

基于动态规划与鲸鱼优化算法的多波束测深测线布局优化模型

摘要

针对**问题 1**, 利用**初等几何**, 借助**正弦定理**分别求出船两侧波束条带覆盖宽度, 再求和得到条带覆盖宽度. 依据重叠率的定义, 推导出两侧条带覆盖宽度不等时重叠率的计算公式. 在已知条件下计算出条带覆盖宽度与重叠率的数值, 并得出结论: 条带宽度与换能器开角以及海底坡度都成正相关.

问题 2 在问题 1 的基础上增加了测线与坡面法向水平投影的夹角这一变量, 从而将问题 1 转变为三维问题. 为此, 利用**空间解析几何**, 建立空间直角坐标系, 先求得同一时刻船的所有波束所在平面的方程, 再结合换能器的开角得到最远两束波所在直线方程, 将其与坡面方程联立求出最远两束波与坡面的交点, 最后利用两点间距离公式求出条带覆盖宽度. 考虑到**题目未说明船的具体位置**, 只给出船与海域中心的距离 ρ , 本文还给出了 ρ 固定时条带宽度与方位角 φ 的关系.

针对**问题 3**, 首先根据文献说明南北走向的测线最能符合题目关于覆盖率的约束条件. 之后利用**跳跃式迭代**、**贪心算法**求解动态规划问题: 在保证相邻条带能够有 10.5% 覆盖率的基础上, 根据上一条测线位置计算出下一条测线的最远位置, 直到增加的条带能够覆盖整个海域为止. 最终得出最少需要 34 条南北走向的测线, 西边稀疏、东边稠密, 测线总长度最短为 68 n mile.

针对**问题 4**, 本文给出 2 种方法. **分块法**依据海底地形将矩形海域分割为西北和东南两部分, 并在各部分中得到曲面的近似平面. 之后同问题 3 的思路, 在两部分中分别构造与等深线走向大体一致的、两两平行的测线. 最终得到的最短测线总长为 3.588×10^5 m, 重叠率超出 20% 的测线总长最短为 31 288.4 m, 而漏测率为 0. 若考虑**曲线测线**, 则采用**曲线微元法**近似优化求解: 采样一系列等深线作为可选测线, 然后将覆盖海域与重叠率作为约束条件, 以测线总长度作为目标函数使用**鲸鱼优化算法**进行 0-1 规划问题的优化求解, 计算出题目要求的 3 个值.

总之, 在多波束测深布线前, 要提前了解待测海域海底地形, 最好提前通过单波束测深等技术粗略得知海底地貌, 之后**因地制宜**布设测线. 为使得测线总长度尽可能短, 节省成本, **地形简单时利用分块法与动态规划**, **地形复杂时利用曲线微元法**. 布设测线时应注意重叠率满足 10% ~ 20% 的约束条件.

关键词: 多波束测深, 测线设计, 初等几何建模, 动态规划, 曲线微元法, 鲸鱼优化算法

一、问题重述

1.1 背景知识

海洋的深度可以通过多种方法来测量。单波束测深是一种使用声波来测量水深的方
法。单波束测深仪是一种简单、经济的测深技术,适用于一般的水深测量任务。但是单
波束测深仪通常只能提供一条声波束的测量结果,对于精确地描绘水底地形和深度分布
可能有限。

多波束测深利用安装于船底或拖体上的声基阵向与航向垂直的海底发射超宽声波
束,接收海底反向散射信号,经过模拟/数字信号处理,形成多个波束,同时获得几十个甚
至上百个海底条带上采样点的水深数据。相比于单波束测深,多波束测深通过同时使用
多个发射和接收波束,可以获得更多的返回信号,从而提高深度测量的精度;通过同时使
用多个波束对水下目标进行更精细的探测和成像,从而增加测量数据的分辨率;可以同
时发射多个波束,从而在单位时间内获得更多的测量数据,提高测量速度;可以通过同时
使用多个波束,覆盖更大的水下区域,提高侧向覆盖范围。

多波束测深的测线设计对获取高质量的海底地形数据非常重要^[1]。测线设计首先要
确保覆盖调查区域,不留空白。在多波束测深中,相邻测线之间需要有一定的重叠区域。
如果相邻测线没有重叠,由于多波束本身的扇形声束,会出现测线之间的间隙区域没有
数据。适当重叠可以避免这种数据空窗。重叠区域获取了两次(及以上)的数据,可以
提高该区域的数据密度,有利于描绘细节地形。重叠区域还可以用来校核相邻测线的数
据精度,发现误差以校正。一定的重叠可以提高数据冗余度,有利于后期的数据处理。所
以,合理的重叠是保证多波束测量质量的重要手段,可以弥补技术缺陷和提高数据可靠
性。但是,若重叠率过高,则数据冗余和重复信息增加,使测量效率下降。因此,一般重叠
率设为 10% ~ 20%^[2]。

1.2 具体问题

1.2.1 问题 1

海底坡面为理想平面,坡度已知。测线方向与海底坡面平行。借助多波束换能器的
开角与某点处海水深度,计算多波束测深的覆盖宽度及相邻条带之间的重叠率。

1.2.2 问题 2

海底坡面为理想平面,坡度已知。测线方向与海底坡面不一定平行,但测线方向已
知。借助多波束换能器的开角与某点处海水深度,计算多波束测深的覆盖宽度。

1.2.3 问题 3

在西深东浅的矩形海域内, 海底坡面为理想平面, 坡度已知. 已知多波束换能器的开角与某点处海水深度, 设计一组测线使得总测线长度最短, 同时满足: 1. 测线可完全覆盖整个待测海域; 2. 相邻条带之间的重叠率位于闭区间 $[10\%, 20\%]$.

1.2.4 问题 4

利用矩形海域内单波束测量的测深数据, 设计多波束测量船的测量布线, 使得: 1. 沿测线扫描形成的条带占总待测海域面积的百分比尽可能大; 2. 在重叠区域中, 重叠率超过 20% 部分的测线总长度尽可能短; 3. 测线的总长度尽可能短. 并给出最优布线的相应指标: 1. 沿测线扫描形成的条带占总待测海域面积的百分比; 2. 在重叠区域中, 重叠率超过 20% 部分的测线总长度; 3. 测线的总长度.

1.3 相关数据

1.3.1 问题 1

- 海底坡面的坡度为 1.5° ;
- 多波束换能器的开角为 120° ;
- 海域中心点处的海水深度为 70 m;
- 9 条测线距中心点处的有向距离分别为 -800 m , -600 m , -400 m , -200 m , 0 , 200 m , 400 m , 600 m , 800 m .

1.3.2 问题 2

- 海底坡面的坡度为 1.5° ;
- 多波束换能器的开角为 120° ;
- 海域中心点处的海水深度为 120 m;
- 测量船距海域中心点处的距离分别为 0 , 0.3 n mile , 0.6 n mile , 0.9 n mile , 1.2 n mile , 1.5 n mile , 1.8 n mile , 2.1 n mile ;
- 测线方向与海底坡面的法向在水平面上投影的夹角有 0° , 45° , 90° , 135° , 180° , 225° , 270° , 315° .

1.3.3 问题 3

- 海底坡面的坡度为 1.5° ;
- 多波束换能器的开角为 120° ;
- 海域中心点处的海水深度为 110 m;

- 矩形海域南北长 2 n mile , 东西宽 4 n mile .

1.3.4 问题 4

- 矩形海域南北长 5 n mile 海里、东西宽 4 n mile 海里;
- 附件.xlsx 提供了矩形海域单波束测量的测深数据, 位置精确到 0.02 n mile .

二、问题分析

2.1 问题 1 分析

如图1所示, 测线即 C 点处垂直于纸面的直线. 假设测线处海水深度 D 已知, 在纸面内过 A 作 AF 平行于海平面, 有 $\angle BAF = \alpha$. CG 为竖直方向, 延长 CG 交 AF 于 E , 则 $CE \perp AF$. 过 B 作 $BF \perp AF$ 于 F . 则解 $\triangle ACG$ 与 $\triangle BCG$ 可得 AG 与 GB 的长度, 进而得到 AE 与 EF 的长度, 记 $W^{(1)} = AE$, $W^{(2)} = EF$, 注意到 AF 与龙骨垂直, 得条带覆盖宽度 $W = W^{(1)} + W^{(2)}$.

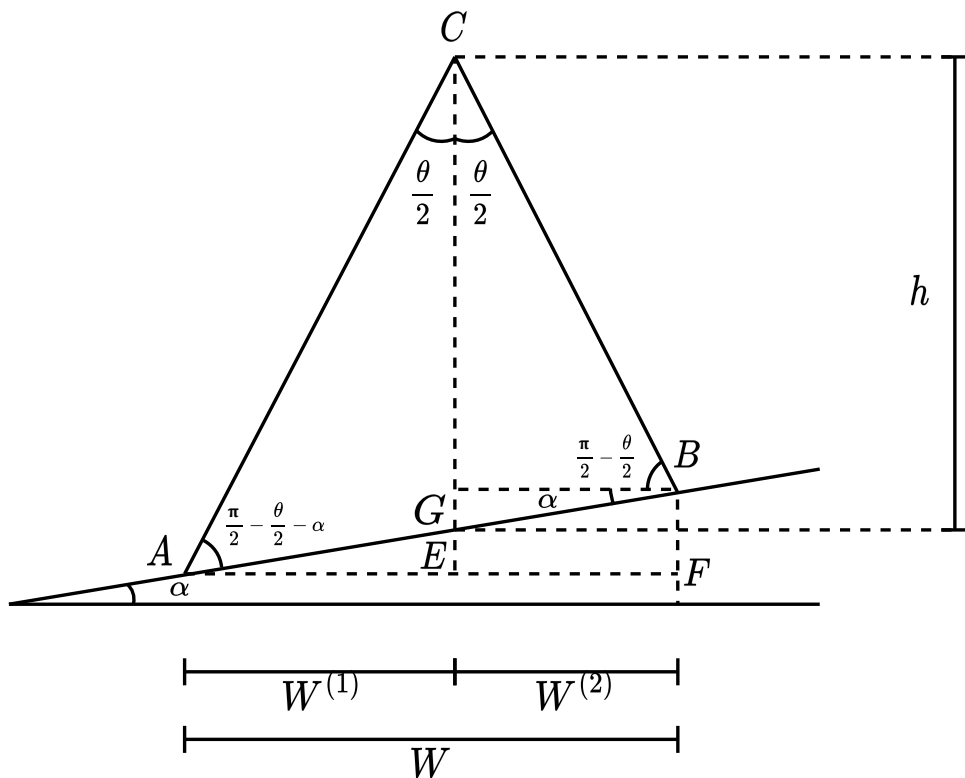


图 1 问题 1 的示意图

在具体问题中, 若已知测线距中心点处 (中心点处深度已知) 的距离, 则可由初等几何计算出测线处海水深度, 转化为上面的情形.

因不同测线处海水深度不同, 则相邻测线条带覆盖宽度不同, 故采用公式(3)计算重叠率.

2.2 问题 2 分析

以海域中心的深 $D_0 = 120\text{ m}$ 处 (即海域中心的底部) 为原点, 以指向天空的坡面法向量在水平面的投影向量为 x 轴正方向, 以水平面内与 x 轴垂直的非零向量为 y 轴正方向, 以竖直向上为 z 轴正方向, 建立如图2所示的右手空间直角坐标系.

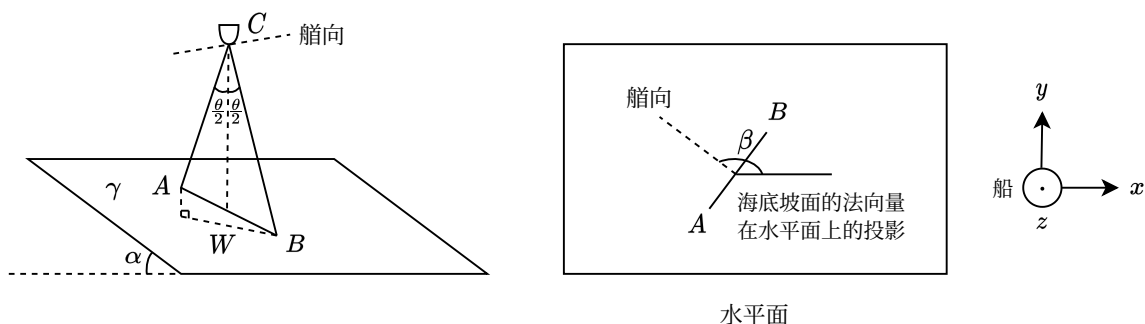


图 2 问题 2 的示意图

如图3, 测量船距海域中心点处的距离 ρ 已知时, 测量船可能的位置为海平面上以一个海域中心点为圆心的圆, 设此时测量船 (换能器) 的坐标为

$$C(\rho \cos \varphi, \rho \sin \varphi, D_0), \quad (1)$$

其中 φ 未知.

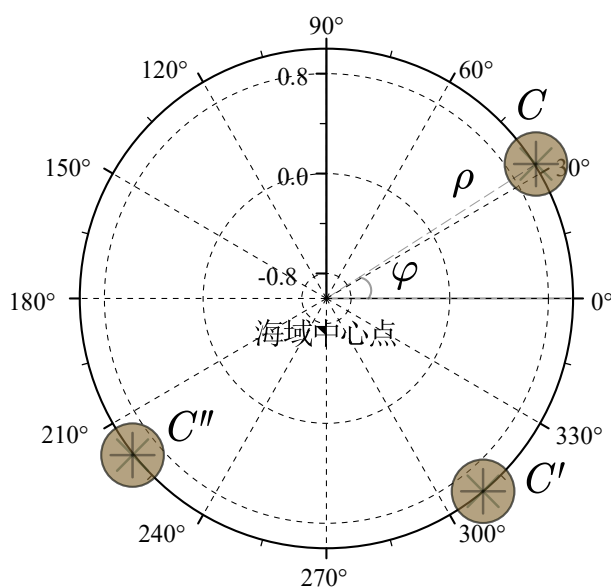


图 3 问题 2 海平面

利用空间解析几何, 先求得 C 处所有波束所在的平面方程, 结合换能器的开角可得最远两束波所在的直线方程, 将其与坡面方程联立, 利用空间与平面两点间距离公式即可求出条带覆盖宽度.

特别地, 当 φ 给定时, 可求得条带覆盖宽度的数值.

2.3 问题 3 分析

测线布设包括测线方向和测线间距选择两个方面的问题. 中华人民共和国交通运输行业标准《多波束测深系统测量技术要求》指出, 测深线布设方向应按工程的需要选择平行于等深线走向、潮流的流向、航道轴线方向或测区的最长边等其中之一布设^[3]. 成芳等指出, 对于测线方向, 通常为使扫测宽度尽可能保持不变, 测线方向尽量与等深线方向保持一致^[1]. 事实上, 若测线不沿等深线, 则在深水区和浅水区的条带覆盖宽度差异悬殊, 导致深水区重叠率过高, 浅水区重叠率过低, 不符合 10% ~ 20% 的标准.

在问题 3 中, 因海底坡面为理想平面, 等深线为直线.

利用问题 2 的方法, 可求得当艏向为北偏东 15° 时, 最大条带覆盖宽度为 427.0 m, 最小条带覆盖宽度为 337.0 m, 差异较大, 这验证了上述规律, 详见附录C的3.3.1.

因此, 下面只考虑艏向沿等深线的情况. 由等深线平行于海底坡面知, 同一测线上各处的条带覆盖宽度相同, 与前一条测线相应位置的重叠率也相同. 可通过调整测线间距来控制重叠率属于闭区间 $[10\%, 20\%]$. 注意到测线宽度越大, 重叠率越小, 为保证测线总长度最短, 利用动态规划尽可能加大测线宽度, 同时使重叠率在标准范围内尽可能小, 从而在矩形海域内减少测线数量, 进而减少测线总长度.

按自西向东的顺序依次选择每一位置是否设置测线. 本阶段测线位置的选择将影响到下一阶段测线位置的选择, 以至于影响测线数量与测线总长度.

动态规划方法基于贝尔曼 (R. Bellman) 等人提出的最优化原理, 即“一个过程的最优策略具有这样的性质: 即无论初始状态及初始决策如何, 对于先前决策所形成的状态而言, 其后的所有决策均应构成最优策略”^[4]. 由于寻优的方向与多阶段决策过程的实际行进方向相反, 基于上述思想的动态规划求解方法称为逆序解法 (后向动态规划方法). 而本文采用顺序解法 (前向动态规划方法): 寻优方向与过程的行进方向相同, 计算时从西开始向东迭代, 矩形海域东边界计算得到的测线总数就是全过程的最少测线数.

2.4 问题 4 分析

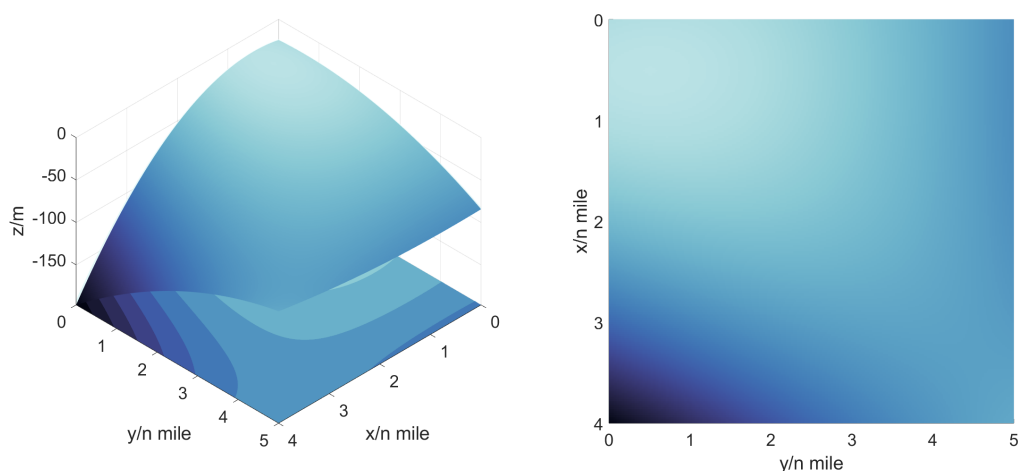


图 4 海底地形图

2.4.1 分块法

观察图4可知,海底平面可沿“对角线”分为两块较平坦的曲面.将每块曲面近似为平面,利用问题 3 的分析思路即可.

事实上,分块法是一种在简单地貌中顶层设计测线的方法,直接使得沿测线扫描形成的条带占总待测海域面积的百分比尽可能大,重叠率超过 20% 部分的测线总长度尽可能短,测线的总长度尽可能短,而无需对上述三个目标进行专门优化.

2.4.2 曲线微元法

根据中华人民共和国交通运输部《多波束测深系统测量技术要求》^[3],测线应该平行于等深线走向.由于问题 4 中海底地形不是斜面且不规则,所以等深线不是直线,沿着等深线意味着要考虑曲线测线的情况.

本文首先采样一系列等高线作为可选的测线,然后这些等高线的组合作为决策变量得到测线.以待测海域与重叠率为约束条件,以测线的总长度为目标函数进行目标优化求解,即将三目标规划问题转化为单目标规划问题.以附件中的海水深度数据建立海底深度的数据集,接着然后利用线性插值法得到海底深度在二维平面上的空间分布函数.对矩形海域的二维平面进行离散化处理,使用微元法(数值积分)近似求解测线的待测海域与重叠率,利用数值曲线积分求解测线的总长度.

三、模型假设

- 因本题待测海域面积较小,将海平面近似为水平面,测线 \subset 海平面;

- 使用单一频率的声波进行测量, 不考虑声速随海水深度的增大而增大, 声波在水中的传播不受海洋生物等的影响;
- 船绝对沿测线航行, 不因风向等因素造成航向偏离;
- 艏向测量仪安装在测量船的艏艉 (龙骨) 线上, 参考方向指向船艏, 将换能器与测量船都近似为测线上一点;
- 忽略问题 4 中单波束测深数据可能的误差;
- 直线测线彼此平行;
- 忽略船的艏摇 (yaw)、横滚 (roll)、俯仰 (pitch)、升沉 (heave)、横荡 (sway)、纵荡 (surge) 对定位和测深的影响 (6 个姿态量各自含义如图5所示)^[5].

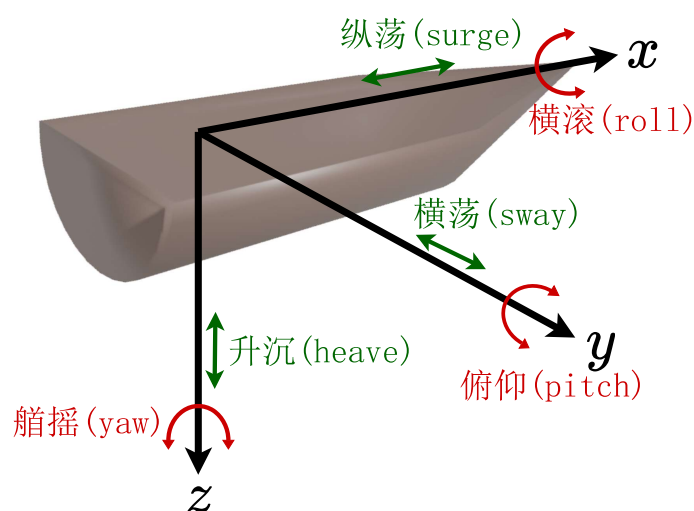


图 5 声呐平台姿态示意图

四、名词解释与符号说明

4.1 名词解释^[2]

声呐 (sonar) 利用声波在水中的传播和反射来进行测距的技术或设备.

波束 (beam) 具有特定空间指向性的声波, 并以此获得采样点水深信息.

多波束测深仪 (multibeam echo sounder) 采用声学波束形成技术, 能同时获得呈条带状分布的多个水深值的装置.

艏向 船头所指方向, 即测线正方向.

多波束测深系统 (multibeam bathymetric system) 由多波束测深仪及其相关外部设备 (定位仪、数字艏向测量仪、运动传感器、表层声速计、声速剖面仪等) 和多波束数据后处理软硬件组成的系统.

多波束水深测量 (multibeam bathymetric survey) 利用多波束测深系统进行海洋水深测量的活动.

条带 (swath) 发射波束与海底反射接收波束相交形成一系列的波束脚印, 它们是接收的声呐 (或声波) 信号中包含海底地形信息的一组采样点, 排列形成一个带状区域.

龙骨 (keel) 船体的基底中央连接船首柱和船尾柱的一个纵向构件, 起支撑和承重作用.

条带覆盖宽度 (swath coverage width) 条带在龙骨垂直方向上的长度.

重叠率 (overlap coverage rate) 相邻条带重叠部分的宽度与条带覆盖宽度的百分比.

- 相邻条带覆盖宽度相等时, 有

$$\text{重叠率} = \frac{\text{重叠部分宽度}}{\text{条带覆盖宽度}} = \frac{\text{条带覆盖宽度} - \text{测线间距}}{\text{条带覆盖宽度}} = 1 - \frac{\text{测线间距}}{\text{条带覆盖宽度}}; \quad (2)$$

- 相邻条带覆盖宽度不等时, 定义平均条带覆盖宽度 = (一侧条带覆盖宽度 + 另一侧条带覆盖宽度)/2, 有

$$\begin{aligned} \text{重叠率} &= \frac{\text{重叠部分宽度}}{\text{平均条带覆盖宽度}} \\ &= \frac{\text{一侧条带覆盖宽度}/2 + \text{另一侧条带覆盖宽度}/2 - \text{测线间距}}{\text{平均条带覆盖宽度}} \quad (3) \\ &= 1 - \frac{\text{测线间距}}{\text{平均条带覆盖宽度}}. \end{aligned}$$

换能器 (transducer) 用于发射声波信号并接收从海底反射回来的声波信号的装置. 它负责将电能转化为声能, 然后再将声能转化为电能, 从而实现声波的发射和接收.

4.2 符号说明

表 1 符号说明

| 符号 | 意义 |
|---------------|------------------------------|
| C | 换能器所在点 |
| γ | 海底坡面（所在平面） |
| l_1, l_2 | 最远波束所在直线 |
| α | 海底坡面与水平面的夹角（rad） |
| θ | 多波束换能器的开角（rad） |
| β | 测线方向与海底坡面的法向在水平面上投影的夹角（rad） |
| φ | 测量船位置相对海域中心点的极角（rad） |
| D | 换能器到海底的距离（m） |
| d | 测线间距（m） |
| i | 问题 1 测线编号 |
| W | 条带覆盖宽度（m） |
| η | 重叠率 |
| ρ | 测量船距海域中心点处的距离（即极径）（n mile） |
| $\pm z_l$ | 最远波束所在直线方向向量的竖坐标 |
| $\Delta \rho$ | ρ 的步长（n mile） |
| i' | 问题 2 极径编号 |
| j | 问题 2 艏向编号 |
| p | 矩形海域东西宽（n mile） |
| q | 矩形海域南北长（n mile） |
| t | 问题 3 中测线与海域中心点的有向距离（“时间”）（m） |
| s_t | t 时刻的状态 |
| $u_t(s_t)$ | t 时刻当状态为 s_t 时的决策变量 |

续表

| 符号 | 意义 |
|-------------------|--|
| $v_t(s_t, u_t)$ | 时刻指标函数 |
| $f_t(s_t)$ | 最优指标函数 |
| t_{\max} | 满足 $u_t = 1$ 的 t 的最大值 (m) |
| t_{last} | 迭代过程中上一个满足 $u_t(s_t) = 1$ 的 t (m) |
| t_{this} | 迭代过程中满足 $u_t(s_t) = 1$ 的待确定位置的 t (m) |
| k | 问题 4 东南部分水平等高线在 xOy 平面内投影的斜率 |
| P | 误差分析中某一半矩形海平面内的采样点 |
| MAE | 平均绝对误差 |
| CV | 变异系数 |

五、模型建立与求解

5.1 问题 1 的建模与求解

5.1.1 模型建立

如图1, $\angle BAF = \alpha$, $CG = D$, 将条带覆盖宽度 $W = AF$ 拆分成左右两个部分 $W^{(1)}, W^{(2)}$, 其中 $W^{(1)} = AE$, $W^{(2)} = AF$.

在 $\triangle ACG$ 中, $\angle CAG = \frac{\pi}{2} - \frac{\theta}{2} - \alpha$, 由正弦定理,

$$\frac{AG}{\sin \frac{\theta}{2}} = \frac{CG}{\sin \angle CAG}. \quad (4)$$

故有

$$AG = \frac{(\sin \frac{\theta}{2})D}{\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha)}. \quad (5)$$

$$W^{(1)} = AE = AG \cdot \cos \alpha = \frac{(\sin \frac{\theta}{2})(\cos \alpha)D}{\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha)}. \quad (6)$$

$\angle ABC = 180^\circ - \angle BAC - \theta = \frac{\pi}{2} - \frac{\theta}{2} + \alpha$. 在 $\triangle BCG$ 中, 由正弦定理,

$$\frac{BG}{\sin \frac{\theta}{2}} = \frac{CG}{\sin \angle ABC}. \quad (7)$$

故有

$$BG = \frac{(\sin \frac{\theta}{2})D}{\sin(\frac{\pi}{2} - \frac{\theta}{2} + \alpha)}. \quad (8)$$

$$W^{(2)} = EF = BG \cdot \cos \alpha = \frac{(\sin \frac{\theta}{2})(\cos \alpha)D}{\sin(\frac{\pi}{2} - \frac{\theta}{2} + \alpha)}. \quad (9)$$

于是条带宽度为

$$\begin{aligned} W &= W^{(1)} + W^{(2)} \\ &= D \left(\sin \frac{\theta}{2} \right) (\cos \theta) \left[\frac{1}{\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha)} + \frac{1}{\sin(\frac{\pi}{2} - \frac{\theta}{2} + \alpha)} \right] \\ &= D \left(\sin \frac{\theta}{2} \right) (\cos \theta) \cdot \frac{\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha) + \sin(\frac{\pi}{2} - \frac{\theta}{2} + \alpha)}{[\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha)] \sin(\frac{\pi}{2} - \frac{\theta}{2} + \alpha)} \\ &= D \left(\sin \frac{\theta}{2} \right) (\cos \theta) \cdot \frac{2 [\sin(\frac{\pi}{2} - \frac{\theta}{2})] \cos(-\alpha)}{-\frac{1}{2} [\cos(\pi - \theta) - \cos(-2\alpha)]} \\ &= D \left(\sin \frac{\theta}{2} \right) (\cos \theta) \cdot \frac{4 (\cos \frac{\theta}{2}) \cos \alpha}{\cos \theta + \cos(2\alpha)} \\ &= D \cdot \frac{2(\sin \theta)(\cos \theta) \cos \alpha}{\cos \theta + \cos(2\alpha)} \\ &= D \cdot \frac{[\sin(2\theta)] \cos \alpha}{\cos \theta + \cos(2\alpha)}. \end{aligned} \quad (10)$$

5.1.2 模型求解

已知海域中心点处的海水深度 $D_0 = 70 \text{ m}$, 测线间距 $d = 200 \text{ m}$, 则“第 i ”条测线 ($i = -4, -3, -2, -1, 0, 1, 2, 3, 4$) 处海水深度为

$$D_i = D_0 - i \cdot d \cdot \tan \alpha. \quad (11)$$

当 $\alpha = 1.5^\circ = \frac{\pi}{120}$, $\theta = 120^\circ = \frac{2}{3}\pi$ 时, “第 i ”条测线处海水深度为

$$W_i = D_i \cdot \frac{[\sin(2\theta)] \cos \alpha}{\cos \theta + \cos(2\alpha)}. \quad (12)$$

由公式(3)得, “第 i ”条测线 ($i = -3, -2, -1, 0, 1, 2, 3, 4$) 与前一条测线的重叠率为

$$\eta_i = 1 - \frac{d}{(W_{i-1} + W_i)/2} \quad (13)$$

$$= 1 - \frac{\cos \theta + \cos(2\alpha)}{[\sin(2\theta)] \cos \alpha} \cdot \frac{2d}{2D_0 - (2i - 1)d \tan \alpha}. \quad (14)$$

利用 Python 求得结果, 见表2.

表 2 问题 1 的计算结果

| 测线编号 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 |
|--------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 测线距中心点处的距离/m | 800 | -600 | -400 | -200 | 0 | 200 | 400 | 600 | 800 |
| 海水深度/m | 90.95 | 85.71 | 80.47 | 75.24 | 70.00 | 64.76 | 59.53 | 54.29 | 49.05 |
| 覆盖宽度/m | 315.71 | 297.53 | 279.35 | 261.17 | 242.99 | 224.81 | 206.63 | 188.45 | 170.27 |
| 与前一条测线的重叠率/% | | 34.77 | 30.66 | 26.00 | 20.66 | 14.49 | 7.29 | -1.25 | -11.51 |

从表2中可看出, 若测线间距不随海水深度的改变而改变, 则重叠率随海水深度的增加而增加, 因此过深与过浅的海水处无法满足 10% ~ 20% 的重叠率要求, 过浅处甚至存在漏测.

5.2 问题 2 的建模与求解

5.2.1 模型建立

由 $x \neq 0$ 时 $\tan \alpha = \frac{-z}{x}$, 得坡面 γ 的方程为

$$\gamma: (\tan \alpha)x + z = 0. \quad (15)$$

艏向单位方向向量为

$$(\cos \beta, \sin \beta, 0). \quad (16)$$

则同一时刻所有波束所在平面方程 ((16)为其一个法向量) 为

$$(\cos \beta)(x - \rho \cos \varphi) + (\sin \beta)(y - \rho \sin \varphi) = 0, \quad (17)$$

即

$$(\cos \beta)x + (\sin \beta)y - \rho \cos(\beta - \varphi) = 0. \quad (18)$$

该平面上最远的波束所在直线 l_1, l_2 必与该平面的法向量垂直, 统一设 l_1, l_2 的方向向量为

$$(\sin \beta, -\cos \beta, \pm z_l). \quad (19)$$

其中 $z_l > 0$. 由

$$\cot \frac{\theta}{2} = \frac{z_l}{\sqrt{\sin^2 \beta + (-\cos \beta)^2}}, \quad (20)$$

知 $z_l = \cot \frac{\theta}{2}$, 故 l_1, l_2 的方向向量为

$$(\sin \beta, -\cos \beta, \pm \cot \frac{\theta}{2}). \quad (21)$$

注意到 $C \in l_1, C \in l_2$, 不妨

$$l_1: \frac{x - \rho \cos \varphi}{\sin \beta} = \frac{y - \rho \sin \varphi}{-\cos \beta} = \frac{z - D_0}{\cot \frac{\theta}{2}}, \quad (22)$$

$$l_2: \frac{x - \rho \cos \varphi}{\sin \beta} = \frac{y - \rho \sin \varphi}{-\cos \beta} = \frac{z - D_0}{-\cot \frac{\theta}{2}}. \quad (23)$$

5.2.2 模型求解

$$\begin{cases} \frac{x - \rho \cos \varphi}{\sin \beta} = \frac{y - \rho \sin \varphi}{-\cos \beta} = \frac{z - D_0}{\cot \frac{\theta}{2}}, \\ (\tan \alpha)x + z = 0. \end{cases} \Rightarrow \begin{cases} x_1 = \frac{\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha}, \\ y_1 = \frac{D_0 \cos \beta + \rho[(\cot \frac{\theta}{2}) \sin \varphi + (\tan \alpha) \cos(\beta - \varphi)]}{\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha}, \\ z_1 = -\frac{\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{(\cot \frac{\theta}{2}) \cot \alpha + \sin \beta}. \end{cases} \quad (24)$$

$$\begin{cases} \frac{x - \rho \cos \varphi}{\sin \beta} = \frac{y - \rho \sin \varphi}{-\cos \beta} = \frac{z - D_0}{-\cot \frac{\theta}{2}}, \\ (\tan \alpha)x + z = 0. \end{cases} \Rightarrow \begin{cases} x_2 = \frac{-\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{-\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha}, \\ y_2 = \frac{D_0 \cos \beta + \rho[-(\cot \frac{\theta}{2}) \sin \varphi + (\tan \alpha) \cos(\beta - \varphi)]}{-\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha}, \\ z_2 = -\frac{-\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{-(\cot \frac{\theta}{2}) \cot \alpha + \sin \beta}. \end{cases} \quad (25)$$

因此,

$$AB = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}. \quad (26)$$

条带覆盖宽度为 AB 在水平面上的投影长度, 即

$$\begin{aligned} W &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ &= \left\{ \left[\frac{\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha} - \frac{-\rho(\cos \varphi) \cot \frac{\theta}{2} - D_0 \sin \beta}{-\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha} \right]^2 + \right. \\ &\quad \left. \left[\frac{D_0 \cos \beta + \rho[(\cot \frac{\theta}{2}) \sin \varphi + (\tan \alpha) \cos(\beta - \varphi)]}{\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha} - \frac{D_0 \cos \beta + \rho[-(\cot \frac{\theta}{2}) \sin \varphi + (\tan \alpha) \cos(\beta - \varphi)]}{-\cot \frac{\theta}{2} + (\sin \beta) \tan \alpha} \right]^2 \right\}^{\frac{1}{2}}. \end{aligned} \quad (27)$$

ρ 的步长 $\Delta \rho = 0.3 \text{ n mile}$, β 的步长 $\Delta \beta = 45^\circ = \frac{\pi}{4}$. $\rho_i = i' \Delta \rho (i' = 0, 1, 2, 3, 4, 5, 6, 7)$, $\beta_j = j \Delta \beta (j = 0, 1, 2, 3, 4, 5, 6, 7)$. 当 φ 已知时, 将基本量代入(27)式即可求得条带覆盖宽度. 特别地, 当 $\varphi = 0$ 时, 利用 MATLAB 求得结果见表3.

表 3 问题 2 的计算结果

| 覆盖宽度 | | 测量船距海域中心点处的距离/n mile | | | | | | | |
|--------------------------|-----|----------------------|--------|--------|--------|--------|--------|--------|--------|
| /m | | 0 | 0.3 | 0.6 | 0.9 | 1.2 | 1.5 | 1.8 | 2.1 |
| 测线 方向 夹角 / ° | 0 | 415.69 | 466.09 | 516.49 | 566.89 | 617.29 | 667.69 | 718.09 | 768.48 |
| | 45 | 416.12 | 466.57 | 517.02 | 567.47 | 617.92 | 668.37 | 718.82 | 769.28 |
| | 90 | 416.55 | 467.05 | 517.55 | 568.06 | 618.56 | 669.06 | 719.57 | 770.07 |
| | 135 | 416.12 | 466.57 | 517.02 | 567.47 | 617.92 | 668.37 | 718.82 | 769.28 |
| | 180 | 415.69 | 466.09 | 516.49 | 566.89 | 617.29 | 667.69 | 718.09 | 768.48 |
| | 225 | 416.12 | 466.57 | 517.02 | 567.47 | 617.92 | 668.37 | 718.82 | 769.28 |
| | 270 | 416.55 | 467.05 | 517.55 | 568.06 | 618.56 | 669.06 | 719.57 | 770.07 |
| | 315 | 416.12 | 466.57 | 517.02 | 567.47 | 617.92 | 668.37 | 718.82 | 769.28 |

由表3, β 变化 180° 时, 在 ρ 一定的条件下, 条带覆盖宽度不变, 这符合实际情况. 由于 α 极小, 在 ρ 一定时, 条带覆盖宽度变化幅度极小.

5.3 问题 3 的建模与求解

5.3.1 模型建立

已知矩形海域南北长 $q = 2 \text{ n mile}$ 、东西宽 $p = 4 \text{ n mile}$, 海域中心点处的海水深度 $D_0 = 110 \text{ m}$.

下面引入“时间”因素, 将该规划问题转为动态规划问题. 设自西向东方向为时间轴 t , 以米作为“时间”单位, 每一“时刻” t ($-\frac{p}{2} \leq t \leq \frac{p}{2}$) 所处的位置 (用东西方向上到海域中心点的有向距离刻画) 的允许决策集合为 $D_t(s_t) = \{1, 0\}$, 其中 1 表示在此处设置测线, 0 表示在此处不设置测线, 每一时刻 t 的状态 s_t 定义为该时刻之前的测线总数 (每条测线长度均为 q). 决策变量 $u_t(s_t) \in D_t(s_t)$. 状态转移方程为

$$s_{t+} = T_t(s_t, u_t) = s_t + u_t. \quad (28)$$

规划目标为

$$\min_{-\frac{p}{2} \leq t \leq \frac{p}{2}} \sum u_t. \quad (29)$$

时刻指标函数为

$$v_t(s_t, u_t) = u_t. \quad (30)$$

最优指标函数 $f_t(s_t)$ 表示在该时刻及之前测线总数的最小值.

在西边界设置测线会造成波束的浪费, 因此 $u_{-\frac{p}{2}} = 0$, 即 $f_{-\frac{p}{2}}(s_{-\frac{p}{2}}) = 0$, 得动态规划的基本方程为

$$\begin{cases} f_t(s_t) = \min_{u_t \in D_t \text{ 且重叠率合适}} [v_t(s_t, u_t) + f_{t-}(s_t)], \\ f_{-\frac{p}{2}}(s_{-\frac{p}{2}}) = 0. \end{cases} \quad (31)$$

设满足 $u_t = 1$ 的 t 的最大值为 t_{\max} , 由(11)式得,

$$D_{t_{\max}} = D_0 - t_{\max} \tan \alpha. \quad (32)$$

由(6)式得,

$$W_{t_{\max}}^{(1)} = \frac{(\sin \frac{\theta}{2})(\cos \alpha) D_{t_{\max}}}{\sin(\frac{\pi}{2} - \frac{\theta}{2} - \alpha)}. \quad (33)$$

5.3.2 模型求解

利用 MATLAB 解方程

$$\frac{p}{2} + t_{\max} = W_{t_{\max}}^{(1)}, \quad (34)$$

得

$$t_{\max} = -3355.0 \text{ m}. \quad (35)$$

下面从 t_{\max} 或之前位置 (时刻) 开始进行跳跃式迭代. 跳跃式迭代的原理是尽可能加大测线宽度, 同时使重叠率在标准范围内尽可能小, 从而在矩形海域内减少测线数量. 但为防止偶然因素导致重叠率低于 10%, 本文以 $\eta_0 = 10.5\%$ 的重叠率, 安排测线间的间隔.

在迭代过程中, 设已确定 $u_{t_{\text{last}}} = 1$, 且已知 t_{last} 与 $W_{t_{\text{last}}}$, 求下一个 t , 即 t_{this} , 使得 $u_t = 1$.

由(11)式得,

$$D_{t_{\text{this}}} = D_0 - t_{\text{this}} \tan \alpha. \quad (36)$$

注意到测线间隔为 $t_{\text{this}} - t_{\text{last}}$, 由(3)式有

$$\eta_0 = 1 - \frac{t_{\text{this}} - t_{\text{last}}}{(W_{\text{last}} + W_{\text{this}})/2}. \quad (37)$$

由(10)及(36)式有

$$W_{\text{this}} = (D_0 - t_{\text{this}} \tan \alpha) \cdot \frac{[\sin(2\theta)] \cos \alpha}{\cos \theta + \cos(2\alpha)}. \quad (38)$$

将(38)代入(37), 即可由 t_{last} 求得 t_{this} , 按此迭代, 直至 $t \geq \frac{p}{2}$ 时终止迭代. 利用 MATLAB 从 $t = -3364.00 \text{ m}$ 进行迭代, 得到在矩形海域海平面内, 南北方向的测线 t 坐标 (单位:

m) 分别为 $-3364.00, -2772.64, -2227.50, -1724.99, -1261.75, -834.73, -441.09, -78.22, 256.28, 564.63, 848.88, 1110.90, 1352.45, 1575.11, 1780.36, 1969.57, 2143.99, 2304.77, 2452.99, 2589.61, 2715.56, 2831.66, 2938.69, 3037.34, 3128.29, 3212.13, 3289.41, 3360.65, 3426.32, 3486.86, 3542.67, 3594.11, 3641.53, 3685.25$, 共 34 条测线, 布局见图6. 此时测线总长度最短, 为 $34q$, 即 68 n mile.

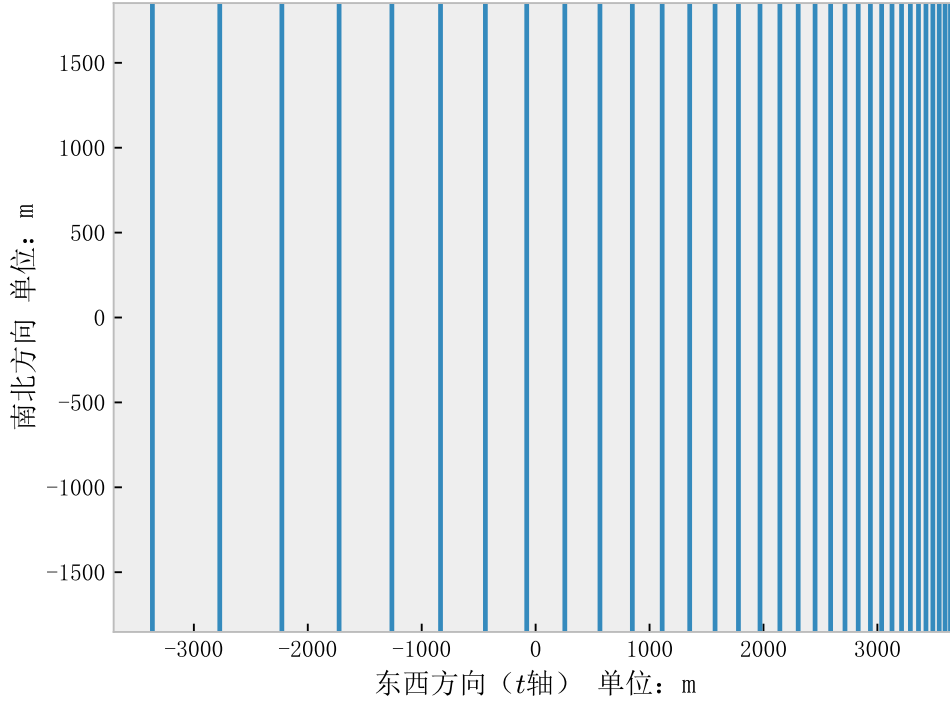


图 6 问题 3 最优测线布局

事实上, 若从 $t = t_{\max}$ 开始迭代, 也可得 34 条测线; 若从 $t = -\frac{p}{2}$ 开始迭代, 可得 35 条测线. 故最少需要 34 条测线, 但测线的位置不唯一.

5.4 问题 4 的建模与求解

5.4.1 分块法模型建立

利用矩形海域单波束测量的测深数据, 以矩形海域西南角为坐标原点, 自西向东为 x 轴正方向, 自南向北为 y 轴正方向, 绘制海底地形等深线图如图7. 在图7中, 浅蓝色表示浅水区, 紫色表示深水区.

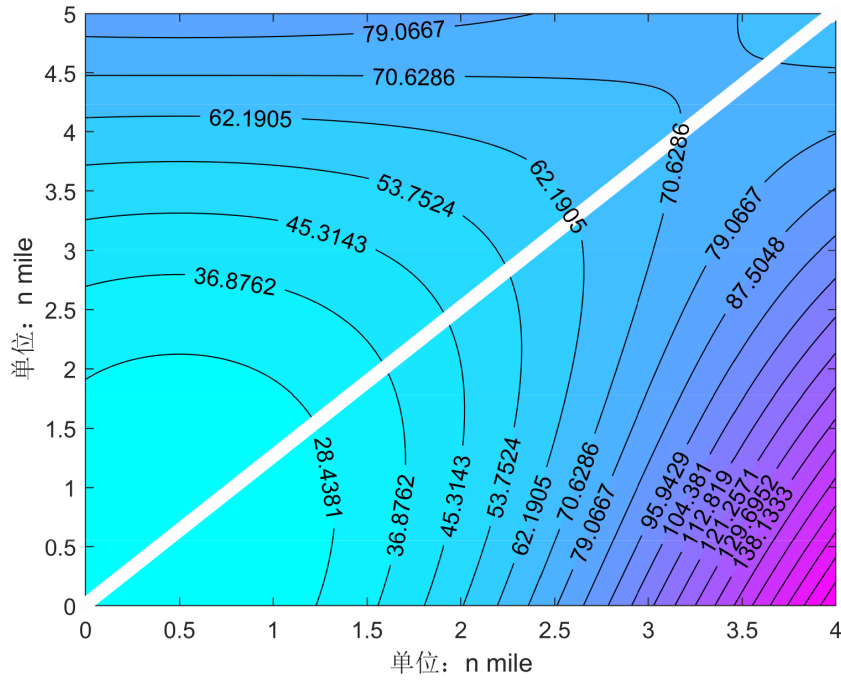


图7 海底地形等深线图

图7的等深线可明显地分为两种走向, 以西南角与东北角的连线为分界线. 其中西北部分等深线几乎沿东西方向, 东南部分等深线方向约为北偏东 15° .

假设测线为直线, 同问题3的分析, 应尽量使测线沿等深线.

观察图4, 发现海底地形“曲面”较为光滑, 且西北部分与东南部分各自较为平坦, 故采用拟合平面进行近似计算. 误差分析详见6.2.

在西北部分区域采用沿东西方向的测线, 拟合平行于东西方向的平面.

在东南部分, 利用东北角、东南角、西南角三点插值得到近似平面方程为

$$x - 0.6111y + 42.87z + 1046.0 = 0. \quad (39)$$

则其法向量为 $(1, -0.6111, 42.87)$. 设水平等高线在 xOy 平面内的投影的斜率为 k , 因插值平面的水平等高线与插值平面的法向量正交, 有

$$(1, k, 0) \cdot (1, -0.6111, 42.87) = 0. \quad (40)$$

解得斜率 $k \approx 1.636$. 之后在东南部分区域采用水平且在水平面投影斜率为 k 的测线.

5.4.2 分块法模型求解

利用之前所述动态规划方法 (优先保证条带覆盖整个待测海域), 迭代过程见附录C 3.4.3, 求得西北部分区域与东南部分区域的测线位置, 结果详见附录B. 分块法得到的最优测线布局见图8, 图中黄色代表西北部分区域的测线, 蓝色代表东南部分区域的测线.

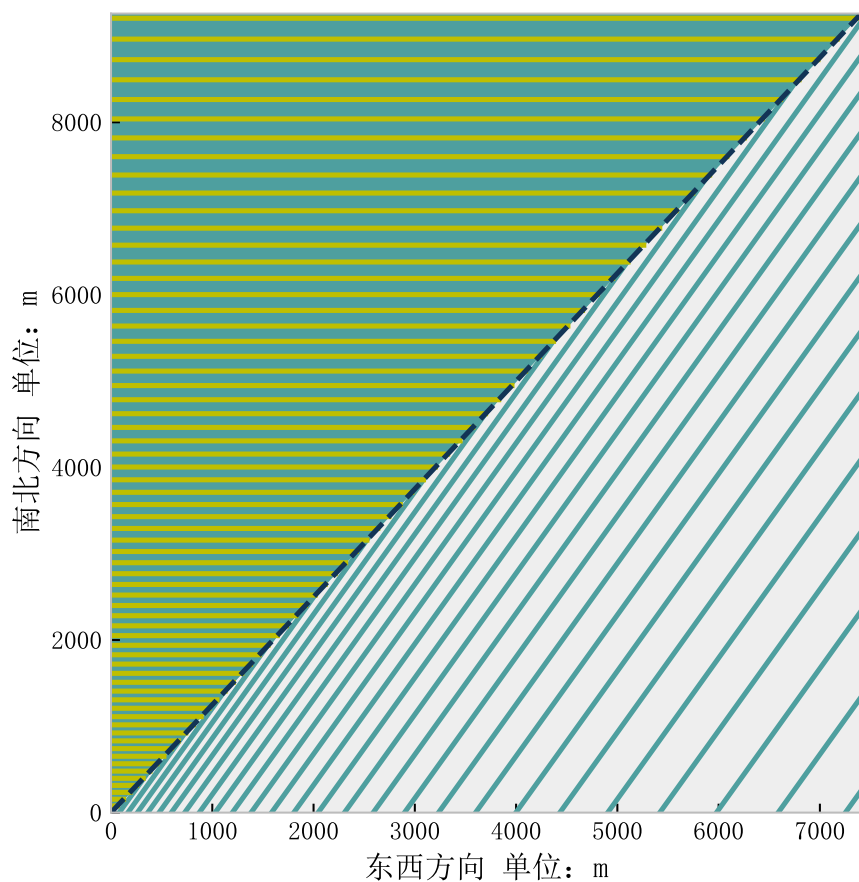


图 8 分块法得到的最优测线布局

在图8中, 由初等几何知识, 利用 MATLAB 计算得两族测线总长度为 $3.588 \times 10^5 \text{ m}$, 重叠率超出 20% 的测线总长最短为 31 288.4 m. 根据动态规划的约束条件, 此时漏测海区占总待测海域面积的百分比为 0%.

5.4.3 曲线微元法模型建立

先对等深线进行采样. 依据题给数据及海底地形图, 本文采用等高采样和等距采样的方法得到一系列等深线作为可选值. 需确保这些等深线的组合对应的测线尽可能覆盖整个海域, 并且重叠率不能过高, 所以可选的等深线要有一定的规模. 所选取的等深线如图9所示.

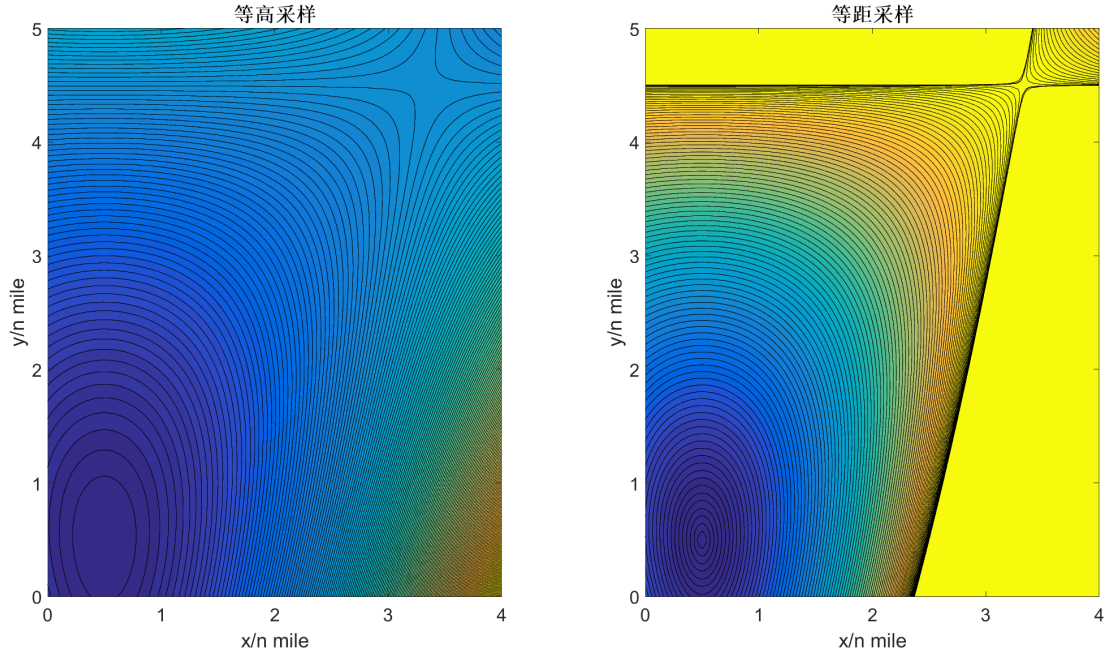


图 9 等高采样与等距采样示意图

以此方式采样, 共获得 324 条等深线. 以 $m_{i''}$ 表示第 i'' 条等深线, 以 $f_{i''}$ 表示第 i'' 条等深线是否被选中, $\mathcal{L}(m_{i''})$ 表示第 i'' 条等深线的长度, M 代表被选中的等深线的集合. 则对应的测线的总长度 L (目标函数) 为

$$F = \sum_{i''=1}^{324} f_{i''} \cdot \mathcal{L}(m_{i''}). \quad (41)$$

需要最小化测线的总长度, 即优化求解目标函数的最小值. 目标为

$$\min F. \quad (42)$$

根据题给附件可得离散分布的海底深度数据, 使用线性插值法可得该空间范围内其他点海底深度的估计值 $\hat{D}(x, y)$. 该海底平面是弯曲的, 由图10, 只需得到某点左右边界位置即可得到某点波束的覆盖范围. 由于本解法中的曲线实际上是用微元法记录的点集, 通过编程实践发现, 使用计算机对每个等深线曲线的点集中的每个点进行模拟求解覆盖边界需要大量算力. 另外, 海平面的整体斜率并不大. 所以这里进行近似处理, 将海底化曲为直, 以利用对称性求解.

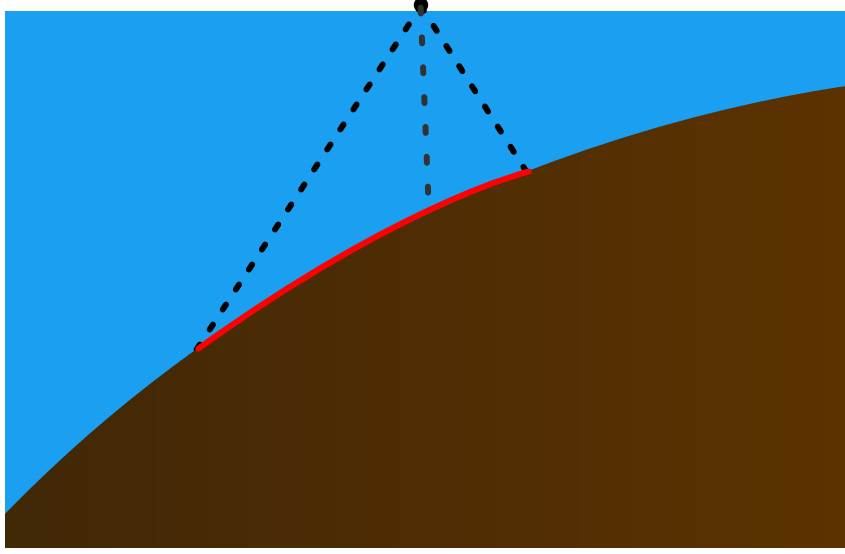


图 10 某点波束覆盖范围示意图

5.4.4 曲线微元法模型求解

对海平面近似处理后, 设测量点 (x, y) 可测量的最大半径为 r , 如图11, 有如下几何关系:

$$r = \hat{D}(x, y) \tan 60^\circ = \sqrt{3} \hat{D}(x, y). \quad (43)$$

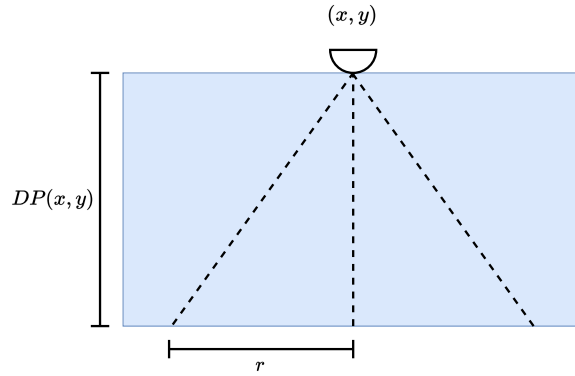


图 11 化曲为直近似截面示意图

为了使用数值方法近似求解测线对应的海域面积的覆盖情况和重叠率, 需要对二维空间进行离散化处理. 沿东西方向选取 400 小段, 在南北方向选取 500 小段, 即以 0.01 n mile 为边长的正方形为一个平面微元, 用于数值积分求解面积.

定义一个 400×500 的二维数组 \mathcal{A} , $\mathcal{A}_{i''', j'''}$ 表示第 i''' 行、第 j''' 列的平面微元被测线

覆盖的次数, 整体海域的覆盖率为

$$C = \frac{\sum_{i'''} \sum_{j'''} \chi_{\{(i''', j''') | A_{i''', j'''} \geq 1\}}(i''', j''')}{400 \times 500} \times 100\%. \quad (44)$$

漏测海区占总海域面积的百分比为

$$C' = (1 - C) \times 100\%. \quad (45)$$

整体的重叠率为

$$\eta = \frac{\sum_{i'''} \sum_{j'''} \chi_{\{(i''', j''') | A_{i''', j'''} \geq 2\}}(i''', j''')}{400 \times 500} \times 100\%. \quad (46)$$

即可得目标问题 (覆盖率 C 和重叠率 η 根据题意选取适当即可) ——

$$\min F, \quad (47)$$

$$\text{s.t.} \begin{cases} C(\mathbf{f}) \geq 95\%, \\ \eta(\mathbf{f}) \leq 20\%. \end{cases} \quad (48)$$

鲸鱼优化算法 (Whale Optimization Algorithm, WOA) 可以应用于解决二进制 (0-1) 规划问题.

以下是将鲸鱼优化算法应用于二进制规划问题的基本步骤:

1. 初始化种群. 随机生成一组初始解作为种群, 每个解都是一个包含 0 和 1 的二进制串, 代表问题的一个可能解.
2. 计算适应度. 根据二进制串的具体含义和问题的目标函数, 计算每个解的适应度.
3. 确定最优解. 找到当前种群中适应度最高的解, 即问题的当前最优解.
4. 模拟鲸鱼运动. 奔向目标根据当前最优解的方向和速度, 更新每个解的位置, 使它们向最优解靠近.
5. 随机探索. 一些解以随机的方式进行探索, 模拟了鲸鱼在搜索食物时的不确定性.
6. 更新位置和速度. 根据奔向目标和随机探索的结果, 更新每个解的位置.
7. 更新最优解. 计算新位置的适应度并更新最优解.
8. 终止条件. 根据一定的停止条件 (如达到最大迭代次数或满足特定的收敛条件), 决定是否结束算法.
9. 输出结果. 算法结束后, 得到的最优解即为二进制规划问题的解.

利用鲸鱼优化算法的求解结果为, 测线总长度最少为 432 765.43 m, 覆盖率为 96.78%, 漏测海区占总待测海域面积的百分比 3.22%, 面积重复率为 17.54%. 求得测线布局见图12.

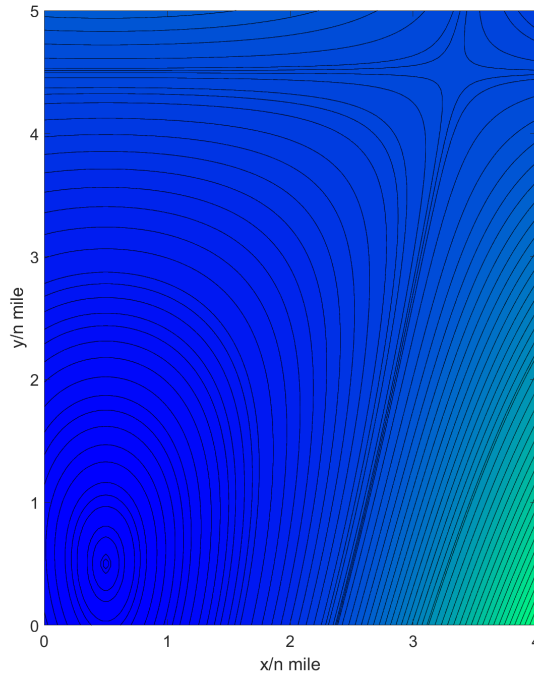


图 12 鲸鱼优化算法求得的测线布局

由于智能优化算法需要对参数进行多次调优, 这个模型结果还可以进一步优化.

六、结果分析

6.1 应用分析

在多波束测深布线前, 要提前了解待测海域海底地形, 最好提前通过单波束测深等技术粗略得知海底地貌, 之后因地制宜布设测线. 为使得测线总长度尽可能短, 节省成本, 地形简单时利用分块法与动态规划, 地形复杂时利用曲线微元法. 布设测线时应注意重叠率满足 10% ~ 20% 的约束条件.

6.2 误差分析

在问题 4 的分块法中, 以 0.04 n mile 为采样步长, 抽样计算每一半矩形海域内实际海底深度与拟合平面处相应深度的平均绝对误差

$$\text{MAE} = \frac{\sum_{P \in \text{某一半矩形海平面}} |D_P - \hat{D}_P|}{\sum_{P \in \text{某一半矩形海平面}} 1}, \quad (49)$$

其中 P 为某一半矩形海平面内的采样点. 为消除量纲影响, 定义变异系数

$$CV = \frac{MAE}{\overline{D_P}}, \quad (50)$$

其中

$$\overline{D_P} = \frac{\sum_{P \in \text{某一半矩形海平面}} D_P}{\sum_{P \in \text{某一半矩形海平面}} 1}. \quad (51)$$

利用 MATLAB 求得西北部分 MAE 为 24.01%, 东南部分 MAE 为 33.09%, 这说明用平面近似曲面有一定的合理性.

6.3 灵敏度分析

在问题 3 中, 假设 α 的值有小范围误差, 利用 MATLAB 研究 α (步长为 0.01°) 对测线数量的影响, 绘制图13.

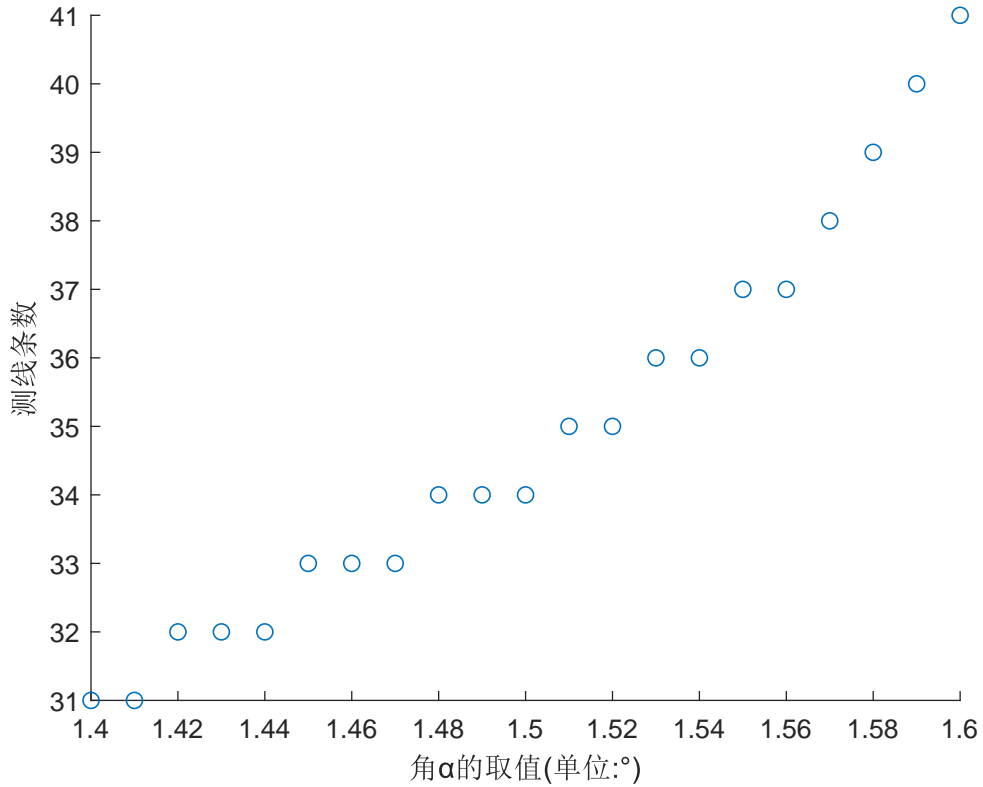


图 13 测线数量随坡度的变化情况

从图13可看出, 当 $1.48^\circ \leq \alpha \leq 1.50^\circ$ 时, 最少需要的测线数量仍是 34 条.

七、模型评价

7.1 优点

- 模型稳健性强, 对小的参数变化保持足够的稳定性, 对输入数据的噪声和异常值的敏感程度较低;
- 符号“因地制宜”原则, 能推广到更广泛的情况, 不仅局限于某一特定海底地形;
- 模型求解速度较快, 程序的时间和空间复杂度可控;
- 模型可扩展性强, 便于增加新变量、改进方程形式等;
- 模型应用场景广泛, 实用性强, 有利于降低测深成本.

7.2 缺点

- 模型仅给出点估计, 未根据参数可能的测量误差给出条带覆盖宽度、重叠率、测线位置的置信区间;
- 未考虑声波在水中受到的反射、遮挡等干扰;
- 未考虑测量船的艏摇、横滚、俯仰、升沉、横荡、纵荡对定位和测深的影响;
- 曲线形状的测线增加了实际航行的难度, 对动态定位系统要求较高.

八、模型改进与推广

8.1 模型改进方向

- 根据参数可能的测量误差给出条带覆盖宽度、重叠率、测线位置的置信区间;
- 开发地形自适应测线方法, 根据探测情况动态修改测线;
- 对于测量船的艏摇、横滚、俯仰、升沉、横荡、纵荡, 研究定位和测深的修正模型;
- 在高关注区域设计交叉测线, 提高重点地区数据冗余度;
- 充分利用卫星图像、航空探测等辅助数据, 校正和补充测线布局;
- 下一步可构建可视化的三维测线布局规划系统, 进行直观的模拟预演;
- 针对不同类型海底地形, 建立典型测线布局模版, 形成案例库;
- 建立测线规划方案评估体系, 进行设计结果评定.

8.2 模型推广

多波束测深系统主要用于水下地形测绘, 其测线布局模型主要应用在与水环境相关的场景中, 但多波束测深系统的一些核心技术原理和方法, 可以推广应用到其他领域中:

三维激光扫描技术 激光雷达扫描原理与多波束声纳工作原理相似, 都可以构建目标物体的三维模型. 激光扫描系统的测线布局也可以参考多波束系统的经验.

探地雷达技术 探地雷达可以探测地下目标, 其数据获取与处理流程类似多波束声纳. 布置不同方位的探测线, 可以更全面地探测地下空间.

航空摄影测量 航拍获得视角不同的图像, 通过像素匹配构建三维模型, 也可参考多波束系统的测线设计经验. 无人机进行低空航拍获取地面图像, 可以设计飞行航线布局, 优化拍摄视角, 获得地面目标的多角度信息. 这类似于多波束测深的测线设计思想.

医学影像 在 CT、MRI 等医学扫描中, 不同角度的扫描可以获得更丰富的体内信息. 可以借鉴多波束系统的测线布局策略.

机器人感知 机器人通过多源传感器获取环境信息, 其测线布局也可参考声纳系统, 提高覆盖效果.

无人驾驶汽车 感知系统获取道路及周边信息, 测线布局可以优化覆盖面和避障能力.

参考文献

- [1] 成芳, 胡迺成. 多波束测量测线布设优化方法研究[J/OL]. 海洋技术学报, 2016, 35 (2): 87-91. <https://d.wanfangdata.com.cn/periodical/ChlQZXJpb2RpY2FsQ0hJTmV3UzIwMjMwODMxEG1oeWpzMjAxNjAyMDE2GghlbnE1ZGFyZW==>. DOI: 10.3969/j.issn.1003-2029.2016.02.016.
- [2] 中国地质调查局广州海洋地质调查局. 海洋多波束水深测量规程: DZ/T 0292-2016 [Z]. 2016: 1—26.
- [3] 中华人民共和国交通运输部. 多波束测深系统测量技术要求[Z]. 2010: 5.
- [4] 胡运权, 郭耀煌. 运筹学教程[M]. 第 5 版. 北京: 清华大学出版社, 2018: 185—193.
- [5] 杨海东. 多波束测深仪姿态对测深影响分析及误差校正技术[J]. 应用声学, 2022: 3.
- [6] 夏伟, 刘雁春, 边刚, 等. 基于海底地貌表示法确定主测深线间隔和测图比例尺[J/OL]. 测绘通报, 2004(3): 24-27. <https://d.wanfangdata.com.cn/periodical/ChlQZXJpb2RpY2FsQ0hJTmV3UzIwMjMwODMxEG1jaHRiMjAwNDZMDA5GggzY25zejh1aQ==>. DOI: 10.3969/j.issn.0494-0911.2004.03.009.

附录 A 支撑材料的文件列表

- 问题 1
 - 问题 1 Python 代码.py
 - result1.xlsx
- 问题 2
 - 求问题 2 符号解 MATLAB 代码.m
 - 求问题 2 数值解 MATLAB 代码.m
 - result2.xlsx
- 问题 3
 - 验证以北偏东 15 度作为船的轨迹会导致条带宽度变化过大 MATLAB 代码.m
 - 问题 3 跳跃式迭代 MATLAB 代码.m
 - 测线布局图绘制 Python 代码.py
- 问题 4
 - 海底地形图绘制 MATLAB 代码.m
 - 海底地形等深线图绘制 MATLAB 代码.m
 - 问题 4 分块法 MATLAB 代码.m
 - 问题 4 分块法测线布局图绘制 Python 代码.py
 - 等高采样与等距采样示意图绘制 MATLAB 代码.m
 - 拟合 MATLAB 代码.m
 - 覆盖率与重叠率计算 MATLAB 代码.m
 - 鲸鱼优化算法 MATLAB 代码.m
 - 线性插值的数据.mat
 - 可选等深线的元胞数组.mat
- 误差分析
 - 误差分析 MATLAB 代码.m
- 灵敏度分析
 - 问题 3 灵敏度分析 MATLAB 代码.m

附录 B 问题 4 分块法求得的测线位置

西北部分区域沿东西方向的测线纵坐标（单位：m）分别为：1.0, 77.37, 155.2, 234.4, 315.1, 397.2, 480.9, 566.2, 653.0, 741.4, 831.4, 923.2, 1017.0, 1112.0, 1209.0, 1307.0, 1408.0, 1510.0, 1614.0, 1721.0, 1829.0, 1939.0, 2051.0, 2165.0, 2282.0, 2400.0, 2521.0, 2644.0, 2769.0,

2896.0, 3026.0, 3158.0, 3293.0, 3430.0, 3570.0, 3712.0, 3857.0, 4005.0, 4155.0, 4308.0, 4464.0, 4623.0, 4785.0, 4949.0, 5117.0, 5288.0, 5462.0, 5639.0, 5820.0, 6003.0, 6191.0, 6381.0, 6576.0, 6773.0, 6975.0, 7180.0, 7389.0, 7602.0, 7819.0, 8039.0, 8264.0, 8493.0, 8726.0, 8964.0, 9206.0.

东南部分区域的测线横截距 (单位: m) 分别为: 1.0, 80.27, 166.6, 260.5, 362.8, 474.2, 595.4, 727.5, 871.2, 1028.0, 1198.0, 1383.0, 1585.0, 1805.0, 2045.0, 2305.0, 2589.0, 2898.0, 3234.0, 3600.0, 3998.0, 4432.0, 4905.0, 5419.0, 5979.0, 6588.0, 7252.0 .

附录 C 源程序代码

3.1 问题 1

3.1.1 问题 1 Python 代码

```
1  # 问题1
2  from math import *
3  # 海底的坡度(弧度制)
4  a = (pi/180)*1.5
5  # 多波束换能器的开角
6  theta = (pi/180)*120
7  # 海域中心的深度
8  middle_h = 70
9  # 两艘船的间距
10 s = 200
11 # 计算w1,w2
12
13
14 def W1(h, a, theta):
15     return h*sin(theta/2)*cos(a)/sin(pi/2-theta/2-a)
16
17
18 def W2(h, a, theta):
19     return h*sin(theta/2)*cos(a)/sin(pi/2+a-theta/2)
20
21
22 # 测线距中心点处的距离
23 d_list = [s*i for i in range(-4, 5)]
24 # 到海底的深度
25 h_list = []
26 # 多波束测深的覆盖宽度
27 w_list = []
28 # 多波束测深的左侧宽度
29 w1_list = []
30 # 多波束测深的右侧宽度
```

```

31 w2_list = []
32
33 for d in d_list:
34     # 此处到海底的深度
35     h = middle_h-d*tan(a)
36     # 此处测线的左宽度
37     w1 = W1(h=h, a=a, theta=theta)
38     # 此处测线的右宽度
39     w2 = W2(h=h, a=a, theta=theta)
40     # 此处测线的宽度
41     w = w1+w2
42     h_list.append(h)
43     w1_list.append(w1)
44     w2_list.append(w2)
45     w_list.append(w)
46
47 # 相邻条带之间重叠率
48 overlapRate_list = []
49
50 for i in range(1, len(w_list)):
51     average_W = (w_list[i-1]+w_list[i])/2
52     # 重叠率
53     overlapRate = 1-s/average_W
54     overlapRate_list.append(overlapRate)
55
56 print("到海底距离分别为", h_list)
57 print("覆盖宽度分别为", w_list)
58 print("重叠率分别为", overlapRate_list)

```

3.2 问题 2

3.2.1 求问题 2 符号解 MATLAB 代码.m

```

1 %求在半径为p海里的圆上的条带宽度的公式---
2 syms x y z p fai beta h0 a theta
3 cond1 = (cot(theta/2)+sin(beta)*tan(a))*x == p*cos(fai)*cot(theta/2)-h0*sin(beta);
4 cond2 = cos(beta)*(x-p*cos(fai)) + sin(beta) * (y-p*sin(fai)) == 0;
5 cond3 = tand(a)*x + z == 0;
6 conds = [cond1 cond2 cond3];
7 sol = solve(conds,x,y,z);
8 %点A的坐标
9 x_1 = sol.x;
10 X_1 = latex(x_1);
11 y_1 = sol.y;
12 Y_1 = latex(y_1);

```

```

13 z_1 = sol.z;
14 Z_1 = latex(z_1);
15
16 syms x y z p fai beta h0 a theta
17 cond1 = (cot(theta/2)-sin(beta)*tan(a))*x == p*cos(fai)*cot(theta/2)+h0*sin(beta);
18 cond2 = cos(beta)*(x-p*cos(fai)) + sin(beta) * (y-p*sin(fai)) == 0;
19 cond3 = tand(a)*x + z == 0;
20 conds = [cond1 cond2 cond3];
21 sol = solve(conds,x,y,z);
22 %点B的坐标
23 x_2 = sol.x;
24 X_2 = latex(x_2);
25 y_2 = sol.y;
26 Y_2 = latex(y_2);
27 z_2 = sol.z;
28 Z_2 = latex(z_2);
29
30 %计算AB两点的投影距离
31 w = sqrt((x_1-x_2)^2+(y_1-y_2)^2);
32 latex(w)
33 %---

```

3.2.2 求问题2数值解 MATLAB 代码

```

1 %第二问
2 %求表格中的条带宽度---
3 %alpha角
4 a = 1.5;
5 %theta角
6 theta = 120;
7 %海域中心的深度
8 middle_h = 120;
9
10 %不同情况下的条带宽度列表
11 Wlist = zeros(8,8);
12
13 for i=0:7
14     for j=0:7
15         %船到海中心的距离
16         d = -0.3*i*1852;
17         %beta角的度数
18         beta = 45*j;
19
20         syms x y z
21         cond1 = tand(a)*x + z == 0;

```

```

22 %若β角为90或270，度tanβ不存，在此时修改方程
23 if(beta == 90 || beta == 270)
24     cond2 = y == 0;
25     cond3 = (x+d)/tand(theta/2) - (z-120) == 0;
26 else
27     cond2 = (x+d) + tand(beta) * y == 0;
28     cond3 = y - (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;
29 end
30 conds = [cond1 cond2 cond3];
31 sol = solve(conds,x,y,z);
32 %点A的坐标
33 x_1 = sol.x;
34 y_1 = sol.y;
35 z_1 = sol.z;
36
37 syms x y z
38 cond1 = tand(a)*x + z == 0;
39 %若β角为90或270，度tanβ不存，在此时修改方程
40 if(beta == 90 || beta == 270)
41     cond2 = y == 0;
42     cond3 = (x+d)/tand(theta/2) + (z-120) == 0;
43 else
44     cond2 = (x+d) + tand(beta) * y == 0;
45     cond3 = y + (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;
46 end
47 conds = [cond1 cond2 cond3];
48 sol = solve(conds,x,y,z);
49 %点B的坐标
50 x_2 = sol.x;
51 y_2 = sol.y;
52 z_2 = sol.z;
53
54 %计算AB两点的投影距离
55 w = sqrt((x_1-x_2)^2+(y_1-y_2)^2);
56 Wlist(j+1,i+1) = w;
57 end
58 end
59 Wlist
60 %---

```

3.3 问题3

3.3.1 验证以北偏东 15° 作为船的轨迹会导致条带宽度变化过大 MATLAB 代码

```

1 %第三问

```

```

2      %如果以北偏东15度作为船的轨迹会导致条带宽度变化过大
3
4      %海底的坡度
5      a = 1.5;
6      %多波束换能器的开角
7      theta = 120;
8      %海域中心的深度
9      middle_h = 110;
10     %矩形海域的长宽
11     p = 4*1852;
12     q = 2*1852;
13     %β角的度数
14     beta = 75;
15
16     %船到海中心的距离(最深)
17     d1 = -1852*cotd(beta);
18     syms x y z
19     cond1 = tand(a)*x + z == 0;
20     cond2 = (x+d1) - tand(beta)*y == 0;
21     cond3 = y - (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;
22     conds = [cond1 cond2 cond3];
23     sol = solve(conds,x,y,z);
24     %点A的坐标
25     x_1 = sol.x;
26     y_1 = sol.y;
27     z_1 = sol.z;
28
29     syms x y z
30     cond1 = sind(a)*x + z == 0;
31     cond2 = (x+d1) - tand(beta)*y == 0;
32     cond3 = y + (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;
33     conds = [cond1 cond2 cond3];
34     sol = solve(conds,x,y,z);
35     %点B的坐标
36     x_2 = sol.x;
37     y_2 = sol.y;
38     z_2 = sol.z;
39
40     %计算AB两点的投影距离
41     w1 = vpa(sqrt((x_1-x_2)^2+(y_1-y_2)^2),3)
42
43     %船到海中心的距离(最浅)
44     d2 = 1852*cotd(beta);
45     syms x y z
46     cond1 = tand(a)*x + z == 0;
47     cond2 = (x+d2) - tand(beta)*y == 0;
48     cond3 = y - (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;

```



```

49     conds = [cond1 cond2 cond3];
50     sol = solve(conds,x,y,z);
51     %点A的坐标
52     x_1 = sol.x;
53     y_1 = sol.y;
54
55     syms x y z
56     cond1 = sind(a)*x + z == 0;
57     cond2 = (x+d2) - tand(beta)*y == 0;
58     cond3 = y + (z-middle_h)/(sqrt(1+tand(beta)^2)*cotd(theta/2)) == 0;
59     conds = [cond1 cond2 cond3];
60     sol = solve(conds,x,y,z);
61     %点B的坐标
62     x_2 = sol.x;
63     y_2 = sol.y;
64
65     %计算AB两点的投影距离
66     w2 = vpa(sqrt((x_1-x_2)^2+(y_1-y_2)^2),3)

```

3.3.2 问题3 跳跃式迭代 MATLAB 代码

```

1     %第三问
2     %求最短路线
3
4     %海底的坡度
5     a = 1.5;
6     %多波束换能器的开角
7     theta = 120;
8     %海域中心的深度
9     middle_D = 110;
10    %矩形海域的长宽
11    p = 4*1852;
12    q = 2*1852;
13    %要求的重叠率
14    overlap_rate = 0.105;
15
16    %求最左侧测线x坐标的最大可能值
17    syms t
18    D = middle_D-t*tand(a);
19    w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
20    cond = p/2+t == w1;
21    max_T = vpa(solve(cond,t),3)
22
23    %不同起始位置下所有测线的位置以及测线的总数量
24    for i = 0:34

```

```

25 %起始位置为-p/2附近
26 T = -p/2+10*i;
27 %上一轮的x坐标
28 last_T = T;
29 %上一轮的水深
30 last_D = middle_D-last_T*tand(a);
31 %上一轮的条带宽度
32 last_w =
    last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2)+1/sind(90+a-theta/2));
33 %上一轮的条带右宽度
34 last_w2 = 0;
35 T_list = [T];
36 while last_T+last_w2 < p/2
37     syms t
38     D = middle_D-t*tand(a);
39     w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
40     w2 = D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
41     w = w1+w2;
42     cond = ((w+last_w)/2)-(t-last_T)==overlap_rate*((w+last_w)/2);
43     T = solve(cond,t);
44     %记录本轮数，据将其作为新一轮的last_数据
45     T_list = [T_list,vpa(T,3)];
46     last_T = T;
47     last_D = middle_D-T*tand(a);
48     last_w1 = last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
49     last_w2 = last_D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
50     last_w = last_w1 + last_w2;
51 end
52 T_list
53 size(T_list)
54 end

```

3.3.3 问题3 最优测线布局绘制 Python 代码

```

1 import matplotlib.pyplot as plt
2 plt.rc('font', family='SimSun')
3 plt.rc('axes', unicode_minus=False)
4 plt.style.use('bmh')
5 t = [-3364.00,
6     -2772.64,
7     -2227.50,
8     -1724.99,
9     -1261.75,
10    -834.73,
11    -441.09,

```

```

12     -78.22,
13     256.28,
14     564.63,
15     848.88,
16     1110.90,
17     1352.45,
18     1575.11,
19     1780.36,
20     1969.57,
21     2143.99,
22     2304.77,
23     2452.99,
24     2589.61,
25     2715.56,
26     2831.66,
27     2938.69,
28     3037.34,
29     3128.29,
30     3212.13,
31     3289.41,
32     3360.65,
33     3426.32,
34     3486.86,
35     3542.67,
36     3594.11,
37     3641.53,
38     3685.25
39 ]
40 n_mile = 1852
41 p=4 * n_mile
42 q=2 * n_mile
43 ax = plt.axes()
44 ax.set_xlabel('东西方向($t$轴) 单位:m')
45 ax.set_ylabel('南北方向 单位:m')
46 ax.set_xlim(-p/2, p/2)
47 ax.set_ylim(-q/2, q/2)
48 ax.vlines(t, -q/2, q/2, colors=None, linestyle='solid', label='') # 画测线
49 ax.grid(False)
50 plt.show()

```

3.4 问题 4

3.4.1 海底地形图绘制 MATLAB 代码

```

1 % 海底地形图绘制代码

```

```

2  m = xlsread('附件.xlsx');
3  m = -m;
4  figure
5  set(gcf, 'Position', [10, 10, 2000, 700])
6  subplot(1,2,1)
7  sf = surf(m);
8  sf(2).Fill = 'on';
9  xlim([2, 202]);
10 ylim([2, 252]);
11 xlabel('x/n mile')
12 ylabel('y/n mile')
13 zlabel('z/m')
14 xticks(linspace(2,202,5));
15 xx = linspace(0,4,5);
16 xticklabels(xx)
17 yticks(linspace(2,252,6));
18 yy = linspace(0,5,6);
19 yticklabels(yy)
20 set(gca, 'FontSize', 18)
21 shading flat
22 colormap(slanCM('ice'))
23 view(135,45)
24 subplot(1,2,2)
25 surf(m)
26 xlim([2, 202]);
27 ylim([2, 252]);
28 xlabel('x/n mile')
29 ylabel('y/n mile')
30 xticks(linspace(2,202,5));
31 xx = linspace(0,4,5);
32 xticklabels(xx)
33 yticks(linspace(2,252,6));
34 yy = linspace(0,5,6);
35 yticklabels(yy)
36 set(gca, 'FontSize', 18)
37 shading flat
38 colormap(slanCM('ice'))
39 view(90,90)
40
41 filename = 'output.png';
42 exportgraphics(gcf, filename, 'BackgroundColor', 'none', 'Resolution', 300);

```

3.4.2 海底地形等深线图绘制 MATLAB 代码

```

1  %海域内水深的等高线图

```

```

2 Z = xlsread('附件.xlsx') ;
3 z = size(Z);
4 X = linspace(0,4,201);
5 Y = linspace(0,5,251);
6 [X,Y] = meshgrid(X,Y);
7 colormap(cool);
8 [C,h] = contourf(X,Y,Z,20);
9 clabel(C,h)
10 xlabel("单位:n mile")
11 ylabel("单位:n mile")

```

3.4.3 问题4 分块法 MATLAB 代码

```

1 %分块法求测线分布
2 %水域横向长度
3 p = 4*1852;
4 %水域纵向长度
5 q = 5*1852;
6 %多波束换能器的开角
7 theta = 120;
8 %要求的重叠率
9 overlap_rate = 0.105;
10
11
12 %所有测线的总长度
13 total_length = 0;
14
15
16 %求区域1的测线---
17 %区域1的近似平面
18 syms x y z
19 plane = ((84.40-24.40)/q)*y+z==-24.40;
20 %近似平面的坡度(角度制)
21 a = atan(54.65/q)*180/pi;
22 %边界区域(x=0,y=0处)的水深
23 boarder_D = 24.40;
24
25 %起始位置(y坐标)
26 T = 1;
27 %上一轮船的坐标
28 last_T = T;
29 %上一轮的水深
30 last_D = boarder_D+last_T*tand(a);
31 %上一轮的条带宽度
32 last_w =

```

```

        last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2)+1/sind(90+a-theta/2));
33 %上一轮的条带右宽度
34 last_w2 = 0;
35 T1_list = [T];
36 while last_T+last_w2 < q
37     syms t
38     %水浅往水深，走深度变大
39     D = boarder_D+t*tand(a);
40     w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
41     w2 = D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
42     w = w1+w2;
43     cond = ((w+last_w)/2)-(t-last_T)==overlap_rate*((w+last_w)/2);
44     T = solve(cond,t);
45     %记录本轮数，据将其作为新一轮的last_数据
46     T1_list = [T1_list,T];
47     last_T = T;
48     last_D = boarder_D+T*tand(a);
49     last_w1 = last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
50     last_w2 = last_D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
51     last_w = last_w1 + last_w2;
52     %此线段的长度加到total_length中
53     length = T;
54     total_length = total_length + length;
55 end
56 T1_list = vpa(T1_list,4)
57 size(T1_list)
58
59
60 %求区域2的测线---
61 %区域2的近似平面
62 syms k1 k2 k3
63 x=0;
64 y=0;
65 z=-24.4;
66 f1 = x+k1*y+k2*z == k3;
67 x=p;
68 y=q;
69 z=-65.20;
70 f2 = x+k1*y+k2*z == k3;
71 x=p;
72 y=0;
73 z=-197.20;
74 f3 = x+k1*y+k2*z == k3;
75
76 cond = [f1 f2 f3];
77 sol = solve(cond,k1,k2,k3);
78 k1 = sol.k1;

```

```

79 k2 = sol.k2;
80 k3 = sol.k3;
81
82 syms x y z
83 plane = x+k1*y+k2*z == k3;
84 %近似平面的坡度(角度制)
85 a = acot(abs(k2/sqrt(1^2+k1^2)))*180/pi;
86 %边界区域(x=0,y=0处)的水深
87 boarder_D = 24.40;
88
89 %起始位置(x坐标)
90 T = 1;
91 %上一轮船的坐标
92 last_T = T;
93 %上一轮的水深
94 last_D = boarder_D+last_T*tand(a);
95 %上一轮的条带宽度
96 last_w =
    last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2)+1/sind(90+a-theta/2));
97 %上一轮的条带右宽度
98 last_w2 = 0;
99 T2_list = [T];
100 while last_T+last_w2 < p
101     syms t
102     %水浅往水深, 走深度变大
103     D = boarder_D+t*tand(a);
104     w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
105     w2 = D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
106     w = w1+w2;
107     cond = ((w+last_w)/2)-(t-last_T)==overlap_rate*((w+last_w)/2);
108     T = solve(cond,t);
109     %记录本轮数, 据将其作为新一轮的last_数据
110     T2_list = [T2_list,T];
111     last_T = T;
112     last_D = boarder_D+T*tand(a);
113     last_w1 = last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
114     last_w2 = last_D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
115     last_w = last_w1 + last_w2;
116     %此线段的长度加到total_length中
117     if (p-T)*(-1/k1) > q
118         x = T/(1+k1);
119         y = x;
120     else
121         x = p;
122         y = (T-p)/k1;
123     end
124     length = sqrt((x-T)^2+y^2);

```

```

125     total_length = total_length + length;
126 end
127 T2_list = vpa(T2_list,4)
128 size(T2_list)
129
130 total_length = vpa(total_length,4)

```

3.4.4 问题4 分块法测线布局图绘制 Python 代码

```

1  import numpy as np
2  import matplotlib.pyplot as plt
3  plt.rc('font', family='SimSun')
4  plt.rc('axes', unicode_minus=False)
5  plt.style.use('bmh')
6  n_mile = 1852
7  p = 4 * n_mile
8  q = 5 * n_mile
9  ax = plt.axes()
10 ax.set_xlabel('东西方向 单位:m')
11 ax.set_ylabel('南北方向 单位:m')
12 ax.set_xlim(0, p)
13 ax.set_ylim(0, q)
14 ax.grid(False)
15 k=1.636
16 X_down=[ 1.0, 80.27, 166.6, 260.5, 362.8, 474.2, 595.4, 727.5, 871.2, 1028.0,
          1198.0, 1383.0, 1585.0, 1805.0, 2045.0, 2305.0, 2589.0, 2898.0, 3234.0, 3600.0,
          3998.0, 4432.0, 4905.0, 5419.0, 5979.0, 6588.0, 7252.0]
17 X_down=np.array(X_down)
18 X_up=X_down+q/k
19 for i in range(len(X_down)):
20 plt.plot([X_down[i],X_up[i]],[0,q],color='#4e9f9f')
21 ax.fill_between([0,p],[0,q],[q,q],color='#4e9f9f')
22
23 y=[1.0, 77.37, 155.2, 234.4, 315.1, 397.2, 480.9, 566.2, 653.0, 741.4, 831.4,
    923.2, 1017.0, 1112.0, 1209.0, 1307.0, 1408.0, 1510.0, 1614.0, 1721.0, 1829.0,
    1939.0, 2051.0, 2165.0, 2282.0, 2400.0, 2521.0, 2644.0, 2769.0, 2896.0, 3026.0,
    3158.0, 3293.0, 3430.0, 3570.0, 3712.0, 3857.0, 4005.0, 4155.0, 4308.0, 4464.0,
    4623.0, 4785.0, 4949.0, 5117.0, 5288.0, 5462.0, 5639.0, 5820.0, 6003.0, 6191.0,
    6381.0, 6576.0, 6773.0, 6975.0, 7180.0, 7389.0, 7602.0, 7819.0, 8039.0, 8264.0,
    8493.0, 8726.0, 8964.0, 9206.0]
24 y=np.array(y)
25 x=y/q*p
26 for i in range(len(y)):
27 plt.plot([0,x[i]],[y[i],y[i]],color='y')
28 ax.plot([0,p],[0,q],color='#123456',linestyle='--')

```


3.4.5 等高采样与等距采样示意图绘制 MATLAB 代码

```

1  % 等高线采样作为可选测线
2  clear
3  clc
4  z0 = xlsread('dt.xlsx');
5  z0 = z0(2:end,2:end);
6  [n,m] = size(z0);
7  set(gcf, 'position', [100 100 1600 1000])
8  nm2m = 1852;
9  subplot(1,2,1)
10 z_max = max(max(z0));
11 z_min = min(min(z0));
12 zd = 1;
13 % 等高线值的选取
14 % 高度等间隔选取
15 c_lv = z_min:zd:z_max;
16 [~, h] = contour(z0,c_lv);
17 % 从等高线对象中提取等高线点坐标
18 c_dt = h.ContourMatrix';
19 contourf(z0, c_lv)
20 xticks(linspace(1,m,5));
21 xx = linspace(0,4,5);
22 xticklabels(xx)
23 yticks(linspace(1,n,6));
24 yy = linspace(0,5,6);
25 yticklabels(yy)
26 xlabel('x/n mile')
27 ylabel('y/n mile')
28 title('等高采样')
29 set(gca, 'FontSize', 16)
30 axis equal
31 c_lv = [];
32 % 距离等间隔选取
33 subplot(1,2,2)
34 [p1x,p1y] = deal(26,26);
35 [p2x,p2y] = deal(170,225);
36 tm = 100;
37 [pdx,pdy] = deal((p2x-p1x)/tm,(p2y-p1y)/tm);
38 while p1x < p2x
39     c_lv = [c_lv,interp2(z0, p1x,p1y, 'linear')];
40     [p1x,p1y] = deal(p1x+pdx,p1y+pdy);
41 end

```

```

42     [~, h] = contour(z0,c_lv);
43     % 添加从等高线对象中提取等高线点坐标
44     c_dt = vertcat(c_dt,h.ContourMatrix');
45     contourf(z0, c_lv)
46     xticks(linspace(1,m,5));
47     xx = linspace(0,4,5);
48     xticklabels(xx)
49     yticks(linspace(1,n,6));
50     yy = linspace(0,5,6);
51     yticklabels(yy)
52     xlabel('x/n mile')
53     ylabel('y/n mile')
54     title('等距采样')
55     set(gca, 'FontSize', 16)
56     axis equal

```

3.4.6 拟合 MATLAB 代码

```

1     function res = fit(f)
2     global qs
3     res = 0;
4     for i = 1:324
5         for j = 2:length(qs{i})
6             if f(i) > 0.5
7                 res = res+sqrt((qs{i}(1,j)-qs{i}(1,j-1)).^2+(qs{i}(2,j)-qs{i}(2,j-1)).^2);
8             end
9         end
10    end
11    end

```

3.4.7 覆盖率与重叠率计算 MATLAB 代码

```

1     function [C,R] = CR(f,d)
2     global qs
3     q = 18.52;
4     G = zeros(400,500);
5     d = 100;
6     for i = 1:324
7         F = zeros(400,500);
8         for j = 1:length(qs{i})
9             p1=qs{i}(1,j);
10            p2=qs{i}(2,j);
11            [ti,tj] = deal(round(p1/q),round(p2/q));

```

```

12     ti = max(1,ti);
13     ti = min(400,ti);
14     tj = max(1,tj);
15     ti = min(500,tj);
16     r = round(d(ti,tj)*tan(pi/3)/q);
17     for i = 0:r
18         for j = 0:r
19             if (ti-i)>=0 && (ti-i) <= 400 && (tj-j) >= 0 && (tj-j)<= 500
20                 F((ti-i),(tj-j)) = 1;
21             end
22         end
23     end
24 end
25 G = G+F;
26 end
27 C = sum(G>0)/200000;
28 R = sum(G>1)/200000;
29 end

```

3.4.8 鲸鱼优化算法 MATLAB 代码

```

1  clear
2  clc
3  load mt % mt???????????
4  global qs
5  load qs % qs???????
6  f = @(x,y) interp2(x0, y0, z0, x, y, 'linear');
7  k = 1852;
8  q = 18.52;
9  a = 400;
10 b = 500;
11 G = zeros(a,b);
12 d = meshgrid(1:a,1:b);
13 for i = 1:a
14     for j = 1:b
15         d(i,j) = f(i*q,j*q);
16     end
17 end
18 num_variables = 324; % ???
19 max_iterations = 100; % ?????
20 num_whales = 30; % ???
21 whales = randi([0, 1], num_whales, num_variables);
22 for iteration = 1:max_iterations
23     fitness = fit(whales);
24     [best_fitness, best_index] = min(fitness);

```

```

25     best_solution = whales(best_index, :);
26     for i = 1:num_whales
27         r1 = rand();
28         r2 = rand();
29         whales(i, :) = whales(i, :) - a * (r1 - 0.5) * (best_solution - whales(i, :))
30         ...
31         + r2 - 0.5;
32
33     whales(i, :) = max(min(whales(i, :), 1), 0);
34
35     flip_mask = rand(size(whales(i, :))) < 0.1;
36     whales(i, flip_mask) = 1 - whales(i, flip_mask);
37 end
38 [C,R] = CR(best_solution,d);
39 while C < 0.95 | R > 0.2
40     for i = 1:num_whales
41         r1 = rand();
42         r2 = rand();
43         whales(i, :) = whales(i, :) - a * (r1 - 0.5) * (best_solution - whales(i,
44             :)) ...
45         + r2 - 0.5;
46
47     whales(i, :) = max(min(whales(i, :), 1), 0);
48
49     flip_mask = rand(size(whales(i, :))) < 0.1;
50     whales(i, flip_mask) = 1 - whales(i, flip_mask);
51 end
52 end
53 a = 2 - iteration * (2 / max_iterations);
54 fprintf('????? = %f\n', best_fitness);

```

3.5 误差分析

3.5.1 误差分析 MATLAB 代码

```

1     %求拟合平面和实际海底面深度的变异系数
2     %实际海域深度数据
3     D_list = xlsread('附件.xlsx');
4     %分块法求测线分布
5     %水域横向长度
6     p = 4*1852;
7     %水域纵向长度
8     q = 5*1852;
9     %多波束换能器的开角

```

```

10  theta = 120;
11
12  %左上区域的所有点深度和
13  total_D1 = 0;
14  %左上区域采样点总数
15  dots_in_area1 = 0;
16  %左上区域各点深度绝对值差之和
17  diff1 = 0;
18
19  %右下区域的所有点深度和
20  total_D2 = 0;
21  %右下区域采样点总数
22  dots_in_area2 = 0;
23  %右下区域各点深度绝对值差之和
24  diff2 = 0;
25
26  %两个区域拟合平面的方程
27  plane1 = ((84.40-24.40)/q)*y+z == -24.40;
28  plane2 = x+k1*y+k2*z == k3;
29
30  for i=1:100
31  for j=1:125
32  x = 0.04*1852*i;
33  y = 0.04*1852*j;
34
35  D = D_list(2*j,2*i);
36  if 1.25*x<y
37  plane1_D = 24.40+((84.40-24.40)/q)*y;
38  delta1_D = abs(D-plane1_D);
39  diff1 = diff1+delta1_D;
40  total_D1 = total_D1+D;
41  dots_in_area1 = dots_in_area1+1;
42  else
43  plane2_D = -(k3-x-k1*y)/k2;
44  delta2_D = abs(D-plane2_D);
45  diff2 = diff2+delta2_D;
46  total_D2 = total_D2+D;
47  dots_in_area2 = dots_in_area2+1;
48  end
49  end
50  end
51
52  %mae
53  mae1 = vpa((diff1/dots_in_area1)/(total_D1/dots_in_area1),4)
54  mae2 = vpa((diff2/dots_in_area2)/(total_D2/dots_in_area2),4)

```

3.6 灵敏度分析

3.6.1 问题3 灵敏度分析 MATLAB 代码

```
1 %第三问
2 %灵敏度分析
3
4 %多波束换能器的开角
5 theta = 120;
6 %海域中心的深度
7 middle_D = 110;
8 %矩形海域的长宽
9 p = 4*1852;
10 q = 2*1852;
11 %要求的重叠率
12 overlap_rate = 0.105;
13
14 a_list = [];
15 num_list = [];
16
17 %海底的坡度小幅度变化
18 for i=-10:10
19     %海底的坡度
20     a = vpa(1.5+0.01*i,5);
21     a_list = [a_list,a]
22     %求最左侧测线x坐标的最大可能值
23     syms t
24     D = middle_D-t*tand(a);
25     w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
26     cond = p/2+t == w1;
27     max_T = vpa(solve(cond,t),3);
28     %起始位置为x坐标的最大可能值
29     T = max_T;
30     %上一轮船的x坐标
31     last_T = T;
32     %上一轮的水深
33     last_D = middle_D-last_T*tand(a);
34     %上一轮的条带宽度
35     last_w =
        last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2)+1/sind(90+a-theta/2));
36     %上一轮的条带右宽度
37     last_w2 = 0;
38     T_list = [T];
39     while last_T+last_w2 < p/2
40         syms t
41         D = middle_D-t*tand(a);
```

```

42     w1 = D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
43     w2 = D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
44     w = w1+w2;
45     cond = ((w+last_w)/2)-(t-last_T)==overlap_rate*((w+last_w)/2);
46     T = solve(cond,t);
47     %记录本轮数，据将其作为新一轮的last_数据
48     T_list = [T_list,vpa(T,3)];
49     last_T = T;
50     last_D = middle_D-T*tand(a);
51     last_w1 = last_D*sind(theta/2)*cosd(a)*(1/sind(90-a-theta/2));
52     last_w2 = last_D*sind(theta/2)*cosd(a)*(1/sind(90+a-theta/2));
53     last_w = last_w1 + last_w2;
54 end
55 num = size(T_list,2);
56 num_list = [num_list,num]
57 end
58 scatter(a_list,num_list)
59 xlabel("角α的取值(单位:°)")
60 ylabel("测线条数")

```