0
5	45	18	21.1	$\pi$
6	47	20.4	21.1	$\pi$
7	49	22.8	21.1	$\pi$
8	52	30	21.1	$\pi$
9	53	34.8	21.1	$\pi$
10	54	39.6	21.1	$\pi$
11	64	15.9	6.7	0
12	67	31.8	6.7	0
13	78	15.9	9.1	$\pi$
14	81	21.2	9.1	$\pi$
15	82	37.1	9.1	$\pi$

从表 6-3 中可以看到，停车场中共剩余车位共 15 个。

我们设置弯道个数辅助判断停车场的位置，对于弯道，令弯道个数为  $A_i$ ，

$$\Delta a_i = a - a_i \quad (6.4)$$

$$A_i = \begin{cases} 0 & \Delta a_i = 0 \\ 1 & \Delta a_i = \pi / 2 \\ 2 & \Delta a_i = \pi \end{cases} \quad (6.5)$$

当无人车在倒车入库前一刻的方向  $\Delta a_i = 0$  时，无车位于下方路段旁边的停车位人

上,此时无人车在倒车过程中需要旋转的角度为 $\frac{\pi}{2}$ ;当无人车在倒车入库前一刻的方向 $\Delta a_i = \frac{\pi}{2}$ 时,无人车位于右方路段旁边的倾斜停车位上,此时无人车在倒车过程中需要旋转的角度为 $\frac{3\pi}{4}$ ;当无人车在倒车入库前一刻的方向 $\Delta a_i = \pi$ 时,无人车位于上方路段旁边的停车位上,此时无人车在倒车过程中需要旋转的角度为 $\frac{\pi}{2}$ 。

弯道路径总和为 $S_{i1} = 2\pi r \times \Delta a_i$ , ( $r$ 为最小转弯半径,由问题一可知 $R_{min} = 5.833m$ )

$$\text{可得 } S_{i1} = \begin{cases} 0 & A=0 \\ 57.5694 & A=1 \\ 115.1388 & A=2 \end{cases} \quad \text{单位 (m)}$$

设该路段的最高速度为 $V_1$ 则该路段的最短时间为

$$t_{i1} = \frac{S_{i1}}{V_1}, \text{由此可算出 } t_{i1} \text{ 共有三个值, } \Delta a_i = 0 \text{ 时, } t_{i1} = 0s, \Delta a_i = \frac{\pi}{2} \text{ 时, } t_{i1} = 2.8785s, \Delta a_i = \pi$$

时,  $t_{i1} = 5.7569s$ 。

由题目中所给信息可知,整个停车场长 79.6m,宽 26.4m,车辆进入停车场时位置为(0,0,0)。

$t_{i1}$ 为弯道行驶过程中所需要的时间。对于直线路径,令 $\Delta x_i = x - x_i$ ,  $\Delta y_i = y - y_i$ 。

其中 $(x, y)$ 表示无人车当前位置,  $(x_i, y_i)$ 表示空余停车位的位置。 $\Delta x_i$ 表示无人车车辆到第*i*个停车位的水平距离, $\Delta y_i$ 表示无人车车辆到第*i*个停车位的垂直距离

$$S_{i2} = \begin{cases} \Delta x & A=0 \\ \Delta x_i + \Delta y_i & A=1 \\ 2c - \Delta x + \Delta y & A=2 \end{cases} \quad (6.6)$$

其中 $c$ 值为 $x$ 轴方向的最大值即 70.1m。通过上述 $S$ 分别对其所需时间进行求解,

设在减速带前 5m 位置内的速度为 $V_2$ , 在其他行车道的速度为 $V_3$ ,

$A=0$ 时,所需时间为 $t_{i2}$ ,

$$t_{i2} = \begin{cases} \frac{\Delta x + 2\pi r}{V_2} & \Delta y < 0 \\ \frac{\Delta x}{V_2} & \Delta y > 0 \end{cases} \quad (6.7)$$

$A=1$ 时,所需时间为

$$t_{i2}' = \begin{cases} \frac{b-x+\Delta y_i}{V_2} & b-x \leq 5 \\ \frac{(b-x-5)}{V_3} + \frac{b-x+\Delta y}{V_2} & b-x > 5 \end{cases} \quad (6.8)$$

$\Delta a_i = \pi$  时，所需时间为

$$t_i = \begin{cases} \frac{b+x-5}{V_3} + \frac{b-V+\Delta y_i}{V_2} + \frac{5}{V_2} + \frac{b-x_i-5}{V_3} & b-x_i \geq 5 \quad b-x > 5 \\ \frac{b+x-5}{V_3} + \frac{b-x+\Delta y_i}{V_2} + \frac{b-x_i}{V_2} & b-x_i \leq 5 \quad b-x > 5 \\ \frac{b+x+\Delta y_i}{V_2} + \frac{5}{V_2} + \frac{b-x_i-5}{V_3} & b-x_i \geq 5 \quad b-x \leq 5 \\ \frac{b+x+\Delta y_i}{V_2} + \frac{5}{V_2} + \frac{b-x_i}{V_2} & b-x_i \leq 5 \quad b-x \leq 5 \end{cases} \quad (6.9)$$

对于倒车路径，所需时间为

$$t_{i3} = \begin{cases} \frac{S_{i3}}{V_2} \\ \frac{S_{i3}}{V_3} \end{cases} \quad (6.10)$$

$$t = \sum_{i=1}^n (t_{i1} + t_{i2} + t_{i3})$$

综上令无人车总时间的目标函数为

。用 MATLAB 可求得无人车

停车过程中的时间。

## 七，问题四模型的建立与求解

### 7.1 自动泊车的轨迹规划

#### 7.1.1 模型的选取与假设

假设我们只考虑信息素浓度对蚂蚁选择路径的影响，并且做出符号说明如表 7-1:

表 7-1

符号	说明
m	蚂蚁数量

$n$	停车位数量
$d_{ij}$	无人车到停车位的距离
$\tau_{ij}(t)$	$t$ 时刻路径上信息素浓度
$\eta_{ij}^k$	启发函数，表示蚂蚁从该停车位转移到其他停车位的期望程度
$\tau_{ij}(t)$	$t$ 时刻路径上信息素浓度
$\eta_{ij}^k$	启发函数，表示蚂蚁从该停车位转移到其他停车位的期望程度
$\alpha$	信息素相对的重要程度
$\beta$	启发式因子的相对重要程度
$allowk$ ( $k=1,2,\dots,n$ )	蚂蚁 $k$ 待访问城市集合
$\rho$	信息素的挥发程度

---

### 7.1.2 模型的建立与求解

#### 模型的建立

##### 第一步：初始化

对一系列相关参数进行初始化，包含蚂蚁初始化规模、启发函数因子，信息素，信息素因子、挥发因子、信息素常数、最大迭代次数等，将数据读入程序，并且对数据进行相关处理，将停车的坐标位置，转为城市间的矩阵。设初始时各路径信息素浓度为相同，设为常数  $\tau_{ij}(0) = \tau_0$ 。

当前无人车车辆在入口处，

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}(t)}{\sum_{i=1}^n s \in allowed_k \tau_{is}(t)} & s, j \in allowed_k \\ 0 & s, i \notin allowed_k \end{cases} \quad (7.1)$$

其中  $allowdk$  表示：第  $(k)$  只蚂蚁暂未访问的停车位的集合，也就是接下来可选的停车位。  $P_{ij}^k$ ：  $t$  时刻第  $k$  只蚂蚁转移到停车位  $j$  的概率。

$P_{ij}^k$  越大，前往停车位  $j$  的概率越大。

##### 改进后的蚁群算法

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha, [\eta_{ij}(t)]^\beta}{\sum_{x \in allow_k} [\tau_{ij}(t)]^\alpha, [\eta_{ij}(t)]^\beta} & j \in allow_k \\ 0 & j \notin allow_k \end{cases} \quad (7.2)$$

蚂蚁在路径上，释放信息素的同时，已经走过的各个城市之间的连接的路径上的信息素挥发，设参数为  $\rho$  ( $0 < \rho < 1$ )，在蚂蚁完成一次循环之后，各个连接路径上的信息素浓度也需要进行实时更新，其信息素更新规则为：

$$\begin{cases} \tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}, 0 < \rho < 1 \\ \Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \end{cases} \quad (7.3)$$

$$\begin{cases} \frac{Q}{L_k}, \text{ 若蚂蚁 } k \text{ 从城市 } i \text{ 访问城市 } j \\ 0, \text{ 否则} \end{cases} \quad (7.4)$$

其中： $\Delta\tau_{ij}$ ：所有蚂蚁在城市 $i$ 与 $j$ 连接路上释放的信息素浓度总和。

$\rho$ ：轨迹的持久性， $1-\rho$ 为轨迹衰减度，表示消减程度。 $Q$ 为正常数， $L_k$ 为蚂蚁 $k$ 在此次运动中所走路径的长度。

第二步：随机地将蚂蚁放于固定出发点，对每个蚂蚁计算其下一个停车位，直至所更新信息素表有蚂蚁访问完所有停车位。

第三步：计算各个蚂蚁经过路径的花费，记录当前迭代次数中的最优解，同时对各个城市连接路径上的信息素浓度进行更新。

第四步：判断是否达到最大迭代次数，若没有达到，则返回第二步，否则终止程序。

第五步：输出程序运行结果，并且根据需要输出蚁群算法运行结果。

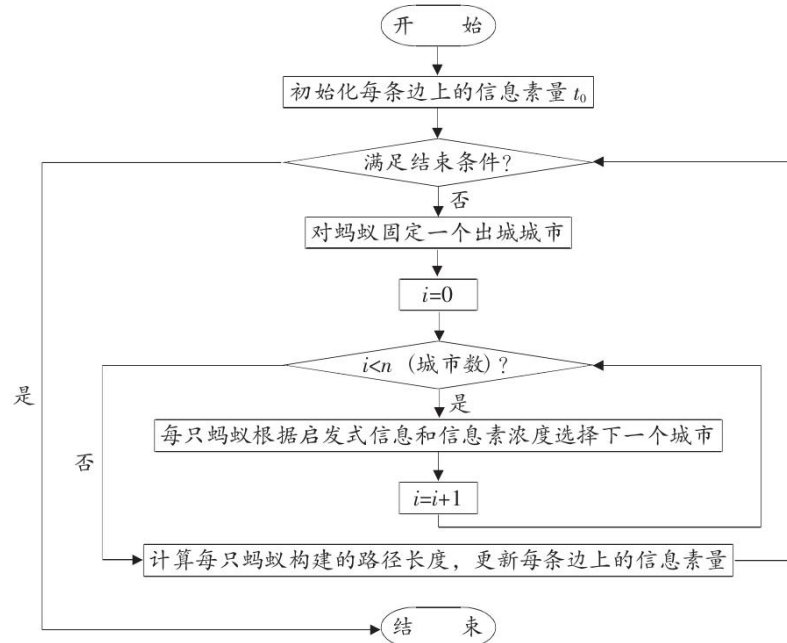


图 7.1 蚁群算法流程图

### 模型的求解

利用 Python 软件进行模拟仿真计算，设各参数分别为：蚁群数量  $m$  为 100， $\alpha$ 、 $\beta$  无限接近 1。



## 八，模型的评价与改进

### 8.1 模型的优点

无人驾驶轨迹规划、路径导航中经常使用到的一种静态的规划算法，就是 A\* (A-Star) 算法，采用 A\* 方法用于移动机器人的路径规划时，机器人首先按照已知的环境地图规划出一条路径，然后沿着这条轨迹运动，当机器人传感器探测到环境信息和原有的环境信息不一致的时候，机器人重新规划从当前位置到目标点的路径。如此循环直至机器人到达目标点或者发现目标点不可达。是一种静态路网中求解最短路径最有效的直接搜索方法。

而 ARA 算法即是其中重要的算法之一，它具有灵活的时间成本，即使在它结束之前被中断，也可以返回路径查找或图形遍历问题的有效解决方案。方法是在逐步优化之前生成快速。既有极大概率求出最优解，又可以减少冗余的时间。在现实世界的规划问题中，问题的解决时间往往是有限的。与时间相关的规划者对这种情况都会比较熟悉：他们能够快速找到可行的解决方案，然后不断努力改进，直到时间用完为止。

启发式搜索 ARA 算法，它根据可用的搜索时间调整其性能边界。它首先使用松散边界快速找到次优解，然后在时间允许的情况下逐渐收紧边界。如果有足够的时间，它会找到可证明的最佳解决方案。在改进其约束的同时，ARA 重复使用以前的搜索工作，因此，比其他随时搜索方法更有效。

### 8.2 模型的缺点

A\* 算法估价值与实际值越接近，估价函数取得就越好，公式表示为： $f(n)=g(n)+h(n)$ ，其中  $f(n)$  是从初始点经由节点  $n$  到目标点的估价函数， $g(n)$  是在状态空间中从初始节点到  $n$  节点的实际代价， $h(n)$  是从  $n$  到目标节点最佳路径的估计代价。保证找到最短路径（最优解的）条件，关键在于估价函数  $f(n)$  的选取：估价值  $h(n) \leq n$  到目标节点的距离实际值，这种情况下，搜索的点数多，搜索范围大，效率低。但能得到最优解。并且如果  $h(n)=d(n)$ ，即距离估计  $h(n)$  等于最短距离，那么搜索将严格沿着最短路径进行，此时的搜索效率是最高的。如果 估价值  $>$  实际值，搜索的点数少，搜索范围小，效率高，但不能保证得到最优解。

因此 A\* 算法通过比较当前路径栅格的 8 个邻居的启发式函数值  $F$  来逐步确定下一个路径栅格，当存在多个最小值时 A\* 算法不能保证搜索的路径最优。

### 8.3 模型的改进

基于 A\* 算法，从蚁群算法优化求解连续空间，蚁群算法是人工智能领域的一个研究热点，蚁群算法是一种最新发展的模拟蚂蚁群体觅食行为的仿生优化算法。

自然界中蚂蚁在觅食过程中主要按照所处环境中的信息量来决定其前进的方向，而

人工蚂蚁是在平面的节点上运动的,因此可把觅食过程抽象成算法中解的构造过程,将信息素抽象为存在于图的边上的轨迹。在每一节点,人工蚂蚁感知连接该节点与相邻节点边上的信息素轨迹浓度,并根据该浓度大小决定走向下一节点的概率。用任意两个节点分别表示蚂蚁的巢穴(初始节点)和食物源(目标节点),人工蚂蚁从初始节点按照一定状态转移概率选择下一节点,依此类推,最终选择行走到目标节点,这样便得到了所求问题的一个可行解。

此外该算法还采用了正反馈并行自催化机制,具有较强的鲁棒性、优良的分布式计算机制、易于与其他方法结合等优点,在解决许多复杂优化问题方面已经展现出其优异的性能和巨大的发展潜力。

### 8.3.1 改进蚁群算法求解连续性空间优化问题

根据具体泊车问题,我们不难看出,通过问题研究的一些抽象的自动泊车过程,可建立一个蚁群算法的基本模型。其问题空间是用图来描述的,解的获取是构造性的,而且在解的构造过程中人工蚂蚁没有接受任何全局的指导信息,因而求解过程是自组织的。在定义了一些规则之后,人工蚂蚁就可求解那些可用图来描述的问题,借鉴蚁群算法的进化思想,提出一种求解连续性空间优化问题的蚁群算法,进一步的去优化求解连续性空间。

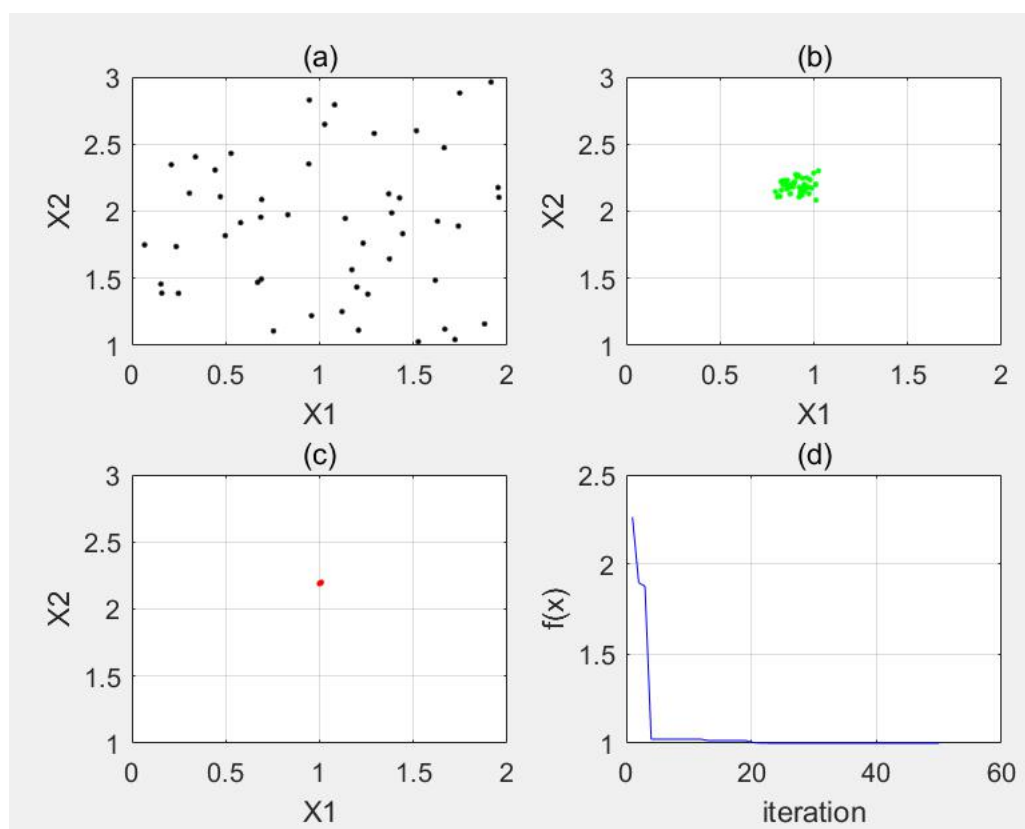


图 8-1 改进蚁群算法求解连续性空间优化问题

### 8.3.2 基于蚁群算法的二维路径规划

根据 MATLAB 绘图软件,初始化各项参数,设置蚂蚁规模和迭代次数,针对改进蚁群算法求解连续性空间优化问题进行绘图,如图 8-1 所示,根据初始位置进行全局转

移概率求取，迭代经过 1/3 时用绿色点表示蚂蚁的分布位置；迭代结束时用红色点表示蚂蚁的分布位置，当有很多蚂蚁觅食时，刚开始每个蚂蚁会随机选择一条路径，并在该路径中释放信息素，路径短的蚂蚁要比路径长的更先到达目的地，往返的频率也更快，所有该路径上的信息素更浓。信息素也会随着时间会有部分挥发。当下一代蚂蚁觅食时，下一代蚂蚁会选择信息素浓度高的路径走，而选择该路径的蚂蚁最多，又会释放更多的信息素，因此由大量蚂蚁组成的蚁群集体行为便表现出一种信息正反馈现象：某一路径上走过的蚂蚁越多，则后来者选择该路径的概率就越大。蚁群算法具有分布计算、信息正反馈和启发式搜索的特征，本质上是进化算法中的一种启发式全局优化算法。

通过各代最短距离与平均距离对比，距离越短，迭代次数越大；迭代次数相同时，平均距离明显大于最短距离。如图 8-2：

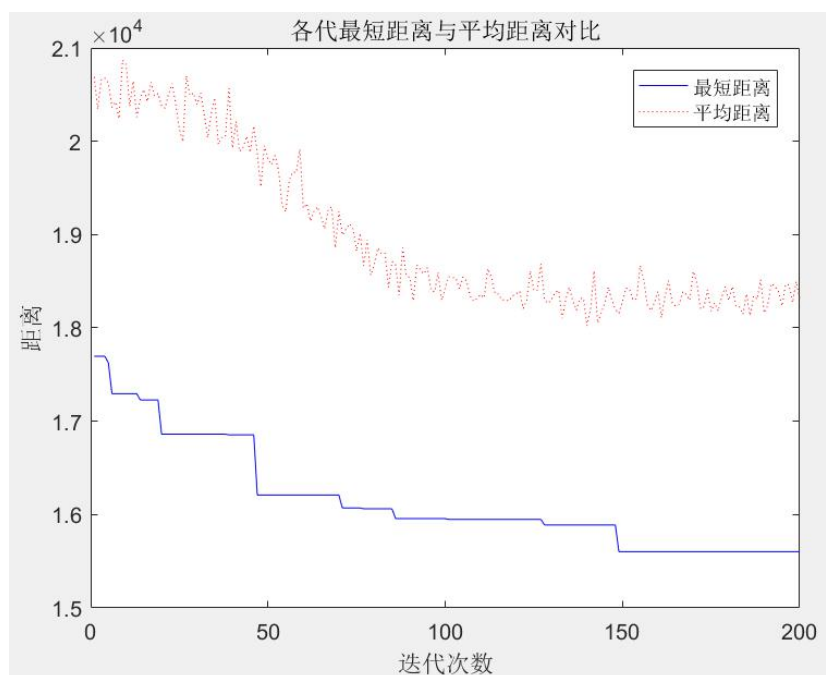


图 8-2 基于蚁群算法的二维路径规划

### 8.3.3 蚁群算法求解 TSP 问题

**TSP 问题：**旅行商问题，即 TSP 问题（Traveling Salesman Problem），给定一组城市 and 每对城市之间的距离，商家从其中一个城市出发，要求每个城市都要经过且每个城市经过一次，最后回到起点城市，求解商家访问这些城市的最短路径。在蚁群算法中，蚂蚁的行走路径表示待优化问题的可行解，整个蚂蚁群体的所有路径构成待优化问题的解空间。蚁群算法求解 TSP 问题的基本原理如下：

- (1) 根据自动泊车具体问题设置蚂蚁种群数量，分头并行搜索。
- (2) 每只蚂蚁完成一次轨迹后，在走过的轨迹上释放信息素，而信息素量与解的质量成正比。
- (3) 蚂蚁路径的选择（当前城市  $i$  到下一城市  $j$ ）根据信息素浓度  $T_{ij}$ ，同时考虑两点之间的距离  $\eta_{ij}$  ( $\eta_{ij}$  为城市  $i$  和城市  $j$  之间距离的倒数)，采用随机的局部搜索策略。这使得距离较短的边，其上的信息素量较大，后来的蚂蚁选择该边的概率也较大。
- (4) 每只蚂蚁只能走合法路线（经过每个城市 1 次且仅 1 次），为此设置禁忌表（即以前走过的城市列表）来控制。

(5) 所有蚂蚁都搜索完一次就是迭代一次，每迭代一次就对所有的边做一次信息素更新，原来的蚂蚁死掉，新的蚂蚁进行新一轮搜索。注意，同一代的蚂蚁互相之间不会受到之前蚂蚁留下的信息素浓度的影响，计算到下一城市  $j$  的概率时所用到的  $T$  是上一代蚂蚁留下的信息素。第一代蚂蚁用的信息素是初始值为 1，所有城市之间信息素浓度一样。

(6) 更新信息素包括原有信息素的蒸发和经过的路径上信息素的增加。

(7) 达到预定的迭代步数，或出现停滞现象（所有蚂蚁都选择同样的路径，解不再变化），则算法结束，以当前最优解作为问题的最优解。

有了基本原理，我们设置各种参数（迭代次数、蚁群数量、城市数量、信息素挥发速率等）根据城市坐标点计算城市之间距离，并且初始化城市间路线上的信息素，进一步求解出 TSP 问题，最后根据 MATLAB 绘图软件得到图 8-3

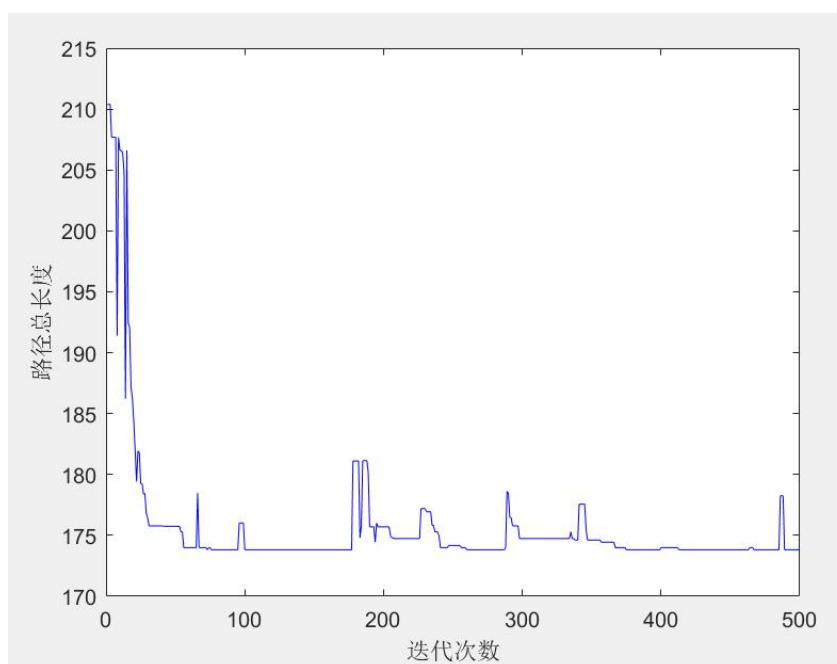


图 8-3 蚁群算法求解 TSP 问题

## 九，参考文献

- [1]张猛. 自动泊车路径规划算法研究[D].太原科技大学,2018.
- [2] 尤田田. 自动泊车路径规划及跟踪控制方法研究 [D]. 合肥工业大学,2021.DOI:10.27101/d.cnki.ghfgu.2021.001928.
- [3]胡圣能,许金良,杨宏志.曲率变化率在公路线形质量设计的应用研究[J].郑州大学学报(工学版),2010,31(03):101-105.
- [4] 王雪焱. 自动驾驶车辆垂直车位自动泊车控制算法研究 [D]. 吉林大学,2021.DOI:10.27162/d.cnki.gjlin.2021.002490.
- [5]谭巍. 城市中心区静态交通空间及停车设施的优化设计研究[D].湖南大学,2004.
- [6]梅振宇. 城市路内停车设施设置优化方法研究[D].东南大学,2006.
- [7]郭宏伟. 城市路内停车影响的建模与特性分析[D].北京交通大学,2011.
- [8]王芳成. 自动平行泊车系统的研究[D].中国科学技术大学,2010.
- [9] 任孝平,蔡自兴.基于阿克曼原理的车式移动机器人运动学建模 [J]. 智能系统学报,2009,4(06):534-537.
- [10]赵景川.汽车内外转向轮转角的关系[J].汽车研究与开发,1995(03):37-39.
- [11]李相彬.汽车列车转弯过程的分析及其最小转弯半径的确定[J].汽车技术,1981(11):18-21.