

“FAST”主动反射面的形状调节

摘 要

本文主要分析研究“FAST”主动反射面的控制调节参数，并针对抛物面几何性质，运用反射定律建立了相关参数调节模型。

对于问题一，根据基准球面坐标和馈源舱位置建立坐标系，考虑到体系的旋转对称性，对其取 xOz 截面分析。综合题目信息可得到主索节点的可行空间，并使较多的理想主索节点位置处于该空间内，即求解理想抛物面使其在可行空间的面积最大。通过假设截面抛物线的顶点位置，并推导出符合要求的抛物线簇，建立以抛物线顶点坐标为参数的非线性规划模型，使用内点法解得最优抛物线并旋

转得到理想抛物面方程为 $z = \frac{x^2 + y^2 - 168181.000558960}{560.502062679404}$ 。

对于问题二，针对已知角度的斜射问题。采用三维笛卡尔坐标变换，将斜射问题通过旋转，变换为正上方入射问题。确定 300 米口径内，692 个反射面板的序号。利用问题一解出的理想抛物面，即可得到在可行空间促动器伸缩的最佳长度。利用逆变换得到顶点坐标 $(-49.269, -36.851, -293.716)$ 。并以主索节点编号 A0 为例，得到调节后其位置坐标为 $(0, 0, -300.207)$ ，对应促动器的伸缩量为 0.193。详细结果置于附件 'result.xlsx' 中。

对于问题三，独立分析每个反射面板。依据反射信号直线传播假设，由反射定律，导出反射向量，得到反射射线方程。定义反射面板的有效区域为射线射入馈源舱的有效截面。遍历反射面板，计算有效面积。分析结果发现当抛物面焦点在馈源舱球面附近移动时可增加信号射入有效区域的数量，并以此对问题一中的理想抛物线加以修正。将馈源舱接收最多反射信号作为目标函数的多目标优化模型，利用局部搜索求解新的最优参数，得到新的理想抛物面方程。针对主索节点和反射面板两种统计对象，分别求解出修改后的接收比为 61.85% 和 27.57%，原基准面的接收比为 21.39% 和 6.38%。将修改前后的情况进行比较，得出基于点与基于面调节后接收比分别较原来提升了 332.29% 和 432.13%。

通过模拟工程误差，对促动器的伸缩量增加高斯噪声，以检验系统对自动控制精度的稳定性。

对结果作图后发现，存在较多反射信号距馈源舱接收区域较近。考虑到工业可行性，进一步分析改变馈源舱有效面积对接收比变化的影响。发现口径在 1.0 至 2.0 米范围内，增大有效面积对接收比有明显改善，并综合接收器制作工艺，建议多馈源舱接收器协同收集信号以大幅提高观测能力。

关键词：非线性目标优化 内点法 三维笛卡尔坐标变换 折射定律 工程误差

一、问题重述

1.1 问题背景

FAST 望远镜由主动反射面、信号接收系统（馈源舱）以及相关的控制、测量和支承系统组成。主动反射面是 FAST 望远镜重要的自主创新部分,其性能将直接关系到整个望远镜的对于天体电磁波接收性能。

主动反射面由反射面板、主索网、促动器、下拉索及支撑结构等主要部件构成。反射面板、主索网结构及其连接的直观示意图如下：

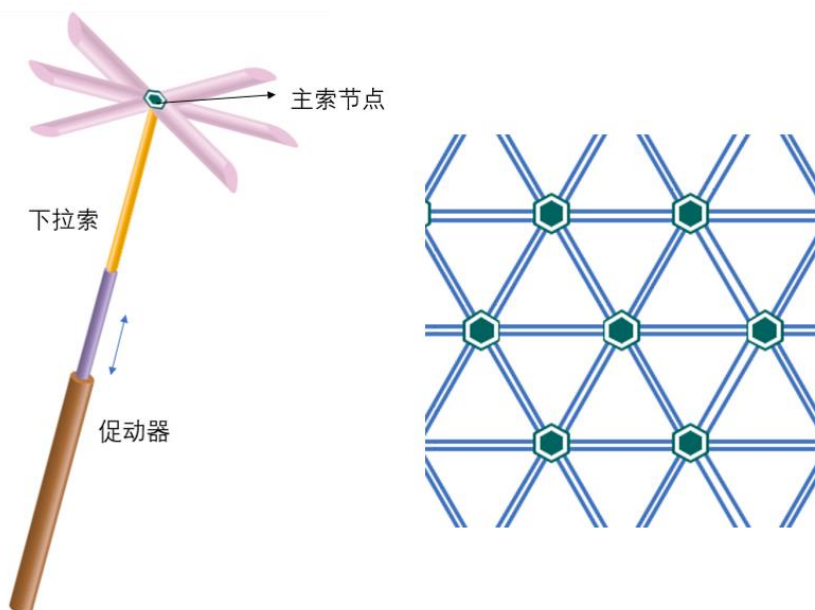


图 1 反射面板、主索网结构及其连接示意图

主动反射面分为基准态和工作态两个状态。给定基准态以及工作态反射面的半径、口径参数值，以及馈源舱接收信号时工作抛物面调整方式。将反射面调整为工作抛物面的过程通过下拉索与促动器配合完成。下拉索长度固定，促动器沿基球面径向伸缩来完成下拉索的调节，从而调节反射面板的位置，最终形成工作抛物面。

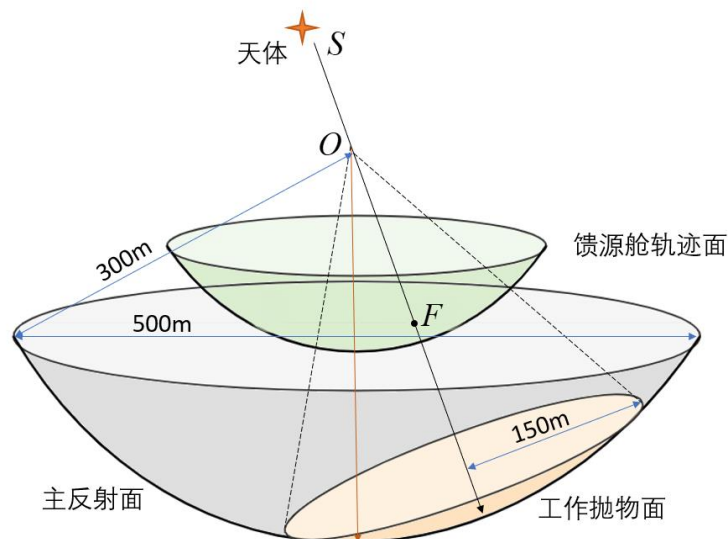


图 2 工作态抛物面及馈源舱轨迹面示意图

本文主要研究将反射面调节为工作抛物面的过程，在反射面调节的约束下，确定一个理想抛物面，通过调节促动器的径向伸缩量，将反射面调节为工作抛物面，使得该工作面尽量贴近理想抛物面，以获得天体电磁波经反射面反射后的最佳接收效果。

1.2 问题提出

(1) 问题一：当待观测天体 S 位于基准球面的正上方，即 $\alpha = 0^\circ$ ， $\beta = 90^\circ$ 时，结合考虑反射面板调节因素，确定理想抛物面。

(2) 问题二：当观测天体 S 位于 $\alpha = 36.795^\circ$ ， $\beta = 78.169^\circ$ 时，确定理想抛物面。建立反射面板调节模型，调节相关促动器的伸缩量，使反射面尽量贴近该理想抛物面。

(3) 问题三：基于问题二的反射面调节方案，计算调节后馈源舱的接收比，即馈源舱有效区域接收到的反射信号与 300 米口径内反射面的反射信号之比。

二、问题分析

2.1 问题一的分析^[2]

对于问题一理想抛物面方程的求解，常用方法有理论推导取特殊值法和列写约束条件最优化法。由于本题中特殊位置并不明显，因此采用后者。促动器伸缩长度的范围和两主索节点间几何关联约束了主索节点的活动空间，即为本题的约束条件。理想抛物面为近似的旋转抛物面，因此可以对体系进行 xO_z 截面，使问题得到简化。根据已给出的基准球面与焦面相关的几何参量，可以列出平面坐标系下抛物线簇的参数方程，进而可以对抛物面簇进行表示与求解。将求解理想抛

物面有效部分面积最大的问题转化为求解使得理想抛物面上有最多主索节点的问题，并选择内点法对非线性规划问题进行求解，从而得到理想抛物面的参数。

2.2 问题二的分析

对于问题二中斜射平行电磁波的情况，可以参考问题一的求解方法。但考虑到该情况下坐标不易表示和计算，因此选择三维笛卡尔坐标变换将体系进行旋转至天体仰角与方向角均为 0（即 $\alpha = \beta = 0$ ）的位置。由于多步绕轴旋转的顺序不同会造成转后的位置不同，因此在求解过程中须严格遵守绕轴旋转的顺序。得到斜射的理想抛物面方程后，令主索节点在约束条件下最大程度贴近理想抛物面。经分析，本题的约束条件为反射面口径需满足小于 300 米且促动器伸缩长度范围需满足在 -0.6~+0.6 米之间。求解得到每个促动器的伸缩长度、调节后反射面 300 米口径内的主索节点编号以及位置坐标。将问题一中理想抛物面的顶点坐标作对应的逆三维笛卡尔坐标变换，即可得到斜射平行光对应理想抛物面的顶点坐标。

2.3 问题三的分析

对于问题三基准反射球面与调节后馈源舱的接收比的计算问题，在绝对理想状态时，可以进行连续曲面积分分析。但为了使结果具有实际工程意义，我们将问题离散化求解。结合反射定律以及空间几何计算技巧，得到反射电磁波方向，进而可以得出每个三角反射面板的反射特性。初步计算后得到的接收比较小，不能很好的符合实际情况。进一步分析，发现焦点不在接收器球面时可能使更多反射信号处于接收器接收范围内。进而对问题一进行修正，修改对焦点位置的约束，以最多的反射信号可处于接收器接收范围内为目标函数，求解得到了更有实际意义的理想抛物面方程。将修正后的数据重新代入，得到接收比，并与基准球面接收比进行对照。最后对出现的结果进行分析。

三、模型假设

1. 假设面板无孔且忽略漫反射，此时可将电磁波信号及反射信号视为直线传播。
2. 由于观测目标天体的距离接近无限远，假设观测到的电磁波为平行信号。
3. 在促动器径向伸缩范围 -0.6~+0.6 米之间，忽略工程误差，即促动器可被精确控制。
4. 假设馈源舱功能完备，可以消除像差、光程差等因素造成的影响。
5. 由于径向伸缩所范围 ± 0.6 米和相邻节点之间的距离与其变化的最大幅度 0.07% 相比过于微小^[1]，因此忽略相邻节点之间距离的变化。

四、符号说明

符号	符号说明
f	焦面与球面半径的半径差
a	理想抛物面顶点 A 的 z 轴坐标
α	天体 S 的方位角
β	天体 S 的仰角
$ x_1 - x_2 $	$Q_1 Q_2$ 最小水平距离
k	促动器径向伸缩长度
α_i	下拉索移动的方向向量
ray_{in}	入射电磁波
ray_{re}	反射电磁波
N	300 米口径工作抛物面内的三角反射面板总数
n_{valid}	300 米口径工作抛物面内参与有效反射的三角反射面板的数目
n_0	基准反射球面上参与有效反射的三角反射面板的数目
P_r	反射面调节后馈源舱的接收比
P_{r0}	基准反射球面的接收比

五、模型准备

为直观表示主索节点的位置，根据附件 1 中提供的主索节点的坐标和编号将主索节点绘制成如下三维空间坐标图：

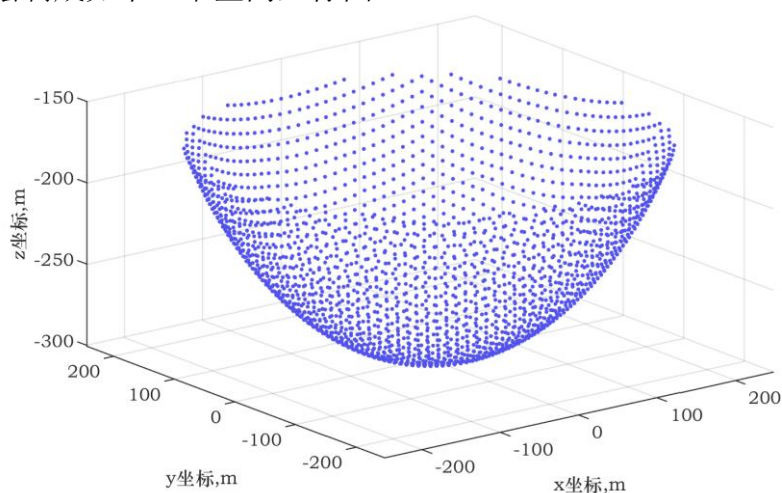


图 3 主索点坐标图

六、模型的建立与求解

为了直观展示模型的建立和求解思路，绘制了如下求解步骤流程图：

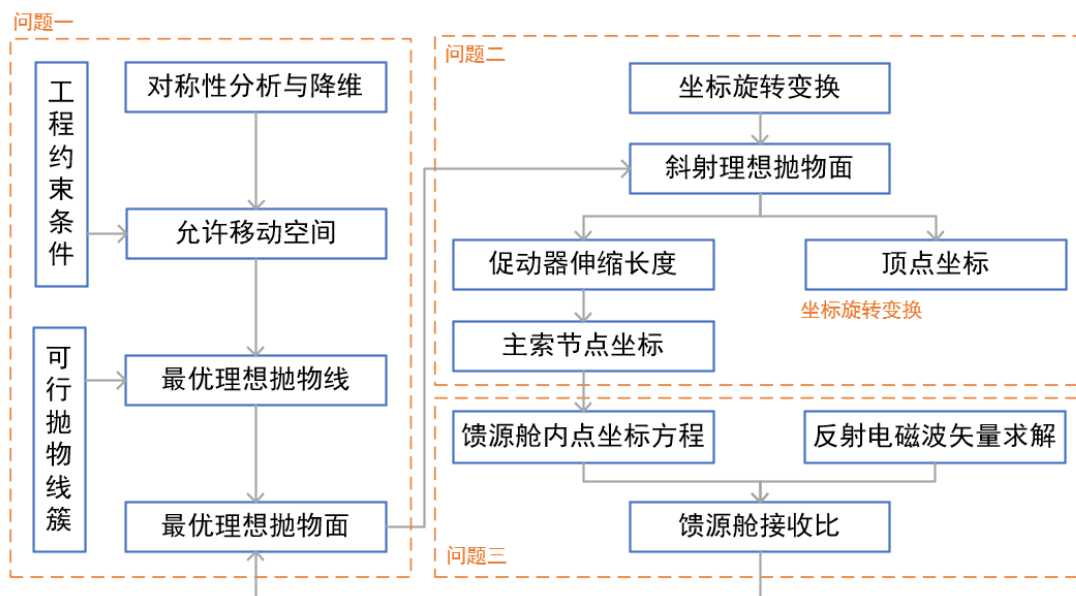


图 4 模型建立与求解总流程图

6.1 问题一的模型建立与求解

6.1.1 确立理想抛物面的非线性优化模型

根据附录中给出的促动伸缩器径向伸缩范围，可推断出反射面的形变范围在径向伸缩范围为-0.6 米及+0.6 米的两个球面之间。考虑到工作状态时反射面的形状为近似旋转抛物面，将问题在平面坐标系下求解。在 $\alpha = 0^\circ, \beta = 90^\circ$ 时旋转抛物面关于 z 轴对称，作理想抛物面及两种边界情况的 xOz 截面，得到示意图如下：

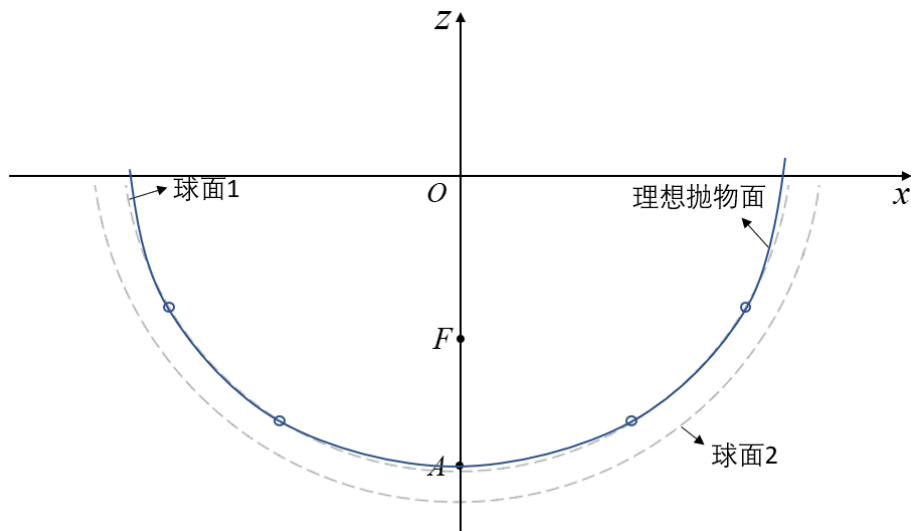


图 5 理想抛物面以及两种边界情况的 $x-z$ 截面

假设 A 在基准球面保持不动，其坐标为 $(0, R)$ 。结合焦面与球面半径的半径差 $f = 0.466R$ ，可以得到工作抛物面的投影抛物线方程：

$$x^2 = 4f(z + R) \quad (1)$$

然而这种情况过于理想，在实际情况中顶点 A 存在 ± 0.6 米的移动，我们将其考虑进模型的建立中。

设工作抛物面的顶点 A 坐标为 $(0, a)$ ，顶点 A 在球面 1 与球面 2 之间，其中，球面 1 和球面 2 分别为径向伸长 $+0.6m$ 和 $-0.6m$ 的球面，因此 a 须满足：

$$-R - 0.6 < a < -R + 0.6 \quad (2)$$

已知焦面与球面半径的半径差为 $0.466R$ ，得到焦点 F 的坐标为 $(0, -0.534R)$ 。

焦面与球面半径的半径差 f 即为焦距，其数值等于顶点坐标与焦点坐标之差，即：

$$f = -0.534R - a \quad (3)$$

由(2)(3)式得焦距 f 需要满足的条件：

$$-0.6 + 0.466R < f < 0.6 + 0.466R \quad (4)$$

设工作抛物面的抛物线簇方程为：

$$x^2 = 4f(z - C_1) \quad (5)$$

整理得：

$$x^2 = 4fz + C \quad (6)$$

对抛物线进行旋转处理，得到理想抛物面簇方程为：

$$x^2 + y^2 = 4fz + C \quad (7)$$

球面 1 和球面 2 在 $x-z$ 平面上的投影的圆方程为：

$$x^2 + z^2 = (R - 0.6)^2 \quad (8)$$

$$x^2 + z^2 = (R + 0.6)^2$$

当抛物面处在两球面之间时，对应的主索节点可以达到理想抛物面处；当抛物面处在两球面之外时，对应的主索节点无法达到理想抛物面处。取其临界情况分析，故联立(6)(8)式得：

$$\begin{aligned} z^2 + 4fz + C - (R - 0.6)^2 &= 0 \\ z^2 + 4fz + C - (R + 0.6)^2 &= 0 \end{aligned} \quad (9)$$

此时

$$\begin{aligned} \Delta_1 &= 16f^2 - 4[(R - 0.6)^2 + C] \\ \Delta_2 &= 16f^2 - 4[(R + 0.6)^2 + C] \end{aligned} \quad (10)$$

其中， Δ_1 与 Δ_2 分别为(9)两式的判别式。

经分析，使得抛物线在可行移动空间内有两种位置情况：

I：抛物线与球面 1 截线和球面 2 截线有两个 z 坐标不同的交点，即 $\Delta_1 > 0, \Delta_2 > 0$

II: 抛物线与球面 2 截线无交点, 与球面 1 截线有两个 z 坐标不同的交点,
即 $\Delta_1 > 0, \Delta_2 < 0$

情况 I 的示意图如图 6 所示, 情况 II 的示意图如图 5 所示。

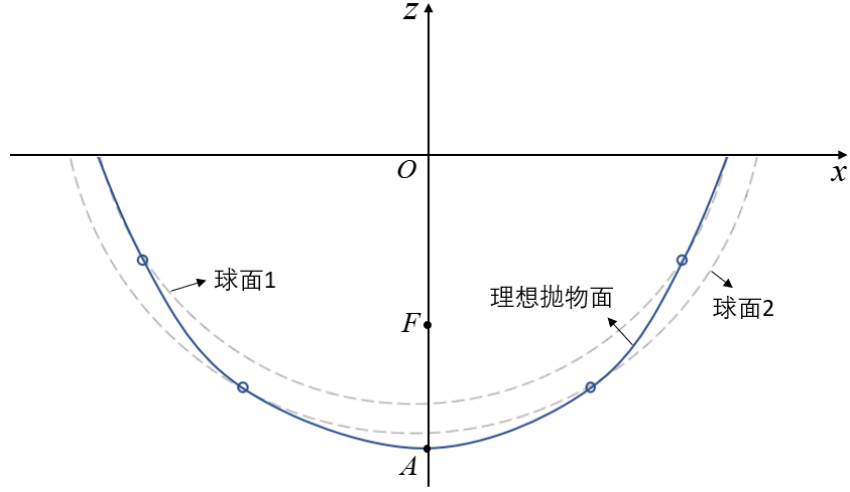


图 6 情况 I 示意图

情况 I 不等式整理得:

$$C < 4f^2 + (R - 0.6)^2 < 4f^2 + (R + 0.6)^2 \quad (11)$$

情况 II 不等式整理得:

$$4f^2 + (R - 0.6)^2 < C < 4f^2 + (R + 0.6)^2 \quad (12)$$

以情况 I 的求解过程为例:

对(9)式进行求解得:

$$\begin{aligned} z &= -2f \pm \sqrt{4f^2 - C + (R - 0.6)^2} \\ z' &= -2f \pm \sqrt{4f^2 - C + (R + 0.6)^2} \end{aligned} \quad (13)$$

分析两球壳之间的抛物面部分, 当此部分面积越大, 面上包含的主索节点越多, 参与反射面板就会越多, 天体电磁波经反射面反射后的接受效果越好。

两球壳之间的理想抛物面投影示意图如下:

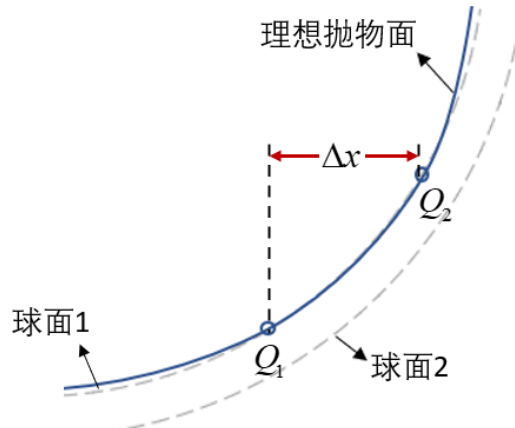


图 7 两球壳之间的工作抛物面投影示意图

工作抛物面的部分可以由抛物线与球面 1 的投影圆的交点 Q_1, Q_2 的水平距离 $|x_1 - x_2|$ 反映, 当 $|x_1 - x_2|$ 越大时, 包含的主索节点数量越多。

将(13)代入(9)解得 Q_1, Q_2 横坐标 x 为:

$$\begin{aligned} x_1 &= \sqrt{4f \cdot [-2f - \sqrt{4f^2 - C + (R+0.6)^2} + C]} \\ x_2 &= \sqrt{4f \cdot [-2f + \sqrt{4f^2 - C + (R+0.6)^2} + C]} \end{aligned} \quad (14)$$

进而得到交点 Q_1, Q_2 的水平距离:

$$|x_1 - x_2| = \sqrt{4f \cdot [-2f + \sqrt{4f^2 - C + (R+0.6)^2} + C]} - \sqrt{4f \cdot [-2f - \sqrt{4f^2 - C + (R+0.6)^2} + C]} \quad (15)$$

当两交点之间的水平距离最大时, 天体电磁波经反射面反射后的接受效果达到最佳^[3]。

因此可以建立以 Q_1, Q_2 最小水平距离为目标函数的非线性规划模型:

$$\begin{cases} \text{目标函数} \begin{cases} \max |x_1 - x_2| \\ x_1 = \sqrt{4f \cdot [-2f - \sqrt{4f^2 - C + (R+0.6)^2} + C]} \\ x_2 = \sqrt{4f \cdot [-2f + \sqrt{4f^2 - C + (R+0.6)^2} + C]} \end{cases} \\ \text{约束函数} \begin{cases} -R - 0.6 < a < -R + 0.6 & \textcircled{1} \\ f = -0.534R - a & \textcircled{2} \\ C < 4f + (R - 0.6)^2 < 4f + (R + 0.6)^2 & \textcircled{3} \end{cases} \end{cases} \quad (16)$$

① 满足顶点 A 在球面 1 与球面 2 之间;

② 满足焦点与顶点之间的关系

③ 满足抛物线与两个球面关于 z 均有两个交点

6.1.2 基于 $fmincon$ 内点算法^[4]的约束最小化求解

本问属于不等式约束优化问题, 选择内点法^[4]对其进行求解。

内点法的求解步骤如下:

Step 1: 给定初始点 $x^{(0)} \in D_0$, 终止误差 $0 \leq \varepsilon \ll 1$, $\tau_1 > 0$, $\rho \in (0, 1)$, 令 $k = 1$ 。

Step 2: 以 $x^{(k-1)}$ 为初始点求解目标函数, 得极小点 $x^{(k)}$ 。

Step 3: 若 $\tau_k \bar{H}(x^{(k)}) \leq \varepsilon$, 则停止计算, 输出 $x \approx x^{(k)}$ 作为原问题的近似极小点; 否则, 转 **Step 4**。

Step 4: 令 $\tau_{k+1} = \rho \tau_k$, $k = k + 1$, 转 **Step 2**。

内点法的迭代过程始终在可行域内进行, 为此把初始点取在可行域内。考虑到本题需要保证最后的最优解为在实数域内, 所以通过在可行域内遍历的手段找到一个满足目标函数的实数初始值。

在可行域的边界上进行设置一道障碍, 使得迭代点靠近边界点时, 给出的新

的目标函数值迅速增大，这样使得迭代点始终留在可行域内。

6.1.3 问题一的结果

综合以上求解步骤，得到理想抛物面的方程为：

$$z = \frac{x^2 + y^2 - 168181.000558960}{560.502062679404} \quad (17)$$

理想抛物面示意图如下：

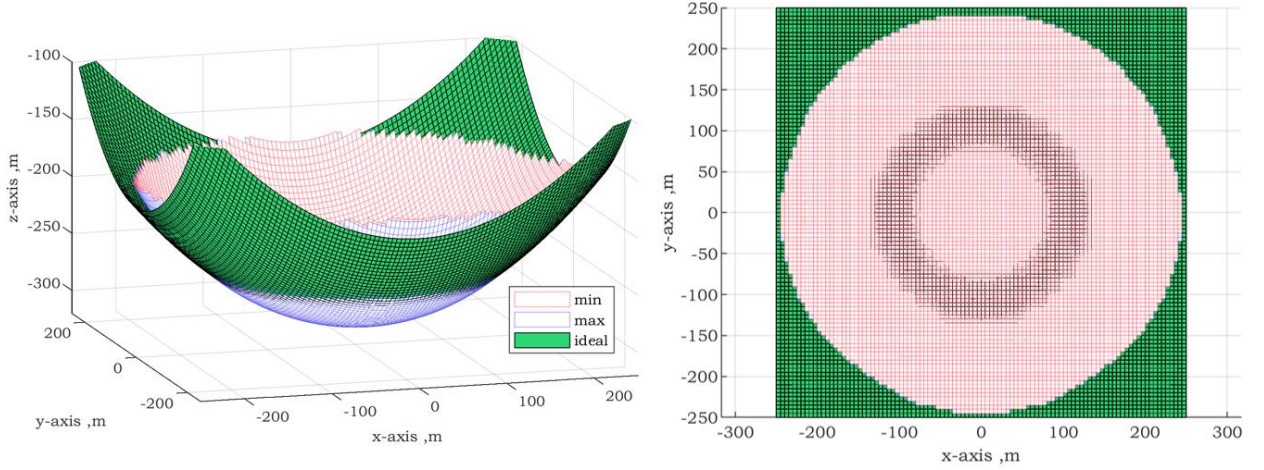


图 8 理想抛物面与边界条件图（左：整体，右：俯视）

6.2 问题二模型的建立与求解

6.2.1 三维笛卡尔坐标变换

问题二要求观测天体在方向角 $\alpha = 36.795^\circ$ ，仰角 $\beta = 78.169^\circ$ 时，确定理想抛物面。在这种情况下，坐标的相关表达式较为复杂，难于计算。因此我们将体系进行合理的旋转^[5]。将坐标进行三维笛卡尔坐标变换，将整个体系旋转到天体位于 $\alpha = 0^\circ$ ， $\beta = 0^\circ$ 的位置，并结合第一问的方法进行求解。

本题分别以坐标轴 y, z 作为旋转轴，在垂直坐标轴的平面上作两次二维旋转，到天体位于 $\alpha = 0^\circ$ ， $\beta = 0^\circ$ 的位置，此过程即为将向量 \overrightarrow{CS} 旋转到 z 轴的过程。

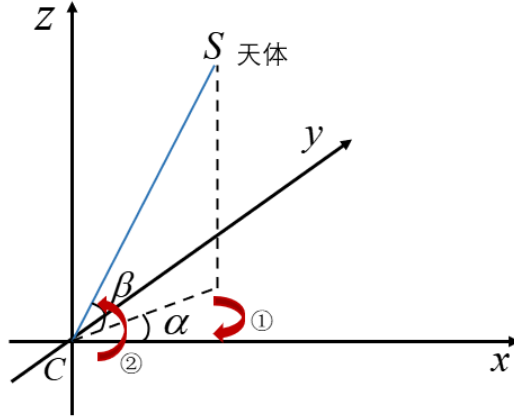


图 9 天体 S 方位角与仰角及旋转顺序示意图

图中， S 为天体， α 为天体 S 方位角， β 为天体 S 仰角。

如图 9 所示，旋转的第一步为绕 z 轴顺时针旋转 α ，第二步为绕 y 轴逆时针旋转 $(90^\circ - \beta)$ 。考虑到几何体旋转的次序不同会产生不同的位置结果，因此变换的次序不可以调换。

三维笛卡尔坐标变换矩阵^[6]如下。

绕 z 轴顺时针旋转 θ 角的旋转矩阵为：

$$R_z(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (18)$$

绕 y 轴逆时针旋转 θ 角的旋转矩阵为：

$$R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (19)$$

结合以上坐标变换矩阵，对体系的原坐标进行变换：

Step 1: 绕 z 轴顺时针旋转 α ：

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (20)$$

Step 2: 绕 y 轴逆时针旋转 $90^\circ - \beta$ ：

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} \cos(90^\circ - \beta) & 0 & -\sin(90^\circ - \beta) \\ 0 & 1 & 0 \\ \sin(90^\circ - \beta) & 0 & \cos(90^\circ - \beta) \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad (21)$$

其中， $\alpha = 36.795^\circ$ ， $\beta = 78.169^\circ$ 。

得到各主索节点对应的新坐标，绘制体系旋转变换前后的三维空间示意图如下：

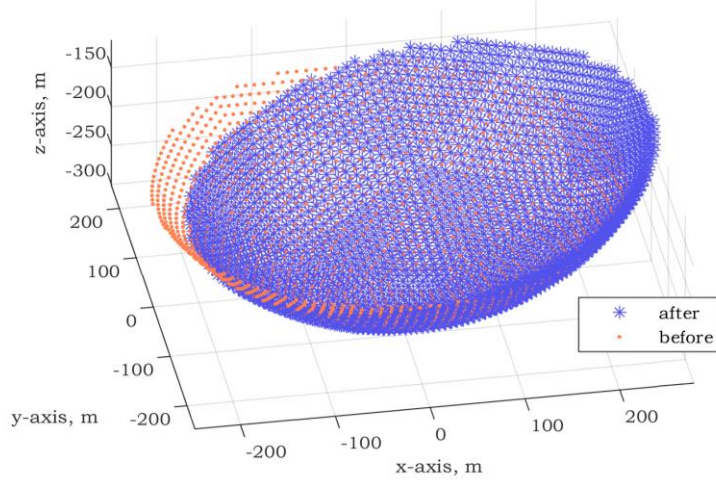


图 10 主索节点旋转变换前后的三维空间示意图

6.2.2 优化模型的建立

在原体系的坐标进行旋转变换后，题目中反射面口径为 300 米的限制条件可由 x, y 直接表示，而不需要 z 坐标的参与，得到不等式如下：

$$x^2 + y^2 < 150^2 \quad (22)$$

设满足以上表达式且在理想抛物面上的点的坐标为 (a_i, b_i, c_i) ，对应基准圆上的点的坐标为 (d_i, e_i, f_i) ，对应主索节点的坐标为 (x_i, y_i, z_i) 。

主索节点坐标由对应基准圆上对应点坐标变换而来，设促动器径向伸缩长度为 k ，下拉索移动的方向向量为 α_i ，则有下列式：

$$\alpha_i = (-d_i, -e_i, -f_i) \quad (23)$$

$$\begin{cases} x_i = d_i + k\alpha_i \cdot x \\ y_i = e_i + k\alpha_i \cdot y \\ z_i = f_i + k\alpha_i \cdot \hat{z} \end{cases} \quad (24)$$

其中， k 的取值范围为 $-0.6 < k < 0.6$ 。

通过第一问的求解，已经得到理想抛物面的全部参数。要使得反射面尽量贴近该理想抛物面，须令理想抛物面上各点与对应的主索节点之间距离最小。因此优化目标为对应的主索节点之间距离平方和最小，即：

$$\begin{cases} \text{目标函数: } \min \sum_{i=1}^N [(x_i - a_i)^2 + (y_i - b_i)^2 + (z_i - c_i)^2] \\ \text{约束条件: } -0.6 < k < 0.6 \end{cases} \quad (25)$$

6.2.3 反射面板调节模型的求解

对上述优化模型求解，得到各促动器的伸缩量后结合(24)式得到各主索节点旋转前位置调节后的坐标。将主索节点坐标以及伸缩移动的距离绘制为如下示意

图：

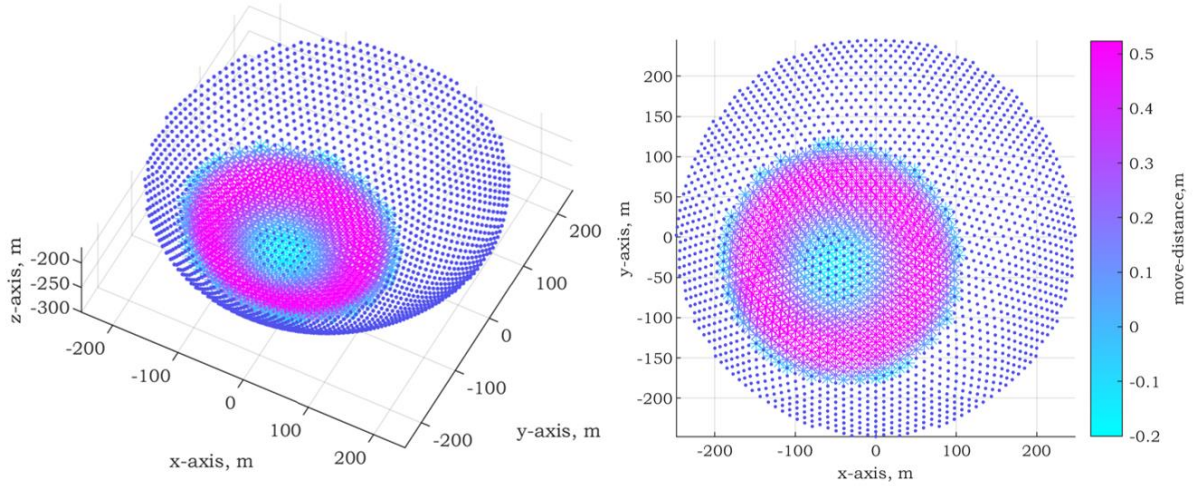


图 11 调节后主索节点示意图（左：整体，右：俯视）

其中，颜色由蓝到紫的深浅变化反映促动器移动距离由短到长的变化。

满足问题二的理想抛物面的顶点坐标可由问题一中求解所得顶点坐标进行与前文对应的坐标逆变换得到。第一步为绕 y 轴顺时针旋转 $(90^\circ - \beta)$ ，第二为绕 z 轴逆时针旋转 α 。设第一次变换后的顶点坐标为 (x_a', y_a', z_a') ，第二次变换后的顶点坐标为 $A'(x_a, y_a, z_a)$ ，结合旋转方向及角度，进行坐标的旋转变换。

绕 y 轴顺时针旋转 $90^\circ - \beta$ ：

$$\begin{bmatrix} x_a' \\ y_a' \\ z_a' \end{bmatrix} = \begin{bmatrix} \cos(\beta - 90^\circ) & 0 & -\sin(\beta - 90^\circ) \\ 0 & 1 & 0 \\ \sin(\beta - 90^\circ) & 0 & -\sin(\beta - 90^\circ) \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ a \end{bmatrix} \quad (26)$$

绕 z 轴逆时针旋转 α ：

$$\begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} = \begin{bmatrix} \cos(-\alpha) & \sin(-\alpha) & 0 \\ -\sin(-\alpha) & \cos(-\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_a' \\ y_a' \\ z_a' \end{bmatrix} \quad (27)$$

求得顶点 A' 的坐标为： $(-49.269343, -36.851473, -293.7163412)$ 。

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见 '*result.xlsx*' 文件，部分结果如下表：

表 1 各促动器的伸缩量

对应主索节点编号	伸缩量（米）
A0	0.193137812
B1	0.292301281
C1	0.258243452
...	...
D268	-0.028002705
D269	-0.059671328

表 2 各主索节点旋转前位置调节后的坐标

节点编号	X 坐标（米）	Y 坐标（米）	Z 坐标（米）
A0	0	0	-300.20686
B1	6.10185686	8.39881965	-299.92807
C1	9.87420419	-3.2082396	-299.96211
...
D268	-139.58181	-139.86204	-226.30179
D269	-148.84306	-131.89519	-225.22283

6.3 问题三模型的建立与求解

6.3.1 信号反射与接收模型的建立

以某一块三角小反射面板为例，分析信号的反射与接收过程。

I 反射过程

设入射电磁波为 ray_{in} ，反射电磁波为 ray_{re} ，入射角与反射角大小相等均为 η ，法线方向的单位向量为 n ，三角反射板的三个顶点坐标为 $D_1(x_{d1}, y_{d1}, z_{d1})$ ， $D_2(x_{d2}, y_{d2}, z_{d2})$ ， $D_3(x_{d3}, y_{d3}, z_{d3})$ 。反射过程示意图如下：

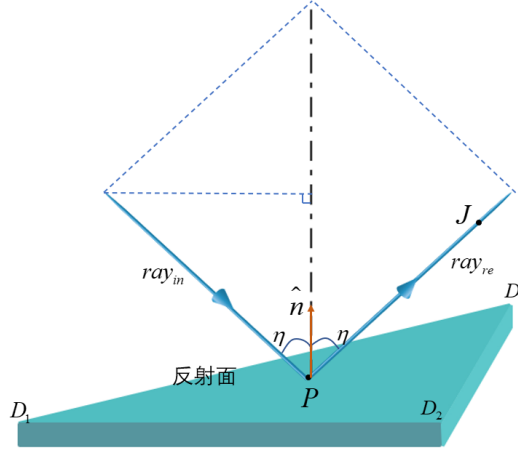


图 12 反射过程示意图

入射点为 P 的坐标为 $\left(\frac{x_{d1} + x_{d2} + x_{d3}}{3}, \frac{y_{d1} + y_{d2} + y_{d3}}{3}, \frac{z_{d1} + z_{d2} + z_{d3}}{3} \right)$ 。

单位法向量 n 为：

$$n = \frac{\overrightarrow{D_1D_2} \times \overrightarrow{D_1D_3}}{|\overrightarrow{D_1D_2}| \cdot |\overrightarrow{D_1D_3}|} \quad (28)$$

其中

$$\begin{aligned}\overrightarrow{D_1D_2} &= (x_{d1} - x_{d2}, y_{d1} - y_{d2}, z_{d1} - z_{d2}) \\ \overrightarrow{D_1D_3} &= (x_{d1} - x_{d3}, y_{d1} - y_{d3}, z_{d1} - z_{d3})\end{aligned}\quad (29)$$

通过平行四边形法则，得到电磁波入射与反射的路径向量以及法线单位向量之间的关系式：

$$\overrightarrow{ray_{in}} + 2(\overrightarrow{ray_{in}} \cdot \mathbf{n}) \cdot \mathbf{n} = \overrightarrow{ray_{re}} \quad (30)$$

设 J 为反射电磁波上任意一点：

$$\overrightarrow{PJ} = \lambda \overrightarrow{ray_{re}} \quad (31)$$

反射电磁波信号路径为一条射线，因此 λ 满足 $\lambda > 0$ 。

II 接收过程

由前两问的分析可知，无论电磁波入射角度为多少，均可通过笛卡尔变换得到垂直入射信号，因此本题中仍在电磁波垂直入射的情况下进行分析。

馈源舱接收信号的有效区域为直径 1 米的中心圆盘。设馈源舱上某点坐标为 $V(x_v, y_v, z_v)$ ，满足下式：

$$\begin{cases} x_v^2 + y_v^2 \leq \left(\frac{1}{2}\right)^2 \\ z_v = -0.556R \end{cases} \quad (32)$$

综合(31)(32)式，找到 300 米口径的工作抛物面内参与有效反射的三角反射面板以及基准反射球面上参与有效反射的三角反射面板。

6.3.2 馈源舱的接收比的计算

馈源舱的接收比等于馈源舱有效区域接收到的反射信号与 300 米口径内反射面的反射信号之比，即反射面中有效反射部分的面积与 300 米口径内反射面的总面积之比。针对实际工程情况，将理想连续平面离散化，馈源舱的接收比转化为参与有效反射的三角反射面板与 300 米口径的工作抛物面内的三角反射面板总数之比。

设 300 米口径的工作抛物面内的三角反射面板总数为 N ，主索节点总数为 N' ；其中参与有效反射的三角反射面板的数目为 n_{valid} ，对应主索节点数为 n'_{valid} ，在基准反射球面上参与有效反射的三角反射面板的数目为 n_0 ，对应主索节点数为 n'_0 。定义两种接收比，记以三角反射面板数目为参考标准的接收比为 P_r ，以有效反射的主索节点为参考标准的接收比为 P'_r 。

则调节后馈源舱的两种接收比为：

$$P_r = \frac{n_{valid}}{N} \quad P'_r = \frac{n'_{valid}}{N'} \quad (33)$$

基准反射球面的两种接收比为：

$$P_{r0} = \frac{n_0}{N} \quad P'_{r0} = \frac{n'_0}{N'} \quad (34)$$

6.3.3 初步结果分析与模型修正

经计算得到调节后馈源舱的接收比 $P_r = 4.32\%$ 。尝试对问题一求解式子中的参量进行调整，发现当 F 点的坐标进行 3% 以内的调整时，馈源舱的接收比会有一定幅度的提升。

对这种现象进行分析解释。当焦点 F 不固定在馈源舱上，即在馈源舱附近有小范围的调整时，反射信号也可以传入馈源舱。假设馈源舱的功能完备，这部分信号也可以被接收利用，此时馈源舱的接收比就得到提升。

对此，对问题一的模型进行修正，放宽问题一优化模型的约束条件，对焦点 F 位置不进行约束，即去掉 $f = -0.534R - a$ 的约束。运用局部搜索算法对最优解进行搜索，在迭代的过程中不断提高最优解的精度。

求解得到问题一修正后的理想抛物面表达式：

$$z = \frac{x^2 + y^2 - 168196.217}{559.5348} \quad (35)$$

修正后求解问题三得到的调节后馈源舱接收比见下表：

表 3 模型修正后馈源舱接收比

	调整后抛物面	原球面	提升
P_r	61.85%	21.39%	289.29%
P'_r	27.57%	6.38%	432.13%

观察表中数据，发现较模型修正前馈源舱的接受比有显著提升。

6.3.4 结果解释与验证

绘制工作抛物面与基准球面上反射点情况图如下：

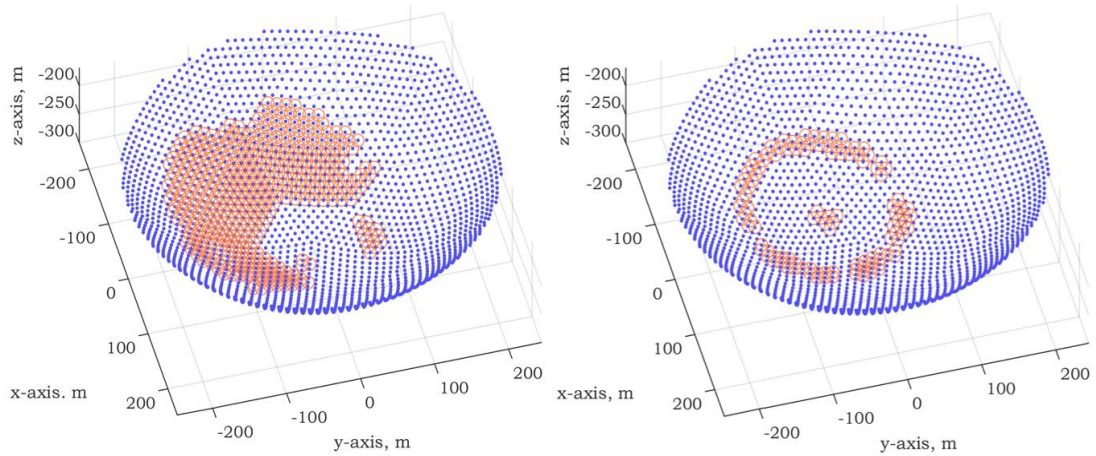


图 13 工作抛物面与基准球面反射点情况

图中工作抛物面与基准球面上能够反射有效信号的三角反射面呈现不对称分布。造成这样结果的原因其一是附件 2 中提供的坐标参数并不严格对称，其中不对称坐标参数最小差值的数量级为 10^{-3} ，由于馈源舱接收信号的有效区域圆盘直径与工作抛物面口径之比为 $1/300$ ，所以模型对原始数据十分敏感，微小的不对称会导致结果出现较为明显的不对称情况。原因其二是在进行体系坐标通过旋转矩阵进行变换时，由于编译软件计算精度有限，会影响数据精度，因此导致旋转后的抛物面不严格对称的结果。

我们对上述的结果进行进一步验证，将馈源舱的有效区域圆盘直径调整为 2 米，计算出此时调节后馈源舱接收比为 70%。对照发现，有效区域圆盘为 1 米时未能传到馈源舱的电磁波信号都十分靠近馈源舱，但由于馈源舱有效区域圆盘的直径过小，导致接收比并不高。

将所有工作抛物面上实现有效反射的面板反射的信号路径绘制成示意图：

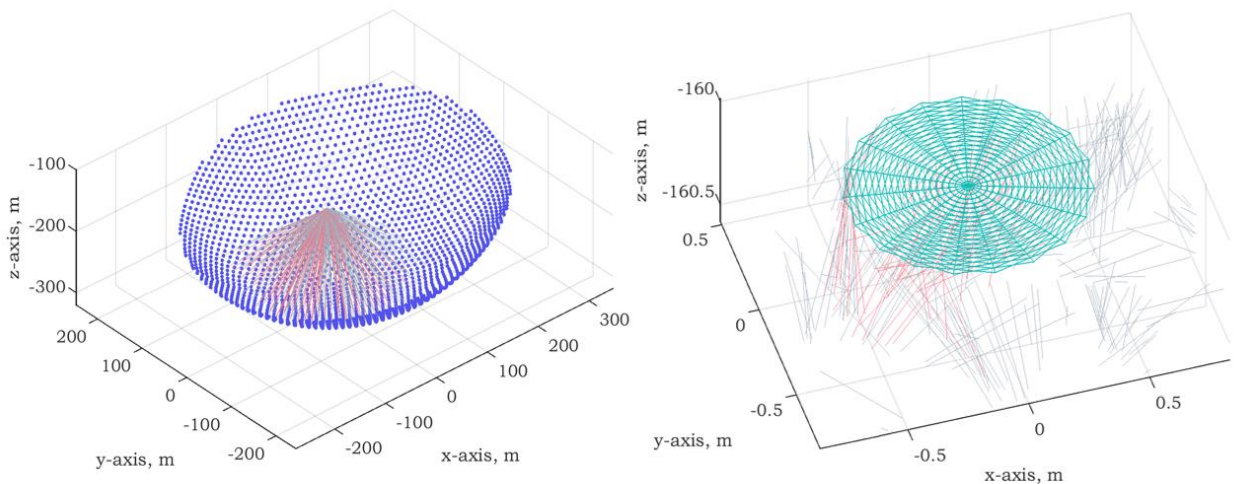


图 14 工作抛物面上有效反射信号路径（左：原图，右：放大 200 倍）

图中可以明显观测到所有有效反射信号都汇聚在直径为 1 米的馈源舱有效接收圆盘上。

综合以上分析，可知此时得到的结果为所给条件下的最优解，即表 3 所得结果。

七、模型的稳定性分析

由于自动控制过程会存在一定的工程误差，导致促动器的伸缩长度与求解出的理论值存在一定偏差^[7]，因此我们采用促动器伸缩长度对系统的影响进行稳定性分析。对工程误差进行模拟，其分布类型为正态分布。

误差 e_i 分布函数 P_{e_i} 可表示为：

$$P_e \sim N(0,1) \quad (36)$$

则推出编号为 i 的促动器实际伸缩长度表达式为：

$$k_i' = k_i + e_i P_{e_i} \quad (37)$$

其中， k_i 为促动器理论伸缩长度。

得到模型稳定性分析图如下：

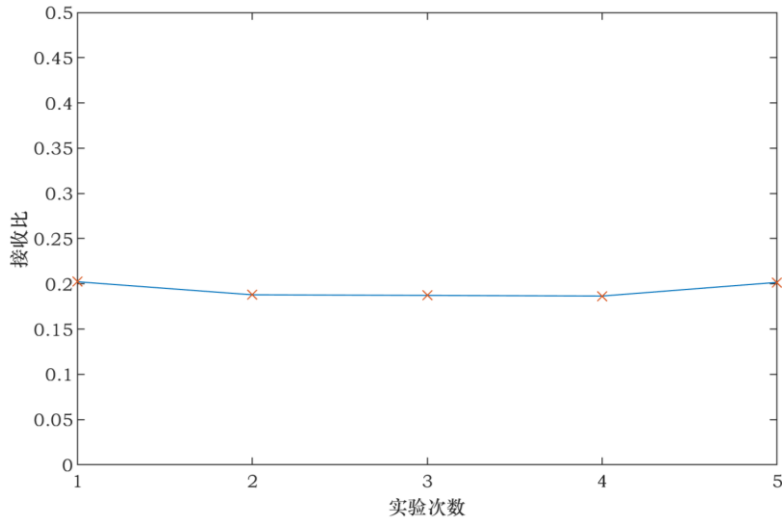


图 15 模型稳定性分析图

进行多次独立重复实验，每次实验随机生成误差 e_i ，得到的接收比结果变化不明显，说明模型在控制精度 ± 6 毫米时稳定。

八、模型的推广

在求解问题三的过程中，通过作图较为明显地发现，对于不完全拟合的类抛物面，反射信号汇聚较为集中。因此，考虑将接收器有效区域进一步扩大，寻找较为合适的接收器口径，使得接收比有较明显的提高。

以步长为 0.2 逐步扩大接收器口径，在前三问的基础上，得到相应的接收比如下图表。

表 4 不同接收区域口径下的信号接收比

口径	接收比	口径	接收比
0.6	7.82%	1.6	64.13%
0.8	16.19%	1.8	70.10%
1.0	27.57%	2.0	73.46%
1.2	41.36%	2.2	76.13%
1.4	54.94%	2.4	78.25%

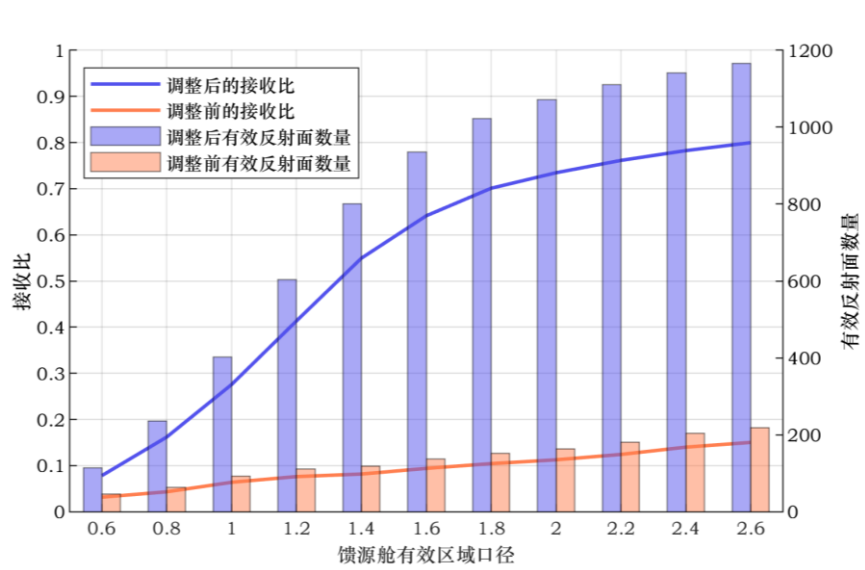


图 16 不同有效区域口径下接收比与有效反射面数量统计图

观察发现，当口径在 1.0 至 1.4 米时，接收比增长幅度较大，且接收比在 1.4 米时以超过 60%，而在 1.4 米以后，接收比增长幅度有明显下降。因此，综合考虑接收器口径与加工难度的关系，不妨将接收器有效区域口径增加至 1.4 米以获得较好的接收比。

另外，也可以将接收器有效区域改进为多个接收器协同作用，以进一步获得更大的接收比。

九、模型的评价

9.1 模型优点

(1) 对于问题一，结合旋转抛物面的对称特性，对其进行截面处理，使得求解更加简便。

(2) 对于问题二，通过体系坐标的旋转变换将方位角以及仰角不为 0 的情况转化为方位角及仰角为 0 的问题，充分利用第一问的求解基础，简化了模型以

及求解的计算。

(3) 对于问题三, 运用矢量运算法则以及空间几何求解技巧, 得出反射电磁波矢量的通式, 进而得到 300 米口径内 1458 个三角反射面板上的反射路径。

(4) 对于问题三, 对初步结果进行分析解释, 进而对问题一的模型进一步优化, 从而得到与客观事实较为符合的接收比, 所得结果较基球面的接收比有较大提升。

(5) 从接收比的角度出发, 考虑到客观加工工艺等因素, 对大幅提升接收比给出了合理客观的建议。

9.2 模型缺点

(1) 在模型的建立中没有考虑三角形反射面板的厚度等因素, 因此其对工作抛物面各参数求解造成的影响未知。

(2) 由于需要解出每个反射点的反射路径方程, 问题规模较大, 致使算法时间复杂度较大。

九、参考文献

- [1] 钱宏亮,范峰,沈世钊,冯若强,陈波.FAST 反射面支承结构整体索网方案研究[J].土木工程学报,2005(12):18-23.
- [2] 孙尚超. 基于 MRAC 理论的 FAST 节点控制系统的建模与仿真研究[D].东北大学,2015.
- [3] 李明辉,朱丽春.FAST 瞬时抛物面变形策略优化分析[J].贵州大学学报(自然科学版),2012,29(06):24-28+43.
- [4] 杨小礼. 线性与非线性破坏准则下岩土极限分析方法及其应用[D].中南大学,2002.
- [5] 王文利,段宝岩.大射电望远镜 FAST 的控制与 GPS 动态检测[J].仪器仪表学报,2001(01):49-53.
- [6] 吕志鹏,伍吉仓.三维坐标变换算法的比较[J].大地测量与地球动力学,2016,36(01):35-39.
- [7] 苏玉鑫,段宝岩.大射电望远镜精调平台运动精度分析[J].中国机械工程,2001(06):115-117+8.
- [8] 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(05):3-20.

十一、附录

11.1 附录 1: Q_1, Q_2 两点间的横坐标之差计算公式

$$|x_1 - x_2| = \frac{1}{2} \times \left(\left[4c - \frac{25632a}{5} - 16a \left(2a - \frac{\sqrt{16a^2 + \frac{25632a}{5} - 4c + \frac{41295363226730493}{53687091200}}}{2} + \frac{1602}{5} \right) \right] + \frac{6408\sqrt{16a^2 + \frac{25632a}{5} - 4c + \frac{41295363226730493}{53687091200}}}{5} - \frac{20531232}{25} \right) - \left(\left[4c - \frac{25632a}{5} - 16a \left(2a + \frac{\sqrt{16a^2 + \frac{25632a}{5} - 4c + \frac{41295363226730493}{53687091200}}}{2} + \frac{1602}{5} \right) \right] - \frac{6408\sqrt{16a^2 + \frac{25632a}{5} - 4c + \frac{41295363226730493}{53687091200}}}{5} - \frac{20531232}{25} \right) \right)$$

11.2 附录 2: 子函数

11.2.1 计算方向单位向量

%计算方向单位向量

```
function [x_dir,y_dir,z_dir] = calculateDirect(pos)
    x = pos(1);
    y = pos(2);
    z = pos(3);
    mo = sqrt(x^2 + y^2 + z^2);
    x_dir = x/mo;
    y_dir = y/mo;
    z_dir = z/mo;
end
```

11.2.2 坐标旋转变换

%按先 z 轴再 y 轴旋转

```
function pos_p = rota(thetay,thetaz,pos)
Rz = [cos(thetaz) sin(thetaz) 0; -sin(thetaz) cos(thetaz) 0; 0 0 1];
Ry = [cos(thetay) 0 -sin(thetay); 0 1 0; sin(thetay) 0 cos(thetay)];
M = Ry*Rz;
pos_p = M*pos;
end
```

11.2.3 坐标旋转逆变换

%按先 y 轴再 z 轴旋转(rota_inv)

```
function pos_p = rota_inv(thetay,thetaz,pos)
Rz = [cos(thetaz) sin(thetaz) 0; -sin(thetaz) cos(thetaz) 0; 0 0 1];
Ry = [cos(thetay) 0 -sin(thetay); 0 1 0; sin(thetay) 0 cos(thetay)];
M = Rz*Ry;
pos_p = M*pos;
end
```

11.2.4 计算反射射线上的两个点

%计算反射射线上的两个点

%flag 画图区分标志

%hanghao 反射直线能够在给定区域口径 1 的面的行号

%cnt 计算的面数量

%区域半径

```
function [Ax,Ay,Az,Bx,By,Bz,flag,hanghao,cnt] =
calculate2ReflectPoint(zhongxin,reflect,node_class,r)
    cnt = 0;
    hanghao = [];
    Ax = [];
    Ay = [];
    Az = [];%作反射直线的一个坐标
    Bx = [];
    By = [];
    Bz = [];%作反射直线的一个坐标
    r=r+0.1;
    for i=1:4300
        if(node_class(i))
            px = zhongxin(i,1);
            py = zhongxin(i,2);
            pz = zhongxin(i,3);
            syms k
```

```

eqns = [ (zhongxin(i,1)+k*reflect(i,1))^2 +
(zhongxin(i,2)+k*reflect(i,2))^2 <=
r^2, (zhongxin(i,3)+k*reflect(i,3))==-0.534*300];
S = solve(eqns, [k]);
if S
    flag(i) = 0;
    cnt = cnt + 1;
    hanghao = [hanghao;i];
    k0=double(S);
    Bx = [Bx;zhongxin(i,1)];
    By = [By;zhongxin(i,2)];
    Bz = [Bz;zhongxin(i,3)];
    Ax = [Ax;zhongxin(i,1) + k0*reflect(i,1)];
    Ay = [Ay;zhongxin(i,2) + k0*reflect(i,2)];
    Az = [Az;zhongxin(i,3) + k0*reflect(i,3)];
end
end
end

end

```

11.2.5 已知三个点求其单位法向量

%已知三个点求其单位法向量

```

function [x,y,z] = pnorm(p1,p2,p3)
    a = p1-p2;
    b = p2-p3;
    c = cross(b,a);
    mo = sqrt(c(1)^2+c(2)^2+c(3)^2);
    x = c(1)/mo;
    y = c(2)/mo;
    z = c(3)/mo;
end

```

11.3 附录 3：问题一主程序

11.3.1

%求解抛物线在范围内的最优解

```

close all;
clear all;
clc

```

%联立方程，求得差为目标函数

```

syms x z a c
R = 300;
eqns = [x^2+z^2 == (R-0.6)^2,x^2==4*(-0.534*R-a)*z+c];
vars = [x,z];
S = solve(eqns,vars);
delta_x = S.x(2,1)-S.x(4,1);

%寻找目标函数的初始值
for a=-301:0.01:-298
    for c=167870:168489
        d = (4*c - (25632*a)/5 - 16*a*(2*a - (16*a^2 + (25632*a)/5 - 4*c
+ 41295363226730493/53687091200)^(1/2)/2 + 1602/5) + (6408*(16*a^2 +
(25632*a)/5 - 4*c + 41295363226730493/53687091200)^(1/2))/5 -
20531232/25)^(1/2)/2 - (4*c - (25632*a)/5 - 16*a*(2*a + (16*a^2 +
(25632*a)/5 - 4*c + 41295363226730493/53687091200)^(1/2)/2 + 1602/5) -
(6408*(16*a^2 + (25632*a)/5 - 4*c +
41295363226730493/53687091200)^(1/2))/5 - 20531232/25)^(1/2)/2;
        if isreal(d)
            fprintf('%d %d\n',a,c);
        end
    end
end
end

```

%最优化求解

```

fx = @(x) -((4*x(2) - (25632*x(1))/5 - 16*x(1)*(2*x(1) - (16*x(1)^2 +
(25632*x(1))/5 - 4*x(2) + 41295363226730493/53687091200)^(1/2)/2 +
1602/5) + (6408*(16*x(1)^2 + (25632*x(1))/5 - 4*x(2) +
41295363226730493/53687091200)^(1/2))/5 - 20531232/25)^(1/2)/2 - (4*x(2)
- (25632*x(1))/5 - 16*x(1)*(2*x(1) + (16*x(1)^2 + (25632*x(1))/5 - 4*x(2)
+ 41295363226730493/53687091200)^(1/2)/2 + 1602/5) - (6408*(16*x(1)^2 +
(25632*x(1))/5 - 4*x(2) + 41295363226730493/53687091200)^(1/2))/5 -
20531232/25)^(1/2)/2);
lb = [-R-0.6, 4*(-0.534*R-(-R+0.6))^2+(R+0.6)^2];
ub = [-R+0.6, 4*(-0.534*R-(-R-0.6))^2+(R-0.6)^2];
fm = fmincon(fx,[-3.003600e+02 168181],[[],[],[],[],[],lb,ub)

```

11.3.2

%绘制抛物面与边界条件

```

clc
clear all
close all

load('mainPoint.mat');
load('move.mat');

```



```

load('node.mat');
R = 300;
F = 0.466 * R;

%初始的球面坐标
x_ini = mainPoint.data(:,1);
y_ini = mainPoint.data(:,2);
z_ini = mainPoint.data(:,3);
%坐标方向
[m,n] = size(x_ini);
for i = 1:m
    mo = sqrt(x_ini(i)^2 + y_ini(i)^2 + z_ini(i)^2);
    x_dir(i) = x_ini(i)/mo;
    y_dir(i) = y_ini(i)/mo;
    z_dir(i) = z_ini(i)/mo;
end
x_dir = x_dir';
y_dir = y_dir';
z_dir = z_dir';
%最小球面坐标
x_min = x_ini - 0.6*x_dir;
y_min = y_ini - 0.6*y_dir;
z_min = z_ini - 0.6*z_dir;
%最大球面坐标
x_max = x_ini + 0.6*x_dir;
y_max = y_ini + 0.6*y_dir;
z_max = z_ini + 0.6*z_dir;

[xq,yq] = meshgrid(-250:5:250,-250:5:250);

a = -300.0837;
c = 168196.217;
zh = (xq.^2+yq.^2 - c)./(4*(-0.534*R-a));
fig1 = figure('numbertitle','off','name','理想抛物线');
surf(xq,yq,zh,'FaceColor','#2ed573');

fig2 = figure('numbertitle','off','name','理想抛物线与边界条件');
zq_min = griddata(x_min,y_min,z_min,xq,yq);
mesh(xq,yq,zq_min,'EdgeColor','#ff4757','EdgeAlpha',0.5);
hold on;
zq_max = griddata(x_max,y_max,z_max,xq,yq);
mesh(xq,yq,zq_max,'EdgeColor','#5352ed','EdgeAlpha',0.5);
surf(xq,yq,zh,'FaceColor','#2ed573','FaceAlpha',1);
legend('min','max','ideal');

```

```

%title('理想抛物线与边界条件','FontSize',16);
xlabel('x-axis ,m');
ylabel('y-axis ,m');
zlabel('z-axis ,m');
zlim([-320 -100]);
xlim([-250 250]);
ylim([-250 250]);

fig3 = figure('numbertitle','off','name','理想抛物线与边界条件 (俯视图)');
mesh(xq,yq,zq_min,'EdgeColor','#ff4757','EdgeAlpha',0.5);
hold on;
mesh(xq,yq,zq_max,'EdgeColor','#5352ed','EdgeAlpha',0.5);
surf(xq,yq,zh,'FaceColor','#2ed573','FaceAlpha',1);
xlim([-250 250]);
ylim([-250 250]);
xlabel('x-axis ,m');
ylabel('y-axis ,m');
view(0,90);
grid on;
axis equal;

```

11.3.3

%暴力搜索，在已知范围内加大精度

```

clc
clear all
clear all

R = 300;
delta = 0.1;
x = -R-0.6:delta:R+0.6;
cnt = 1;

for a=-300.083-0.001:0.0001:-300.083+0.001
    for c=168196.216-0.001:0.00001:168196.216+0.001
        f = (-0.534*R-a);
        % c = -4*a*f;
        z = (x.^2-c)./(4*f);
        n = length(z);
        ljb = [];
        for i=1:n
            if(z(i) < -169.019 && x(i)^2+z(i)^2>=(R-0.6)^2 &&
x(i)^2+z(i)^2<=(R+0.6)^2)
                ljb(i,:) = [x(i) z(i)];
            else

```

```

        ljb(i,:) = [0 0];
    end
end
dz = diff(ljb(:,2));
S = delta*trapz((1+dz.^2).^0.5);
res(cnt,1) = a;
res(cnt,2) = c;
res(cnt,3) = S;
cnt = cnt+1;
end
end

```

11.4 附录 4：问题二的附录

11.4.1

```

%坐标旋转
clc
clear all
close all

load('mainPoint.mat');
load('move.mat');
load('node.mat');
mainPoint_ori = mainPoint.data;%备份原坐标, n*3
res1 = [];
res2 = [];%第二张表格的结果包括 change_idx 和 res2
res3 = [];
alpha = deg2rad(36.795);
beta = deg2rad(78.169);
R = 300;
F = 0.466 * R;
%Q1 中解得的理想抛物面参数
a = -300.0837;
c = 168196.217;
%x,y,z:原坐标
x = mainPoint.data(:,1);
y = mainPoint.data(:,2);
z = mainPoint.data(:,3);
%对原有坐标旋转, 使得 S 位于 z 轴上
thetaz = alpha;
thetay = pi/2-beta;
[m,n] = size(x);

```

```

for i=1:m
    pos = [x(i) y(i) z(i)]';
    pos_p = rota(thetay,thetaz,pos);
    xx(i) = pos_p(1);
    yy(i) = pos_p(2);
    zz(i) = pos_p(3);
end

mainPoint_rot = [xx; yy; zz]';%旋转后的主索节点坐标
fig1 = figure('numbertitle','off','name','旋转后的主索节点坐标');
scatter3(xx,yy,zz,'*','MarkerEdgeColor','#5352ed');
hold on;
scatter3(x,y,z,'.','MarkerEdgeColor','#ff7f50');
legend('after','before');
%title('旋转后的主索节点坐标','FontSize',16);
xlabel('x-axis, m');
ylabel('y-axis, m');
zlabel('z-axis, m');
grid on;
axis equal ;

%%
%寻找在口径 300 内的主索节点（编号）
D = 300;%口径
mainPoint_rotIndex = zeros(m,1);%在口径内的主索节点编号
change_idx = [];%主索节点编号（有序，string），1*n
for i=1:m
    x = mainPoint_rot(i,1);
    y = mainPoint_rot(i,2);
    if x^2 + y^2 <= (D/2)^2
        mainPoint_rotIndex(i) = 1;
        change_idx = [change_idx ;string(mainPoint.textdata(i))];
    end
end

%检查找到的主索节点是否正确
%xx,yy,zz: 提取旋转后的坐标且在口径 300 内
xx = [];
yy = [];
zz = [];
for i=1:m
    if mainPoint_rotIndex(i)
        xx = [xx; mainPoint_rot(i,1)];
        yy = [yy; mainPoint_rot(i,2)];
        zz = [zz; mainPoint_rot(i,3)];
    end
end

```

```

end
fig2 = figure('numbertitle','off','name','检查找到的主索节点是否正确');
scatter3(xx,yy,zz, '.', 'MarkerEdgeColor','#5352ed');
%%
%在相同的径向下，计算由 Q1 解得的理想抛物面的 z 坐标 z_ideal
%计算径向方向向量
for i = 1:sum(mainPoint_rotIndex)
    [xx_dir(i),yy_dir(i),zz_dir(i)] = calculateDirect([xx(i) yy(i)
zz(i)]);
end
%理想抛物面上的坐标
x_ideal = [];
y_ideal = [];
z_ideal = [];
k = [];%缩放调节量
for i=1:sum(mainPoint_rotIndex)
    syms p
    eqns = [(zz(i)+p*zz_dir(i)) == ((xx(i)+p*xx_dir(i)).^2 +
(yy(i)+p*yy_dir(i)).^2 - c)/(4*(-0.534*R-a))];
    S = solve(eqns,p);
    lambda = vpa(S(2,1));
    if lambda<-0.6
        lambda = -0.6;
    end
    if lambda >0.6
        lambda = 0.6;
    end
    tx = vpa(xx(i) + lambda*xx_dir(i));
    ty = vpa(yy(i) + lambda*yy_dir(i));
    tz = vpa(zz(i) + lambda*zz_dir(i));
    x_ideal = [x_ideal; tx];
    y_ideal = [y_ideal; ty];
    z_ideal = [z_ideal; tz];
    k = [k;lambda];
end
k = double(k);
k = -1*k;
%绘制
fig3 = figure('numbertitle','off','name','理想抛物面与实际的主索节点');
scatter3(x_ideal,y_ideal,z_ideal,50, '.');
hold on;
scatter3(xx,yy,zz);
%%
%求解调整后的主索节点坐标

```

```

tmpx = [];
tmpy = [];
tmpz = [];
for i = 1:m
    [x_dir(i),y_dir(i),z_dir(i)] = calculateDirect(mainPoint_ori(i,:));
end

cnt = 1;%调节的主索节点个数
mainPoint_change = mainPoint_ori;%调节后的主索节点坐标
for i=1:m
    if mainPoint_rotIndex(i)
        %按径向修改节点坐标
        tx = mainPoint_ori(i,1) + (-k(cnt))*x_dir(i);
        ty = mainPoint_ori(i,2) + (-k(cnt))*y_dir(i);
        tz = mainPoint_ori(i,3) + (-k(cnt))*z_dir(i);
        cnt = cnt+1;
        tmpx = [tmpx;tx];
        tmpy = [tmpy;ty];
        tmpz = [tmpz;tz];
        mainPoint_change(i,:) = [tx ty tz];
    end
end
res2 = [tmpx tmpy tmpz];
%理想抛物面顶点坐标需要反相旋转回来
A = rota_inv(-thetay,-thetaz,[0 0 -300.0913]');
A = A';
res1 = A;
res3 = k;

%绘制修改过主索节点后的坐标
fig4 = figure('numbertitle','off','name','调节后主索节点坐标');
scatter3(mainPoint_ori(:,1),mainPoint_ori(:,2),mainPoint_ori(:,3),'.','MarkerEdgeColor','#5352ed');
hold on
scatter3(res2(:,1),res2(:,2),res2(:,3),50,k,'*');
%title('调节后主索节点坐标','FontSize',16);
xlabel('x-axis, m');
ylabel('y-axis, m');
zlabel('z-axis, m');
h = colorbar;
set(get(h,'label'),'string','move-distance,m');
colormap('cool')
grid on;
axis equal;

```

```

save('index.mat','change_idx');%导出修改的坐标序号集合
    save('new.mat','mainPoint_change');%导出修改后主索节点坐标

```

11.5 附录 5：问题三的主程序

```

close all
clear all
clc

load('index.mat');
load('mainPoint.mat');
load('node.mat');
load('new.mat');

alpha = deg2rad(36.795);
beta = deg2rad(78.169);
thetaz = alpha;
thetay = pi/2-beta;

%%
%旋转球面
mainPoint_change_rot = [];%修改后的主索节点做旋转
[m,n] = size(mainPoint_change);
for i=1:m
    pos = [mainPoint_change(i,1) mainPoint_change(i,2)
mainPoint_change(i,3)]';
    pos_p = rota(thetay,thetaz,pos);
    mainPoint_change_rot = [mainPoint_change_rot;pos_p'];
end
%%
%根据附件 3，判断三角面中有多少个点在 Q2 解得的口径 D 内的点的集合内
[m,n] = size(node);
node_class = zeros(m,1);%对三角面分类，三个节点全在口径 300 内记为 3
for i=1:m
    cnt = 0;
    for j = 1:3
        [h,l] = find(change_idx == node(i,1));
        if ~(isempty(h) && isempty(l))
            cnt = cnt + 1;
        end
    end
    node_class(i) = cnt;
end
end

```

```

%计算三角形面对应的中心
zhongxin = [];%三角面中心, n*3
node_norm = [];%三点的法向量
reflect = [];%反射方向向量
ru = [0 0 -1];%入射方向向量, 由于经过旋转, 使得方向竖直向下
for i=1:m
    if(node_class(i))
        [h1,l] = find(mainPoint.textdata == node(i,1));
        [h2,l] = find(mainPoint.textdata == node(i,2));
        [h3,l] = find(mainPoint.textdata == node(i,3));
        %取一个面的三个点
        p1 = mainPoint_change_rot(h1,:);
        p2 = mainPoint_change_rot(h2,:);
        p3 = mainPoint_change_rot(h3,:);
        %计算面法向量
        [node_norm(i,1) node_norm(i,2) node_norm(i,3)] = pnorm(p1,p2,p3);
        zhongxin(i,:) = (p1+p2+p3)/3;
        reflect(i,:) = -2*dot(ru,node_norm(i,:))*node_norm(i,:) + ru;
    else
        zhongxin(i,:) = NaN;
        node_norm(i,:) = [NaN NaN NaN];
        reflect(i,:) = [NaN NaN NaN];
    end
end

%%
%求解能够使反射到达有效区域的三角面数量
R = 300;
cnt = 0;
hanghao = [];%反射直线能够在给定区域口径 1 的面的行号
huax = [];
huay = [];
huaz = [];%作反射直线的一个坐标
zx = [];
zy = [];
zz = [];%作反射直线的一个坐标
flag = node_class;%分区画图标记
[huax,huay,huaz,zx,zy,zz,flag,hanghao,cnt] =
calculate2ReflectPoint(zhongxin,reflect,node_class,0.5);

%%
%作反射直线当符合口径 1
fig1 = figure('numbertitle',1,'name','抛物面反射直线');

```



```

axis equal;
hold on
N = 20;
er = 0.5;
r0 = linspace(0,er,N);
the = linspace(0,4*pi,N);
[r,theta] = meshgrid(r0,the);
z = -0.534*300*r.^0;
[x,y] = pol2cart(theta,r);
D = mesh(x,y,z,'FaceAlpha',0.1);
for i=1:3:size(zx)

plot3([huax(i),zx(i)],[huay(i),zy(i)],[huaz(i),zz(i)'],'Color','#ff
6b81','LineWidth',0.15);
end
xlabel('x-axis, m');
ylabel('y-axis, m');
zlabel('z-axis, m');
view(3);
grid on;
hold on;
rcv_cnt = cnt;%接收到的面数量
%%
%其他点作图
huax = [];
huay = [];
huaz = [];
zx = [];
zy = [];
zz = [];
[huax,huay,huaz,zx,zy,zz,flag,~,cnt] =
calculate2ReflectPoint(zhongxin,reflect,node_class,10);

%%
for i=1:5:size(zx)

plot3([huax(i),zx(i)],[huay(i),zy(i)],[huaz(i),zz(i)],'-','Color','#a
4b0be','LineWidth',0.05);
end
scatter3(mainPoint_change_rot(:,1),mainPoint_change_rot(:,2),mainPoin
t_change_rot(:,3),'.','MarkerEdgeColor','#5352ed');
view(3);
grid on;

```

```

%%
fig2 = figure('numbertitle',2,'name','反射点情况');
fanshe = [];
for i=1:rcv_cnt
    idx = hanghao(i)
    p1 = node(idx,1)
    p2 = node(idx,2)
    p3 = node(idx,3)
    [h1,1] = find(mainPoint.textdata==p1);
    fanshe = [fanshe;mainPoint_change(h1,:)];
    [h1,1] = find(mainPoint.textdata==p2);
    fanshe = [fanshe;mainPoint_change(h1,:)];
    [h1,1] = find(mainPoint.textdata==p3);
    fanshe = [fanshe;mainPoint_change(h1,:)];
end
scatter3(fanshe(:,1),fanshe(:,2),fanshe(:,3),'o','MarkerEdgeColor','#
ff7f50');
hold on;
scatter3(mainPoint_change(:,1),mainPoint_change(:,2),mainPoint_change
(:,3),'.','MarkerEdgeColor','#5352ed');
axis equal;
xlabel('x-axis. m');
ylabel('y-axis, m');
zlabel('z-axis, m');

%%
%计算未经修改过球面
mainPoint_change_rot = [];
[m,n] = size(mainPoint_change);
for i=1:m
    pos =
[mainPoint.data(i,1),mainPoint.data(i,2),mainPoint.data(i,3)]';
    pos_p = rota(thetay,thetaz,pos);
    mainPoint_change_rot = [mainPoint_change_rot;pos_p'];
end

%%
%根据附件 3，判断三角面中有多少个点在 Q2 解得的口径 D 内的点的集合内
[m,n] = size(node);
node_class = zeros(m,1);%对三角面分类，三个节点全在口径 300 内记为 3
for i=1:m
    cnt = 0;
    for j = 1:3
        [h,1] = find(change_idx == node(i,1));

```

```

        if ~(isempty(h) && isempty(l))
            cnt = cnt + 1;
        end
    end
    node_class(i) = cnt;
end

%计算三角形面对应的中心
zhongxin = [];%三角面中心, n*3
node_norm = [];%三点的法向量
reflect = [];%反射方向向量
ru = [0 0 -1];%入射方向向量, 由于经过旋转, 使得方向竖直向下
for i=1:m
    if(node_class(i))
        [h1,l] = find(mainPoint.textdata == node(i,1));
        [h2,l] = find(mainPoint.textdata == node(i,2));
        [h3,l] = find(mainPoint.textdata == node(i,3));
        %取一个面的三个点
        p1 = mainPoint_change_rot(h1,:);
        p2 = mainPoint_change_rot(h2,:);
        p3 = mainPoint_change_rot(h3,:);
        %计算面法向量
        [node_norm(i,1) node_norm(i,2) node_norm(i,3)] = pnorm(p1,p2,p3);
        zhongxin(i,:) = (p1+p2+p3)/3;
        reflect(i,:) = -2*dot(ru,node_norm(i,:))*node_norm(i,:) + ru;
    else
        zhongxin(i,:) = NaN;
        node_norm(i,:) = [NaN NaN NaN];
        reflect(i,:) = [NaN NaN NaN];
    end
end

%%
%求解能够使反射到达有效区域的三角面数量
R = 300;
cnt = 0;
hanghao = [];%反射直线能够在给定区域口径 1 的面的行号
huax = [];
huay = [];
huaz = [];%作反射直线的一个坐标
zx = [];
zy = [];
zz = [];%作反射直线的一个坐标
flag = node_class;%分区画图的标记

```

```

[~,~,~,~,~,~,~,hanghao,ori_cnt] =
calculate2ReflectPoint(zhongxin,reflect,node_class,0.5);
%%
fig3 = figure('numbertitle',3,'name','球面反射点情况');
fanshe2 = [];
for i=1:ori_cnt
    idx = hanghao(i)
    p1 = node(idx,1)
    p2 = node(idx,2)
    p3 = node(idx,3)
    [h1,1] = find(mainPoint.textdata==p1);
    fanshe2 = [fanshe2;mainPoint_change(h1,:)];
    [h1,1] = find(mainPoint.textdata==p2);
    fanshe2 = [fanshe2;mainPoint_change(h1,:)];
    [h1,1] = find(mainPoint.textdata==p3);
    fanshe2 = [fanshe2;mainPoint_change(h1,:)];
end
scatter3(fanshe2(:,1),fanshe2(:,2),fanshe2(:,3),'o','MarkerEdgeColor'
,'#ff7f50');
hold on;
scatter3(mainPoint.data(:,1),mainPoint.data(:,2),mainPoint.data(:,3),
'.','MarkerEdgeColor','#5352ed');
axis equal;
xlabel('x-axis, m');
ylabel('y-axis, m');
zlabel('z-axis, m');
%%
%计算接受比
tot = sum(node_class)/3;
yita_change = rcv_cnt/tot;
yita_unchange = ori_cnt/tot;

```