

基于目标规划模型的主动反射面形状调节的研究

摘要

本文针对 FAST 理想抛物面确定问题, 基于目标规划模型、遍历法、蒙特卡略随机模拟和空间几何知识等, 得出了工作抛物面的变化模型。

针对问题一, 要求在反射面板可调范围内, 天体 S 位于基准球面正上方时的理想抛物面。为方便求解, 使理想抛物面与基准球面尽量贴合, 建立以工作态和基准态时主索节点的位置位移量最小为目标的单目标规划模型。分析促动器、下拉索、主索节点相对位置, 基本判定节点满足与基准球心共线。暂不考虑约束条件, 假设理想抛物面光滑, 分类讨论工作状态下 FAST 竖直剖面上对称轴与照明区域边界的夹角取不同数值时坐标的变化关系。通过优化算法得出理想抛物面的顶点坐标 $(0, 0, -300.8446)$, 焦距 $f = 140.431m$, 表达式为 $z = \frac{x^2 + y^2}{561.724} - 300.8446$ 。

检验口径为 300 米处抛物面和球面切线的斜率分别为 0.5400、0.5780, 两斜率差值为 0.0380 小于 0.01, 证明该结果合理。

针对问题二, 求待观测物体天体 S 位于 $\alpha = 36.795^\circ, \beta = 78.169^\circ$ 时的理想抛物面。待观测天体并非位于坐标系 z 轴的正上方, 为在第一问的模型基础上进行研究, 考虑旋转坐标系, 即先将坐标系先绕 z 轴逆时针旋转, 再绕 y 轴逆时针旋转。此时, 坐标变换均绕坐标轴进行, 圆心 C 点仍为坐标原点。而基准球面上的点坐标发生偏转, 因此对附件中的坐标均进行变换。将相邻节点距离变化幅度和促动器径向伸缩范围作为约束条件, 利用问题一中的模型求解出此时的理想旋转抛物面。又由于促动器沿基准球面径向安装、下拉索径向伸缩来完成, 可认为主索点仅沿径向产生位移, 不考虑切向位移。计算各点移动位置, 并将各点坐标还原为起始坐标系下的点。最终, 求得旋转后的理想抛物面顶点坐标为 $(-49.3930, -36.9440, -294.4536)$, 理想抛物面的顶点坐标、调节后反射面对应的主索节点编号、位置坐标、各促动器伸缩量等结果保存在“result.xlsx”中。

针对问题三, 要求基于第二问的反射面板调节方式时的馈源舱的接收比, 并与基准球面接收比进行比较分析。因此, 可将坐标系进行旋转后再求解。在理想抛物面坐标下, 反射面板中仍有点无法反射电磁波至馈源舱。考虑到上述情况, 采用蒙特卡略随机模拟, 在每块反射板上随机取 10000 个点代表该板的反射波在馈源舱的接受情况。反射板固定在三个主索节点上, 其厚度可忽略不计。将反射面板抽象为三角形, 由已知主索节点坐标求得面积。利用三节点的坐标可确定反射板所在小平面的法线方程, 由入射光线与反射光线关于法线对称, 入射电磁波是近似于从天体发射出的平行电磁波, 求得过三角形中心的反射电磁波交馈源器直径为 1 米圆盘所在平面的交点。通过建立物理反射模型, 求解得调节前 (基准球面) 反射的接收比为 0.742%, 调整优化后的工作抛物面接收比为 1.317%, 提升了 0.575%。

关键词: 目标规划模型 遍历法 旋转坐标系 蒙特卡略随机模拟

一、 问题重述

1.1 问题背景

自 1994 年中国天文界提出建造天眼起，中国科学院天文台携天文界科学家历时 16 年，完成 500 米口径球面射电望远镜——中国天眼（FAST）的建造计划。^[1]天眼，迄今为止世界范围内单口径最大、灵敏度最高的射电望远镜，拥有我国自主知识产权。它的出现推动了我国科研技术的进步，为我国实现现代化建设强国的目标添砖加瓦。

1.2 问题提出

主动反射面、信号接收系统及相关的控制、测量和支承系统构成 FAST。

主动反射面有基准态和工作态两个状态。在基准态时，反射面是半径约 300 米、口径为 500 米的球面；在工作态时，反射面是一个 300 米口径的近似旋转抛物面。基准球面的球心记作点 C，馈源舱接收平面的中心在焦面上进行移动，焦面和基准球面同心，半径差为 $F=0.466R$ 。半径为 0.5 米的中心圆盘是馈源舱接收信号的有效区域。FAST 观测天体 S 时，直线 SC 与焦面的交点 P 为馈源舱接收平面的中心，调节基准球面上的部分反射面板，使其形成以直线 SC 为对称轴、以 P 为焦点的近似旋转抛物面，反射汇聚天体的平行电磁波至馈源舱的有效区域。

建立模型求解以下问题：

（1）天体 S 为待观测物体，当它处于基准球面正上方时，此时， $\alpha=0^\circ, \beta=90^\circ$ 。综合反射面板调节因素，给出理想抛物面。

（2）天体 S 为待观测物体，当它位于 $\alpha=36.795^\circ, \beta=78.169^\circ$ 时，给出理想抛物面。为使反射面尽可能贴合理想抛物面，通过调节改变促动器伸缩量，同时限制相邻主索节点之间的距离变化幅度，来求解建立出反射面板调节模型。在“result.xlsx”中保存理想抛物面的顶点坐标、调节后各反射面对应的主索节点编号、位置坐标、促动器伸缩量等结果。

（3）在第二问反射面板调节模型已建立并得到求解的情况下，求出调节至工作抛物曲面后馈源舱的接收比，将其和基准反射球面的接收比进行对比。

二、 问题分析

2.1 问题一的分析

问题一要求天体 S 位于基准球面正上方时的理想抛物面。题目已知天体 S 距基准球面无限远，可将天体发射的电磁波视为平行电磁波，汇聚于馈源舱的中心点 P，点 P 即为焦点。为方便求解理想抛物面的方程，假设理想抛物面与基准球面尽可能贴合，以工作状态时主索节点和基准态时主索节点的位移量为目标函数，

找到使目标值最小时的对应理想曲面抛物方程。不考虑约束，求解最优光滑抛物面。讨论工作态下 FAST 竖直剖面上对称轴与照明区域边界的夹角取不同数值时坐标的变化关系。综合上述条件，通过优化算法，求得理想抛物面。

2.2 问题二的分析

问题二要求求解当待观测物体天体 S 位于 $\alpha = 36.795^\circ$, $\beta = 78.169^\circ$ 时的理想抛物面。此时的待观测天体并非位于坐标系 z 轴的正上方，可将坐标系先绕 z 轴逆时针旋转 α 度，再绕 y 轴逆时针旋转 β 度。由于此时的坐标变换均绕坐标轴进行，则圆心 C 点仍为原点，但附件 1、附件 2 中 $\left(\frac{\pi}{2} - \beta\right)$ 基准球面上的点坐标发生偏转，所以对附件中的坐标均进行变换。变换后可利用问题 1 中的目标函数求解出此时的理想旋转抛物面。由附录知，相邻节点的变化幅度不超过 0.07%，促动器径向伸缩范围为 -0.6~+0.6 米，将主索节点距离变化幅度和促动器径向伸缩范围作为约束条件。由题目可知促动器沿基准球面径向安装、下拉索径向伸缩来完成，则理解为主索点仅延径向移动，不考虑切向位移。待计算好各点移动位置后，还需将各点坐标还原为原始坐标系下的点。

2.3 问题三的分析

问题三是在第二问的基础上进行研究的，仍可将其进行旋转后求解。然而，在理想抛物面的坐标下，各反射面板中部分点不可将电磁波反射至馈源舱。因此，采用蒙特卡略随机取点代表反射波接受情况。由于反射板是固定在三个主索节点上的，并可忽略其厚度，则可利用三节点的坐标可确定反射板所在小平面的法线方程、反射板的面积即三角形的中心，由入射光线与反射光线关于法线对称，入射光线是近似于从天体发射出的平行光，求过三角形中心的反射光线交馈源器直径为 1 米圆盘所在平面的交点，若该交点到 p 点的距离小于 0.5，则认为对应放射板为有效放射板，该版面积为有效面积。

三、 模型假设

- 1、已知圆孔与反射无关，本文忽略圆孔的影响。
- 2、可认为目标天体 S 距天眼无穷远，因此，假设信号直线传播，且天体 S 发射出的信号为平行信号。
- 3、忽略反射面板自身厚度。
- 4、假设三角反射面板为刚体，不发生形变。
- 5、主索节点仅沿径向移动，忽略切向位移的影响。

四、 符号说明

符号	符号说明
$card\theta$	主索节点坐标的集合个数
q_i^0	基准状态时主索节点 i 的位置
q_i	工作状态时主索节点 i 的位置

R	基准球面半径
F	两同心球面的半径
α	x 与 y 轴的夹角
β	CS 线与 xy 平面的夹角
$\cos \varphi$	天体在直角坐标系中 y 轴的方向余弦
θ	竖直剖面对称轴与照明区域边界夹角
\vec{n}	反射面板的法向量

五、模型的建立与求解

5.1 问题一：均方差最小的优化模型

在问题一中，已知目标天体 S 为待观测物体，且天体 S 位于基准球面正上方。为求解此刻的理想抛物面，本文进行单目标规划处理。

5.1.1 模型预处理

（一）基准球面的确定

根据附件 1 中的数据，可绘出反射面板链接主索节点的位置示意图。根据示意图，初步确定基准球面的大致形状。下图即为基准球面示意图。

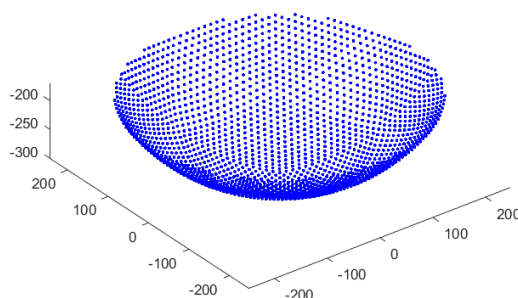


图 1 主索节点的空间分布示意图

（二）SC 直线及焦点 P 位置的确定

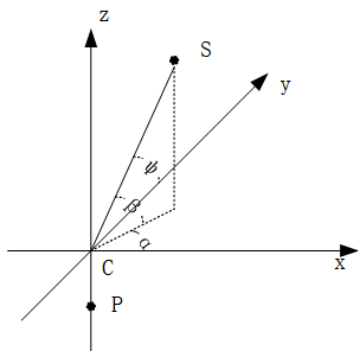


图 2 天体 S 方位角与仰角

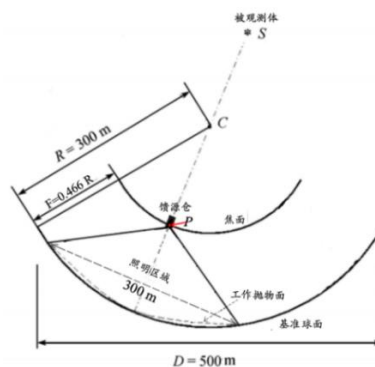


图 3 FAST 剖面示意图

设点 P 为 (x, y, z) ，已知存在角 α ， β 。其中， α 为 x 与 y 轴的夹角； β 为 CS 线与 xy 平面的夹角。 $\cos \varphi$ 为天体在直角坐标系中 y 轴的方向余弦。

已知，方向余弦的平方和等于一，得出以下公式：

$$\cos^2 \alpha + \cos^2 \varphi + \cos^2 \left(\frac{\pi}{2} - \beta \right) = 1 \quad (1)$$

变换公式，即可得，

$$\cos \varphi = \sqrt{1 - \cos^2 \alpha - \cos^2 \left(\frac{\pi}{2} - \beta \right)}$$

将 $P(x, y, z)$ 进行参数化处理，

$$\begin{cases} x = t \cos \alpha \\ y = t \cos \varphi \\ z = t \cos \left(\frac{\pi}{2} - \beta \right) \end{cases} \quad (2)$$

其中， t 为点 P 与点 C 的距离。且

$$t = -(R - F) \quad (3)$$

其中， $\begin{cases} R = 300m \\ F = 0.446R \end{cases}$ ，带入求得点 P 的坐标。

在第一问中，由于待测天体 S 位于 C 点的正上方，故可直接写出抛物线焦点 P 的坐标，即

$$(x_p, z_p) = (0, -(R - F))$$

因此，可求得抛物线顶点 Q 的坐标为

$$(x_Q, z_Q) = (0, -(R - F) - f)$$

抛物线准线方程为

$$z_{ij} = -(R - F) - 2f$$

抛物线方程为

$$(z - z_l)^2 = x^2 + (z - z_p)^2$$

带入可得

$$(z + (R - F) + 2f)^2 = x^2 + (z + (R - F))^2$$

化简得，

$$z = \frac{x^2}{4f} - f - (R - F)$$

(三) 将各主索节点化为极坐标

在现阶段求出的抛物线方程中，虽只需确定焦距即可解出理想抛物线方程，但由于在该方程中，仅可知抛物线上点的坐标，无法很好的与原基准球面上点的坐标对应，即无法表达出该抛物线上的点是从原半圆上某一点伸缩变化而来，故需将该方程化为极坐标方程。在极坐标下，由于点只径向移动即对应点的 θ 值相同，便于后续求解。

$$\begin{cases} x = \rho \cos \theta \\ z = \rho \sin \theta \end{cases} \quad (4)$$

带入抛物线方程，得

$$\frac{\rho^2 \cos^2 \theta}{4f} - \rho \sin \theta - f - (R - F) = 0 \quad (5)$$

(四) 促动器、下拉索，主索节点相对位置分析

理想状态下促动器低端、促动器顶端、主索节点应在一条直线上。但现实中

由于施工误差、重力等因素，会造成三点不在一条直线的情况，在场根据附件 2 和附件 3 的数据对基准球面状态下分析实际 3 点状态。

$$r_1 = \arccos \frac{|\overrightarrow{CL_i Z_i} \cdot \overrightarrow{CL_i C}|}{|\overrightarrow{CL_i Z_i}| \cdot |\overrightarrow{CL_i C}|}, r_2 = \arccos \frac{|\overrightarrow{CH_i Z_i} \cdot \overrightarrow{CH_i C}|}{|\overrightarrow{CH_i Z_i}| \cdot |\overrightarrow{CH_i C}|} \quad (6)$$

其中， Z_i 代表主索节点， C 为原点， CL_i 为促动器底端， CH_i 为促动器顶端。经计算所有夹角的均小于 10^{-5} ，因此在后续建立模型求解过程中，可假设促动器低端、促动器顶端、主索节点在一条直线上。

5.1.2 目标规划模型建立

当使用天眼观测目标天体 S 时，若 FAST 为光滑的抛物曲面，来自目标天体的所有平行电磁波均能反射汇聚到焦点 P 处， P 为馈源舱接收平面的中心。此时，认为该抛物面为理想抛物面。[2]

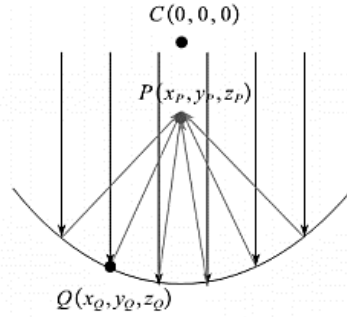


图 4 天体 S 信号发射示意图

已知抛物线的焦点，以焦距为自变量，可唯一确定一条抛物线，进而可绕 z 轴旋转成抛物曲面。当工作状态下主索节点位于某抛物面上，所有节点的伸缩量平方和最小时，则认为该抛物曲面为理想抛物面。基准状态与抛物面对应点极坐标下极径的差值即为伸缩量。

故目标函数为：

$$\min \sum_{t=1}^n (\rho_{0t} - \rho_{jt})^2$$

其中， ρ_{0t} 是基准态第 t 个点， ρ_{jt} 是工作状态第 t 个点。

5.1.3 模型求解

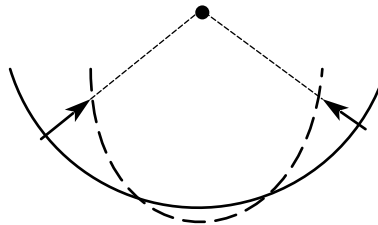


图 4 促动器伸缩示意图

(一) 遍历法求解步骤

Step1: 将所有主索节点的三维直角坐标 (x_i, y_i, z_i) 转化为其对应剖面上的二维极

坐标 (ρ_i, θ_i) ;

Step2:通过抛物线几何关系筛选出在 300m 照明区域内的节点, 标记为 1;

Step3:取 $\min = 100$, 对焦距 f 在 $[F-0.6, F+0.6]$ 的区间范围内以步长为 0.0001 进行迭代, 每次迭代使各点在保证 θ_i 不变前提下得到新的 ρ_i' ;

Step4: 计算各点处促动器的伸缩量 $\Delta = \rho_i - \rho_i'$, 再求所有点处伸缩量的平方总和, 与 \min 进行比较, 更新较小的 \min , 并记录此时的 f 和 ρ_i ;

Step5:遍历结束, 输出最新的 \min 和对应的 f , 最后将此时各点的二维极坐标还原为三维直角坐标。

5.1.4 模型结果

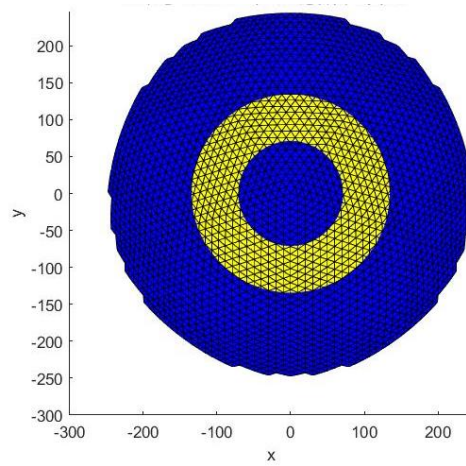


图 6 基准球面（蓝色）与理想抛物面（黄色）

利用 MTALAB 进行求解, 筛选出在 300m 照明区域内的主索节点共有 706 个, 最终得到理想抛物面的焦距 $f=140.4310\text{m}$, 焦点坐标为 $(0, 0, -160.4136)$; 顶点坐标为 $(0, 0, -300.8446)$, 较原先位置下移了 0.4446m . 故理想抛物曲面方程为:

$$z = \frac{x^2 + y^2}{561.724} - 300.8446$$

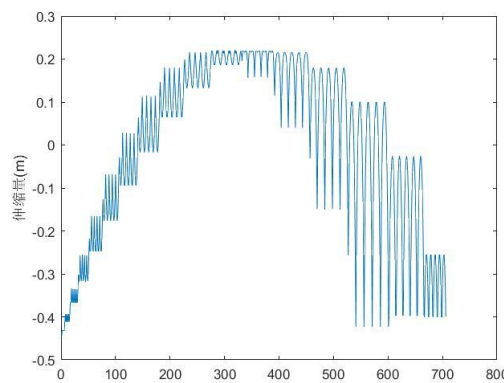


图 7 300 米照明区伸缩量范围变化图

最后再次检验 300m 照明区域内 706 个节点的伸缩量的变化范围为 $(-0.4446, 0.2197)$ 。如上图所示, 均未超过 $(-0.6, +0.6)$ 的约束范围。

5.1.5 模型精确度检验

对问题一的模型检验, 就是检验理想抛物面方程的准确性, FAST 的工作状态

时是将口径为 500 米的基准球面照射部分拉伸成抛物面，其余的基准球面部分可视为静止不动，为保证主索点间距离尽可能的小，抛物面口径为 300 米处切线斜率与基准球面口径为 300 米处切线斜率基本相同。

该过程中，所运用到的表达式有

$$z = \frac{x^2 + y^2}{561.724} - 300.8446 \quad (7)$$

$$\frac{dz}{dx} = 0.0036x \quad (8)$$

$$x^2 + z^2 = R^2 \quad (9)$$

令抛物面 $y = 0$ ，将其降维为二维平面，令 $x = 150$ ，代入得 $z = -260.7893$ ，其抛物线斜率为 0.5400；令基准球面 $y = 0$ ，也将其降维为二维平面，令 $x = 150$ ，代入得 $z = -260.27$ ，其圆形斜率为 0.5780。两斜率差值为 0.0380 小于 0.01，证明该结果检验合理。

5.2 问题二：旋转后理想抛物面的确定

对于问题二，题目要求一般状况下的理想抛物面。运用问题一求得的模型，对理想抛物曲面进行旋转即可求得一般情形。

5.2.1 旋转矩阵的求解步骤

依据右手螺旋进行方向判断，拇指指向坐标轴正方向，旋转方向与四指指向一致则判断为顺旋转。以原始位点 $H(x, y, z)$ 绕坐标轴旋转 γ 角为例，设其旋转后所得的坐标为点 $H'(x', y', z')$ 。则

(1) 点 H 绕 x 轴顺旋转时，

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = H_x \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (10)$$

(2) 点 H 绕 y 轴顺旋转时，

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos \gamma & 0 & \sin \gamma \\ 0 & 1 & 0 \\ -\sin \gamma & 0 & \cos \gamma \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = H_y \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (11)$$

(3) 点 H 绕 z 轴顺旋转时，

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = H_z \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (12)$$

5.2.2 模型准备——基准球面与抛物面的对应点

现已知基准球面各点的坐标及其编号，通过旋转坐标系后，代入模型一即可求出理想旋转抛物曲面。在求解工作状态下主索节点的坐标及伸缩量时，需知道原基准球面各个节点在抛物面上的理想坐标。由于节点仅通过径向移动，则在球坐标系下，根据原有节点计算出 θ, φ 值后，节点变化过程中，坐标的唯一变量为

ρ 值，即 ρ 值的变化量等于促动器的伸缩量。

$$\begin{cases} x = \rho \cdot \sin \varphi \cdot \cos \theta \\ y = \rho \cdot \sin \varphi \cdot \sin \theta, \theta \in [0, 2\pi], \varphi \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \\ z = \rho \cdot \cos \varphi \end{cases} \quad (13)$$

变换求解可得，

$$\begin{cases} \theta = \arctan \frac{y}{x} \\ \varphi = \arccos \frac{z}{\rho} \\ \rho = \sqrt{x^2 + y^2 + z^2} \end{cases} \quad (14)$$

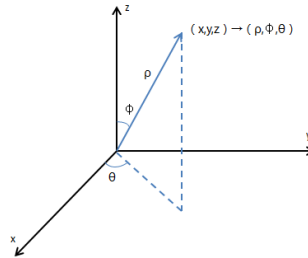


图 8 三维坐标系下节点坐标图

5.2.3 目标优化模型的建立

(一) 目标函数

第二问所用目标函数与第一问相同，即以工作状态下主索节点和基准态时主索节点的均方差为目标函数。

即，

$$\min \sum_{t=1}^n (\rho_{0t} - \rho_{jt})^2$$

(二) 约束条件

(1) 约束条件一：主索节点变化幅度的约束

由题目已知，在主索节点调节后，相邻节点之间的距离可能会发生微小变化，变化幅度不超过0.07%。设出两相邻主索节点 i, j 。其中， (x_i^0, y_i^0, z_i^0) 为基准态时主索节点 i 的位置， (x_i, y_i, z_i) 为工作状态时主索节点 i 的位置。

$$d_0 = \sqrt{(x_i^0 - x_j^0)^2 + (y_i^0 - y_j^0)^2 + (z_i^0 - z_j^0)^2}$$

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$$

相邻节点之间的距离可用 $d - d_0$ 表示。即

$$-0.07\%d_0 \leq d - d_0 \leq 0.07\%d_0$$

(2) 约束条件二：促动器径向伸缩范围的约束

在基准状态下，促动器顶端径向伸缩量为 0。促动器的径向伸缩为 $\rho_{0t} - \rho_{jt}$ ，范围为 $-0.6 \sim +0.6 \text{ m}$ 。

其中， ρ_{0t} 为基准球态第 t 个点球坐标的 ρ 值， ρ_{jt} 为工作态对应第 t 个点球坐标的 ρ 值。

综上，目标规划模型为，

$$\min \sum_{t=1}^n (\rho_{0t} - \rho_{jt})^2$$

$$s.t. \begin{cases} \theta = \arctan \frac{y}{x} \\ \varphi = \arccos \frac{z}{\rho} \\ \rho = \sqrt{x^2 + y^2 + z^2} \\ -0.07\% d_0 \leq d - d_0 \leq 0.07\% d_0 \\ |\rho_{0t} - \rho_{jt}| \leq 0.6, t = 0, 1, 2, \dots, n \end{cases}$$

5.2.4 模型的求解



图9 问题二求解流程图

(一) 旋转矩阵

在问题一的模型下，对所有坐标点进行旋转变换。由于 α 为 x 与 y 轴的夹角， β 为CS线与 xy 平面的夹角，因而，先将曲面绕 z 轴逆时针旋转 α 角，再将曲面绕 y 轴逆时针旋转 $\left(\frac{\pi}{2} - \beta\right)$ 角。设第一次旋转前点的坐标为 $(x_{i_1}, y_{i_1}, z_{i_1})$ ，第一次旋转所得（即第二次旋转前）的坐标为 $(x_{i_2}, y_{i_2}, z_{i_2})$ ，第二次旋转所得（即第三次旋转前）的坐标为 $(x_{i_3}, y_{i_3}, z_{i_3})$ 。

(1) 曲面绕 z 轴逆时针旋转 α 角

$$\begin{pmatrix} x_{i_1} \\ y_{i_1} \\ z_{i_1} \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{i_2} \\ y_{i_2} \\ z_{i_2} \end{pmatrix} \quad (15)$$

$$\begin{pmatrix} x_{i_2} \\ y_{i_2} \\ z_{i_2} \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{i_1} \\ y_{i_1} \\ z_{i_1} \end{pmatrix} \quad (16)$$

(2) 曲面绕 y 轴逆时针旋转 $\left(\frac{\pi}{2} - \beta\right)$ 角

$$\begin{pmatrix} x_{i_2} \\ y_{i_2} \\ z_{i_2} \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{2} - \beta\right) & 0 & \sin\left(\frac{\pi}{2} - \beta\right) \\ 0 & 1 & 0 \\ -\sin\left(\frac{\pi}{2} - \beta\right) & 0 & \cos\left(\frac{\pi}{2} - \beta\right) \end{pmatrix} \begin{pmatrix} x_{i_3} \\ y_{i_3} \\ z_{i_3} \end{pmatrix} \quad (17)$$

$$\begin{pmatrix} x_{i_3} \\ y_{i_3} \\ z_{i_3} \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{2} - \beta\right) & 0 & \sin\left(\frac{\pi}{2} - \beta\right) \\ 0 & 1 & 0 \\ -\sin\left(\frac{\pi}{2} - \beta\right) & 0 & \cos\left(\frac{\pi}{2} - \beta\right) \end{pmatrix} \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{i_1} \\ y_{i_1} \\ z_{i_1} \end{pmatrix} \quad (18)$$

在MATLAB求解过程中，难以实现直接对坐标轴进行旋转，因此，采用旋转所有坐标方向，先绕 z 轴顺旋转 $-\alpha$ ，再绕 y 轴顺旋转 $-\left(\frac{\pi}{2} - \beta\right)$ 。以此达到相同的效果。此时， α 和 β 换成弧度制。

再通过对300m照明区域的旋转，筛选出此时照明区域范围内的主索节点共有687个，接下来将优先考虑对范围内的节点进行调节来贴合理想抛物面，若难以满足则还需考虑对范围外的节点进行调节。

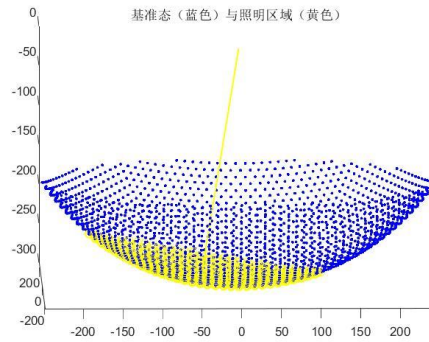


图 10 照明区域旋转后的示意图

(二) 二分法

已知理想旋转曲面，可计算照射范围内基准球面上各点到对应理想抛物面上对应点的距离，采用遍历的思想，逐步增加各点下拉索的变化长度，但照射范围内主索点个数庞大，且单独移动某一个点，很容易便超出相邻主索点间距离变化的约束，计算量庞大。这里可利用二分法的思想进行求解。

Step1: 计算出基准球面上各点延径向到理想旋转曲面的距离。

Step2: 判断目标点延径向在基准球面的上方还是下方，如果目标点的3坐标值均小于原始点的3坐标值，则该目标点在原始点的下方取 $g=1$ ，反之则 $g=0$ ，以此确定移动方向；

Step3: 将整体所有点移动与目标点距离的二分之一；

Step4: 检验在该情况下是否满足相邻点距离的约束和和促动器伸缩量的约束，若符合约束则进行 Step5，否则进行 Step6；

Step5: 将此时的坐标更新为原始点数据，跳转至 Step1；

Step6: 将此时的坐标更新为目标点数据，跳转至 Step1；

将该循环重复一定次数，即可逼近理想旋转曲面，若所有点整体移动的效果不好，还可将争光照明区域分多个小块重复上述过程，确定较为满意的移动方案后，再重新组合成新的原始点数据，代入 Step1 迭代多次。

5.2.5 模型的结果

通过矩阵的旋转，初步求解得在 $\alpha = 36.795^\circ$, $\beta = 78.169^\circ$ 时的理想抛物面，如

下图 10、11 所示。

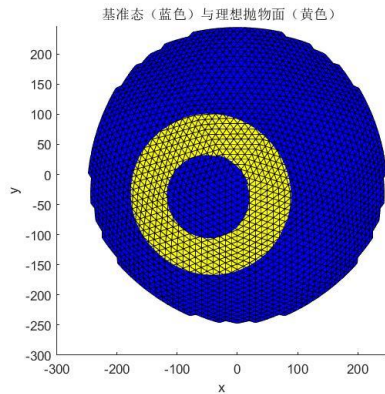


图 11 基准态与理想抛物面（初次）

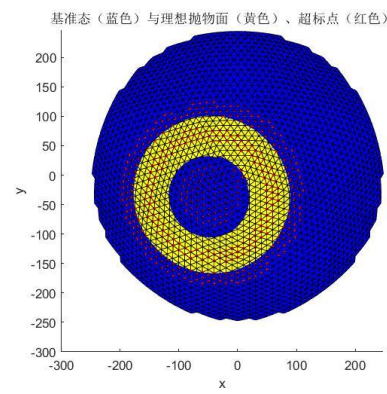


图 12 超出相邻限制的节点标记

求解得到此时理想抛物面的焦点坐标为（-26.337， -19.699， -157.006），顶点坐标为（-49.393， -36.944， -294.454）。

理想抛物面的函数表达式为

$$z = 0.001719x^2 + 0.001736y^2 + 0.03832x - 0.02876y - 300$$

同时，在 result.xlsx 中，保存了求得理想抛物面的顶点坐标及仅考虑对 300m 范围内的节点调节后的反射面 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。

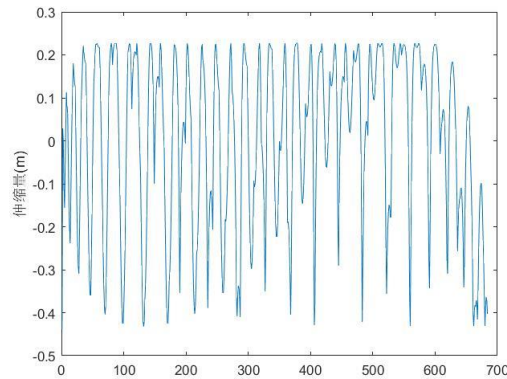


图 13 各促动器伸缩量结果

表 1 主索节点编号、位置坐标、各促动器伸缩量结果

节点编号	X 坐标（米）	Y 坐标（米）	Z 坐标（米）	伸缩量（米）
A0	0.000	0.000	-300.838	-0.438
B1	6.107	8.406	-300.192	0.028
C1	9.883	-3.211	-300.224	-0.003
D1	0.000	-10.395	-300.337	-0.117
...
E427	-187.195	156.703	-175.050	0.000
E428	-181.263	165.587	-173.104	0.000
E429	-175.262	173.878	-171.143	0.000
E430	-168.969	181.996	-169.019	0.000

题目要求相邻主索节点之间距离变化幅度不超过 0.07%，促动器径向伸缩范围在（-0.6,0.6）间波动。通过对所求结果检查发现，所有点都满足径向伸缩要求，但 300m 范围内不满足相邻距离约束的超标点占比近一半，对此，对模型结果进行迭代优化处理，考虑对 300m 区域外的主索点进行调节。在进行模型结果优化后，满足相邻主索节点之间距离变化幅度不超过 0.07% 约束的点占比为 81.34%，且径向伸缩依旧满足要求。

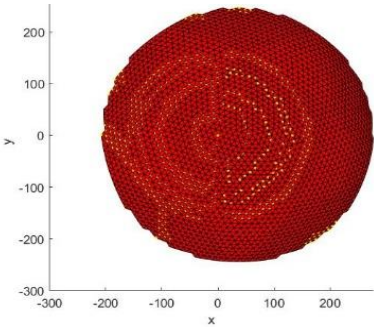


图 14 优化后的超标点

表 2 优化后主索节点编号、位置坐标、各促动器伸缩量结果

节点编号	X 坐标（米）	Y 坐标（米）	Z 坐标（米）	伸缩量（米）
A0	0.000	0.000	-300.017	-0.071
B1	6.100	8.396	-299.827	0.021
C1	9.873	-3.208	-299.931	-0.011
D1	0.000	-10.380	-299.892	-0.124
...
E427	-203.273	129.227	-179.378	0.073
E428	-201.912	143.722	-169.623	0.073
E429	-198.299	138.299	-178.177	0.073
E430	-192.851	147.569	-176.714	0.145

5.3 问题三：馈源舱的接收比的求解

问题三要求调节后馈源舱的接收比，由于题目已知馈源舱的直径，可大致求得接受信号的范围。即使入射光线与 P 点所在切面的交点与基准态该点对应的位置距离不超过 0.5m 为有效。

本问将在第二问旋转坐标系下求解，我们已将各反射面板调整到最接近理想抛物面的坐标下，但各反射面板不是所有点都可以将电磁波反射至馈源舱。每块反射板上可能部分点可以将电磁波反射至馈源舱，而剩下点则不能。本问将采用蒙特卡洛随机模拟的方法，在每块反射板上随机取 10000 个点代表该板的反射波在馈源舱的接受情况。

5.3.1 模型的准备

(一) 反射面板法向量的确定

由附件 1 可知各主索节点的具体坐标。在同一反射面板中，令三角形反射面板的三个节点为节点 $J_1(x_1, y_1, z_1)$, $J_2(x_2, y_2, z_2)$, $J_3(x_3, y_3, z_3)$ 。由于目标天体 S 发射的电磁波可视为平行电磁波，因而可以假设每束电磁波都经由一块反射面板。则本文需求解出每块反射面板的法线，平行电磁波过法线与板面的交点反射至接收器的中心 P 点处^[3]。

因此，建立表达式求解反射面板的垂线（法线）。

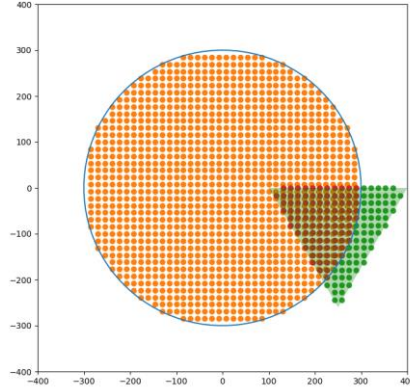


图 15 照明区域边界示意图

在反射面板内， $\overrightarrow{J_1J_2}$, $\overrightarrow{J_1J_3}$ 为平面内两个不共线的向量，

$$\overrightarrow{J_1J_2} = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

$$\overrightarrow{J_1J_3} = (x_3 - x_1, y_3 - y_1, z_3 - z_1)$$

当空间中某一直线与平面内两条相交直线都垂直时，该直线与平面垂直。由此可以推得， $\overrightarrow{J_1J_2}$ 与 $\overrightarrow{J_1J_3}$ 的数乘垂直于反射面板，即反射面板的法向量可表示为 $\overrightarrow{J_1J_2}$ 与 $\overrightarrow{J_1J_3}$ 的数乘。

$$\vec{n} = \overrightarrow{J_1J_2} \times \overrightarrow{J_1J_3} = \begin{vmatrix} i & j & k \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \end{vmatrix} = a\vec{i} + b\vec{j} + c\vec{k} = (u, v, w) \quad (19)$$

$$u = (y_2 - y_1) \cdot (z_3 - z_1) - (y_3 - y_1) \cdot (z_2 - z_1)$$

$$v = (z_2 - z_1) \cdot (x_3 - x_1) - (z_3 - z_1) \cdot (x_2 - x_1)$$

$$w = (x_2 - x_1) \cdot (y_3 - y_1) - (x_3 - x_1) \cdot (y_2 - y_1)$$

为了便于后续的简化求解，将法向量转化为单位法向量 \vec{N} ，即

$$\vec{N} = \frac{\vec{n}}{\sqrt{u^2 + v^2 + w^2}} \quad (20)$$

(二) 寻找每块反射板调整后的球心

本问假设反射板都是基准球面上的一部分，在基准状态下，4300 块反射板共同组成半径为 300.4 球面的一部分。通过对主索节点的调节，每块反射板对应

的球心位置发生变化，为便于寻找反射板任意点的法向量，先求出每板的球心，随后任一点到球心即为该曲面的法向量，后续可求出该点的反射光。

三点确定外接圆的圆心

外接圆的圆心到三点的距离都相等，可以列出下列方程：

$$\begin{cases} (x_1 - x_2)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2 = (x_2 - x_0)^2 + (y_2 - y_0)^2 + (z_2 - z_0)^2 \\ (x_2 - x_0)^2 + (y_2 - y_0)^2 + (z_2 - z_0)^2 = (x_3 - x_0)^2 + (y_3 - y_0)^2 + (z_3 - z_0)^2 \end{cases}$$

求解得

$$\begin{aligned} x_p &= \frac{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix} \begin{vmatrix} \frac{x_1^2 + y_1^2 + z_1^2}{2} & 1 & z_1 \\ \frac{x_2^2 + y_2^2 + z_2^2}{2} & 1 & z_2 \\ \frac{x_3^2 + y_3^2 + z_3^2}{2} & 1 & z_3 \end{vmatrix} + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \begin{vmatrix} \frac{x_1^2 + y_1^2 + z_1^2}{2} & y_1 & 1 \\ \frac{x_2^2 + y_2^2 + z_2^2}{2} & y_2 & 1 \\ \frac{x_3^2 + y_3^2 + z_3^2}{2} & y_3 & 1 \end{vmatrix}}{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}^2} \\ y_p &= \frac{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \begin{vmatrix} 1 & \frac{x_1^2 + y_1^2 + z_1^2}{2} & z_1 \\ 1 & \frac{x_2^2 + y_2^2 + z_2^2}{2} & z_2 \\ 1 & \frac{x_3^2 + y_3^2 + z_3^2}{2} & z_3 \end{vmatrix} + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix} \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \begin{vmatrix} x_1 & \frac{x_1^2 + y_1^2 + z_1^2}{2} & 1 \\ x_2 & \frac{x_2^2 + y_2^2 + z_2^2}{2} & 1 \\ x_3 & \frac{x_3^2 + y_3^2 + z_3^2}{2} & 1 \end{vmatrix}}{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}^2} \\ z_p &= \frac{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \begin{vmatrix} 1 & y_1 & \frac{x_1^2 + y_1^2 + z_1^2}{2} \\ 1 & y_2 & \frac{x_2^2 + y_2^2 + z_2^2}{2} \\ 1 & y_3 & \frac{x_3^2 + y_3^2 + z_3^2}{2} \end{vmatrix} + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix} \begin{vmatrix} x_1 & 1 & \frac{x_1^2 + y_1^2 + z_1^2}{2} \\ x_2 & 1 & \frac{x_2^2 + y_2^2 + z_2^2}{2} \\ x_3 & 1 & \frac{x_3^2 + y_3^2 + z_3^2}{2} \end{vmatrix} + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}}{\begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix}^2 + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}^2} \end{aligned}$$

外接圆的半径为：任意点到圆心的距离 r 。

过三角形外接圆圆心，以三角形法线方向为方位向量可写出直线方程：

$$\frac{x - x_0}{x_f} = \frac{y - y_0}{y_f} = \frac{z - z_0}{z_f} \quad (21)$$

其中， (x_f, y_f, z_f) 前文中计算求出的空间中反射面板的法向量； (x_0, y_0, z_0) 为外接圆圆心。

通过分析可知，球心在该直线上，并距离圆心有：

$$d = \sqrt{R^2 - r^2} \quad (22)$$

其中 R 为球的半径， $R=300.4$ 。

可计算出该直线上距离圆心 d 的有两个点，经分析，取 z 值较大的点。

$$\begin{cases} x = x_0 + x_f \cdot t \\ y = y_0 + y_f \cdot t \\ z = z_0 + z_f \cdot t \end{cases}$$

将公式变换可得，

$$\begin{cases} (x - x_0)^2 = (x_f \cdot t)^2 \\ (y - y_0)^2 = (y_f \cdot t)^2 \\ (z - z_0)^2 = (z_f \cdot t)^2 \end{cases}$$

推知，

$$d^2 = (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = t(x_f^2 + y_f^2 + z_f^2) \quad (23)$$

$$t = \pm \sqrt{\frac{d^2}{x_f^2 + y_f^2 + z_f^2}} \quad (24)$$

(三) 反射向量的求解

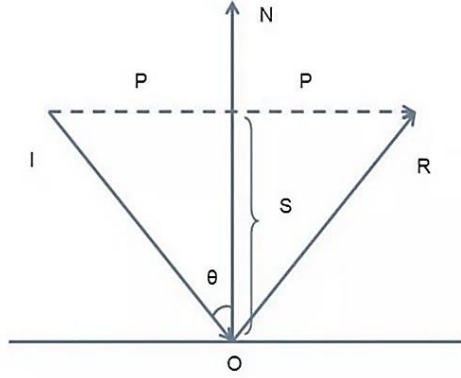


图 16 平面中入射向量与反射向量示意图

设入射光线向量 \vec{I} 和反射平面的法向量 \vec{N} 之间的夹角为 θ ，连接 \vec{I} 的始端和末端，有

$$\vec{R} = 2\vec{P} - \vec{I}$$

此时，问题变为如何求出点 P。设入射点 O 到 P 与 N 的交点的向量为 \vec{S} ，则有

$$\vec{P} = \vec{I} + \vec{S}$$

现在，问题变为如何求取向量 \vec{S} ，由于向量 \vec{S} 与向量 \vec{N} 方向相反，则向量 \vec{S} 即向量 $-\vec{N}$ 在向量 \vec{N} 上的投影。根据向量的投影公式有

$$\vec{S} = -\frac{\vec{I} \cdot \vec{N}}{|\vec{N}|^2} \vec{N}$$

由于向量 \vec{N} 为单位向量，简化上述表达式可得

$$\vec{S} = -(\vec{I} \cdot \vec{N}) \vec{N}$$

由此求得，

$$\vec{R} = \vec{I} - 2(\vec{I} \cdot \vec{N}) \vec{N} \quad (25)$$

(四) 各板随机取点模型

判断由三个坐标点围成的反射面板为曲面，可近似认定其为球面状。如图 17 所示，对该曲面进行投影可得到近似的投影三角区域，则在投影区域内上取任意一点，总能在曲面上找到一一对应的反射点^[4]。

由附件 1 可知各主索节点的具体坐标。在同一反射面板中，令三角形反射面板的三个节点为节点 $J_1(x_1, y_1, z_1)$, $J_2(x_2, y_2, z_2)$, $J_3(x_3, y_3, z_3)$ ，即每块板抽象成三角形^[5]。三角形区域内任意一点 $E(x_e, y_e, z_e)$ 与节点 J_1 的连线做直线，必交以 J_2, J_3 为端点的边于点 $F(x_f, y_f, z_f)$ 。

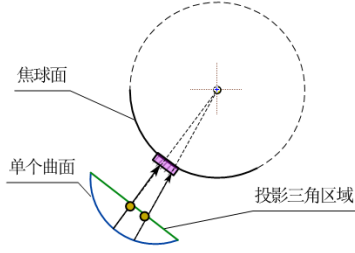


图 17 三角形与反射板对应点关系示意图

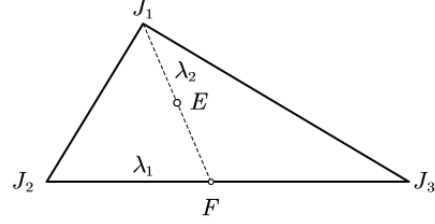


图 18 三角形反射面板各点示意图

由图易知，F 点坐标可表示为：

$$\lambda_1 = \frac{J_2 F}{J_2 J_3}$$

$$\begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} = (\lambda_1, 1 - \lambda_1) \begin{pmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix} = \begin{pmatrix} \lambda_1 x_2 + (1 - \lambda_1) x_3 \\ \lambda_1 y_2 + (1 - \lambda_1) y_3 \\ \lambda_1 z_2 + (1 - \lambda_1) z_3 \end{pmatrix}$$

E 点坐标可表示为：

$$\lambda_2 = \frac{J_1 E}{J_1 F}$$

$$\begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix} = (\lambda_2, 1 - \lambda_2) \begin{pmatrix} x_1 & y_1 & z_1 \\ \lambda_1 x_2 + (1 - \lambda_1) x_3 & \lambda_1 y_2 + (1 - \lambda_1) y_3 & \lambda_1 z_2 + (1 - \lambda_1) z_3 \end{pmatrix}$$

因此，通过随机取 (0,1) 之间的任意两个数，可确定平面上随机一点。套用本问前述公式，即可求出 E 点的对应点 E'。又由于本问是旋转坐标系使原点到待观测天体方向为 z 轴的情况下，则该 E 点的入射向量 \vec{I} 为

$$\vec{I} = \vec{OE} = (x_e, y_e, z_e) \quad (26)$$

(四) 三角形面积的公式

1) 在二维平面中，

当三个点 A, B, C 的坐标分别为 $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$ 时，以 A, B, C 为顶点的三角形面积为 $S = (x_1 y_2 - x_1 y_3 + x_2 y_3 - x_2 y_1 + x_3 y_1 - x_3 y_2)$ 。设三个点 A, B, C 的坐标分别为 $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$ 。 A, B, C 三点可围成一个三角形。AC 与 AB 边的夹角为 $\angle A$ 。则向量 $\vec{AB} = (x_2 - x_1, y_2 - y_1)$ ，向量 $\vec{AC} = (x_3 - x_1, y_3 - y_1)$ 。

令向量 $\vec{AB} = \vec{a}$ ，向量 $\vec{AC} = \vec{b}$ ，由向量运算法则可得，

$$|\vec{a} \cdot \vec{b}| = |\vec{a}| \cdot |\vec{b}| \cdot |\cos A|$$

变换公式可得,

$$\cos A = \frac{|\vec{a} \cdot \vec{b}|}{|\vec{a}| \cdot |\vec{b}|}, \quad \sin A = \frac{\sqrt{(|\vec{a}| \cdot |\vec{b}|)^2 - (|\vec{a} \cdot \vec{b}|)^2}}{|\vec{a}| \cdot |\vec{b}|}$$

由此可得, 三角形的面积为

$$S = |\vec{a}| \cdot |\vec{b}| \cdot \sin A = \sqrt{(|\vec{a}| \cdot |\vec{b}|)^2 - (|\vec{a} \cdot \vec{b}|)^2} S$$

又因为

$$\vec{a} \cdot \vec{b} = (x_2 - x_1) \cdot (x_3 - x_1) + (y_2 - y_1) \cdot (y_3 - y_1)$$

综上, 推导出三角形的面积公式, 表示为,

$$S = (x_1 y_2 - x_1 y_3 + x_2 y_3 - x_2 y_1 + x_3 y_1 - x_3 y_2) \quad (27)$$

2) 推广至三维空间中,

(1) 三维中的两点之间距离之式为:

$$d = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2 + (z_0 - z_1)^2} \quad (28)$$

(2) 叉乘。叉乘在定义上有: 两个向量进行叉乘得到的是一个向量, 方向垂直于这两个向量构成的平面, 大小等于这两个向量组成的平行四边形的面积。

在直角坐标系 $[0; i, j, k]$ 中, i, j, k 分别为 X 轴, Y 轴, Z 轴上向量的单位向量。

设 $P_0(0, 0, 0)$, $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$ 。

从原点出发, 向量 $\overrightarrow{P_0 P_1}$ 可简记为 \vec{P}_1 , 向量 $\overrightarrow{P_0 P_2}$ 可简记为 \vec{P}_2 。则有,

$$\vec{P}_1 \times \vec{P}_2 = \begin{vmatrix} i & j & k \\ x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \end{vmatrix}$$

展开, 可得:

$$\begin{aligned} \vec{P}_1 \times \vec{P}_2 &= i y_1 z_2 + j z_1 x_2 + k x_1 y_2 - k y_1 x_2 - j x_1 z_2 - i z_1 y_2 \\ &= (y_1 z_2 - y_2 z_1) i + (x_2 z_1 - x_1 z_2) j + (x_1 y_2 - x_2 y_1) k \end{aligned}$$

按照规定, 有, 单位向量的模为 1。因而可以得出, 叉积的模为:

$$\begin{aligned} |\vec{P}_1 \times \vec{P}_2| &= y_1 z_2 - y_2 z_1 + x_2 z_1 - x_1 z_2 + x_1 y_2 - x_2 y_1 \\ &= (y_1 z_2 + x_2 z_1 + x_1 y_2) - (y_2 z_1 + x_1 z_2 + x_2 y_1) \end{aligned} \quad (29)$$

(3) 设三角形的三个顶点分别为 $A(x_0, y_0, z_0)$, $B(x_1, y_1, z_1)$, $C(x_2, y_2, z_2)$ 。将三角形的两条边 AB 和 AC 看成是向量。以 A 为原点, 进行坐标平移, 得到向量

\vec{B} , \vec{C} 。其中, 向量 $\vec{B} = (x_1 - x_0, y_1 - y_0, z_1 - z_0)$, 向量 $\vec{C} = (x_2 - x_0, y_2 - y_0, z_2 - z_0)$ 。

① 在三维的情况下, 直接代入公式, 可得向量 \vec{B} 和向量 \vec{C} 叉乘结果的模为:

$$\begin{aligned} |\vec{B} \times \vec{C}| &= ((y_1 - y_0) \cdot (z_2 - z_0) + (z_1 - z_0) \cdot (x_2 - x_0) + (x_1 - x_0) \cdot (y_2 - y_0)) - ((y_2 - y_0) \cdot (z_1 - z_0) + (z_2 - z_0) \cdot (x_1 - x_0) + (x_2 - x_0) \cdot (y_1 - y_0)) \\ &= \begin{vmatrix} 1 & 1 & 1 \\ x_1 - x_0 & y_1 - y_0 & z_1 - z_0 \\ x_2 - x_0 & y_2 - y_0 & z_2 - z_0 \end{vmatrix} \end{aligned} \quad (30)$$

它的一半即为所要求的三角形面积 S。即,

$$S = \frac{1}{2} \cdot \begin{vmatrix} 1 & 1 & 1 \\ x_1 - x_0 & y_1 - y_0 & z_1 - z_0 \\ x_2 - x_0 & y_2 - y_0 & z_2 - z_0 \end{vmatrix} \quad (31)$$

5.3.2 模型的建立: 空间几何模型的建立

由于馈源舱接收平面的中心在与基准球面同心的一个球面（焦面）上移动, 且馈源舱接收信号的有效区域为直径 1 米的中心圆盘。设馈源舱接收平面的中心点坐标为 $P(x_P, y_P, z_P)$, 即可表示出 P 点出焦面的切面方程。

P 点处切面方程为

$$\cos \alpha (x - x_P) + \cos \varphi (y - y_P) + \cos \left(\frac{\pi}{2} - \beta \right) (z - z_P) = 0 \quad (32)$$

接着, 判断有效区域能否接受反射面板反射出的信号。

已知三角形三个顶点坐标 $J_1(x_1, y_1, z_1)$, $J_2(x_2, y_2, z_2)$, $J_3(x_3, y_3, z_3)$, 即可求解得三角形中心点坐标 $M(x_M, y_M, z_M)$, 且三角形的法线过中心点 M。

$$\begin{aligned} x_M &= \frac{x_1 + x_2 + x_3}{3} \\ y_M &= \frac{y_1 + y_2 + y_3}{3} \\ z_M &= \frac{z_1 + z_2 + z_3}{3} \end{aligned}$$

除此以外, 还需判断过 M 点的反射线 \vec{R} 是否穿过有效区域。

设反射线 $\vec{R} = (x_R, y_R, z_R)$

则可以得出反射直线的参数方程,

$$\begin{cases} x = x_M + x_R t \\ y = y_M + y_R t \\ z = z_M + z_R t \end{cases}$$

为使求解更为简便, 尽可能消去未知数。将表达式进行联立, 得

$$t = \frac{(x_P - x_M) \cdot \cos \alpha + (y_P - y_M) \cdot \cos \varphi + (z_P - z_M) \cdot \cos \left(\frac{\pi}{2} - \beta \right)}{\cos \alpha \cdot x_R + \cos \varphi \cdot y_R + \cos \left(\frac{\pi}{2} - \beta \right) \cdot z_R} \quad (33)$$

将所求得的 t 带入表达式即可求解出入射光线与 P 点所在切面的交点

$E(x_E, y_E, z_E)$ 。

由于馈源舱接受信号的有效区域为以点 P 为圆心、1m 为直径坐落在切面上的圆，利用该条件进行约束，表达式为

$$(x_E - x_P)^2 + (y_E - y_P)^2 + (z_E - z_P)^2 \leq (0.5)^2 \quad (75)$$

统计各反射板上随机 10000 点中共有 m 个点可反射至馈源舱的有效区域内，则各板的有效面积为：

$$\frac{m}{10000} \times S$$

馈源舱的接收比模型为：

$$\frac{\sum_{i=1}^n \frac{m}{10000} S_i}{\sum_{i=1}^n S_i}$$

5.3.3 模型的求解

Step1: 计算单个三角面元面积，近似认为每个投影面上的三角面元面积为单个反射面板面积；

Step2: 通过入射角和反射角的关系，在已知三角面元摆放位置状态和射入信号的路径时，通过物理光学知识，得出射出信号的路径并判断路径是否通过圆盘；

Step3: 面积反射可认为是多个射线反射的集合，所有打在曲面上能进行下一步反射的入射信号的集合称为当前角度下总有效面积；

Step4: 根据馈源舱吸收反射信号的总强度，将其数值等价于一束束平行的信号射线打在一平面上的面积，将此面积称为该曲面的等效面积；

Step5: 在当前角度下，通过曲面等效面积与总有效面积作比，即可得出该角度下该曲面作为反射源时馈源舱的接收面；

Step6: 在上述步骤的基础上，分别求出调节后的工作抛物面及基准反射球面对信号的接收比。

5.3.4 模型的结果及分析

在观测天体 S 位于 $\alpha = 36.795^\circ$, $\beta = 78.169^\circ$ 时，300m 照明区域范围内，计算基准球面授，总有效面积为 $70098.8481m^2$ ，理想基准球面反射的有效面积为 $520.1335m^2$ ，接收比为 0.742%；计算调整优化后的抛物面时，总有效面积为 $622736.7138m^2$ ，抛物面反射的有效面积为 $8201.443m^2$ ，接收比为 1.317%，提升了 0.575%。

六、 模型评价与推广

6.1 模型的优点

- 1、本文在对问题进行求解的过程中，对模型的算法进行优化，使所求得理想抛物面更为精确。
- 2、本文通过模型的得到结果，并能联系全文不同模型所得结果，合理的分析，反复推测，最终验证模型的可行性。
- 3、本文所建立的目标规划模型规律简单、易懂，而且能够运用该模型以及模型

求解算法得出比较理想的抛物面，说明了模型的实用性和算法有效性

6.2 模型的缺点

- 1、由于实际生活中球体很难将平行光束汇聚到同一点，因此，本文对抛物面近似体的选取易造成一定误差。
- 2、本文所使用遍历法、迭代法编程用时较长，复杂程度较高，不够简明扼要。
- 3、因缺少相关数据，未考虑促动器纵向移动情况，所建模型具有一定的局限性，在实际问题的应用中，易造成误差。

6.3 模型的推广

- 1、本模型中最主要的是反射面板理想抛物面的确定及主动反射面板的调节，可运用于光伏发电板的安装调节，针对光伏阵列最大功率点的跟踪，计算出每时刻点最大功率所处的曲面，进而对发电板进行方向上的调节，以便产生更多的电量。
- 2、在通信工程中基于抛物面解决天线馈源的横偏及散射问题，例如大型广告牌的接受天线不便转动，天线波瓣难以测量，可将其天线面轮廓分为多个抛物面，进行调整。

七、 参考文献

- [1] 南仁东. 500 米球反射面射电望远镜 FAST. 中国科学, G 辑物理学力学天文学, 2005.
- [2] 钱宏亮, 范峰, 沈世钊, 王启明. FAST 反射面支撑结构整体索网分析[J]. 哈尔滨工业大学学报, 2005.
- [3] 张志伟. 基于 CAN 总线技术的天文望远镜索网促动控制系统的研制. 硕士学位论文. 沈阳:东北大学信息科学与工程学院, 2006.
- [4] 朱丽春. 500 米口径球面射电望远镜(FAST)主动反射面整网变形控制[J]. 科研信息化技术与应用, 2012.
- [5] 杜敬利, 保宏, 杨东武, 崔传贞. 索网主动反射面的形状精度调整研究[J]. 工程力学, 2012.

八、 附录

附件清单

1. 数据预处理准备工作
2. 问题一主程序
3. 问题二主程序
4. 问题二检验
5. 问题二优化过程
6. 问题三主程序 1
7. 问题三主程序 2
8. 程序运行所需 m 函数文件

1. 数据预处理准备工作 (Prepare. m)

```

%找出三角板、节点、节点坐标的调用关系，即 MeshGrid、Node_ID、
NodeLoc
clear;clc;close all;
%载入节点数据，[num,txt]=xlsread('xxx')
[NodeLoc,NodeId1]=xlsread('附件 1.csv');
[NodeLoc2,NodeId2]=xlsread('附件 2.csv');
[~,NodeId3]=xlsread('附件 3.csv');
R=abs(min(NodeLoc(:,3)));%记录离原点（球心）最远的坐标点
f=0.466;%焦距
NodeNum=size(NodeLoc,1);%记录主索节点个数，size(n,1)返回行数
MeshNum=size(NodeId3,1)-1;%记录反射面板个数
MeshGrid=zeros(MeshNum,3);%预分配理想矩阵与反射面板中节点一一对
应
for ii=1:NodeNum%乱序节点遍历
    NodeId1Name=NodeId1{ii+1,1};%对应节点编号
    Row=NodeId1Name(1)-'A'+1;%首字母编号转化为数字编号，A为1，B
    为2...
    Col=str2double(NodeId1Name(2:end))+1;
    Node_ID(Row,Col)=ii;%使 NodeLoc 中每一行坐标对应其节点编号
end
%使三角面板（MeshGrid）与节点编号（Node_ID）一三对应
for ii=1:MeshNum%三角反射面板遍历
    for jj=1:3
        NodeId1Name=NodeId3{ii+1,jj};%对应三角面板中某一个节点
        Row=NodeId1Name(1)-'A'+1;
        Col=str2double(NodeId1Name(2:end))+1;
        MeshGrid(ii,jj)=Node_ID(Row,Col);
    end
end
save NodeInfo NodeLoc NodeLoc2 NodeId1 NodeId2 NodeId3...
Node_ID MeshGrid NodeNum R f

```

2. 问题一主程序 (Q1.m)

```

clear;clc;close all;
load NodeInfo
%将节点三维直角坐标转化成相应的二维极坐标
Loc_ji=zeros(NodeNum,3);
for ii=1:NodeNum
    Loc_ji(ii,1)=atan(norm(NodeLoc(ii,1:2))/abs(NodeLoc(ii,3)));%θ,
    表示节点与球心间的连线和 Z 坐标轴的夹角
    Loc_ji(ii,2)=norm(NodeLoc(ii,:));%ρ, % 表示节点与球心间的距离
end
%更新坐标
[detail,LocNew,ff]=Update(NodeLoc);
%绘图
figure('color','w');

```

```

figure(1);clf;hold on;
PlotModel(LocNew,MeshGrid,'y');
PlotModel(NodeLoc,MeshGrid,'b');
title('基准态（蓝色）与理想抛物面（黄色）')
% 300m 内伸缩范围
detall2=detall1;
for ii=1:NodeNum
    if Loc_ji(ii,3)==0
        detall2(ii,:)=0;
    end
end
detall2(all(detall2==0,2),:)=[];%删除全 0 行
figure('color','w');
x=1:length(detall2);
plot(x,detall2')
ylabel('伸缩量(m)')
%输出
format short
disp('搜寻到焦距')
ff
disp('搜寻到顶点 z 坐标为')
peak=LocNew(1,3)

```

3. 问题 2 主程序 (Q2.m)

```

clc;clear;close all;
load NodeInfo
LocOld=NodeLoc;
%旋转坐标系
NodeLoc=xz(NodeLoc);
%绘图
figure('color','w');
figure(1);
scatter3(LocOld(:,1),LocOld(:,2),LocOld(:,3),'r.')
axis([-320 320 -320 320 -320 0]);
axis equal
title('原坐标系')
figure('color','w');
figure(2)
scatter3(NodeLoc(:,1),NodeLoc(:,2),NodeLoc(:,3),'b.');
axis([-320 320 -320 320 -320 0]);
axis equal
title('旋转坐标系后')
%旋转后就可以套用问一的模型了
[detall,LocNew,ff,Loc_ji]=Update(NodeLoc);
%进行坐标系的还原
LocNew=hy(LocNew);
peak=[0,0,-300.8446]; %抛物线顶点坐标转化

```

```

peak=hy(peak);
%剔除 300m 外的点
LocNew2=LocNew;
for ii=1:NodeNum
    if Loc_ji(ii,3)==0
        LocNew2(ii,:)=0;
    end
end
LocNew2(all(LocNew2==0,2),:)=[];%删除全 0 行
%绘图
figure(3)
figure('color','w');
scatter3(LocOld(:,1),LocOld(:,2),LocOld(:,3),'b.')
hold on
scatter3(LocNew2(:,1),LocNew2(:,2),LocNew2(:,3),'y')
axis([-320 320 -320 320 -320 0]);
hold on;
tu=[peak;[0,0,0]];
plot3(tu(:,1),tu(:,2),tu(:,3),'y','Linewidth',1.2);
axis equal
grid off
title('基准态（蓝色）与照明区域（黄色）')
figure('color','w');clf;hold on;
PlotModel(LocNew,MeshGrid,'y');
PlotModel(LocOld,MeshGrid,'b');
title('基准态（蓝色）与理想抛物面（黄色）')
%300m 内伸缩范围
detall2=detall;
for ii=1:NodeNum
    if Loc_ji(ii,3)==0
        detall2(ii,:)=0;
    end
end
detall2(all(detall2==0,2),:)=[];%删除全 0 行
figure('color','w');
x=1:length(detall2);
plot(x,detall2);
ylabel('伸缩量(m)')

```

4. 问题 2 检验 (Jury.m)

```

for ii=1:MeshNum*3
    if jury2(ii)==0
        count=count+1;
    end
end
rate=count/(MeshNum*3);%所有点的达标率
%记录超标的点
trial=LocNew;

```



```

Chao=zeros(MeshNum,1);
for ii=1:MeshNum
    if jury2(ii,1)==1
        Chao(MeshGrid(ii,1),1)=1;
        Chao(MeshGrid(ii,2),1)=1;
    end
    if jury2(ii,2)==1
        Chao(MeshGrid(ii,1),1)=1;
        Chao(MeshGrid(ii,3),1)=1;
    end
    if jury2(ii,3)==1
        Chao(MeshGrid(ii,2),1)=1;
        Chao(MeshGrid(ii,3),1)=1;
    end
end
%保留不达标的点，删除达标的点
for ii=1:NodeNum
    if Chao(ii)==0
        trial(ii,:)=0;
    end
end
trial(all(trial==0,2),:)=[];
%绘制区域图
figure('color','w');clf;hold on;
PlotModel(LocNew,MeshGrid,'y');
PlotModel(LocOld,MeshGrid,'b');
scatter3(trial(:,1),trial(:,2),trial(:,3),'r');
title('基准态（蓝色）与理想抛物面（黄色）、超标点（红色）')
%300m 范围内的达标率
rate2=1-length(trial)/sum(detall~=0);
%检验伸缩量是否超标 0.6m
jury1=zeros(MeshNum,3);
jury1(abs(detall)-0.6>0)=1;%超出约束
jury1(abs(detall)-0.6>0)=1;%没有超出约束

```

5. 问题二优化过程（Q2_Opt.m）

```

clc,clear;
load NodeInfo
%save MeshGrid MeshGrid
load MeshGrid
figure(1)
plot3(NodeLoc(:,1),NodeLoc(:,2),NodeLoc(:,3),'r')
a=-36.795/180*pi;
b=-(90-78.169)/180*pi;
Hz=[cos(a),-sin(a),0;sin(a),cos(a),0;0,0,1];
Hy=[cos(b),0,sin(b);0,1,0;-sin(b),0,cos(b)];
%Hx=[1,0,0;0,cos(b),-sin(b);0,sin(b),cos(b)];
[m,n]=size(NodeLoc);

```

```

XZ=zeros(m,n);
for i=1:m
    XZ(i,:)=Hz*NodeLoc(i,:);
    XZ(i,:)=Hy*XZ(i,:);%XZ 旋转后基准态的坐标
end
figure(2)
plot3(XZ(:,1),XZ(:,2),XZ(:,3),'b')
xlabel('x 轴'); ylabel('y 轴'); zlabel('z 轴'); % 加上坐标轴的标
签
hold on
ff=140.431;
figure(2)
plot3(XZ(:,1),XZ(:,2),XZ(:,3),'b')
hold on
x=-150:1:150;
y=-150:1:150;
[x,y]=meshgrid(x,y);
z=(x.^2+y.^2)/(4*ff)-300.8446;
plot3(x,y,z,'r')
xlabel('x 轴'); ylabel('y 轴'); zlabel('z 轴'); % 加上坐标轴的标
签
%此时顶点坐标为(0,0,-3000.8446)
t5=[0,0,-300.8446]';
a=36.795/180*pi;%还原
b=(90-78.169)/180*pi;
Hz=[cos(a),-sin(a),0;sin(a),cos(a),0;0,0,1];
Hy=[cos(b),0,sin(b);0,1,0;-sin(b),0,cos(b)];
tr=Hy*t5;
tr=Hz*tr;%还原后的顶点坐标
JZ=XZ;%XZ 旋转后坐标赋值给基准态
%load Loc_ji
%ID=[1:1:2226]';
%ID(Loc_ji(:,3)==0,:)=[];%保留点对应的编号
%JZ(Loc_ji(:,3)==0,:)=[];%将不在 300 米内的点删去
[m,n]=size(JZ);
%XZ 旋转后基准态的坐标转化为球坐标,先求第 5 步
JZ_Q=zeros(m,n);%基准态球坐标
for i=1:m
    x=XZ(i,1);y=XZ(i,2);z=XZ(i,3);
    cita=atan(y/x);
    rou=sqrt(x^2+y^2+z^2);
    fai=acos(z/rou);
    JZ_Q(i,:)=[cita,fai,rou];
end
LX=zeros(m,3);%理想抛物面坐标
LX_Q=zeros(m,3);%理想抛物面坐标
%在抛物面  $z=(x.^2+y.^2)/(4*ff)-300.8446$ ;找到对应节点

```

```

for i=1:m%筛选出 300 内的点后，把 2226 改成点的个数
    cita=JZ_Q(i,1);fai=JZ_Q(i,2);
    rou=Lrou(fai);%旋转抛物面 rou 值的确定
    x=rou*sin(fai)*cos(cita);
    y=rou*sin(fai)*sin(cita);
    z=rou*cos(fai);
    %z1=(x^2+y^2)/561.724-300.8446%检验
    LX(i,:)=[x,y,z];
    LX_Q(i,:)=[cita,fai,rou];
end
format short g
%伸缩相差量 6、7 向上为负
cha=zeros(m,1);
load Net
B=zeros(4300,3);
load bc
load MeshGrid
for t=1:100
    B=zeros(4300,3);
    for i=1:m%伸缩相差量 6、7 向上为负
        cha(i)=JZ_Q(i,3)-LX_Q(i,3);
        if cha(i)>0.6%伸
            cha(i)=0.6;
            LX_Q(i,3)=JZ_Q(i,3)-cha(i);%更新理想球坐标
            LX(i,:)=qiu2zhi(LX_Q(i,1),LX_Q(i,2),LX_Q(i,3));
        end
        if cha(i)<-0.6%缩
            cha(i)=-0.6;
            LX_Q(i,3)=JZ_Q(i,3)-cha(i);%更新理想球坐标
            LX(i,:)=qiu2zhi(LX_Q(i,1),LX_Q(i,2),LX_Q(i,3));
        end
    end
end
%现更新所有板的理想坐标
%LX_totle=XZ;
%for j=1:m
%    LX_totle(ID(j),:)=LX(j,:);
%end
%MeshGrid_2=MeshGrid;
%ID_max=max(ID);
%MeshGrid_2(MeshGrid_2(:,1)>ID_max,:)=[];
bc_i=solvebc(LX);
num=zeros(2226,1);%该点是不符合约束，1 不符合
num_c=zeros(2226,1);%该点是符合约束，1 符合
B=zeros(4300,3);
for i=1:4300
    for j=1:3
        temp1=abs(bc_i(i,j)-bc(i,j));%距离差的绝对值
        temp2=bc(i,j)*0.0007;
    end
end

```

```

        if temp1<temp2
            B(i,j)=0;%满足条件等于 0
        else
            B(i,j)=1;%不满足条件等于 1
            %找出是哪两个点的边不符合条件
            temp_h=i;%MeshGrid 中的第 i 行
            if j==1
                temp_y=[1,2];%MeshGrid 中的第 1,2 列
            end
            if j==2
                temp_y=[1,3];%MeshGrid 中的第 1,3 列
            end
            if j==3
                temp_y=[2,3];%MeshGrid 中的第 2,3 列
            end
            temp_num=MeshGrid(temp_h,temp_y(1));
            num(temp_num)=1;
            temp_num=MeshGrid(temp_h,temp_y(2));
            num(temp_num)=1;
        end
    end
end
SU_b=sum(B(:));
SU_d=sum(num);
num(num==0)=2;
num_c(num==2)=1;%该点是符合约束，1 符合
num(num==2)=0;%将 num 还原
trial=zeros(2226,3);%初始化不满足条件点的坐标
%迭代过程。更新 LX 矩阵。LX 与 JZ 的中间点存入 LX 矩阵中
for i=1:2226%更新不满足的点
    if num(i)~=0 %不满足条件更新 LX 矩阵。LX 与 JZ 的中间点存入 LX
        矩阵中
            trial(i,:)=LX(i,:);%记录不满足条件点的坐标，后面画图
            LX(i,:)=0.5*(LX(i,:)+JZ(i,:));%直角坐标
            LX_Q(i,3)=0.5*(LX_Q(i,3)+JZ_Q(i,3));
        end
    end
end

for i=1:2226
    if num_c(i)==1
        %先更新满足约束点的坐标。对应 JZ 中的点
        JZ(i,:)=LX(i,:);
        JZ_Q(i,3)=LX_Q(i,3);
        %JZ(i,:)=0.9*LX(i,:)+0.1*JZ(i,:);
        %JZ_Q(i,3)=0.9*LX_Q(i,3)+0.1*JZ_Q(i,3);
        %JZ_Q(i,:)=zhi2qiu(JZ(i,1),JZ(i,2),JZ(i,3)); %球坐标
    end
end

```

```

end
end
PlotModel(XZ,MeshGrid,'b');
hold on
scatter3(trial(:,1),trial(:,2),trial(:,3),'y.');
```

title('基准态（蓝色）、超标点（黄色）')

6. 问题三主程序 1 (Q3_1.m)

```

clear all;clc;close all;
load JZ
load XZ
load NodeInfo
%NodeLoc=JZ;%求工作状态的
%NodeLoc=XZ;%求刚旋转后的
%save MeshGrid MeshGrid
load MeshGrid
%备注将坐标系旋转后，300 口径范围内的点代入
for ii=1:MeshNum
    if (Loc_ji(MeshGrid(ii,1),3)==0)||(Loc_ji(MeshGrid(ii,2),3)==0)...
        ||(Loc_ji(MeshGrid(ii,3),3)==0)
        MeshGrid(ii,:)=0;
    end
end
MeshGrid(all(MeshGrid==0,2),:)=[];%删除全零行
MeshNum=length(MeshGrid);
format long g
R=300;
F=0.466*R;
Z=-(R-F);-300.4-Z+0.6;
P=[0,0,-(R-F)];
n=10000;%每块板找点数
triangle=zeros(3,3);%三角板顶点坐标，一行为一个坐标
sum1=0;sum2=0;%1 有效面积，2 总面积
for i=1:MeshNum%三角板个数 4300 块
    %此次循环的 triangle,%相连接的三个点
    %测试用的一个三角板的坐标，每行表示一个点的坐标
    triangle(1,:)=LocNew1(MeshGrid(i,1),:);%三角板第一个点坐标
    triangle(2,:)=LocNew1(MeshGrid(i,2),:);
    triangle(3,:)=LocNew1(MeshGrid(i,3),:);
    %triangle=[0,0,-300.4;6.1078,8.407,-300.2202;9.8827,-3.211,-
300.2202];
    count_m=0;
    p1=triangle(1,:);
    p2=triangle(2,:);
    p3=triangle(3,:);
    fdir=cross((p1-p2),(p3-p2));%三角板的法向量
    if fdir(3)<0 %保证法向量指向圆心
```

```

        fdir=-fdir;
    end
    mj=area(p1,p2,p3);
    %[f,mj,mdloc]=meshrefcal(triangle(1,:),...
    triangle(2,:),triangle(3,:));
    for i=1:n
        r1=rand(1);r2=rand(1);
        E=zeros(3,1);%E 点
        F=zeros(3,1);%F 点
        F=[r1,1-r1]*triangle(2:3,:);
        E=[r2,1-r2]*[triangle(1,:);F];
        Out=E-2*dot(E,fdir)*fdir;%fdir 法向量, Out 反射光
        xp=P(1);yp=P(2);zp=P(3);
        xe=E(1);ye=E(2);ze=E(3);
        xr=Out(1);yr=Out(2);zr=Out(3);
        %t=((xp-xe)+(yp-ye)+(zp-ze))/(xr+yr+zr);公式错误
        t=(Z-ze)/zr;
        M=E+t*Out;%焦点平面焦点求解公式错误
        xm=xe+t*xr;ym=ye+t*yr;zm=ze+t*zr;
        temp=(xm-xp)^2+(ym-yp)^2+(zm-zp)^2;
        if temp<=0.5^2
            count_m=count_m+1;
        end
    end
    sum1=sum1+count_m/n*mj;
    sum2=sum2+mj;
end
y=sum1/sum2 %接收比

```

7. 问题三主程序 2 (Q3_2.m)

```

%进行坐标系的还原,找焦点坐标
clear,clc;
load NodeInfo
load NodeInfo2
load LocNew1
MeshNum=4300;
number=0;
for ii=1:MeshNum
    if (Loc_ji(MeshGrid(ii,1),3)==0)||(Loc_ji(MeshGrid(ii,2),3)==0)...
    ||(Loc_ji(MeshGrid(ii,3),3)==0)
        MeshGrid(ii,:)=[0,0,0];
    end
end
MeshGrid(all(MeshGrid==0,2),:)=[];%删除全零行
MeshNum=length(MeshGrid);
F=0.466*300.4;
focus=[0,0,-300.4+F];
focus=hy(focus);

```

```

for ii=1:MeshNum
    n=1e4;
    nmd1=unifrnd(0,1,[1,n]);
    nmd2=unifrnd(0,1,[1,n]);
    p1=LocOld(MeshGrid(ii,1),:);
    p2=LocOld(MeshGrid(ii,2),:);
    p3=LocOld(MeshGrid(ii,3),:);
    for jj=1:1e4
        [XX,YY,ZZ]=Carol(p1,p2,p3,nmd1(jj),nmd2(jj));%随机点 e 的坐标,也是定点
        V=[XX,YY,ZZ];
        t=(norm(focus)^2-sum([XX,YY,ZZ].*focus))/sum(V.*focus);
        x=XX+V(1)*t;
        y=YY+V(2)*t;
        z=ZZ+V(3)*t;
        D=norm([focus(1)-x, focus(2)-y, focus(3)-z]);%交点与焦点距离
        if D<0.5
            number=number+1;
        end
    end
end
rate=number/(n*MeshNum)

```

8. 程序运行所需 m 函数文件

Update.m

```

function [detail,LocNew,ff,Loc_ji]=Update(NodeLoc)
NodeNum=size(NodeLoc,1);%记录主索节点个数, size(n,1) 返回行数
R=abs(min(NodeLoc(:,3)));%记录基准抛物面最低点的 z 值
F=0.466*R;
min_all=9e10;%预赋值所有伸缩量的平方和的开根, 用于比小
LocNew=zeros(NodeNum,3);
rou=zeros(NodeNum,1);
    %将节点三维直角坐标转化成相应的二维极坐标
Loc_ji=zeros(NodeNum,3);
for ii=1:NodeNum
    Loc_ji(ii,1)=atan(norm(NodeLoc(ii,1:2))/abs(NodeLoc(ii,3)));%θ, 表示节点与球心间的连线和 Z 坐标轴的夹角
    Loc_ji(ii,2)=norm(NodeLoc(ii,:));%ρ, % 表示节点与球心间的距离
end

%焦距遍历
for f=135:0.001:145
%找出 300m 口径内的点
z=150^2/(4*f)-f-(R-f);%边界点的 z 坐标
xita=asin(150/sqrt(150^2+z^2));%推导公式
%通过 θ 与抛物线方程找到新的节点坐标 (θ, ρ), 再还原成三维直角坐标
detail=zeros(NodeNum,1);%伸缩量

```

```

for ii=1:NodeNum
    if Loc_ji(ii,1)>xita
        Loc_ji(ii,3)=0;%判断是否在 300m 内, 0 为不在
    else
        Loc_ji(ii,3)=1;
    end
    a=cos(pi/2-Loc_ji(ii,1))^2/(4*f);%推导公式的求解
    b=sin(pi/2-Loc_ji(ii,1));
    c=-f-(R-F);
    rou(ii)=(-b+sqrt(b^2-4*a*c))/(2*a);%只保留正解
    rou(1)=-c;%A0 节点在正下方, x、y 坐标为 0, 不能用二次方程公式求解
    detal(ii)=Loc_ji(ii,2)-rou(ii);
end
detal(Loc_ji(:,3)==0)=0;%不考虑不在 300m 内的点
if norm(detal)<min_all
    min_all=norm(detal);%判断条件
    for ii=1:NodeNum
        LocNew(ii,:)=(Loc_ji(ii,2)-
detal(ii))/Loc_ji(ii,2).*NodeLoc(ii,:);%更新坐标, 伸缩方向各坐标成比例
        detal1=detal;
        ff=f;
    end
end
end
end
end

```

XZ. m

```

%坐标系旋转
%先绕 z 轴顺旋转 alpha=36.795°(右手螺旋)
%再绕 y 轴顺旋转 (90-bata) 度
function [XZ]=xz(NodeLoc)
    alpha=36.795;%单位为角度
    bata=78.169;
    [m,n]=size(NodeLoc);%返回行数、列数
    XZ=zeros(m,n);
    a=-alpha/180*pi;
    b=-(90-bata)/180*pi;
    Hz=[cos(a),-sin(a),0;sin(a),cos(a),0;0,0,1];%旋转矩阵
    Hy=[cos(b),0,sin(b);0,1,0;-sin(b),0,cos(b)];
    %Hx=[1,0,0;0,cos(b),-sin(b);0,sin(b),cos(b)];
    for i=1:m
        XZ(i,:)=Hz*NodeLoc(i,:);
        XZ(i,:)=Hy*XZ(i,:);
    end
end
end

```

hy. m


```

%坐标系还原
%再绕 y 轴逆旋转 (90-bata) 度
%先绕 z 轴逆旋转 alpha=36.795°(右手螺旋)
function [HY]=hy(LocNew)
    alpha=36.795;          %单位为角度
    bata=78.169;
    [m,n]=size(LocNew);%返回行数、列数
    HY=zeros(m,n);
    a=alpha/180*pi;%还原
    b=(90-bata)/180*pi;
    Hz=[cos(a),-sin(a),0;sin(a),cos(a),0;0,0,1];
    Hy=[cos(b),0,sin(b);0,1,0;-sin(b),0,cos(b)];
    for i=1:m
        HY(i,:)=Hy*LocNew(i,:);
        HY(i,:)=Hz*HY(i,:);
    end
end

```

PlotModel

```

function PlotModel(coord,facet,c)
%顶点和切面的数量
nvert=size(coord,1);
ntria=size(facet,1);
%所有顶点的坐标
xpts = coord(:,1);
ypts = coord(:,2);
zpts = coord(:,3);
%所有切面的坐标
node1 = facet(:,1);
node2 = facet(:,2);
node3 = facet(:,3);
%储存 vind 数组中每个顶点的节点
for i = 1:ntria
    pts = [node1(i) node2(i) node3(i)];
    vind(i,:) = pts;
end
x = xpts; y = ypts; z = zpts;
% define X,Y,Z arrays and plot them
%定义 X,Y,Z 数组并绘画他们
for i = 1:ntria
    X = [x(vind(i,1)) x(vind(i,2)) x(vind(i,3)) x(vind(i,1))];
    Y = [y(vind(i,1)) y(vind(i,2)) y(vind(i,3)) y(vind(i,1))];
    Z = [z(vind(i,1)) z(vind(i,2)) z(vind(i,3)) z(vind(i,1))];
    fill3(X,Y,Z,c)
    if i == 1
        hold on
    end
end
axis square

```

```

xlabel('x'); ylabel('y'); zlabel('z');
%这是为了避免任意一维上的最大值和最小值都为 0
xmax = max(xpts); xmin = min(xpts);
ymax = max(ypts); ymin = min(ypts);
zmax = max(zpts); zmin = min(zpts);
dmax = max([xmax ymax zmax]);
dmin = min([xmin ymin zmin]);
%这是为了避免任意一维上的最大值和最小值都为 0
xmax = dmax; ymax = dmax; zmax = dmax;
xmin = dmin; ymin = dmin; zmin = dmin;
axis([xmin, xmax, ymin, ymax, zmin, zmax]);

```

Neighbor.m

```

%节点的相邻矩阵
%load MeshGrid
Nei_t=zeros(NodeNum,18);
Nei=zeros(NodeNum,7);
Nei(:,1)=1:NodeNum;%自己的节点编号
for ii=1:NodeNum%排节点编号
    m=1;
    for jj=1:MeshNum%找节点编号
        for t=1:3
            if MeshGrid(jj,t)==ii
                Nei_t(ii,m:m+2)=MeshGrid(jj,:);
                m=m+3;
            end
        end
    end
end
for ii=1:NodeNum
    k=unique(Nei_t(ii,:));%删除重复项
    k(k==0)=[]; %删除多余的 0
    k(k==ii)=[]; %删除自己的节点编号
    a=zeros(1,6-length(k));%用于补全空缺，方便合并
    Nei(ii,2:end)=[k,a];
end

```