

报名序号： 1199

论文题目：露天停车场停车位的优化设计

# 露天停车场停车位的优化设计

## 摘 要

停车场受场地条件限制，仅能提供有限的停车位，如何综合考虑各方面因素设计停车场的停车位，使之能够获得较大的停车能力是一个值得研究的课题。

对于第一问，求解给定矩形停车场能够容纳的最多停车位数量。不失一般性，假设道路和停车带均与矩形长边平行，每一条停车带内车辆与道路的夹角为 $\theta_i$ ，根据不同的 $\theta_i$ 求出相应停车带与道路的宽度，列出规划方程组，使停车位最大化。结合实际，假设每条停车带与道路的夹角相同均为 $\theta$ ，求出停车场能容纳的最大停车带条数。对于剩余宽度，定义补偿停车带的概念，允许调节夹角 $\theta'$ ，再设置一条或两条停车道，从而实现对停车场面积的最大化利用，求出最佳倾斜角 $\theta$ 。改变停车场的长度和宽度进行灵敏度分析，得到结论：实际停车场建设过程中，为了方便施工，可将停车位的倾斜角设置为 $0^\circ$ 和 $90^\circ$ 。在停车场的扩建过程中，一次性将宽度增加 20m 最佳。

对于第二问，当停车场为不规则形状时，将停车场划分为三个矩形区域和一个三角形区域。考虑消防通道后对求解区域进行了修改。在三个矩形区域内利用第一问的结论求出最佳倾斜角和道路数、停车带数；将三角形区域划分成长度变化的矩形区域进行求解，对于顶端剩余三角形进行补偿求解。同时，考虑不同区域之间的通道连接，分析连接通道的设计位置对车位数量的影响。最后，将四个区域的停车位数求和，减掉连接通道引起的车位损失，即得总车位数量。改变消防通道宽度进行灵敏度分析，发现当消防通道宽度设置为下降临界值时，停车场的空间利用率最佳。

对于第三问，当停车场的出入口位置未定时，考虑出入口位置对停车位数量的影响主要在通道走向以及不同区域之间通道连接两方面。对此，提出了几种优化的出入口设计方案，使通道分布及连接对停车位的影响尽可能小。分别对几种方案进行区域划分，最佳倾斜角求解，连接通道影响分析，得出总停车位数量。对比几种方案的总停车位数量，得到最优解方案。

**关键词：**补偿停车带；消防通道；区域划分；通道连接；出入口设置；规划

# 1. 问题重述

## 1.1 背景

停车场的车辆容纳能力主要和停车场的形状、长宽度、出入口的位置以及停车位的设计倾斜角度有关。在规划设计停车场时，应在保证驾驶者出入方便的前提下，尽可能地提供足够多的停车位，以提高整个停车场面积利用率。

## 1.2 问题提出

**问题一**给出特定矩形停车场的最大车位数设计方案。

**问题二**给出特定不规则形状停车场最大车位数设计方案。

**问题三**给出不限定出入口不规则形状停车场最大车位数设计方案。

## 1.3 问题分析

**问题一**由于题目中已给出停车场的形状、长宽度以及出入口的位置。因此，在第一问的分析中，主要考虑停车位设计倾角对停车位数量的影响。由题目 1 当中给出的停车场长宽度数据，可以得到不同倾斜角下所能容纳的最大通道数和最大停车带数，其中，通道数和停车带数应满足一定的约束关系。通过停车带数和每一个停车带所能容纳的停车位数量，能得到总的停车位。遍历所有的倾斜角，得到最大的停车位数和最佳倾斜角设计。

**问题二**停车场具有不规则的形状。考虑消防通道的影响，对停车场面积进行修改。把停车场分割成几个独立的小区域，对于规则的矩形区域中，利用问题 1 的结论，得到最佳的倾斜角设计。对于三角形区域，采用微分的思想，将三角形区域划分成许多个变长度的矩形区域。同时，考虑不同区域之间的连接通道和对停车位数量的影响。综合上述三点分析，可以得到不规则形状停车场的最佳停车位设计。

**问题三**不限定停车场出入口的位置，分析出入口位置对通道走向，连接通道的影响。在通道对停车位的影响尽可能小的前提下，提出几种方案，分别求出总的停车位数量，比较得出最好的方案。

# 2. 模型假设

- 假设所有车辆的停车过程是相互独立的，不存在停车过程之间的相互影响。
- 假设设计停车场优化方案时，不考虑实际费用及通行效率的影响。

- 假设停车场为平面结构，不考虑地形变化对停车位数量的影响。
- 假设所有驾驶人员的驾驶技术均娴熟，不考虑驾驶技术对停车位空间的影响。
- 假设出入口为一维结构，不考虑出入口占据的空间。
- 为了实现最大停车位，假设中间过道都是单行单车道，且宽度是相应最小转弯半径。

### 3. 符号说明

$m_i$	第i个区域停车带数量
$n_i$	第i个区域车辆通道数
$\theta_i$	第i个区域停车位倾斜角
$R_i$	第i个通道宽度
$L_j$	第j个停车带宽度
$W_j$	第j个停车带单位停车位宽度
$C_1$	最大转弯半径
$C_2$	最小转弯半径
$L_d$	停车位末端距离
$A_i$	第i个区域的长度
$B_i$	第i个区域的宽度
$x_j$	第j个停车带停车位数量
$X_i$	第i个区域停车位总数
$X_{loss}$	因通道连接损失的停车位
$\theta_c$	停车位最佳倾斜角
$X$	露天停车场总的停车位数

### 4. 模型建立与求解

#### 4.1 问题一矩形停车场

#### 4.1.1 模型建立与分析

本文以家用小轿车为例，家用小轿车的转弯半径为5.5米，当垂直停放时需要长度为5.5米，宽度为2.5米的位置。对于优化模型，需要求解出停车带数和通道数与车辆停放倾斜角的函数关系，以求出最佳的倾斜角设计。现就问题一做出如下假设：

- 假设同一停车带上停车位的倾斜角度是一致的，以求得最大的空间利用率。
- 由于车辆出入过程对称，可假设车辆出入方向为从西向东。

对于每一个停车位，为了便于该车位上的小轿车自由进出，必须有一条边是靠近通道一侧的。如图 1 所示，停车位的倾斜角为 $\theta$ ，即该矩形停车位的长边与通道的夹角为 $\theta$ （ $0 \leq \theta \leq \frac{\pi}{2}$ ）。车辆的最小转弯半径定义是汽车转弯时转向中心到汽车外侧转向车轮轨迹间的最小距离<sup>[1]</sup>。小轿车驶入车位时，最外端在半径为最小转弯半径 $C_1$ 的圆周下移动，最内端在半径为最小转弯半径 $C_2$ 的圆周上移动，顺时针转弯 $\theta$ 角度驶入车位。保留一定的裕度，假设车辆行驶过程中所需的道路宽度等于垂直停放时所需停车位的宽度，即 $C_2 = C_1 - 2.5$ 。可得，通道最小宽度 $R = C_1 - C_2 \cos \theta$ ，将题目中所给的数据代入，得到最小宽度 $R = 5.5 - 3 \cos \theta$ 。

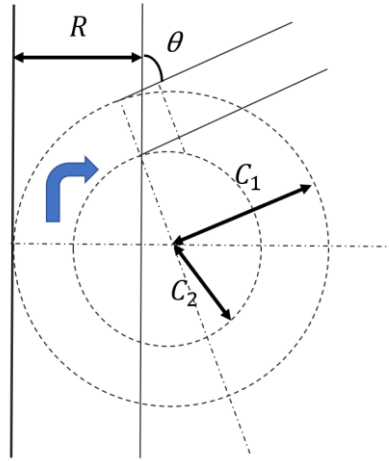


图 1 转弯半径图

为了使车辆能够自由出入，应考虑停车带宽度 $L$ 和单位停车位宽度 $W$ 与倾斜角 $\theta$ 之间的关系。当 $\theta$ 较大时（ $\theta \geq 24.44^\circ$ ），单位停车位宽度 $W$ 取决于停车位的宽度，如图2所示，根据图中的几何关系，可以得到(1)、(2)、(3)式：

$$W = \frac{2.5}{\sin \theta} \quad (1)$$

$$L = 5.5 \sin \theta + 2.5 \cos \theta \quad (2)$$

$$L_d = 5.5 \cos \theta - \frac{2.5}{\sin \theta} + 2.5 \sin \theta \quad (3)$$

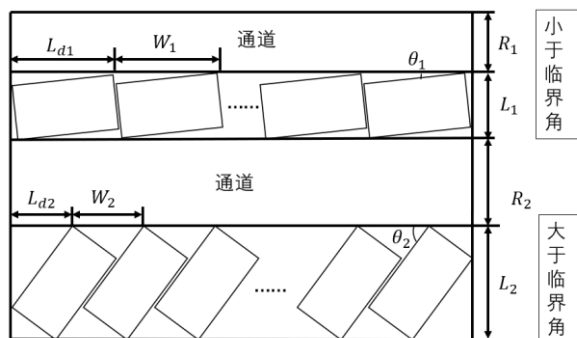


图 2 停车位尺寸图

当 $\theta$ 较小时 ( $0 \leq \theta \leq 24.44^\circ$ )，如图2所示，单位停车位宽度 $W$ 取决于停车位的长度，可修改得到停车位长度 $W$ 和 $L_d$ 的表达式为式(4)、(5)。

$$W = \frac{2.5}{\cos \theta} \quad (4)$$

$$L_d = 0 \quad (5)$$

平行式停放：

如图3所示，平行式停放相当于斜放式中 $\theta = 0^\circ$ 的情况，此时，通道的宽度 $R=2.5\text{m}$ ，停车带宽度 $L=2.5\text{m}$ ，单位停车位宽度 $W=5.5\text{m}$ 。

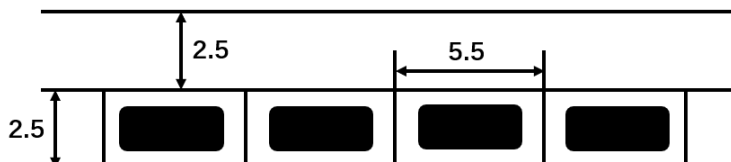


图 3 平行式停放

垂直式停放：

如图4所示，垂直式停放相当于斜放式中 $\theta = \frac{\pi}{2}$ ，此时，通道的宽度 $R=5.5\text{m}$ ，停车带宽度 $L=5.5\text{m}$ ，单位停车位宽度 $W=2.5\text{m}$ 。

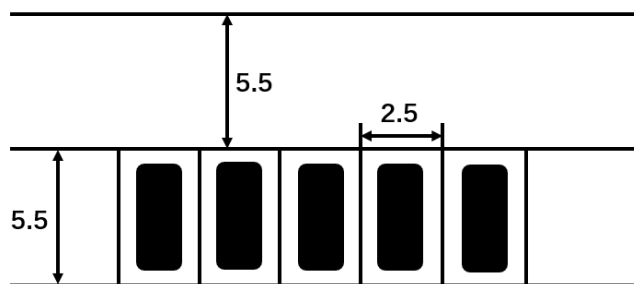


图 4 垂直式停放

为了得到尽可能多的停车位数量，应使通道的利用率尽可能的高。先考虑最简单的拓扑模型，有以下两种。

基本拓扑模型：

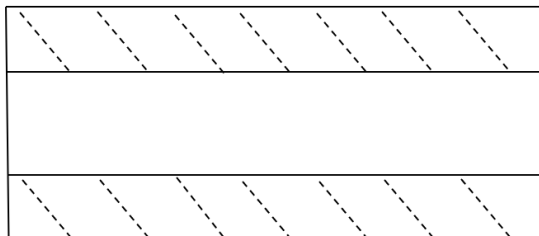


图 5 偶数车道数简图

假设  $n = 1, m = 2$ ，如图(5)所示，两条停车带共用一个通道，此时满足  $m = 2n$ ，停车位的宽度如(6)式

$$B = 2L + R = 11 \sin \theta + 2 \cos \theta + 5 \quad (6)$$

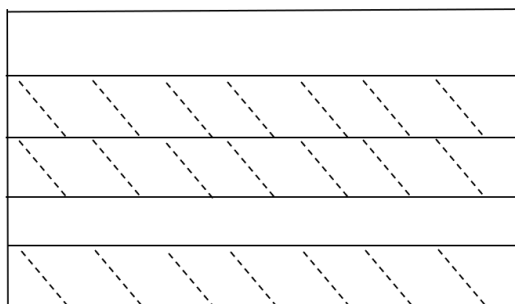


图 6 奇数车道数简图

假设  $n = 1, m = 2$ ，如图(6)所示，停车场下方均为两条停车带共用一个通道，最上方一条通道为一条停车带独享，此时满足  $m = 2n - 1$ ，停车位的宽度如(7)式

$$B = 16.5 \sin \theta + 1.5 \cos \theta + 11 \quad (7)$$

对于实际的停车场排布，可以分解成以上两种基本拓扑模型的叠加。

**实际优化模型：**

根据题目中所给的停车场长宽度数据，有  $A=79m, B=26.5m$ ，总共可容纳  $m$  个通道， $n$  个停车带， $x_i$  为第  $i$  个停车带所容纳的停车位个数，停车位的倾角为  $\theta$ ，可得到优化模型为：

$$\left\{ \begin{array}{l} \text{Max} \sum_{i=1}^n x_i \\ W_i = \begin{cases} \frac{2.5}{\sin \theta_i} & \theta_i \geq 24.44^\circ \\ \frac{2.5}{\cos \theta_i} & \theta_i \leq 24.44^\circ \end{cases} \\ L_j = \max\{(5.5 \sin \theta_{2j-1} + 2.5 \cos \theta_{2j-1}), (5.5 \sin \theta_{\max(n,2j)} + 2.5 \cos \theta_{\max(n,2j)})\} \\ L_{di} = \begin{cases} 5.5 \cos \theta_i - \frac{2.5}{\sin \theta_i} + 2.5 \sin \theta_i, & \theta_i \geq 24.44^\circ \\ 0 & \theta_i \leq 24.44^\circ \end{cases} \\ x_i = (A - L_{di}) / W_i \\ \sum_{i=1}^m R_i + \sum_{j=1}^n L_i = B \\ 0 \leq \theta_i \leq \frac{\pi}{2} \\ 2n - 1 \leq m \leq 2n \\ m, n \in Z^+ \end{array} \right. \quad (8)$$

由于该优化模型既涉及到 $\theta$ 的变化，同时又涉及到 $m$ 和 $n$ 的变化，结合实际情况，在停车场的规划设计中，可将前面几个停车带的停车位倾斜角设为相同值，最后一到两个停车带可设置不同的倾斜角来补偿剩余的宽度，以求得空间的最大利用率。因此，在给定的倾斜角 $\theta$ 的情况下，可先求出相同倾斜角时最大能够摆放的停车带个数 $m_{max}$ ，再根据剩余宽度决定补偿停车带的倾角。

求解过程分为两部分：

第一步：给定倾斜角 $\theta$ ，先求出最大能够设置的停车带个数以及相应的行车道个数。

第二步：根据剩余宽度决定是否继续设置更多车位以及补偿停车带的倾角 $\theta'$ ，当补偿前的 $m=2n$ 时，偶数条车道两边界都是只有一条车道，这时经分析补偿车道数可以1或2，补偿通行道数只能是1；当补偿前  $m=2n+1$  时，奇数条车道有一个边界是通行道，经分析补偿车道数只可能1，补偿通行道数也只能是1。得到以下公式：



$$\left\{ \begin{array}{l}
\text{Max } mx \\
\text{st. } R = 5.5 - 3\cos\theta \\
W = \begin{cases} \frac{2.5}{\sin\theta} & \theta \geq 24.44^\circ \\ \frac{2.5}{\cos\theta} & \theta \leq 24.44^\circ \end{cases} \\
L = 5.5 \sin\theta + 2.5 \cos\theta \\
L_d = \begin{cases} 5.5 \cos\theta - \frac{2.5}{\sin\theta} + 2.5 \sin\theta & \theta \geq 24.44^\circ \\ 0 & \theta \leq 24.44^\circ \end{cases} \\
x = (A - L_d)/W \\
mR + nL \leq B \\
0 \leq \theta \leq \frac{\pi}{2} \\
2n - 1 \leq m \leq 2n \\
m, n \in Z^+
\end{array} \right. \quad (9)$$

$$\left\{ \begin{array}{l}
\text{Max } m'x \\
\Delta B = B - mR - nL \\
\text{st. } R' = 5.5 - 3\cos\theta' \\
W' = \begin{cases} \frac{2.5}{\sin\theta'} & \theta' \geq 24.44^\circ \\ \frac{2.5}{\cos\theta'} & \theta' \leq 24.44^\circ \end{cases} \\
L' = 5.5 \sin\theta' + 2.5 \cos\theta' \\
L'_d = \begin{cases} 5.5 \cos\theta' - \frac{2.5}{\sin\theta'} + 2.5 \sin\theta' & \theta' \geq 24.44^\circ \\ 0 & \theta' \leq 24.44^\circ \end{cases} \\
x = (A - L'_d) / W' \\
m'R' + n'L' = \Delta B \\
0 \leq \theta' \leq \frac{\pi}{2} \\
m', n' \in Z^+
\end{array} \right. \quad (10)$$

建模使用matlab编码得到结果如下：

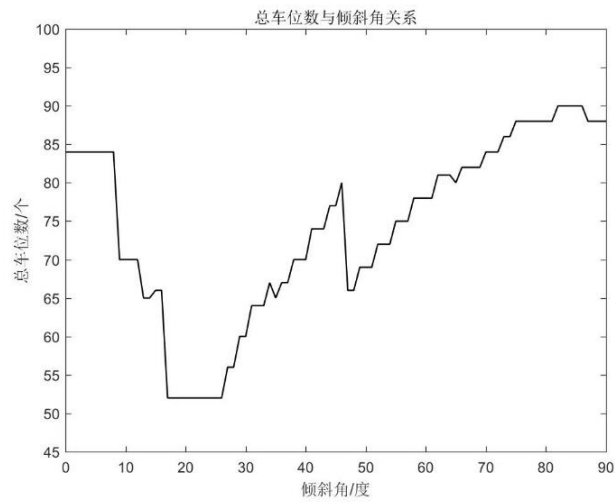


图 7 总车位数与倾斜角关系

倾角 $\theta$	81	82	83	84	85	86	87
总车道数 $m$	4	4	4	4	4	4	4
总通道数 $n$	2	2	2	2	2	2	2
停车位总数 $X$	88	90	90	90	90	90	88

表 1 第一问求解结果

#### 4.1.2 求解结果

倾角 $\theta/^\circ$	82-86
总车道数 $m$	4
总通道数 $n$	2
停车位总数 $X$	90

表 2 第一问优化方案

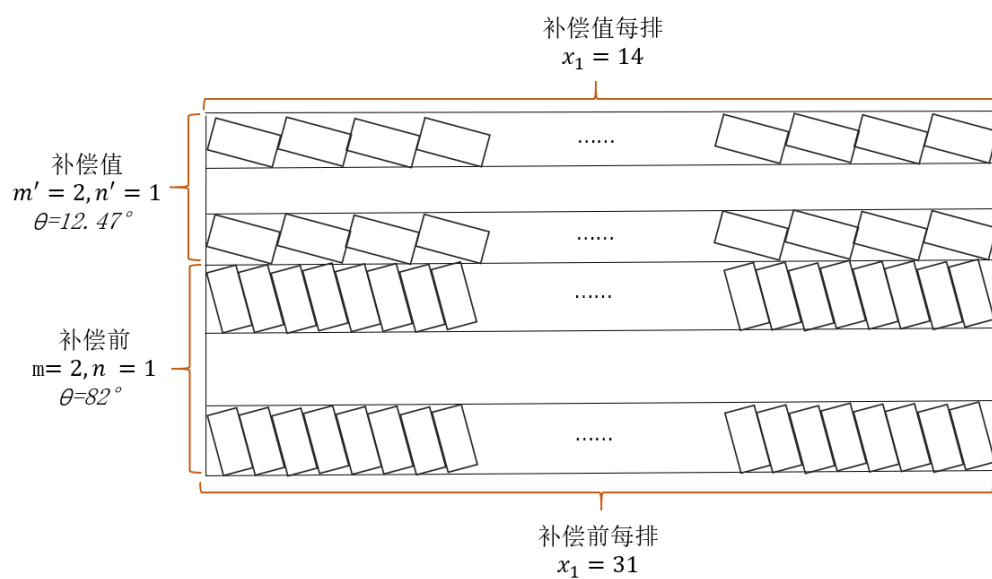


图 8 问题一平面设计图

当 $\theta$ 角取 $0^\circ \sim 90^\circ$ 之间的不同值时，在所给的矩形停车场上能够设置的最多停车位如图所示，可见在 $\theta = 82^\circ \sim 86^\circ$ 时停车位数量达到最大，共90个停车位。在 $46^\circ$ 时会有一个局部最优解，但显然不是我们要找的。

#### 4.1.3 灵敏度分析

首先把长度 $A=79\text{m}$ 设为不变，修改停车场的宽度，分析最优 $\theta$ 角和最大停车数的变化情况，如图(9)：

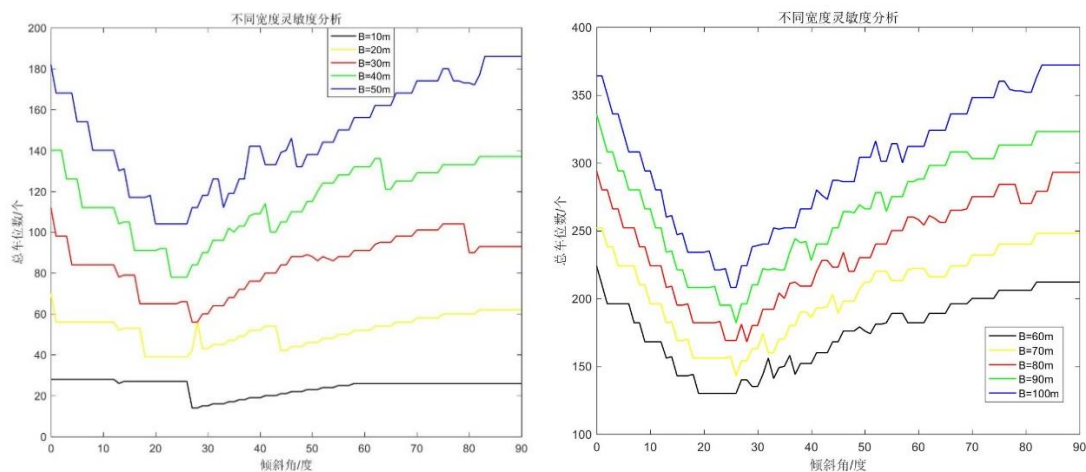


图 9 宽度灵敏度

宽度/m(长度79m)	10	20	30	40	50	60	70	80	90	100
最大角度/ $^\circ$	0	0	0	0	90	0	0	0	0	90
最大停车数	28	70	112	140	186	224	252	294	324	372

表 3 宽度灵敏度分析结果

调整矩形停车场的宽度，发现当宽度从10m增加到20m,20m到30m, 40m到50m, 70m到80m,90m到100m这个增加过程中，平均停车位增加量较多，即实际在考虑停车场扩建时，应考虑宽度一次扩建20m，获得的边际效益最大。

同时，大部分情况下，可以看出当 $\theta = 0^\circ$ 或 $90^\circ$ 时，能够设置的停车位最多，即使当宽为26.5m，长为79m时， $\theta = 82^\circ \sim 86^\circ$ 停车位最多，共90个，但 $\theta = 90^\circ$ 时亦可设置88个停车位，差别并不是特别大。可以得出结论，在相同的矩形面积下，要想设置尽可能多的停车位，应优先考虑 $\theta$ 角等于0度或者90度，即平行式停放或者垂直式停放。也可以方便建设。

宽度定为26.5米，调整长度做出以下数据：

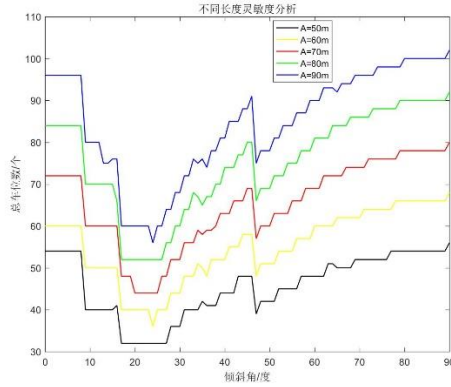


图 10 长度灵敏度

长度/m(宽度26.5m)	50	60	70	80	90	100
最大角度/°	90	90	90	90	90	90
最大停车数	56	68	80	92	102	114

表 4 长度灵敏度分析结果

调整矩形停车场的长度，如图(10)，发现长度每增加10m,增加的最大停车数均为12个，且最佳倾斜角 $\theta_c$ 均为90°。这个结论与实际情况相符，增加停车场长度相当于线性增加每一个停车带停车位的数量，并不会改变最佳倾斜角的设计。

## 4.2 问题二不规则停车场

### 4.2.1 模型建立与分析

如图(11)所示的大型商场周围的停车场的设计。问题2的关键在于如何对不规则的形状进行规划，形状的不规则造成了约束方程很难得到甚至无法得到，难以仿照矩形区域的建模方法，对此，做出以下假设：

- 为了节省通道空间和统一规划，假设通道之间均为直角连接。
- 由于通道宽度和单位停车带宽度数值大致相同，假设在不同倾斜角下，通道连接对停车位数量变化的影响相同。
- 假设不同区域之间的规划过程除了通道连接以外均无相互影响。

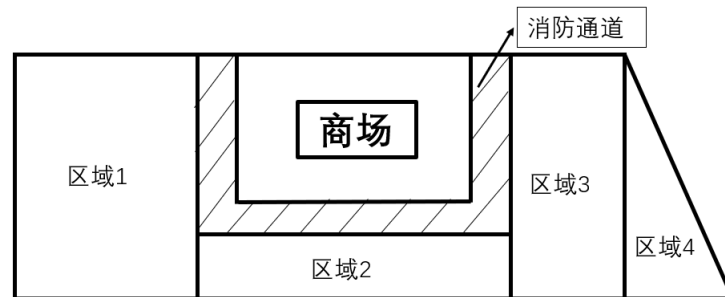


图 11 区域划分示意图

如图(11)所示，将此停车场划分为四块规则的形状，同时考虑消防通道的影响，由

三块矩形区域和一块三角形区域组成。分别求解每块规则形状的最优解，然后再将几个部分的通道连接起来（通道连接过程会伴随着停车位的减少），问题二的模型求解可以分解成三部分：矩形区域的规划、三角形区域的规划和区域之间通道连接的影响。

对于消防通道宽度，根据国家标准GB50067-2014《汽车库、修车库、停车场防火规范》中的规定：停车数量超过400辆的停车场属于一级防火等级，消防车道的净宽度和净空高度均不应小于4.0m<sup>[2][3]</sup>。所以合理的设计是围绕商场设置一圈宽度为4米的消防通道，等效为停车场的可规划面积减小。

### 矩形区域求解：

在将露天停车场划分成四块区域后，其中区域1，区域2，区域3均为矩形区域，利用问题一的结论可以对三块区域进行规划设计。

对于区域1，长度 $A_1=270\text{m}$ ,宽度 $B_1=39\text{m}$ ,利用图(12)的结果，得到当停车位的倾斜角 $\theta_1 = 90^\circ$  时，区域1停车位的数量最多。在这个倾斜角下，可以求得停车带的数量 $m = 10$ ,  $n = 5$ ，区域1的停车位数 $X = 490$ 。

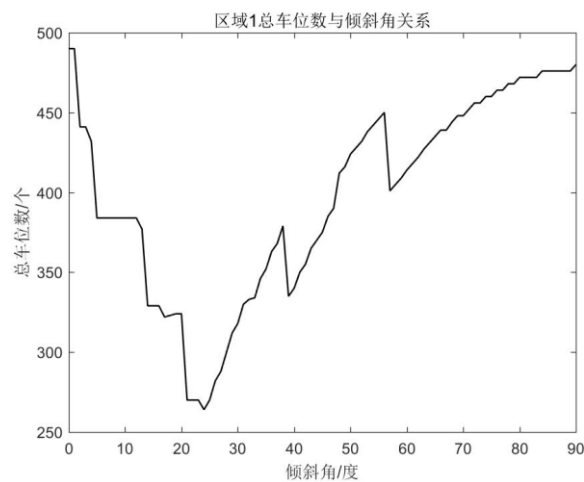


图 12 区域 1 求解结果

对于区域2，长度 $A_2=358\text{m}$ ,宽度 $B_2=16\text{m}$ ，由图(13)，得到当停车位的设计倾斜角 $\theta_2 = 0^\circ$  时，区域2的停车位的数量最多。在这个倾斜角下，可以求得停车带的数量 $m_2 = 4$ ,通道的数量 $n_2 = 2$ ,区域2的停车位数 $X_2 = 260$ 。

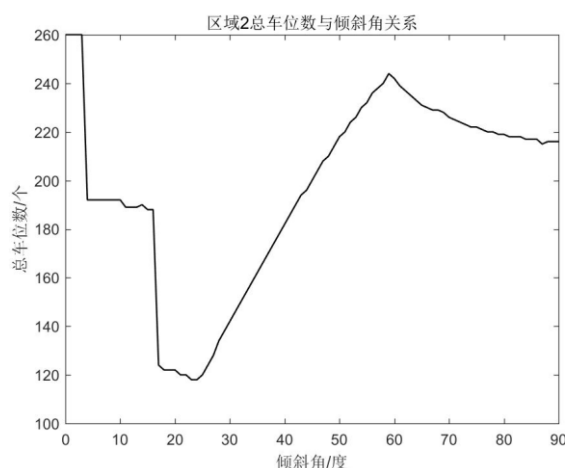


图 13 区域 2 求解结果

对于区域3, 长度 $A_3=270\text{m}$ , 宽度 $B_3=17\text{m}$ , 利用图(14)的结果, 得到当停车位的设计倾斜角 $\theta_1 = 90^\circ$  时, 区域3停车位的数量最多。在这个倾斜角下, 可以求得停车带的数量 $m_3 = 2$ , 通道的数量 $n_3 = 1$ , 区域3的停车位数 $X_3 = 216$ 。

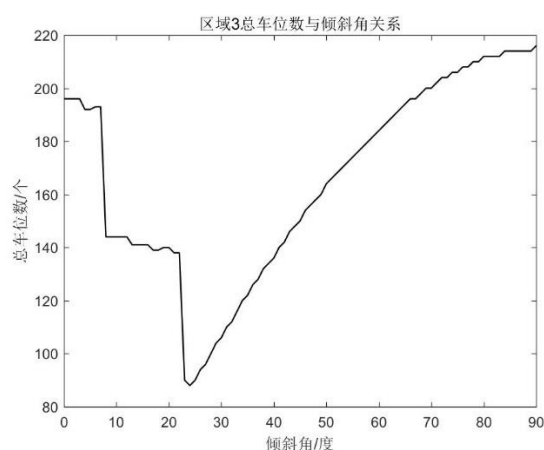


图 14 区域 3 求解结果

### 三角形区域求解:

根据题中给出的区域划分, 得到的三角形如图(15), 定义此直角三角形宽 270m, 长 36m。考虑到不同车道对应的车道长度不同, 因此每个车道车位数不一样。

建模求解时, 我们首先假设所有通道车位倾斜角相同, 利用矩形区域求解方法, 所不同的就是每排车位的长度不同, 设定坐标系如下:

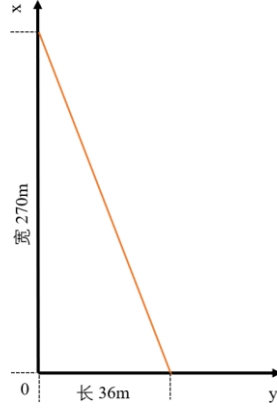


图 15 三角区域坐标系

得到以下不同宽度对应长度值方程：

$$y = -\frac{36}{270}x + 36 \quad (11)$$

以最大车位排数为目标，第一个约束是最上端车位约束条件设为此倾角下容不下一辆车，第二个约束所有车和车道总宽度不大于 270m。得到以下规划方程：

$$\left\{ \begin{array}{l} \text{Max } m \\ \text{st. } R = L_0 - 3\cos\theta \\ W = L_0 \sin\theta + W_0 \cos\theta \\ L' = W_0 \sin\theta + L_0 \cos\theta \\ -\frac{36}{270}(mL + nR) + 36 \leq W1 \\ mR + nL \leq 270 \\ 0 \leq \theta \leq \frac{\pi}{2} \\ 2n - 1 \leq m \leq 2n \\ m, n \in Z^+ \end{array} \right. \quad (11)$$

这里求得每个角度下最大车道数目后，每个车道长度与斜角边相接触的车位，接触点不同影响该车道总长度，因为三角形  $\alpha$  约为  $82.4^\circ$  近似垂直，因此可以近似认为车道中间点长度为车道总长，求得总车位数：

$$\left\{ \begin{array}{l} A_i = -\frac{36}{270}\left(m_i\left(L - \frac{L}{2}\right) + n_i R\right) + 36 \\ L_{di} = \begin{cases} 5.5 \cos\theta - \frac{2.5}{\sin\theta} + 2.5 \sin\theta & \theta_i \geq 24.44^\circ \\ 0 & \theta_i \leq 24.44^\circ \end{cases} \\ x_i = \left\lfloor \frac{A_i - L_d}{W} \right\rfloor \\ X_4 = \sum_{i=1}^n x_i \end{array} \right. \quad (12)$$

计算结果显示： $\theta = 90^\circ$ 时总车位数最大，因此对于最上方多出的未分配的小三角形

如图(16)，可以假定 $\theta_2 = 90^\circ$ 。对于剩下的小三角形经仿真得出最大宽度 50 米，此时 $m = 2n - 1$ ，其剩余宽度同时增加该角度的一个车道和一个通道已不足，但是对于垂直车位足够。

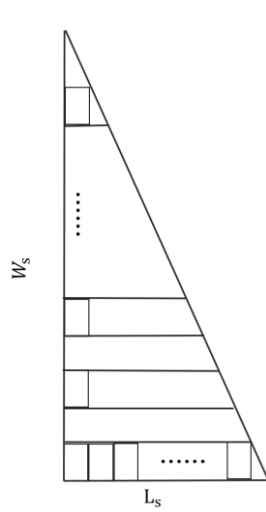


图 16 三角区域剩余待补偿三角形

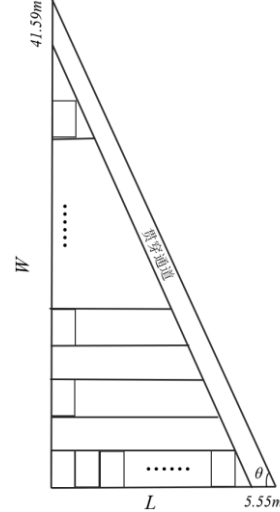


图 17 考虑通道三角区域

因此小三角形车位数设定以下条件，以最后剩余长度不足 $W_0$ 约束，得到以下方程就可以求解剩余三角形容纳最大车位数

$$\left\{ \begin{array}{l} \theta = \frac{\pi}{2} \\ W = R = L_0 \\ L_d = 0 \\ m'R + n'L \leq W_s \\ x_{i2} = \left\lfloor \frac{A_i - L_d}{W} \right\rfloor \\ X_2 = \sum_{i=1}^n x_{i2} \end{array} \right. \quad (13)$$

总车位数：

$$X = X_1 + X_2 \quad (14)$$

设计通道时，考虑到通道都为单行，而且补偿三角形车辆转弯半径为 5.5m，因此设计一条紧靠斜边的 5.5m 宽贯穿通道如图(16)以供车辆驶出，得到区域 4 参数为：长度 $A_4 = 36 - 5.5/\sin \theta_{tri} = 30.45\text{m}$ ，宽度 $B_4 = 270 - 5.5/\cos \theta_{tri} = 228.41\text{m}$ ，使用 matlab 工具结合线性规划和循环判断算法编程求解，得到  $\theta_1 = 90^\circ$  时，补偿停车位数量最多。在这个倾斜角下，可以求得停车带的数量 $m = 25$ ， $n = 13$ ，区域 4 的停车位数 $X = 155$ 。



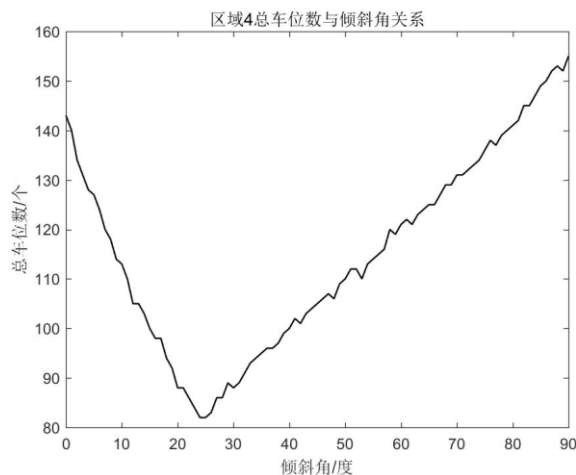


图 18 区域 4 求解结果

总的结果汇总如下：

项目 \ 区域	区域1	区域2	区域3	区域4	所有区域
总车道数m	10	4	2	25	41
总通道数n	5	2	1	13	21
最佳倾斜角 $\theta_c$	0	0	90	90	——
停车位总数 X	490	260	216	155	1121

表 5 问题二总区域求解结果

### 考虑不同区域之间通道连通：

根据每个矩形区域的规划设计，发现每个矩形区域的停车带数  $m$  和通道数  $n$  均满足  $n=2m$  的数学关系，即所有通道均为两条停车带共用。

如图(19)所示，考虑最简单的通道连通情况，即竖直方向和水平方向均为一条通道和两条停车带。考虑到行车通道为单向通道，为了使受影响的停车位数量尽可能小，应使竖直方向的连接通道安排在最右方，而使水平方向的连接通道安排在最下方。对比通道连接前和连接后的停车位变化，由于左下角的停车位无法利用，可得出在这种情况下，受影响的停车位数量为 1 个。

如图(20)所示，水平方向通道数和停车带数量不变，考虑在竖直方向增加一条通道和两条停车带，即竖直方向的  $m=2, n=4$  水平方向的  $m=1, n=2$ 。对比通道连接前和连接后的停车位变化，可得受影响的停车位数量为 2 个。

如图(21)所示，竖直方向通道数和停车带数量不变，考虑在水平方向增加一条通道和两条停车带，即竖直方向的  $m=1, n=2$ ，水平方向的  $m=2, n=4$ 。对比通道连接前和连接后的停车位变化，可得受影响的停车位数量为 2 个。

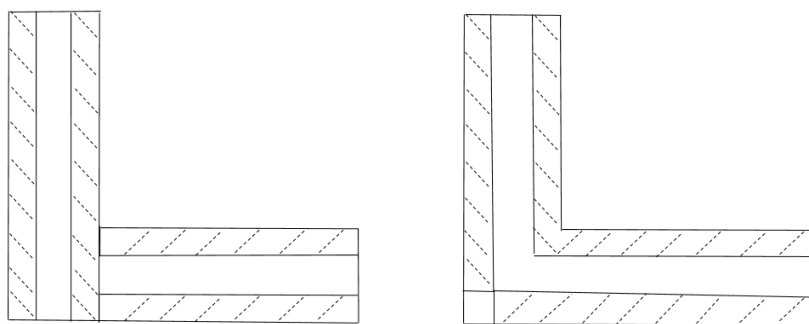


图 19 简单通道连接

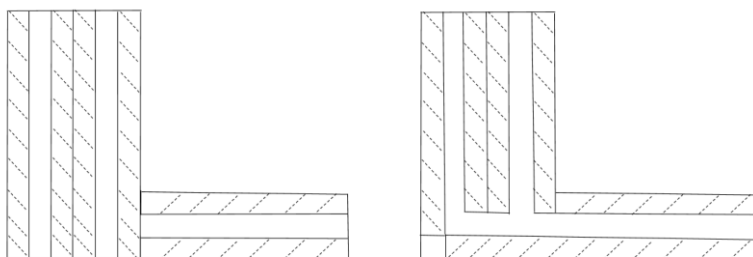


图 20 简单通道连接修改一

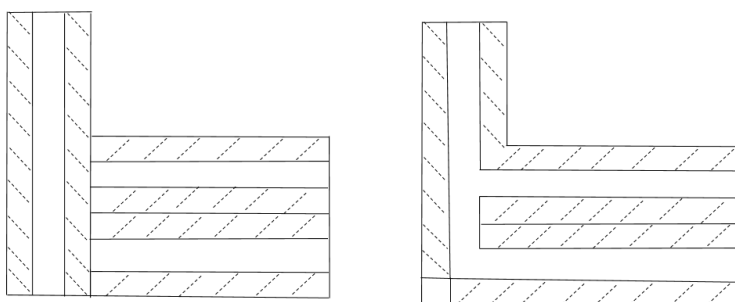


图 21 简单通道连接修改二

如图(22)所示，若竖直水平方向的通道数和停车带数同时增加，即竖直方向的  $m=2, n=4$ ，水平方向的  $m=2, n=4$ ，此时，受影响的停车位数量为 3 个。

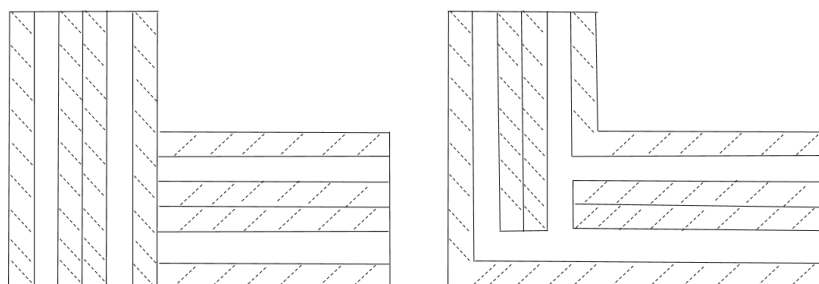


图 22 简单通道连接修改三

综合上述分析，可以得到受影响的停车位数量等于竖直方向和水平方向的通道数之和减 1。

将所求解的四个区域规划代入分析，可得，对于区域 1 和区域 2 的连接，区域 1 连

接通道应设置在最右方，区域 2 连接通道应设置在最下方，此时，因区域 1,2 连接而受影响的停车位为 6 个。

对于区域 2、3 之间的连接，同样由上述的结论可得，因区域 2、3 连接而受影响的停车位的数量为 2 个。

对于区域 3、4 之间的连接，如图(23)所示，由于此时存在区域 2、3 之间通道的连接，因区域 3、4 连接而受影响的停车位等于区域 4 的通道数减 1。

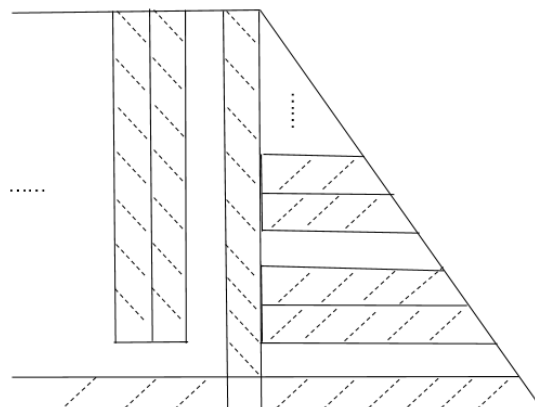


图 23 区域 3、4 连接

在考虑区域之间的连接后，受影响的停车位数量一共有  $13-1=12$  个，总的停车位数量为  $1121-6-2-12=1101$ 。

#### 4.2.2 求解结果

区域 项目	区域1	区域2	区域3	区域4	所有区域 (已减 $X_{loss}$ )
总车道数m	10	4	2	25	41
总通道数n	5	2	1	13	21
最佳倾斜角 $\theta_c$	0	0	90	90	——
停车位总数 X	490	260	216	155	1101

表 6 问题二求解结果

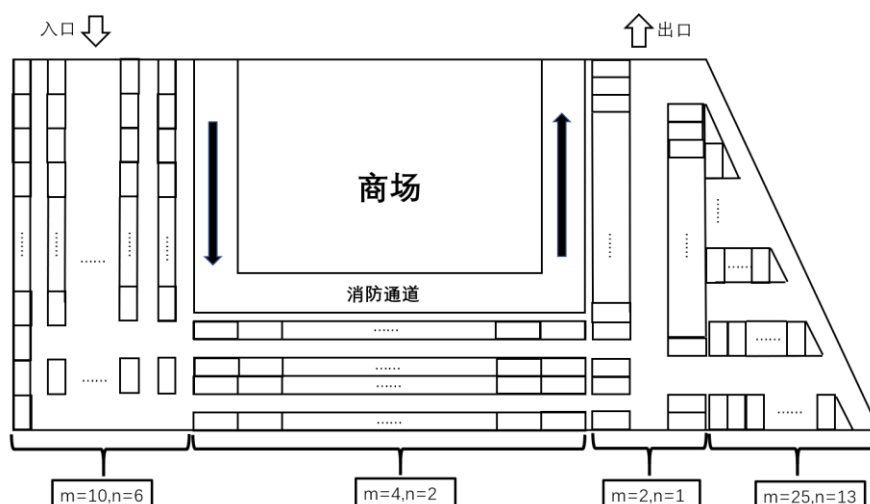


图 24 问题二平面设计图

### 4.2.3 灵敏度分析

问题二着重考虑消防通道宽度对停车场设计的影响，以下我们改变消防通道宽度，通过仿真得到不同消防通道宽度对应得最大车位数方案。

消防通道宽度 $B_x$ /m(宽度26.5m)	4	4.2	4.4	4.6	4.8	5	5.2	5.4	5.6	5.8	6
最优角度 $\theta_c /^\circ$	0	0	0	0	0	0	0	0	0	0	0
最大停车数	1106	1106	1106	1041	992	992	943	946	946	880	880

表 8 消防通道宽度灵敏度分析

从问题二的灵敏度分析我们可以得到，随着消防通道的增加，最大停车位数量基本上呈现逐渐减少的趋势，但停车位数量的减少是一个阶梯式下降的过程，即存在一个下降过程的最小步长。因此，在实际消防通道的设计过程中，可考虑将消防通道的宽度设置在下降的临界值上，以保证在取得最大停车位数量的前提下，尽可能增加消防通道的宽度，以求得较好的安全系数。

## 4.3 问题三出入口位置设计

### 4.3.1 模型建立与分析

较于问题二，问题三中停车场的形状和长宽度并没有改变，影响停车位数量的是出入口的设置位置。仿照问题二的方法，在增加消防通道之后，将停车场分区域进行优化，再考虑区域之间通道连接对停车位数量的影响。考虑到通行车道为单向单车道。

分析出入口位置对停车位的影响，得到出入口位置主要影响通道的走向以及区域之间的连通方式。在考虑出入口的设置时，应使通道所占的面积尽可能地小，同时，减小区域之间连通对停车位数量的影响。根据以上分析，可得到出入口的设置位置应该使出入口相对部分尽可能地多。因此，有以下 3 种出入口设置方案。

## 方案一

区域的划分和通道走向如图(25)所示，将停车场的南北边设计为出入口位置，由于车辆的驶入与驶出是可逆的，不妨将停车场的入口设置在北边的三条边上，而将出口设置在南边，其中，区域 2 的通道为东西走向。

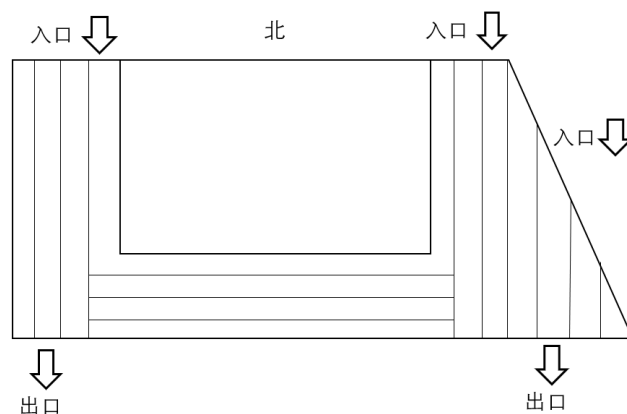


图 25 方案一布局图

利用问题一和二结论，可得四个区域各自的通道数，停车带数，最佳倾斜角，停车位总数如表(9)，此时，区域 1、2、3 的通道数  $n$ ，停车带数  $m$  均满足  $m=2n$  的关系，即区域两侧均有设置停车带。得到结果如下

项目 \ 区域	区域1	区域2	区域3	区域4	所有区域 (已减 $X_{loss}$ )
总车道数 $m$	10	4	2	9	25
总通道数 $n$	5	2	1	5	13
最佳倾斜角 $\theta_c$	0	0	90	0	—
停车位总数 $X$	490	260	216	207	1169

表 9 方案一求解结果

此时不存在区域 3、4 之间的连通。因通道连接而受影响的停车位有 4 个

综上所述，方案一总的停车位数：**1169**。

## 方案二

区域的划分和每个区域的通道走向如图(26)所示，将停车场的入口设置在北边的三条边上，而将出口设置在南边，与方案一不同的是，此时区域 2 的通道为南北走向。

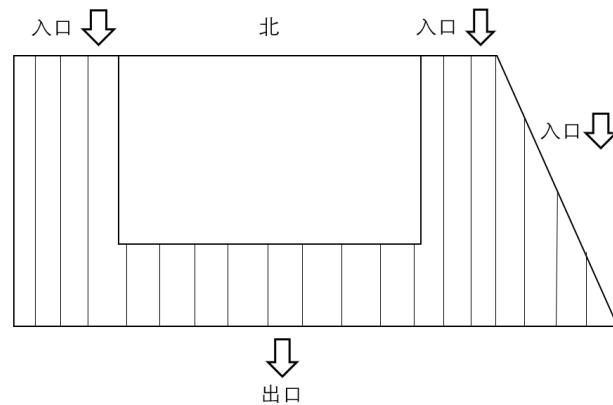


图 26 方案二布局图

同样利用问题一和二结论,可得四个区域各自的通道数,停车带数,最佳倾斜角,停车位总数如表(10)。此时区域 1、3 的通道数  $n$ , 停车带数  $m$  均满足  $m=2n$  的关系,即区域两侧均有设置停车带。

与方案一相同,此时不存在区域 3、4 之间的连通。但由于此时区域 2 的通道走向为南北向,应考虑在区域 2 的增设一条连接通道,以使区域 2 的通道同时具有驶入和驶出方向。容易得到因增设此连接通道而受影响的停车位数量等于区域 2 的停车带数加 2,不存在区域 3、4 之间连接通道的影响。得到结果为:

区域 项目	区域1	区域2	区域3	区域4	所有区域
总车道数 $m$	10	43	2	9	64
总通道数 $n$	5	22	1	5	33
最佳倾斜角 $\theta_c$	0	90	90	0	——
停车位总数 $X$	490	258	216	207	1126

表 10 方案二求解结果

考虑 $X_{loss} = m_2 + 2 = 45$ , 综上所述, 方案二总的停车位数: 1126。

### 方案三

在考虑消防通道之后,区域的划分和每个区域的通道走向如图(27)所示,将停车场的北边以及东边的一部分,西边的一部分设计为入口,将停车场东边的一部分设计为出口。

对于区域 1, 通道为南北走向, 长度为 254m, 宽度为 39m; 对于区域 2, 通道为东西走向, 根据三角形相似可求得通道的长度为 447.87m, 宽度为 16m; 对于区域 3, 通道为南北走向, 长度为 254m, 宽度为 17m; 对于区域 4, 通道为南北走向, 高度为 33.3m, 底边长度为 250m。

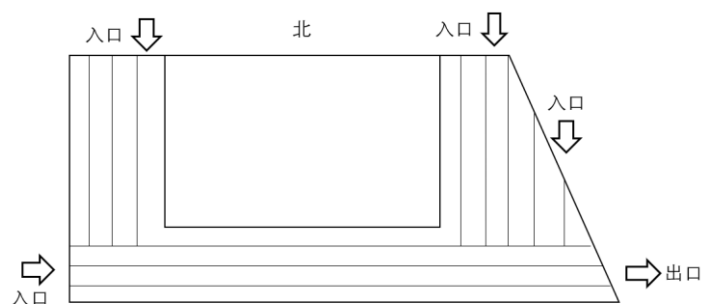


图 27 方案三布局图

对于区域 5，通道为东西走向，高度为 16m，底边长度为 2.13m。由于区域 5 的底边长度过小，无法容纳停车位。因此将区域 5 作为缓冲地区使用，只设置通道，不设置停车位。

求解得出区域 3 的通道数和停车带数满足  $n_3 = 2m_3$ ，即区域 3 通道两侧均设计停车带。在这种情况下，因区域间通道连接而受影响的通道数为区域 1、3、4 的通道数之和。带入求解结果如表(11)：

项目 \ 区域	区域1	区域2	区域3	区域4	所有区域
总车道数m	10	4	2	4	20
总通道数n	5	2	1	2	10
最佳倾斜角 $\theta_c$	0	0	90	90	——
停车位总数 X	460	308	202	200	1162

表 11 方案三求解结果

综上所述，方案三的总停车位数为 1162。

### 方案比较：

通过对上述三种方案总停车数的比较，可以看出，方案一的总停车位数量最多，选择方案一，区域设计如图(28)所示，此时，停车场总停车位数量为 1169。

### 求解结果

项目 \ 区域	区域1	区域2	区域3	区域4	所有区域 (已减 $X_{loss}$ )
总车道数m	10	4	2	9	25
总通道数n	5	2	1	5	13
最佳倾斜角 $\theta_c$	0	0	90	0	——
停车位总数 X	490	260	216	207	1169

表 12 问题三求解结果

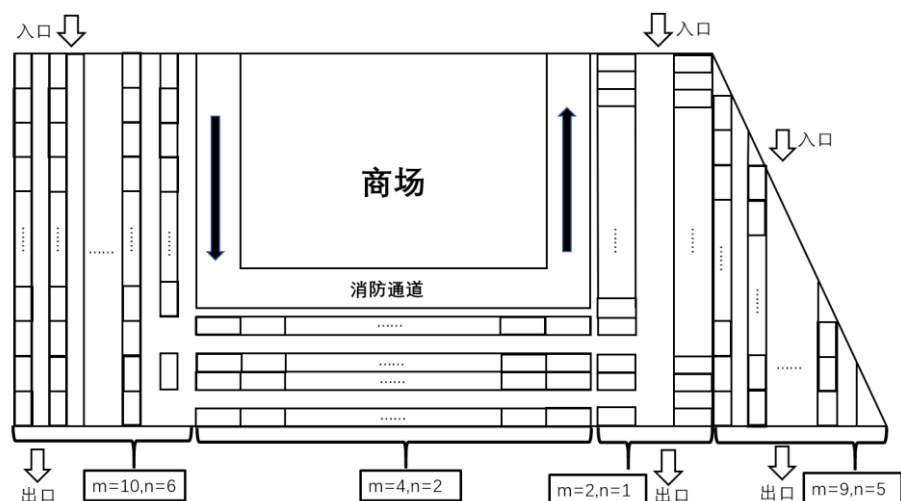


图 28 问题三平面设计图

## 5. 模型的分析与推广

### 5.1 模型分析

#### 优点：

针对问题一，为了利用停车场剩余的空间，提出了补偿停车带的概念，使停车场的空间利用率最大化。并且对停车场的长度和宽度进行了灵敏度分析，增强了模型的普适性，也可用于指导停车场扩建。

针对问题二，对区域进行划分，考虑三角形停车带的特殊性，对补偿停车带的算法进行了改进。同时，分析区域间通道连接对停车位数量的影响。在保证一定合理性的前提下尽可能提高停车位的数量。

针对问题三，为了减小通道对停车位数量的影响，提出了几种出入口设计的优化方案，并求出每一种方案各区域的设计方法以及总的停车位数量，使停车场的空间利用率进一步提高。

为了防止规划求解陷入局部最优值，遍历了所有角度值。

#### 缺点：

没有考虑通道宽度对车辆行驶效率的影响，因此，在分析过程中均使用最小通道宽度。而在生活实际中，应保留一定的通道宽度裕度，以提高车辆通行效率和保证行驶安全性。

为了获得最大停车位数量，忽略出入口数量对停车场管理效率和建设费用的影响。

### 5.2 模型改进



考虑其他诸如车辆行驶效率以及停车场建设费用等因素,可对模型进行改进,将原有的单目标优化规划方程变成多目标优化,采用层次分析法、帕累托最优解、权重分析法得出综合考虑所有因素后的最佳方案。

### 5.3 模型推广

在实际生活中,如何合理规划停车场与城市规划和交通道路规划有着密切的联系。该优化模型是针对大型商场的露天停车场设计的,属于建筑物配建停车场。大型商场的特点是人员出行活动频繁,因此对停车位的需求也较高。通过该优化模型的设计,可有效提高停车场的空间利用率,实用性较强。同时,该模型的设计思想,优化方案以及结果,对于其他类型建筑物配建停车场的设计也有一定的参考价值。考虑不同建筑物各自人员的活动特点,可将该模型推广到例如住宅配建,医院配建等等一类建筑物配建停车场。

随着停车设施的不断发展,新的技术不断出现,例如升降型停车场、立体螺旋形停车场,结合了人工智能、大数据分析等新兴技术,在提高停车效率的同时,更加智能化、人性化,为用户提供更加的停车体验<sup>[4][5]</sup>。

## 6. 总结

本文利用非线性单目标规划模型,分析了停车场在规则形状、不规则形状(出入口位置已定)和不规则形状(出入口位置未定)三种情况下最优的设计方案,以求得最大的停车位数量。并在这个模型的基础上,对停车场的长宽度,消防通道宽度进行灵敏度分析,验证这个模型更具有普适性,同时,方便对原停车场进行扩建。但本文只分析了停车位数量对优化方案的影响,在实际生活中,还应考虑停车场通行效率、停车场建设费用对设计方案进行综合优化。

### 参考文献

- [1]宋作忠,何文章.基于遗传算法的交易中心停车场优化设计[J].数学的实践与认识,2004(01):19-23.
- [2]GB 50067-2014. 汽车库、修车库、停车场防火规范[S]. 2014
- [3]胡冬梅. 地下停车场消防安全评价[D].西南交通大学,2008.
- [4]马晓,周长杰,谷锦彪.垂直停车位宽度和通道宽度对驾驶人停车行为的影响[J].智能城市,2018,4(16):10-13.
- [5]谭武龙,丘志敏.高效率停车场研究设计探讨[J].内燃机与配件,2018(21):165-167.

# 附录

## 附录 1 矩形区域区域求解程序

```
clc;clear all;close;

Width0=2.5;Length0=5.5;%车尺寸

Width01=3;

Width1=26.5;Length1=79;%停车场尺寸,与车道平行长度为

%x1 车位排数, x2 通道数

theta0=atan(Width0/Length0);%theta 最小值, 也就是 theta=0;

delta_theta=1;%角度步长

k=1;

ParkRow=[];%每个角度对应车位排数和通道数

ParkRow2=[];%补偿后总数

ParkRowadd=[];%添加的车道数和通道数

CarNumPerLine=[];%每个角度对应一排车数

CarNumTotal=[];%每个角度对应总车数

WidthPass=[];%通道宽度矩阵

WidthDelta=[];%通道宽度矩阵

WidthDelta2=[];%包括添加的车在内的最后剩余宽度

LengthDelta=[];

WidthPark=[];%每车所占垂直通道宽度矩阵

WidthCar=[];%每车所占平行通道长度矩阵

Ld=[];%补偿最后一个车位平行通道长度

WidthPark2=[];%多出来一辆车

WidthPass2=[];

theta20=[];

CarNumPerLine2=[];

for theta=0:delta_theta:90%floor 为向正无穷取整

    theta_rad=theta*pi/180;%角度化为弧度

    WidthPass(1,k)=Length0-Width01*cos(theta_rad);%转弯所需最小通道宽度

    WidthPark(1,k)=Width0*cos(theta_rad)+Length0*sin(theta_rad);%每车所占垂直通道宽度

    fl=[-1,0];

    intcon1=[1,2];%x1 是车位排数, x2 是通道数

    A1=[WidthPark(1,k),WidthPass(1,k);

        1,-2; -1,2];

    b1=[Width1;0;1];

    Aeq1=[];

    beq1=[];

    LB1=[0,0];

    UB1=[];

    [ParkRow(1:2,k),FVAL1] = intlinprog(fl,intcon1,A1,b1,Aeq1,beq1,LB1,UB1);

    %解出排数和通道数

    Parktemp=int8(ParkRow(1,k));
```

```

if mod(Parktemp,2)==0%判断车道排数奇偶性
    WidthDelta(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-ParkRow(1,k)/2*WidthPass(1,k);
    if WidthDelta(1,k)>=5.5
        if WidthDelta(1,k)>=7.5
            theta20(1,k)=asin((WidthDelta(1,k)-5.5)/11.18)*180/pi-10.305;
            ParkRowadd(1,k)=2;
            %                ParkRow(2,k)=ParkRow(1,k)/2+1;

        else
            theta20(1,k)=asin((WidthDelta(1,k)-5.5)/5.5227)*180/pi+5.19;
            ParkRowadd(1,k)=1;
            %                ParkRow(2,k)=ParkRow(2,k)+1;

        end
        WidthPass2(1,k)=Length0-Width01*cos(theta20(1,k)*pi/180);
        WidthPark2(1,k)=Width0*cos(theta20(1,k)*pi/180)+Length0*sin(theta20(1,k)*pi/180);
    else
        theta20(1,k)=-1;
        ParkRowadd(1,k)=0;
        WidthPark2(1,k)=0;
        WidthPass2(1,k)=0;
    end
else%奇数车道
    WidthDelta(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-(ParkRow(1,k)-1)/2*WidthPass(1,k);
    if WidthDelta(1,k)>=(Width0+WidthPass(1,k))
        thetatem=0;
        while 1
            thetaremp_rad=thetatem/180*pi;
            WidthPass2(1,k)=Length0-Width01*cos(thetaremp_rad);
            WidthPark2(1,k)=Width0*cos(thetaremp_rad)+Length0*sin(thetaremp_rad);
            if ((WidthDelta(1,k)-WidthPark2(1,k))< WidthPass(1,k))||((WidthDelta(1,k)-WidthPark2(1,k))< WidthPass2(1,k))
                thetatem=thetatem-1;
                thetaremp_rad=thetatem/180*pi
                WidthPass2(1,k)=Length0-Width01*cos(thetaremp_rad);
                WidthPark2(1,k)=Width0*cos(thetaremp_rad)+Length0*sin(thetaremp_rad);
                theta20(1,k)=thetatem;
                ParkRowadd(1,k)=1;%奇数车道加一行
                break;
            else
                thetatem=thetatem+1;
            end
        end
    end
    theta20(1,k)=-1;
    ParkRowadd(1,k)=0;
    WidthPark2(1,k)=0;
    WidthPass2(1,k)=0;

```

```

        end
    end
    if theta_rad<theta0
        WidthCar(1,k)=Length0/cos(theta_rad);
        Ld(1,k)=0;
    else
        WidthCar(1,k)=Width0/sin(theta_rad);%每车所占平行通道长度
        Ld(1,k)=Length0*cos(theta_rad)-(Width0/sin(theta_rad)-Width0*sin(theta_rad));%+Width0*cot(theta_rad)*cos(theta_rad);%最后一
车多余长度
    end
    CarNumPerLine(1,k)=fix((Length1-Ld(1,k))/WidthCar(1,k));
    LengthDelta(1,k)=Length1-CarNumPerLine(1,k)*WidthCar(1,k)-Ld(1,k);
    CarNumTotal(1,k)=theta;
    CarNumTotal(2,k)=CarNumPerLine(1,k)*ParkRow(1,k);
    if theta20(1,k)>=0
        if theta20(1,k)/180*pi<theta0
            WidthCar2=Length0/cos(theta20(1,k)/180*pi);
            Ld2=0;
        else
            WidthCar2=Width0/sin(theta20(1,k)/180*pi);%每车所占平行通道长度
            Ld2=Length0*cos(theta20(1,k)/180*pi)-(Width0/sin(theta20(1,k)/180*pi)-Width0*sin(theta20(1,k)/180*pi));%最后一车多余长
度
        end
        CarNumPerLine2(1,k)=fix((Length1-Ld2)/WidthCar2);%多于一行车数目
        CarNumTotal(2,k)=CarNumTotal(2,k)+ CarNumPerLine2(1,k)*ParkRowadd(1,k);
    else
        CarNumPerLine2(1,k)=0;
    end
    ParkRow2(1,k)=ParkRowadd(1,k)+ParkRow(1,k);
    Parktemp=int8(ParkRow2(1,k));
    if mod(Parktemp,2)==0%判断车道排数奇偶性
        ParkRow2(2,k)=ParkRow2(1,k)/2;
    else
        ParkRow2(2,k)=(ParkRow2(1,k)+1)/2;
    end
    if theta20(1,k)<0
        WidthDelta2(1,k)=WidthDelta(1,k);
    else
        WidthDelta2(1,k)=WidthDelta(1,k)-ParkRowadd(1,k)*WidthPark2(1,k)-1*WidthPass2(1,k);
    end
    k=k+1;
end
plot(CarNumTotal(1,:),CarNumTotal(2,:), 'k','linewidth',1);
xlabel('倾斜角/度');
ylabel('总车位数/个');

```

```
title('区域 3 总车位数与倾斜角关系');
```

```
axis([0 90 45 100]);
```

## 附录 2 三角形区域求解程序

```
clc;clear all;close;
```

```
Width0=2.5;Length0=5.5;%车尺寸
```

```
Width01=3;
```

```
Width1=270-41.58;Length1=36-5.55;%停车场尺寸
```

```
theta0=atan(Width0/Length0);%theta 最小值, 也就是 theta=0;
```

```
theta1=atan(270/36);%斜边临界角
```

```
delta_theta=1;%角度步长
```

```
k=1;
```

```
ParkRow=[ ; ];%每个角度对应车位排数和通道数
```

```
ParkNumAdd=[];%添加的车道数和通道数
```

```
CarNumPerLine=[];%每个角度对应一排车数
```

```
CarNumTotal=[];%每个角度对应总车数
```

```
WidthPass=[];%通道宽度矩阵
```

```
WidthDelta=[];%通道宽度矩阵
```

```
WidthDelta2=[];%包括添加的车在内的最后剩余宽度
```

```
LengthDelta=[];
```

```
WidthPark=[];%每车所占垂直通道宽度矩阵
```

```
WidthCar=[];%每车所占平行通道长度矩阵
```

```
Ld=[];%补偿最后一个车位平行通道长度
```

```
WidthPark2=[];%多出来一辆车
```

```
WidthPass2=[];
```

```
theta20=[];
```

```
CarNumPerLine2=[];
```

```
for theta=0:delta_theta:90%floor 为向正无穷取整
```

```
    theta_rad=theta*pi/180;%角度化为弧度
```

```
    WidthPass(1,k)=Length0-Width01*cos(theta_rad);%转弯所需最小通道宽度
```

```
    WidthPark(1,k)=Width0*cos(theta_rad)+Length0*sin(theta_rad);%每车所占垂直通道宽度
```

```
    Wtemp=Length0*cos(theta_rad)+Width0*sin(theta_rad);
```

```
    f1=[-1,0];
```

```
    intcon1=[1,2];%x1 是车位排数, x2 是通道数
```

```
    A1=[WidthPark(1,k)*Length1/Width1,WidthPass(1,k)*Length1/Width1;
```

```
        1,-2;-1,2];
```

```
    b1=[(Length1-Wtemp);0;1];
```

```
    Aeq1=[];
```

```
    beq1=[];
```

```
    LB1=[0,0];
```

```
    UB1=[];
```

```
    [ParkRow(1:2,k),FVAL1] = intlinprog(f1,intcon1,A1,b1,Aeq1,beq1,LB1,UB1);
```

```
    Parktemp=int8(ParkRow(1,k));
```

```
    k2=1;
```

```
    if mod(Parktemp,2)==0%判断车道排数奇偶性
```

```

WidthDelta2(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-ParkRow(1,k)/2*WidthPass(1,k);
WidthDelta(1,k)=WidthDelta2(1,k);
while 1
    if ((WidthDelta2(1,k)-Length0)/tan(theta1)>Width0) && ((WidthDelta2(1,k)-Length0)>0)
        ParkNumAdd(k2,k)=fix((WidthDelta2(1,k)-Length0)/tan(theta1)/Width0);
        WidthDelta2(1,k)=WidthDelta2(1,k)-Length0;
        k2=k2+1;
        if ((WidthDelta2(1,k)-2*Length0)/tan(theta1))>Width0 && ((WidthDelta2(1,k)-2*Length0)>0)
            ParkNumAdd(k2,k)=fix((WidthDelta2(1,k)-2*Length0)/tan(theta1)/Width0);
            WidthDelta2(1,k)=WidthDelta2(1,k)-2*Length0;
            k2=k2+1;
        else
            ParkNumAdd(k2,k)=0;
            break;
        end
    else
        ParkNumAdd(k2,k)=0;
        break;
    end
end
if k2==2
    ParkRow(2,k)=ParkRow(1,k)/2+1;
elseif (k2==4)||(k2==3)
    ParkRow(2,k)=ParkRow(1,k)/2+2;
end
else%奇数车道
    WidthDelta2(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-(ParkRow(1,k)-1)/2*WidthPass(1,k);
    WidthDelta(1,k)=WidthDelta2(1,k);
    WidthDelta2(1,k)=WidthDelta2(1,k)-WidthPass(1,k);
    while 1
        if (WidthDelta2(1,k)-Length0)/tan(theta1)>Width0 && (WidthDelta2(1,k)-Length0)>0
            ParkNumAdd(k2,k)=fix((WidthDelta2(1,k)-Length0)/tan(theta1)/Width0);
            WidthDelta2(1,k)=WidthDelta2(1,k)-Length0;
            k2=k2+1;
            if (WidthDelta2(1,k)-2*Length0)/tan(theta1)>Width0&&(WidthDelta2(1,k)-2*Length0)>0
                ParkNumAdd(k2,k)=fix((WidthDelta2(1,k)-Length0)/tan(theta1)/Width0);
                WidthDelta2(1,k)=WidthDelta2(1,k)-2*Length0;
                k2=k2+1;
            else
                ParkNumAdd(k2,k)=0;
                break;
            end
        else
            ParkNumAdd(k2,k)=0;
            break;
        end
    end
end

```

```

        end

    end

    if k2==2||k2==3%补偿后总通道数
        ParkRow(2,k)=(ParkRow(1,k)-1)/2+2;
    elseif k2==4
        ParkRow(2,k)=(ParkRow(1,k)-1)/2+2;
    end

end

if theta_rad<theta0
    WidthCar(1,k)=Length0/cos(theta_rad);
    Ld(1,k)=0;
else
    WidthCar(1,k)=Width0/sin(theta_rad);%每车所占平行通道长度
    Ld(1,k)=Length0*cos(theta_rad)-(Width0/sin(theta_rad)-Width0*sin(theta_rad));%+Width0*cot(theta_rad)*cos(theta_rad);%最后一车多余长度
end

CarNumPerLine(1,k)=theta;
CarNumTotal(1,k)=theta;
Lengthtemp2=0;
Widthtemp=0;
for k1=1: ParkRow(1,k)
    Widthtemp=fix(k1/2)*WidthPass(1,k)+(k1-0.5)*WidthPark(1,k);
    Lengthtemp2=-Length1/Width1*Widthtemp+Length1;
    CarNumPerLine(k1+1,k)=fix((Lengthtemp2-Ld(1,k))/WidthCar(1,k));
    LengthDelta(k1+1,k)=Lengthtemp2-CarNumPerLine(k1+1,k)*WidthCar(1,k)-Ld(1,k);
end

CarNumTotal(2,k)=sum(CarNumPerLine(2:int8(ParkRow(1,k))+1,k));
CarNumTotal(2,k)=CarNumTotal(2,k)+sum(ParkNumAdd(1:k2,k));
ParkRow(1,k)=ParkRow(1,k)+k2-1;%补偿后总车道数
k=k+1;
end

plot(CarNumTotal(1,:),CarNumTotal(2,:));

```

### 附录 3 灵敏度分析代码

```

clc;clear all;close;

Width0=2.5;Length0=5.5;%车尺寸
Width01=3;
Width1=26.5;Length1=79;%停车场尺寸,与车道平行长度
k3=2;
CarNumTotal=[];%每个角度对应总车数
for Length1=50:10:100
    theta0=atan(Width0/Length0);%theta 最小值, 也就是 theta=0;
    delta_theta=1;%角度步长
    k=1;
    ParkRow=[];%每个角度对应车位排数和通道数

```

```

ParkRow2=[];%补偿后总数
ParkRowadd=[];%添加的车道数和通道数
CarNumPerLine=[];%每个角度对应一排车数
WidthPass=[];%通道宽度矩阵
WidthDelta=[];%通道宽度矩阵
WidthDelta2=[];%包括添加的车在内的最后剩余宽度
LengthDelta=[];
WidthPark=[];%每车所占垂直通道宽度矩阵
WidthCar=[];%每车所占平行通道长度矩阵
Ld=[];%补偿最后一个车位平行通道长度
WidthPark2=[];%多出来一辆车
WidthPass2=[];
theta20=[];
CarNumPerLine2=[];
for theta=0:delta_theta:90%floor 为向正无穷取整
    theta_rad=theta*pi/180;%角度化为弧度
    WidthPass(1,k)=Length0-Width01*cos(theta_rad);%转弯所需最小通道宽度
    WidthPark(1,k)=Width0*cos(theta_rad)+Length0*sin(theta_rad);%每车所占垂直通道宽度
    fl=[-1,0];
    intcon1=[1,2];%x1 是车位排数, x2 是通道数
    A1=[WidthPark(1,k),WidthPass(1,k); 1,-2; -1,2];
    b1=[Width1;0;1];
    Aeq1=[];
    beq1=[];
    LB1=[0,0];
    UB1=[];
    [ParkRow(1:2,k),FVAL1] = intlinprog(fl,intcon1,A1,b1,Aeq1,beq1,LB1,UB1);
    Parktemp=int8(ParkRow(1,k));
    if mod(Parktemp,2)==0%判断车道排数奇偶性
        WidthDelta(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-ParkRow(1,k)/2*WidthPass(1,k);
        if WidthDelta(1,k)>=5.5
            if WidthDelta(1,k)>=7.5
                theta20(1,k)=asin((WidthDelta(1,k)-5.5)/11.18)*180/pi-10.305;
                ParkRowadd(1,k)=2;
            else
                theta20(1,k)=asin((WidthDelta(1,k)-5.5)/5.5227)*180/pi+5.19;
                ParkRowadd(1,k)=1;
            end
            WidthPass2(1,k)=Length0-Width01*cos(theta20(1,k)*pi/180);%多出来一排车转弯所需最小通道宽度
            WidthPark2(1,k)=Width0*cos(theta20(1,k)*pi/180)+Length0*sin(theta20(1,k)*pi/180);%多出来一排车所占垂直通道宽度 nd
        else
            theta20(1,k)=-1;
            ParkRowadd(1,k)=0;
            WidthPark2(1,k)=0;
            WidthPass2(1,k)=0;
        end
    end
end

```



```

end
else%奇数车道
    WidthDelta(1,k)=Width1-ParkRow(1,k)*WidthPark(1,k)-(ParkRow(1,k)-1)/2*WidthPass(1,k);
    if WidthDelta(1,k)>=(Width0+WidthPass(1,k))
        thetatemp=0;
        while 1
            thetaremp_rad=thetatemp/180*pi;
            WidthPass2(1,k)=Length0-Width01*cos(thetaremp_rad);%多出来一排车转弯所需最小通道宽度
            WidthPark2(1,k)=Width0*cos(thetaremp_rad)+Length0*sin(thetaremp_rad);
            If ((WidthDelta(1,k)-WidthPark2(1,k))<WidthPass(1,k))||((WidthDelta(1,k)-WidthPark2(1,k))< WidthPass2(1,k))
                thetatemp=thetatemp-1;
                thetaremp_rad=thetatemp/180*pi;
                WidthPass2(1,k)=Length0-Width01*cos(thetaremp_rad);%多出来一排车转弯所需最小通道宽度
                WidthPark2(1,k)=Width0*cos(thetaremp_rad)+Length0*sin(thetaremp_rad);
                theta20(1,k)=thetatemp;
                ParkRowadd(1,k)=1;%奇数车道加一行
                break;
            else
                thetatemp=thetatemp+1;
            end
        end
    end
    theta20(1,k)=-1;
    ParkRowadd(1,k)=0;
    WidthPark2(1,k)=0;
    WidthPass2(1,k)=0;
end
end
if theta_rad<theta0
    WidthCar(1,k)=Length0/cos(theta_rad);
    Ld(1,k)=0;
else
    WidthCar(1,k)=Width0/sin(theta_rad);%每车所占平行通道长度
    Ld(1,k)=Length0*cos(theta_rad)-(Width0/sin(theta_rad)-Width0*sin(theta_rad));%+Width0*cot(theta_rad)*cos(theta_rad);%最后一
车多余长度
end
CarNumPerLine(1,k)=fix((Length1-Ld(1,k))/WidthCar(1,k));
LengthDelta(1,k)=Length1-CarNumPerLine(1,k)*WidthCar(1,k)-Ld(1,k);
CarNumTotal(1,k)=theta;
CarNumTotal(k3,k)=CarNumPerLine(1,k)*ParkRow(1,k);
if theta20(1,k)>=0
    if theta20(1,k)/180*pi<theta0
        WidthCar2=Length0/cos(theta20(1,k)/180*pi);
        Ld2=0;
    else

```

```

        WidthCar2=Width0/sin(theta20(1,k)/180*pi);%每车所占平行通道长度
        Ld2=Length0*cos(theta20(1,k)/180*pi)-(Width0/sin(theta20(1,k)/180*pi)-Width0*sin(theta20(1,k)/180*pi));
    end
    CarNumPerLine2(1,k)=fix((Length1-Ld2)/WidthCar2);%多于一行车数目
    CarNumTotal(k3,k)=CarNumTotal(k3,k)+ CarNumPerLine2(1,k)*ParkRowadd(1,k);%加上多余的 n 行
else
    CarNumPerLine2(1,k)=0;
End
    k=k+1;
end
    k3=k3+1;
end
k3=k3-2;
    plot(CarNumTotal(1,:),CarNumTotal(2,:),'k','linewidth',1);
    xlabel('倾斜角/度');
    ylabel('总车位数/个');
    title('不同长度灵敏度分析');
    hold on
    plot(CarNumTotal(1,:),CarNumTotal(3,:),'y','linewidth',1);
    hold on
    plot(CarNumTotal(1,:),CarNumTotal(4,:),'r','linewidth',1);
    hold on
    plot(CarNumTotal(1,:),CarNumTotal(5,:),'g','linewidth',1);
    hold on
    plot(CarNumTotal(1,:),CarNumTotal(6,:),'b','linewidth',1);
    legend('A=50m','A=60m','A=70m','A=80m','A=90m','A=100m');

```