

赛区评阅编号（由赛区组委会填写）：

---

## 2023 高教社杯全国大学生数学建模竞赛

### 承 诺 书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》（以下简称“竞赛章程和参赛规则”，可从 <http://www.mcm.edu.cn> 下载）。

我们完全清楚，在竞赛开始后参赛队员不能以任何方式，包括电话、电子邮件、“贴吧”、QQ 群、微信群等，与队外的任何人（包括指导教师）交流、讨论与赛题有关的问题；无论主动参与讨论还是被动接收讨论信息都是严重违反竞赛纪律的行为。

我们完全清楚，在竞赛中必须合法合规地使用文献资料和软件工具，不能有任何侵犯知识产权的行为。否则我们将失去评奖资格，并可能受到严肃处理。

**我们以中国大学生名誉和诚信郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。**

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号（从 A/B/C/D/E 中选择一项填写）： A

我们的报名参赛队号（12 位数字全国统一编号）： 2101105

参赛学校（完整的学校全称，不含院系名）： 哈尔滨工业大学（深圳）

参赛队员（打印并签名）：1. 成员 A

2. 成员 B

3. 成员 C

指导教师或指导教师组负责人（打印并签名）： 陈勇勇

（指导教师签名意味着对参赛队的行为和论文的真实性负责）

日期： 2023 年 07 月 30 日

（请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。）

赛区评阅编号：  
(由赛区填写)

全国评阅编号：  
(全国组委会填写)

2023 高教社杯全国大学生数学建模竞赛

编 号 专 用 页

赛区评阅记录（可供赛区评阅时使用）：

评阅人						
备注						

送全国评阅统一编号：  
(赛区组委会填写)

(请勿改动此页内容和格式。此编号专用页仅供赛区和全国评阅使用，参赛队打印后装订到纸质论文的第二页上。注意电子版论文中不得出现此页。)

# 基于目标优化的 FAST 主动反射面调节模型

## 摘要

本文基于 FAST 射电望远镜的工作原理,通过对空间图形、单目标优化、坐标变换等方法,建立了反射面板调节模型,并利用蒙特卡罗模拟方法,对不同条件下馈源舱的接收比进行了研究和计算。

针对问题 1,本文首先根据天体  $S$  位置、焦点  $P$  等信息,确定了抛物线方程各参数之间的函数关系。然后通过研究促动器伸缩量的要求区间,得出了参数的取值范围,获得包含理想抛物面的抛物面曲面簇。建立了以变步长的抛物线方程参数为决策变量,以工作状态 300 米口径内促动器伸缩量满足  $[-0.6, 0.6]$  为约束条件,以促动器伸缩量绝对值总和最少为优化目标的单变量单目标优化模型。并且根据抛物面的性质得出抛物面上的点到原点距离的变化关系,把约束条件改进为仅针对其中两类点进行判断,大幅度降低了程序复杂度。最后本文求解出的理想抛物面方程为  $z = 0.0017795(x^2 + y^2) - 300.9017$ 。

针对问题 2,对于问题所给新的方位角和仰角,首先判断所给角度是否会使 300 米口径工作状态超出 500 米口径的主动反射面。在确定不会超出边界后,求得将原始坐标系旋转到以天体  $S$  与原点  $C$  连线为  $z$  轴的新坐标系的旋转矩阵,然后将其作用于附件 1.csv 所有的主索节点到新坐标系上,用问题 1 的方法求出新坐标系下的理想抛物面方程,根据线面关系求得其余各种问题所求物理量,再逆变换回到原始坐标系下。最终把理想抛物面顶点坐标  $(-49.3997, -36.9491, -294.4974)$ 、调节后反射面 300 米口径内的主索节点编号、位置坐标、各促动器的伸缩量等结果保存在 result.xlsx 中。

针对问题 3,在问题 2 的条件下,本文采用蒙特卡罗模拟,在 300 米口径工作状态内生成随机点发出竖直向下的入射光线,确定射入的三角形反射面,根据其平面方程计算出反射光线,最终确定其是否落在直径为 1 米的馈源舱中心圆盘内,记录成功接收次数比去试验总次数的值,即得接收比。最终本文求得的问题 2 条件下馈源舱的接收比为 1.1353%,基准球面的接收比为 0.7625%,提升的幅度为 48.69%,效果显著。

最后本文对上述模型做了模型评价和进一步的模型推广。

**关键字:** 单目标优化模型 变步长搜索 旋转矩阵 蒙特卡罗模拟

## 1 问题重述

FAST 射电望远镜由主动发射面、馈源舱等部分组成，来自天体的电磁波经过主动反射面的反射汇聚至馈源舱内成像。主动反射面工作的关键在于将反射面调节为工作抛物面，这一环节主要通过促动器顶端伸缩带动反射面板的移动完成。

### 1.1 问题 1:

当待测天体位于 FAST 射电望远镜正上方时，根据主动反射面对电磁波的反射原理确定理想抛物面。

### 1.2 问题 2:

当待测天体相对  $C$  点的方位角为  $\alpha = 36.795^\circ$ ,  $\beta = 78.169^\circ$  (如图 1) 时，确定理想抛物面，给出调节后  $300m$  口径反射面内的主索节点编号和坐标，以及对应促动器顶端的伸缩量等物理量。

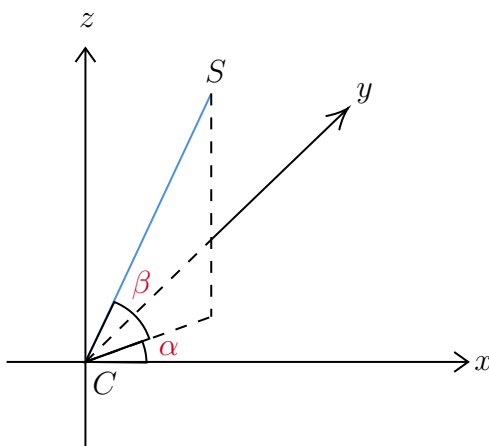


图 1 天体  $S$  方位角与仰角示意图

### 1.3 问题 3:

根据第二问建立的射电望远镜物理模型，计算调节后的接受比，并与初始时球面反射面的接受比进行比较（接受比定义为馈源舱有效区域接收到的反射信号总量与  $300m$  口径反射面内反射出的信号总量的比值）。

## 2 问题分析

根据旋转抛物面关于轴的对称性, 本文将问题化为放在  $yOz$  平面上的抛物线来解决, 从而实现了问题的降维。

### 2.1 问题 1 的分析:

本文把问题 1 转化为三个子问题:

- (1) 求解满足以  $SC$  (即  $z$  轴) 为轴, 焦点在焦面上的旋转抛物面簇;
- (2) 给定步长遍历 (1) 中得到的抛物面簇, 对每一个被遍历的抛物面, 根据附件 1.xlsx 所给的主索节点坐标, 求对应促动器顶端沿径向伸缩后的新的坐标, 并取处在工作态范围内的点为有效节点;
- (3) 根据有效节点伸缩前后坐标的变化, 计算促动器的伸缩量, 筛选出满足促动器的变化范围要求的抛物面, 并以促动器伸缩量绝对值的总和或促动器伸缩量绝对值的最大值等为指标, 确定最终的理想抛物面。

对于子问题 (1), 设抛物面方程  $S: z = a(x^2 + y^2) + b$ , 则  $yOz$  截面上的抛物线方程为  $l: z = ay^2 + b$ , 由  $z$  轴上的点  $C$ 、 $P$  以及抛物线顶点  $A$  之间的位置关系 (如图 2), 可得出参数  $a$  与  $b$  之间的函数关系, 即可获得对应的抛物面簇。

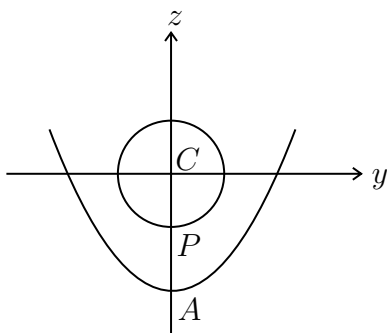


图 2  $z$  轴上各点之间关系

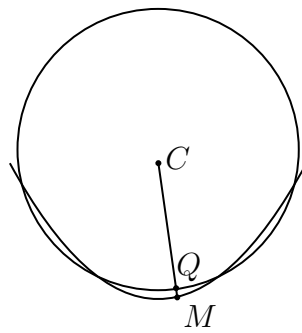


图 3 求解伸缩后主索节点坐标

对于子问题 (2), 为求解促动器顶端伸缩后的坐标, 可以筛选出可能伸缩到工作态范围内的基准球面上的主索节点, 对其中的每一个主索节点  $Q$ , 可确定直线  $CQ$ , 并求出  $CQ$  与被遍历的抛物面的交点坐标  $M$  (如图 3), 即为促动器伸缩后的新主索节点坐标。

对于子问题 (3), 由于主索节点的变化方向是径向的, 因此可以认为新旧主索节点坐标之间的距离即为促动器的伸缩量。按照促动器的变化范围要求, 可通过判断伸缩量是否在  $[-0.6, 0.6]$  之间来筛选出满足条件的抛物面。通过对不同指标实际意义的比较, 本文最终以促动器伸缩量绝对值总和最小为确定最终理想抛物面的指标。

## 2.2 问题 2 的分析:

问题 2 要求在  $\alpha = 36.795^\circ, \beta = 78.169^\circ$  的条件下确定理想抛物面。注意到旋转抛物面沿轴的对称性, 本文考虑在原始坐标系与以  $SC$  为  $z$  轴的坐标系之间建立一个旋转变换关系 (如图 4), 使得问题 2 经旋转变换后可转化为问题 1 的情景, 从而求得在新坐标系下的理想抛物面方程和其它待求量。再通过逆向旋转变换, 最终得到再原始坐标系下的理想抛物面方程和其它待求量。

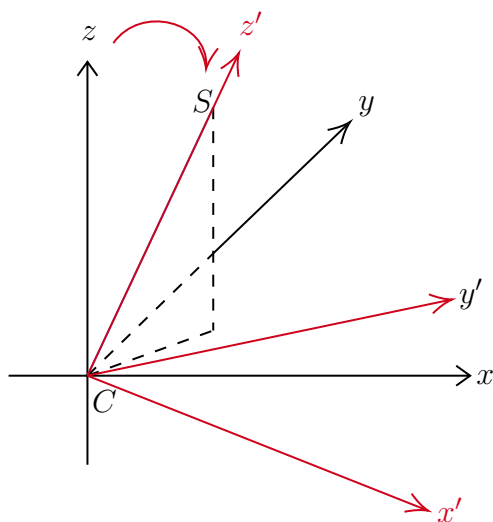


图 4 新旧坐标系的旋转变换关系

## 2.3 问题 3 的分析:

问题 3 要求在问题 2 情况下馈源舱的接收比, 并与基准态球面的接收比做比较。本文拟采用蒙特卡罗方法, 在问题 2 新建的坐标系下研究, 取一条工作态范围内竖直向下的入射光线, 通过它和与之对应的反射面三角形所在平面的几何关系获得反射光线, 再与馈源舱所在的平行于  $xOy$  的平面获得交点, 观察其是否处在馈源舱的有效区域内。该流程经过大量试验后即可获得接收比。

3 符号说明

符号	说明	单位
$S$	被观测体	
$C$	基准球面的球心	
$P$	馈源舱接收平面中心点	
$R$	基准球面半径	$m$
$r$	工作状态范围半径	$m$
$F$	基准球面与交面半径之差	$m$
$D$	FAST 反射面口径	$m$
$p$	馈源舱接收圆盘半径	$m$
$d$	单个促动器伸缩量	$m$
$\Sigma$	工作状态内所有促动器伸缩量绝对值总和	$m$
$\alpha$	天体 $S$ 的方位角	$^{\circ}$
$\beta$	天体 $S$ 的仰角	$^{\circ}$

4 模型假设

- 1. 对问题 1 和问题 2，仅考虑把工作状态反射面理想化为一个光滑理想抛物面；
- 2. 假设促动器顶端伸缩后相邻主索节点间距离的微小变化幅度小于等于 0.07%；
- 3. 不考虑反射过程中电磁波的能量损失对信号强度的影响；
- 4. 不考虑电磁波的二次反射；
- 5. 假设电磁波信号在介质中沿直线传播。

## 5 模型建立与求解

### 5.1 问题 1 的模型建立与求解

#### 5.1.1 旋转抛物面簇的确定

由题目中的方位角和仰角大小  $\alpha = 0^\circ$ 、 $\beta = 90^\circ$  可知，旋转抛物面对应的旋转轴即为  $z$  轴，因此设其方程为  $z = a(x^2 + y^2) + b$ 。根据旋转抛物面的对称性质，本文取  $yOz$  平面与抛物面截得的抛物线  $z = ay^2 + b$  来讨论，实现问题的降维。

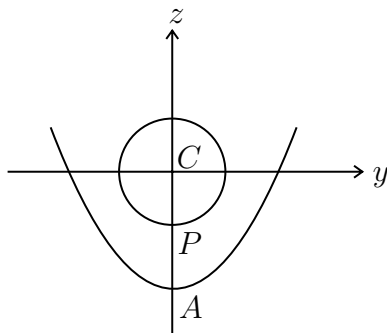


图 5  $yOz$  平面内的二维直角坐标系

如图 5 所示，本文基于题目中所给出的坐标系，建立  $yOz$  面上，以  $C$  为原点的二维直角坐标系，水平向右为  $y$  轴正向，竖直向上为  $z$  轴正向。抛物线开口向上并以  $z$  轴为对称轴， $P$  在焦面球与  $yOz$  面相交的圆周上，是抛物线的焦点， $A$  为抛物线的顶点，位于  $z$  轴的负半轴上。易知  $A$  点坐标为  $(0, b)$ ， $b < 0$ ，并且有数量关系： $|CP|$  加上基准球面与交面半径之差  $F$  即为基准球面半径  $R$ ，即

$$|CP| = (1 - 0.466)R \quad (1)$$

又根据抛物线的方程和其性质，抛物线焦距  $f$  满足：

$$f = \frac{1}{4a} \quad (2)$$

根据  $C$ 、 $P$ 、 $A$  间的位置关系有： $|b| = |CP| + f$ ，结合 (1) (2) 整理得到抛物线参数  $a$ 、 $b$  之间的关系：

$$-b = 0.534R + \frac{1}{4a} \quad (3)$$

从基准球面调整为工作态抛物面后，由于顶点  $A$  处的促动器的变化限制， $b$  的取值可以确定为  $[-R - 0.6, -R + 0.6]$ ，由 (3) 即把原抛物线方程转化成了只含单个参数的方程：

$$z = -\frac{1}{4(b + 0.534R)}y^2 + b, \quad b \in [-301.0, -299.8] \quad (4)$$

通过  $b$  值的变化，即可获得对应的旋转抛物面簇，所求的理想抛物面即在该抛物面簇中。因此，本文将  $b$  作为决策变量，对该抛物面簇进行优化。



### 5.1.2 单目标优化模型的建立

本文根据理想抛物面的要求,以工作态内所有促动器伸缩量在  $[-0.6, 0.6]$  之间为约束条件,工作态内所有促动器伸缩量绝对值总和  $\Sigma$  最小为优化目标,建立单目标单变量优化模型。设变化前位于基准球面上的主索节点坐标为  $(x_0, y_0, z_0)$ ,沿径向变化后的工作态抛物面上的新的坐标为  $(x_1, y_1, z_1)$ 。

注意到  $\sqrt{x_0^2 + y_0^2 + z_0^2} = R$ , 得出目标函数如下:

$$\Sigma = \sum |R - \sqrt{x_1^2 + y_1^2 + z_1^2}|, \quad \forall (x_1, y_1, z_1) | x_1^2 + y_1^2 \leq r^2 \quad (5)$$

约束条件为:

$$s.t. \quad -0.6 \leq R - \sqrt{x_1^2 + y_1^2 + z_1^2} \leq 0.6, \quad \forall (x_1, y_1, z_1) | x_1^2 + y_1^2 \leq r^2 \quad (6)$$

由于工作态内的点太多,以上的约束条件将需要对大量的数据进行处理,降低程序的运行效率,因此拟对该约束条件进行改进。本文从抛物线的性质出发,获得抛物线上的动点到原点的距离变化关系,总结获得引理 1 并给出相应的证明 1:

**引理 1** 对于抛物线方程  $z = ay^2 + b$ ,  $a > 0$ ,  $b < 0$ , 且  $b < -\frac{1}{2a}$ 。当动点在抛物线上从顶点  $A$  沿着一边运动时,其与原点的距离先减小后增大,并在  $z = -\frac{1}{2a}$  处取得最小值。

**证明 1** 设动点坐标为  $(y_0, z_0)$ , 到原点  $C$  的距离的平方满足  $d_0^2 = y_0^2 + z_0^2$ , 和抛物线方程联立得:  $d_0^2 = z_0^2 + \frac{1}{a}z_0 - \frac{b}{a}$ , 根据二次函数性质得知  $d_0$  在  $z = -\frac{1}{2a}$  处取最小值, 当动点在抛物线上从顶点  $A$  沿着一边运动时, 其与原点的距离先减小后增大, 证毕。

根据图 6, 以下对题中所给信息是否满足引理 1 的条件展开讨论:

1) 首先讨论  $b < -\frac{1}{2a}$  是否成立。因为  $b \in [-301.0, -299.8]$ , 根据  $a$ 、 $b$  间的函数关系即公式 (3) 可以得到  $-\frac{1}{2a} \in (-281.2, -278.7)$ ; 所以理想抛物面必满足  $b < -\frac{1}{2a}$ 。

2) 判断工作态边界圈, 即  $x^2 + y^2 = r^2$  与工作态抛物面相交的圆圈所在的面高度  $z_{\text{bound}}$  与面  $z = -\frac{1}{2a}$  的大小关系。根据抛物面方程  $z = a(x^2 + y^2) + b$  可得出边界圈满足  $z_{\text{bound}} = ar^2 + b$ , 将公式 (3) 代入得到

$$z_{\text{bound}} = b - \frac{r^2}{4(b + 0.534R)} \quad (7)$$

再由  $b$  的取值范围得到  $z_{\text{bound}} \in (-261.1, -259.4)$ , 满足  $-\frac{1}{2a} < z_{\text{bound}}$ 。

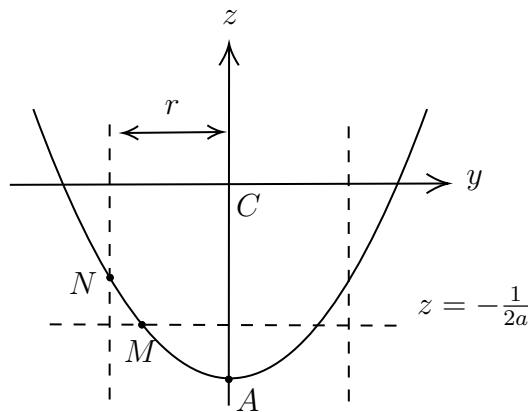


图 6 抛物线上 A、M、N 与原点 C 的位置关系

如图 6 所示, A 点为抛物线的顶点, M 是抛物线上  $z$  坐标为  $-\frac{1}{2a}$  的点, N 则是距离  $z$  轴为  $r$  的点, 即工作状态边界点, 所以根据引理 1 有: 动点在抛物线上从 A 出发, 向着  $y$  坐标减小的方向运动, 在 AM 段与原点 C 的距离逐渐减小, 在 M 处取得最小值, 在 MN 段逐渐增大。结合抛物面方程  $z = a(x^2 + y^2) + b$ , 在曲线 AN 段上有: 对于 M 满足  $z_M = -\frac{1}{2a}$  时有  $|MC|$  最小:

$$|MC| = \sqrt{x_M^2 + y_M^2 + z_M^2} = \sqrt{z_M^2 + \frac{1}{a}z_M - \frac{b}{a}} = \sqrt{-\frac{1}{4a^2} - \frac{b}{a}} \quad (8)$$

对于 N 满足  $x_N^2 + y_N^2 = r^2$  有  $|NC|$  为边界极值:

$$|NC| = \sqrt{x_N^2 + y_N^2 + z_N^2} = \sqrt{r^2 + (ar^2 + b)^2} \quad (9)$$

由 (8) (9), 本文把先前的约束条件 (6) 改进为:

$$s.t. \quad \begin{cases} d_{\min} = \sqrt{-\frac{1}{4a^2} - \frac{b}{a}} - R \geq -0.6 \\ d_{\max} = \sqrt{r^2 + (ar^2 + b)^2} - R \leq 0.6 \end{cases} \quad (10)$$

大幅度降低了对约束条件的判断复杂度。

### 5.1.3 单目标优化模型的求解

本文建立的基于变步长遍历计算的单目标优化模型求解算法步骤如下:

1) **预处理:** 设置半径  $R = 300.4m$ , 工作状态边界半径  $r = 300m \times 0.5 = 150m$ 。加载数据附件 1.csv, 获取所有可能伸缩到工作状态  $300m$  口径内的点。由于在基准球面中  $x^2 + y^2$  略大于  $r^2$  的点也可能在工作态时收缩到  $300m$  口径内, 因此本文以满足  $x^2 + y^2 \leq (r + 0.6)^2$  为筛选条件对数据进行筛选。

2) **遍历求得目标函数**: 以  $b$  为决策变量, 设置步长为 0.01, 对抛物面簇进行遍历寻优:

对于  $b$  的每一次取值, 都有一个确定的抛物面  $S_0$  与之对应。先利用约束条件 (10) 筛选掉不符合促动器伸缩量在  $[-0.6, 0.6]$  的抛物面。再求第一步预处理后的点移动到工作态抛物面后的新的坐标, 进而求促动器的伸缩量, 并且对于每个  $b$  都要初始化工作态内所有促动器伸缩量绝对值总和  $\Sigma$  为 0。以下开始对这些点进行遍历:

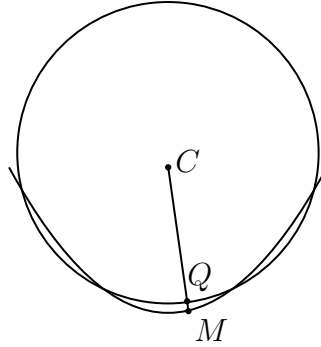


图 7 求解促动器伸缩后的主索节点坐标

如图 7 所示, 设该基准球面上的主索节点坐标为  $Q(x_Q, y_Q, z_Q)$ , 沿径向伸缩后落在抛物面  $S_0$  的  $M$  上。联立

$$\begin{cases} l_{CQ} : \frac{x}{x_Q} = \frac{y}{y_Q} = \frac{z}{z_Q} \\ S_0 : z = a(x^2 + y^2) + b \end{cases} \quad (11)$$

解得  $M$  的坐标  $(x_M, y_M, z_M)$ , 根据相似三角形的原理, 计算其促动器伸缩量  $d_M$ :

$$d_M = \left( \frac{z_M}{z_Q} - 1 \right) R \quad (12)$$

若  $M$  在工作态范围内, 目标函数  $\Sigma$  加上  $d_M$  的绝对值:

$$\Sigma = \sum |d_M|, \quad \forall M(x_M, y_M, z_M) |x_M^2 + y_M^2 \leq r^2 \quad (13)$$

遍历决策变量范围内所有的值之后, 即可获得  $(b, \Sigma)$  之间的键值对关系。

3) **变步长与结果分析**: 在这些键值对关系中, 通过简单比较即可获得最小的目标函数  $\Sigma_{\min}$ , 记录对应步长的最优解  $b_{\text{best}}$ 。在下一轮中缩小决策变量区间, 分别更换遍历步长为 0.001、0.0001, 最终本文确定的理想抛物面为:

$$z = 0.0017795(x^2 + y^2) - 300.9017$$

## 5.2 问题 2 的模型建立与求解

### 5.2.1 模型准备

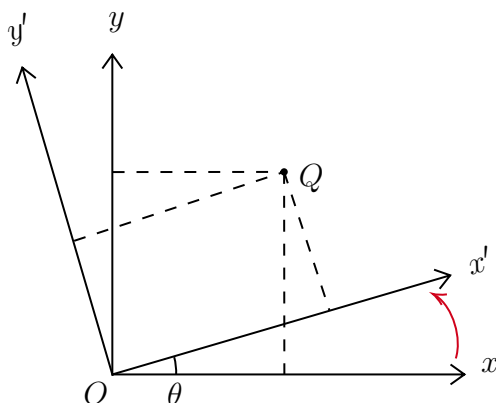


图 8 二维坐标系旋转推导

#### 1) 旋转矩阵

如图 8 所示, 考虑二维坐标系旋转, 设点  $Q$  在坐标系  $xOy$  中的坐标为  $(x, y)$ , 在相对  $xOy$  逆时针旋转  $\theta$  角后的新坐标系  $x'Oy'$  中的坐标为  $(x', y')$ 。根据图中的几何关系, 满足

$$\begin{cases} x' = x \cos\theta + y \sin\theta \\ y' = y \cos\theta - x \sin\theta \end{cases}$$

转化为矩阵形式表示即为

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

故坐标系  $x'Oy'$  相对于坐标系  $xOy$  的旋转变换矩阵为  $\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$ , 它可以 将坐标系  $xOy$  下的向量映射为坐标系  $x'Oy'$  下的对应向量。

类似地, 考虑三维坐标系下绕单坐标轴的旋转。不妨假设坐标系  $xOy$  绕坐标轴  $z$  逆时针旋转  $\theta$  角变为坐标系  $x'Oy'$ , 由于  $z$  轴固定, 故  $z$  坐标不变, 而  $xOy$  平面绕原点逆时针旋转了  $\theta$  角, 与图 8 所示情景完全相同, 故满足

$$\begin{cases} x' = x \cos\theta + y \sin\theta \\ y' = y \cos\theta - x \sin\theta \\ z' = z \end{cases}$$

即可导出对应的旋转矩阵为  $\begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$ 。



### 5.2.2 模型建立

#### 1) 找出待求理想抛物面顶点处的主索节点

假设理想抛物面顶点处存在主索节点  $M$ ，根据旋转抛物面的对称性， $M$  应位于直线  $SC$  上。由于促动器顶端只能沿径向伸缩，则  $M$  在移动前也应位于直线  $SC$  上，即直线  $SC$  与基准球面的交点。仍在第一问坐标系下研究，由  $\alpha = 36.795^\circ, \beta = 78.169^\circ$  可知直线  $SC$  的参数方程为

$$l_{SC} : \begin{cases} x = \cos\beta \cos\alpha t \\ y = \cos\beta \sin\alpha t \\ z = \sin\beta t \end{cases} \quad (14)$$

代入基准球面方程

$$x^2 + y^2 + z^2 = R^2$$

可解得交点坐标为

$$(-49.3200, -36.8894, -294.0185)$$

附件 1.csv 中  $D_{27}$  的坐标为  $(-49.3194, -36.889, -294.0187)$ ，与解出的交点坐标相同，故理想抛物面顶点处存在主索节点  $M$  的猜想成立，且  $M$  为  $D_{27}$ 。

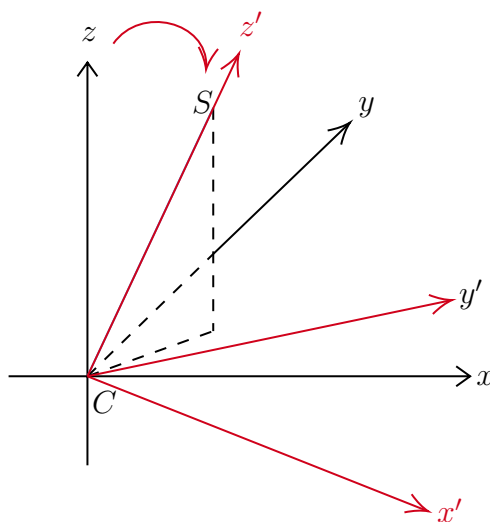
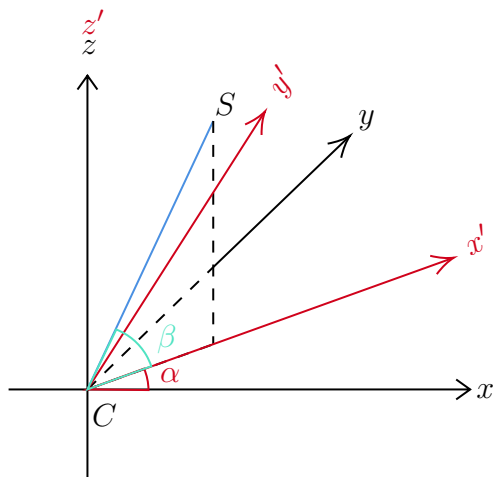


图 10 新旧坐标系的旋转变换关系

#### 2) 导出将原始坐标系旋转到以 $SC$ 为 $z$ 轴的坐标系的旋转矩阵

注意到旋转抛物面沿轴的对称性，本文考虑在原始坐标系与以  $SC$  为  $z$  轴的坐标系之间建立一个旋转变换关系，如图 10 所示。

根据图 10 以及已知条件  $\alpha = 36.795^\circ, \beta = 78.169^\circ$ ，要将原始坐标系旋转变换为

图 11 绕  $z$  轴旋转后的坐标系

以  $SC$  为  $z$  轴的坐标系，首先需要将原始坐标系绕  $z$  轴逆时针旋转  $\alpha$  角（如图 11），使得旋转后的坐标轴  $x'$  与直线  $SC$  在平面  $xOy$  上的投影重合；由图 11 中直线  $SC$  与  $x'$  轴的夹角  $\beta$ （图中以青色标出），再将坐标系绕  $y'$  逆时针旋转  $(90^\circ - \beta)$  角即可让坐标系的  $z$  轴与直线  $SC$  重合，从而得到以  $SC$  为  $z$  轴的坐标系。

根据模型准备中对矩阵旋转的推导，第一次将坐标系绕  $z$  轴逆时针旋转  $\alpha$  角对应的旋转矩阵为

$$R_z = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

第二次将坐标系绕  $y$  轴逆时针旋转  $(90^\circ - \beta)$  角对应的旋转矩阵为

$$R_y = \begin{pmatrix} \cos(90^\circ - \beta) & 0 & -\sin(90^\circ - \beta) \\ 0 & 1 & 0 \\ \sin((90^\circ - \beta)) & 0 & \cos(90^\circ - \beta) \end{pmatrix}$$

从而，将原始坐标系旋转到以  $SC$  为  $z$  轴的坐标系对应的旋转矩阵为

$$M = R_y R_z = \begin{pmatrix} \cos(90^\circ - \beta) & 0 & -\sin(90^\circ - \beta) \\ 0 & 1 & 0 \\ \sin((90^\circ - \beta)) & 0 & \cos(90^\circ - \beta) \end{pmatrix} \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (15)$$

### 3) 基于坐标系旋转变换的理想抛物面计算

把 2) 中得到的旋转矩阵作用在附录 1.csv 给出的主索节点坐标上，即得主索节点在以  $SC$  为  $z$  轴的坐标系下的新坐标，再根据模型准备中的结论，300 米口径的旋转抛物面不会超出主动反射面的边界，因此问题 2 经旋转变换后可转化为问题 1 的情景。

将旋转矩阵作用后的所有主索节点坐标代入问题 1 的代码中进行求解, 可以得到在以  $SC$  为  $z$  轴的坐标系下的理想抛物面方程为  $z = 0.00178(x^2 + y^2) - 300.89$ , 根据理想抛物面方程, 新坐标系下理想抛物面的顶点坐标为  $(0, 0, -300.89)$ 。与问题 1 中的做法相同, 可以通过求直线与抛物面交点来得到新坐标系下反射面 300 米口径内主索节点调节后的坐标, 并根据距离公式得到促动器的伸缩量。

由于求得理想抛物面顶点坐标和反射面 300 米口径内主索节点调节后的坐标都是基于以  $SC$  为  $z$  轴的坐标系, 还要将这些坐标通过**逆向旋转**变换回原始坐标系下的坐标。逆向旋转变换对应的旋转变换矩阵为  $M$  的逆矩阵, 即

$$\begin{aligned}
 M^{-1} &= \left( \begin{pmatrix} \cos(90^\circ - \beta) & 0 & -\sin(90^\circ - \beta) \\ 0 & 1 & 0 \\ \sin((90^\circ - \beta)) & 0 & \cos(90^\circ - \beta) \end{pmatrix} \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \right)^{-1} \\
 &= \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} \cos(90^\circ - \beta) & 0 & -\sin(90^\circ - \beta) \\ 0 & 1 & 0 \\ \sin((90^\circ - \beta)) & 0 & \cos(90^\circ - \beta) \end{pmatrix}^{-1} \quad (16)
 \end{aligned}$$

将矩阵  $M^{-1}$  作用在上述坐标上即可得到原始坐标系下的理想抛物面顶点坐标和原始坐标系下反射面 300 米口径内主索节点调节后的坐标, 以及主索节点  $D_{27}$  的新的坐标  $(-49.3997, -36.9491, -294.4974)$  一并按要求存放在附件 4.xlsx。

### 5.3 问题 3 的模型建立与求解

问题 3 是在问题 2 的条件下求取馈源舱的接收比, 并和基准反射球面的接收比进行比较。考虑到落入每个反射面板上的入射光线的量不一致, 且反射之后在馈源舱所属平面内的投影三角形与  $P$  上的中心圆盘的公共面积较难求取, 本文采用**蒙特卡罗模拟方法**, 在工作态范围内生成竖直向下的入射光线, 经反射后观察其最终是否打在中心圆盘上, 用圆盘接收到的次数比去实验总次数即得接收比。

本问题的研究都基于问题 2 建立的以  $SC$  为  $z$  轴的空间直角坐标系。

#### 5.3.1 数据预处理

问题 2 求得在以  $SC$  为  $z$  轴的空间直角坐标系下反射面 300 米口径内主索节点调节后的坐标后, 经过旋转矩阵  $M^{-1}$  的作用, 最后获得各点在原始坐标系下反射面 300 米口径内主索节点调节后的坐标。问题 3 则对上述过程进行改变, 获得以  $SC$  为  $z$  轴的空间直角坐标系下反射面 300 米口径内主索节点调节后的坐标后, 不再用  $M^{-1}$  作用于各点, 直接存放在 `success.csv` 待使用。



对于附件 3.csv 中给定的 4300 块反射面板对应的主索节点编号, 分别在附件 1.csv、success.csv 中找到三角形的三个顶点坐标。把每个三角反射面的三个顶点  $A$ 、 $B$ 、 $C$  以及与  $z$  轴正向所成角小于  $\frac{\pi}{2}$  的法向量  $\vec{n} : (x_1, y_1, z_1), z_1 > 0$  记录在一个三角形字典中。其中用附件 1.csv 求得的是基准球面的接收比, success.csv 则是问题 2 条件下的接收比。

### 5.3.2 利用蒙特卡罗模拟方法生成随机点

设置蒙特卡罗次数为  $1 \times 10^6$ 。在工作态 300m 口径  $x^2 + y^2 \leq r^2$  范围内随机生成  $x$  坐标与  $y$  坐标, 获得入射光出射点  $N(x, y, 0)$ 。入射光线从  $N$  点出发竖直向下, 必与一三角形反射面交于反射点  $N'(x, y, z)$ 。须确定对应哪一个三角形, 才能求得反射光线。

### 5.3.3 确定入射光线对应的反射面板和入射点坐标

根据以  $z$  轴为对称轴的抛物面的性质, 抛物面中所有的三角形反射面板投影到  $xOy$  平面都是紧密无缝且不重叠的, 因此通过二维随机点  $P(x, y)$  对应的投影后的三角形面板  $S'_\Delta$ , 可以确定它原本的三角形  $S_\Delta$ , 这是一种单射的关系, 找到的三角形必唯一。本文通过遍历三角形字典, 每次都把三个顶点  $A$ 、 $B$ 、 $C$  投影到  $xOy$  平面内获得  $A'$ 、 $B'$ 、 $C'$ , 如图 12 所示用向量法对  $P$  进行判断。

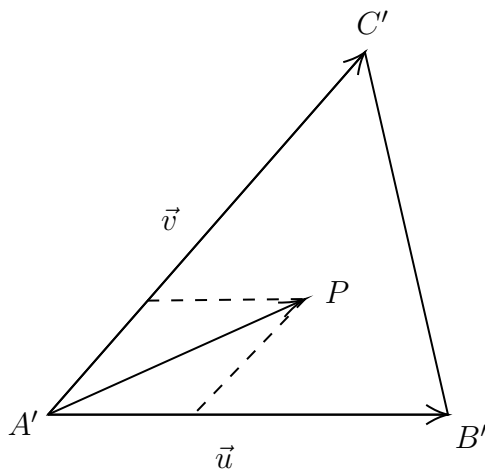


图 12 用向量法判定点  $P$  是否在  $\triangle A'B'C'$  中

根据平面中的任意向量可以由两不共线非零向量的线性组合表示, 在  $\triangle A'B'C'$  中, 有  $\vec{A'P} = \mu \vec{A'C'} + \lambda \vec{A'B'}$ 。那么  $P$  在  $\triangle A'B'C'$  中的充分必要条件是:  $\mu > 0$ ,  $\lambda > 0$  且  $\mu + \lambda < 1$ 。根据上述方法对每次取得的三角形进行判断, 当满足以上所有条件时, 即找到了三维随机点  $N$  竖直向下发出入射光线所对应的反射面三角形  $S_{\Delta_{in}}$ , 停止遍历。

在先前创建的三角形字典中获取  $S_{\Delta_{in}}$  的其中一个顶点  $A(x_A, y_A, z_A)$  及其法向量  $\vec{n}(x_1, y_1, z_1)$ , 得出  $S_{\Delta_{in}}$  所在平面  $S_{ABC}$  的方程:

$$S_{ABC}: x_1(x - x_A) + y_1(y - y_A) + z_1(z - z_A) = 0 \quad (17)$$

由于入射光线竖直向下, 将  $N(x, y, 0)$  代入即可得出反射点  $N'(x, y, z)$ , 其中

$$z = \frac{x_1(x_A - x) + y_1(y_A - y)}{z_1} + z_A \quad (18)$$

#### 5.3.4 利用几何关系判断反射光线是否打在馈源舱内

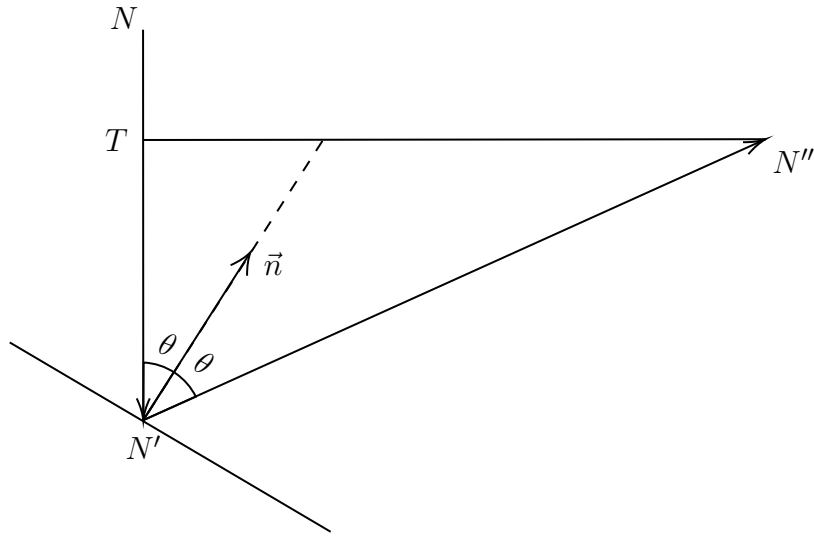


图 13 入射光线的反射图像

根据 5.3.3, 点  $N'$  的坐标为  $(x, y, z)$ , 法向量  $\vec{n} = (x_1, y_1, z_1)$ , 因此由图 13 的几何关系可得

$$\cos\theta = \frac{(x_1, y_1, z_1) \cdot (0, 0, 1)}{\sqrt{x_1^2 + y_1^2 + z_1^2}} = \frac{z_1}{\sqrt{x_1^2 + y_1^2 + z_1^2}} \quad (19)$$

直线  $TN''$  位于馈源舱所在平面上, 故有

$$|TN'| = |z| - |CP| \quad (20)$$

又由于

$$|TN''| = |TN'| \cdot \tan 2\theta \quad (21)$$

将 (19) (20) 代入 (21) 中可得

$$|TN''| = (|z| - |CP|) \cdot \frac{2z_1\sqrt{x_1^2 + y_1^2}}{z_1^2 - x_1^2 - y_1^2} \quad (22)$$

在馈源舱所处的二维  $xOy$  平面内, 对  $TN''$  算出在  $x$  与  $y$  方向上的分量

$$\overrightarrow{TN''} = (|TN''| \frac{x_1}{\sqrt{x_1^2 + y_1^2}}, |TN''| \frac{y_1}{\sqrt{x_1^2 + y_1^2}}) \quad (23)$$

而点  $T$  的坐标为  $(x_1, y_1)$ , 故点  $N''$  的坐标为

$$\begin{aligned} (x_{N''}, y_{N''}) &= \left( x_1 + |TN''| \frac{x_1}{\sqrt{x_1^2 + y_1^2}}, y_1 + |TN''| \frac{y_1}{\sqrt{x_1^2 + y_1^2}} \right) \\ &= \left( x_1 + \frac{2(|z| - |CP|)x_1z_1}{z_1^2 - x_1^2 - y_1^2}, y_1 + \frac{2(|z| - |CP|)y_1z_1}{z_1^2 - x_1^2 - y_1^2} \right) \end{aligned} \quad (24)$$

由于馈源舱的接受范围为  $x^2 + y^2 \leq p^2$  ( $p$  为馈源舱的半径, 大小为 0.5 米), 因此只有满足  $x_{N''}^2 + y_{N''}^2 \leq p^2$  的入射光线才能成为有效的入射光线, 代入 (24) 中的  $(x_{N''}, y_{N''})$  即可得解。

### 5.3.5 问题 3 的结果分析

	基准球面接收比	调整面板后接收比
1	0.7695%	1.1572%
2	0.7518%	1.1307%
3	0.7716%	1.1268%
4	0.7647%	1.1274%
5	0.7600%	1.1345%
平均值	0.7635%	1.1353%

表 1 五次蒙特卡罗模拟结果及平均值

经过以上方法计算, 获得的数据存放于表 1。最终结论为: 基于问题 2 的反射面调节方案, 所得的馈源舱接收比为 **1.1353%**, 基准反射球面的接收比为 **0.7625%**, 调节之后的接收比提升率为 **48.69%**, 提升效果显著。

## 6 模型评价与推广

### 6.1 模型优点

1. 在对  $b$  进行变步长搜索的过程中, 通过研究抛物面上的点到  $C$  点距离的单调性, 只需对特定位置的三个点的伸缩量进行检查, 避开了对所有主索节点伸缩量的依次检查, 极大减小了算法的时间复杂度。

2. 问题 3 采用蒙特卡罗模拟的方法, 通过在馈源舱范围内生成随机点, 来模拟光线的入射, 一方面省却了繁复的数学演算过程, 另一方面降低了问题的建模难度, 不必考虑不同三角形反射面在反射后于馈源舱平面处的投影三角形与中心圆盘的重合面积的问题。

### 6.2 模型缺点

1. 在寻找理想抛物面时, 没有根据要求的“相邻主索节点坐标之间距离变化幅度不超过 0.07%”将“相邻主索节点坐标之间距离变化尽可能小”作为优化条件之一。

2. 在寻找理想抛物面时, 仅从几何的角度考虑问题, 而没有从力学角度考虑各主索节点伸缩时的受力情况, 从减小受力的角度对模型进行进一步优化。

### 6.3 模型的改进和推广

1. 可以对主索节点的伸缩过程进行应力分析, 从而找到对索网结构破坏最小的伸缩方案。

2. 由于该模型可以给出不同情况下馈源舱有效区域的接收比, 因此基于计算问题 1、问题 2 的模型可以进一步对每一块三角形做单独调整, 使其顶点不必位于理想抛物面内, 以使得馈源舱有效区域可获得最大接收比。

## 参考文献

- [1] 南仁东. 500m 球反射面射电望远镜 FAST[J]. 中国科学 G 辑: 物理学、力学、天文学, 2005.
- [2] 钱宏亮. FAST 主动反射面支承结构理论与试验研究[D]. 哈尔滨工业大学, 2007.

# 问题一核心代码

```
import pandas as pd
import numpy as np
```

```
data = pd.read_csv('data1.csv', usecols=range(1, 4),
encoding="gbk", dtype=float)
```

# 新建一个存放b所对应的促动器变化值总和的字典

```
b_dict = {}
```

# b在[-301, -299.8]的范围之间变动, 把0.001作为变化步长进行遍历:

```
# -301, -299.8, 0.01
```

```
# -301, -299.8, 0.001
```

```
# -300.95, -300.85, 0.0001
```

```
for b in np.arange(-301, -299.8, 0.001):
```

# 针对第一问的条件, 由  $f + CP = |b|$  可得a关于b的函数关系

```
a = 1/(4*(-b - 0.534*300.4))
```

```
"""
```

在yOz平面中研究促动器伸缩量, 取 $z = a*y^2 + b$ 平面进行研究

通过观察, 发现理想抛物面上的点到原点C的距离d满足从顶点A0'到 $z =$

$-1/2a$ 的点先变小,

再到工作态抛物面边界的点(满足 $\sqrt{x^2 + y^2} = 150$ )又变大的单调性关系

因此确定筛选条件: 1.  $\sqrt{z^2 + y^2} - R \geq -0.6$  ,  $z = -1/2a$

2.  $\sqrt{z^2 + y^2} - R \leq +0.6$  ,  $y = 150$

```
"""
```

```
if (-b/a - 1/(4*a**2))**0.5 - 300.4 < -0.6 or (150**2 +
```

```
(a*150**2+b)**2)**0.5 - 300.4 > 0.6:
```

```
continue
```

```

else:
    b_dict[b] = 0
    for row in data.iloc:
        x0 = row[0]
        y0 = row[1]
        z0 = row[2]
        if x0**2 + y0**2 <= 22591: # 根据极端情况 筛掉无用的点
            # 计算新的主索节点的纵坐标z1

            """
            先转化为求平面上的交点的问题
            联立: 1.  $z = a*y^2 + b$ 
            2.  $z = z0/\sqrt{x0^2 + y0^2} * y$ 
            解得关于z的一元二次方程  $Az^2 + Bz + C = 0$ 
            """
            A = a*((x0*x0+y0*y0)/(z0*z0))
            B = -1
            C = b
            if A != 0:
                # 所求的点在x0y平面以下, 满足 $z < 0$ , 故取较小的z1
                z1 = (-B-(B*B-4*A*C)**0.5)/(2*A)
                y1 = (z1/z0) * y0
                x1 = (z1/z0) * x0
                # 根据相似关系  $d = R - R(z1/z0)$ 
                d = 300.4*(1-z1/z0)
                # 仅取工作态范围内的点
                if x1**2 + y1**2 <= 22500:
                    b_dict[b] += abs(d)
            else:
                # 最终加上顶点的变化量(因为顶点的xy坐标均为0, 需要单独计算)
                b_dict[b] += abs(300.4 + b)

            # final_b为最终确定的最优参数
            final_b = 0
            min_sum = np.inf
            # 遍历字典求得最小变化量所对应的b
            for (key, value) in b_dict.items():

```

```
if value < min_sum:
    min_sum = value
    final_b = key
    final_a = 1/(4*(-final_b - 0.534*300.4))

print(f"z = {round(final_a,7)}(x^2+y^2)+{round(final_b,4)}")
```

```
# 问题2核心代码
import pandas as pd
import numpy as np
import openpyxl as op

alpha = np.deg2rad(36.795)
gamma = np.deg2rad(90 - 78.169)

# 计算旋转矩阵
Rz = np.array([[np.cos(alpha), np.sin(alpha), 0],
               [-np.sin(alpha), np.cos(alpha), 0],
               [0, 0, 1]])

Ry = np.array([[np.cos(gamma), 0, -np.sin(gamma)],
               [0, 1, 0],
               [np.sin(gamma), 0, np.cos(gamma)]])

R = Ry @ Rz
R_inv = np.linalg.inv(R)

data = pd.read_csv('data1.csv', usecols=range(0, 4), encoding="gbk")

a_dict = {}

# i表示数据表中的行索引
i = 0
```

```

for row in data.iloc:
    x0 = row[1]
    y0 = row[2]
    z0 = row[3]
    # 求出每点在新坐标系下的坐标, 并保存至a_dict
    A_0 = R @ [[x0], [y0], [z0]]
    a_dict[i] = A_0
    i = i + 1

# 新建一个存放b所对应的促动器变化值总和的字典
b_dict = {}

# b在[-301, -299.8]的范围之间变动, 把0.001作为变化步长进行遍历:
for b in np.arange(-301, -299.8, 0.001):

    # 针对第一问的条件, 由  $f + CP = |b|$  可得a关于b的一次函数关系
    a = 1 / (4 * (-b - 0.534 * 300.4))

    """
    在yOz平面中研究促动器伸缩量, 取  $z = a * y^2 + b$  平面进行研究
    通过观察, 发现理想抛物面上的点到原点C的距离d满足从顶点A0'到  $z = -1/2a$  的点先变小,
    再到工作态抛物面边界的点(满足  $\sqrt{x^2 + y^2} = 150$ )又变大的单调性关系
    因此确定筛选条件: 1.  $\sqrt{z^2 + y^2} - R \geq -0.6$  ,  $z = -1/2a$ 
    2.  $\sqrt{z^2 + y^2} - R \leq +0.6$  ,  $y = 150$ 
    """
    if (-b/a - 1/(4*a**2))*0.5 - 300.4 < -0.6 or (150**2 +
        (a*150**2+b)**2)*0.5 - 300.4 > 0.6:
        continue
    else:
        b_dict[b] = 0
        for j in range(0, i):
            x0 = a_dict[j][0][0]
            y0 = a_dict[j][1][0]
            z0 = a_dict[j][2][0]
            # 仅取工作态范围内的点 计算新的主索节点的纵坐标z1
            if x0**2+y0**2 <= 22591:

```



```

"""
先转化为求平面上的交点的问题
联立：1.  $z = a \cdot y^2 + b$ 
2.  $z = z_0 / \sqrt{x^2 + y^2} \cdot y$ 
解得关于 $z$ 的一元二次方程  $Az^2 + Bz + C = 0$ 
"""

A = a * ((x0 * x0 + y0 * y0) / (z0 * z0))
B = -1
C = b

if A != 0:
    # 所求的点在x0y平面以下，满足 $z < 0$ ，故取较小的 $z_1$ 
    z1 = (-B - (B * B - 4 * A * C) ** 0.5) / (2 * A)
    x1 = (z1 / z0) * x0
    y1 = (z1 / z0) * y0
    # 根据相似关系  $d = R - R(z_1 / z_0)$ 
    d = 300.4 * (1 - z1 / z0)

    if x1 ** 2 + y1 ** 2 <= 22500:
        b_dict[b] += abs(d)
    else:
        # 最终加上顶点的变化量(因为顶点的xy坐标均为0，需要单独计算)
        b_dict[b] += abs(300.4 + b)

# final_b为最终确定的最优参数
final_b = 0
min_sum = np.inf
# 遍历字典求得最小变化量所对应的b
for (key, value) in b_dict.items():
    if value < min_sum:
        min_sum = value
        final_b = key
        final_a = 1 / (4 * (-final_b - 0.534 * 300.4))

print(f"z = {round(final_a, 6)}(x^2+y^2){round(final_b, 3)}")

# 写.xlsx文件前的准备工作

```

```

workbook = op.Workbook()
worksheet1 = workbook.active
worksheet1.title = "调整后主索节点编号及坐标"
worksheet2 = workbook.create_sheet(title="促动器顶端伸缩量")
worksheet3 = workbook.create_sheet(title="理想抛物面顶点坐标")
dat1 = [["节点编号", "X坐标 (米)", "Y坐标 (米)", "Z坐标 (米)"]]
dat2 = [["对应主索节点编号", "伸缩量 (米)"]]
A_3 = R_inv@[[0], [0], [-300.89]]
dat3 = [["X坐标 (米)", "Y坐标 (米)", "Z坐标 (米)"]]
dat3.append([A_3[0][0], A_3[1][0], A_3[2][0]])

'''
新坐标系下点(x0, y0, z0)对应的直线参数方程为 $x=x_0*t$ ,  $y=y_0*t$ ,  $z=z_0*t$ 
与求得的抛物线 $z = 0.00178(x^2+y^2)-300.89$ 联立可得一元二次方程:
 $(0.00178*(x_0^2 + y_0^2)) * x^2 + (-z_0) * x + (-300.89) = 0$ 
即可解出交点坐标
'''

for j in range(0, i):
    x0 = a_dict[j][0][0]
    y0 = a_dict[j][1][0]
    z0 = a_dict[j][2][0]
    # 求解一元二次方程
    D = 0.00178*(x0**2 + y0**2)
    E = -z0
    F = -300.89
    if D != 0:
        t = (-E + (E**2 - 4 * D * F)**0.5)/(2*D)
        x1 = t*x0
        y1 = t*y0
        z1 = t*z0
        if x1**2+y1**2 <= 22500:
            A_1 = R_inv @ [[x1], [y1], [z1]]
            dat1.append([data.iloc[j][0], A_1[0][0], A_1[1][0], A_1[2][0]])
            dat2.append([data.iloc[j][0], (x1**2+y1**2+z1**2)**0.5-300.4])
        else:
            A_1 = R_inv @ [[0], [0], [final_b]]
            dat1.append([data.iloc[j][0], A_1[0][0], A_1[1][0], A_1[2][0]])

```

```
dat2.append([data.iloc[j][0], (x1**2+y1**2+z1**2)**0.5-300.4])
```

```
# 将获得的结果写入result.xlsx文件
```

```
for row in dat1:
```

```
    worksheet1.append(row)
```

```
for row in dat2:
```

```
    worksheet2.append(row)
```

```
for row in dat3:
```

```
    worksheet3.append(row)
```

```
workbook.save(filename="result.xlsx")
```

```
# 问题3核心代码
```

```
import numpy as np
```

```
import pandas as pd
```

```
import random
```

```
import winsound
```

```
alpha = np.deg2rad(36.795)
```

```
gamma = np.deg2rad(90 - 78.169)
```

```
# 计算旋转矩阵
```

```
Rz = np.array([[np.cos(alpha), np.sin(alpha), 0],  
               [-np.sin(alpha), np.cos(alpha), 0],  
               [0, 0, 1]])
```

```
Ry = np.array([[np.cos(gamma), 0, -np.sin(gamma)],  
               [0, 1, 0],  
               [np.sin(gamma), 0, np.cos(gamma)]])
```

```
R = Ry @ Rz
```

```
position_dict = {}
```

```
position_dict_strict = {}
```

```
# 基准态球面
```

```
data1 = pd.read_csv('data1.csv', usecols=range(0, 4), encoding="gbk")
```

```
# 问题2情况计算
```

```
# data1 = pd.read_csv('success.csv', usecols=range(0, 4), encoding="gbk")
```

```
data3 = pd.read_csv('data3.csv', usecols=range(0, 3), encoding="gbk")
```

```
for row in data1.iloc:
    x0 = row[1]
    y0 = row[2]
    z0 = row[3]
    # 求出每点在新坐标系下的坐标, 并保存至a_dict
    A_0 = [[x0], [y0], [z0]]
    # 去掉内层中括号
    A_1 = [A_0[0][0], A_0[1][0], A_0[2][0]]
    # 取处在工作态及附近的点
    if x0**2+y0**2 <= 180**2:
        position_dict[row[0]] = A_1
    if x0**2+y0**2 <= 150**2:
        position_dict_strict[row[0]] = A_1

triangle_dict = {}
i = 0
for row in data3.iloc:
    A = row[0]
    B = row[1]
    C = row[2]

    if (A in position_dict_strict) or (B in position_dict_strict) or (C in
        position_dict_strict):
        A_position = position_dict[A]
        B_position = position_dict[B]
        C_position = position_dict[C]
        # 求三角形重心
        G = [(A_position[0] + B_position[0] + C_position[0])/3, (A_position[1] +
            B_position[1] + C_position[1]) /
            3, (A_position[2] + B_position[2] + C_position[2])/3]
        # 求沿z轴正方向的法向量

        VEC_AB = np.array([B_position[0]-A_position[0],
            B_position[1]-A_position[1],
            B_position[2]-A_position[2]])
```

```

VEC_AC = np.array([C_position[0]-A_position[0],
C_position[1]-A_position[1],
C_position[2]-A_position[2]])
n = np.cross(VEC_AB, VEC_AC)
if (n[2] < 0):
n = [-n[0], -n[1], -n[2]]
else:
n = [n[0], n[1], n[2]]
# 将A\B\C和法向量存入三角形字典
triangle_dict[i] = [A_position,B_position,C_position, n]
i = i + 1

# 根据蒙特卡罗法, 随机生成位于工作态内的坐标(x, y)
# 记录距离随机点最近的三角形
triangle_close = []
righttimes = 0
wrongtimes = 0
# 设置蒙特卡洛实验次数
mc_times = 1000000
for j in range(0, mc_times):
print(j)
dis = np.inf
x = random.randrange(-150, 150)
y = random.randrange(-150, 150)

if ((x**2 + y**2) > 150**2):
continue
for num, tri in triangle_dict.items():
u = x - tri[0][0]
v = y - tri[0][1]
m = tri[1][0] - tri[0][0]
n = tri[1][1] - tri[0][1]
r = tri[2][0] - tri[0][0]
s = tri[2][1] - tri[0][1]
a = (s * u - r * v) / (s * m - r * n)
b = (n * u - m * v) / (n * r - m * s)
if(a > 0) and (b > 0) and (a + b < 1):

```

```
triangle_close = tri
break
A_close = triangle_close[0]
n_close = triangle_close[3]
z = (A_close[0] * n_close[0] + A_close[1] * n_close[1] +
A_close[2] * n_close[2] - n_close[0]*x - n_close[1]*y)/n_close[2]
# 计算光线投影在馈源舱平面的坐标

x1 = n_close[0]
y1 = n_close[1]
z1 = n_close[2]
z_prime = abs(z)-0.534 * 300.4
x_reflect = x + 2 * x1 * z1*z_prime/(z1*z1-(x1*x1+y1*y1))
y_reflect = y + 2 * y1 * z1*z_prime/(z1*z1-(x1*x1+y1*y1))

if (x_reflect**2+y_reflect**2) <= 0.5**2:
    righttimes = righttimes + 1
else:
    wrongtimes = wrongtimes + 1
print(righttimes / (righttimes + wrongtimes))
```