深海地形研究中的多波束测线技术

摘要

深海地形研究是海洋科学领域的重要研究方向之一。了解深海地形对于理解海洋地球系统、研究海洋生态系统以及开发海洋资源具有重要意义。然而，由于深海环境的复杂性和不可见性，传统的地形测量方法受到了很大的限制。为了克服这些限制，多波束测线技术应运而生。

在问题一中，我们可以利用三角函数和几何模型的方法，通过解决一系列方程问题和添加辅助线，建立多波束测深的数学模型，从而有效地解决海水深度、覆盖宽度和侧线重叠率等问题。

在问题二中，我们可以利用三角函数和几何关系的方法，建立多波束测深覆盖宽度的数学模型。该模型考虑侧线方向与海底坡面法向的夹角以及船距海域中心点的距离，通过统一计算不同海域中心点处的距离和不同测线方向夹角下的覆盖宽度，从而得到多波束测深的覆盖宽度。

问题三中，我们需要在满足约束条件的前提下，在矩形海域内找到一条合适的航行轨迹，该轨迹能够穿过整个海域，并且相邻条带之间的重叠率在 10%到 20%之间。为了实现这个目标，我们使用迭代的方法来寻找最佳航行轨迹，从而建立一个全覆盖待测海域的数学模型。最终通过附录 4 中的代码得出坡面法向的投影与航线的夹角 91°，重叠率为 13.54%，总侧线大小为 3679.10，平均宽度为 510.99，所需条带数为 7。

问题四中，首先我们用 matlab 分析附件中的数据，得出附件中的深度图，然后我们根据深度图的特点，建立了深度图的平面直角坐标系，并根据其四个较为特殊的点，进行构造三角形，将曲面分割为两个平面坡面，利用三角函数求出对应的角度与边，并根据角度与边求出两个变得坡度，并假设出多波束能器的开角，通过问题三的模型计算出对应的数据。根据附录 5 中的代码，我们可以求出当α=1.39°，测线总长度 28099.16 米，漏测的百分比为 13.04%，超过 20%部分的总长度为 2864.79 米。当α=4.42°，总长度为 119661.55 米，漏测百分比为 16.67%，超过 20%部分的总长度为 4513.49 米。

关键词：三角函数；几何关系数学模型；迭代优化；多目标规划；

一、　问题分析

1.1 问题一的分析

本题的目的是为了测量条带的宽度以及相邻两条测线覆盖率，并求解距离中心点不同距离的测线所测海域海水的深度，它的本质上是使用方程思想对几何问题的求解。又因为测线方向时固定的，因此我们可以建立测线及海底横截面海水深度模型，利用三角函数对距中心点不同距离的测线测得海水深度并求解。再根据所解出海水的深度建立测量条带覆盖宽度模型，运用三角函数正切关系解出深侧和较浅侧条带覆盖的宽度，两侧覆盖宽度相加就可以得到这个位置测量条带覆盖宽度。之后再根据相邻的两条测线之间的间距，使用题目所给重叠率计算公式再解出相对应的测线测量条带重叠率。并在已给分析出的条件下，建立测线和海底及海底横截面的深度的数学模型，最后求出海水深度、覆盖宽度以及侧线的重叠率。

1.2 问题二的分析

通过读题可以分析出来本题的要求是在测线方向可变及海底的地形为矩形待测海域坡面的前提下。在给出条件已知多波束换能器开角为 120°，坡度为 1.5°，海域内中心的海水深度为 120m 的前提下建立船在不同测线方向β下以及不同船据海域中心点的距离下建立多波束测深覆盖宽度的数学模型。

首先对测线方向夹角进行分类，设 0°为一类，90°、270°为一类，45°、135°、180°、225°、315°作为一类。由题可知在测线方向夹角为 0°，90°，270°时为一个特殊值，测线方向夹角为 0°船在行驶时，假设距离为 x，从而说明该航线与坡面法向的投影方向相同。测线方向夹角为 90°，270°的特殊值时，该值结果相等，因为 90°与 270°在同一个轴上。当测线方向为其他值没有特殊值时我们就要根据不同测线方向以及在不同的覆盖宽度下与海底坡面水平面上的夹角上并计算覆盖宽度。

1.3 问题三的分析

通过读题可知该题的目的是在满足题目约束条件矩形海域的前提下在矩形海域内寻找出一条合适的航行轨迹，该航行轨迹能穿过这片矩形海域且满足在相邻条带之间的重叠率满足 10%—20%，我们需要跟据第二问得出的结论对第三问进行求解。因为坡面法向β以及间距 d 并没有给出准确数值，对此我们需要建立相关约束并保证测线的长度最短。由于β角关于海域中心对称，因此需要求出 0-180°的最优解。得出η范围是否正确，如果正确我们将 d 保留，如果η范围不正确，我们将 d 舍去。之后使用迭代优化在该β角度已知的情况下持续进行迭代，并重复上述操作，设计出长度最短且覆盖整个待测海域的测线。

1.4 问题四的分析

通过审题我们可以首先对给出的海水深度附件进行数据的预处理，接着再进行海水深度进行可视化分析，得出深度的 3D 模型，并根据 3D 模型建立深度的坐标系，并根据坐标系求出每条边对应的坡度α，之后我们可以根据四条边的坡度α，引用第三问中的数学模型进行求解。

二、　模型假设 I

1.设模型海底坡度在测线范围内是均匀；

2.假设模型测量参数准确，如测线距离中心点的水平距离、换能器的开角等是准确

的，没有误差；

3.假设在迭代过程中，船只的速度和航向可以根据当前状态进行调整。

4.假设测深任务的约束条件是可满足的，不存在无解的情况

5.假设测深任务的目标函数是可微分的，可以通过梯度下降法进行优化。

三、 模型的建立与求解

3.1 问题一模型的建立与求解

3.1.1 模型的建立与分析

由于该海底并非为平面而是一个有坡度 α 的坡面，因此我们在这里我们根据高度构建出新的三角形，以海域中心为原点，建立直角坐标系,如图 1 所示：

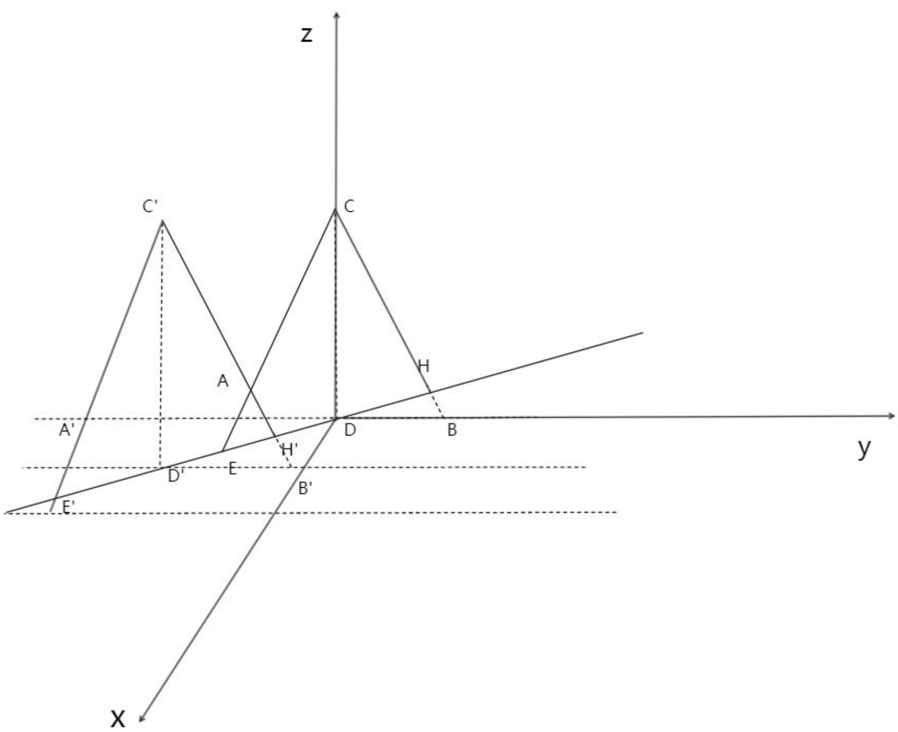


图 1

我们将 C`C 视为航船距离海域中心间距，并延长 DB 于 C`E`交于 A`。

首先，我们考虑单个待测海域的宽度，重叠率，这里我们首先考虑海域中心，如图 2 所示：

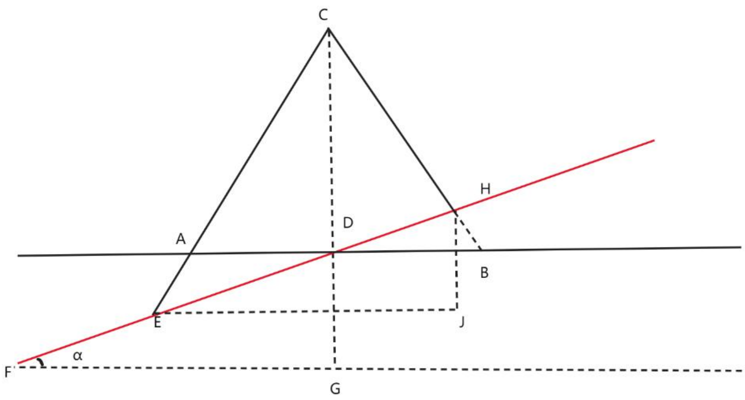


图 2

由题目给出

$$\angle ACB = \theta^{\circ} \text{、} CD = D$$

通过辅助线得出

$$\angle CAD = \angle DBH$$

∵辅助线 AB 平行于 FG 且 EJ 也平行于 FG
由于 ADE 与 HDB 为对顶角

$$\therefore \angle \mathrm{EFG} = \angle \mathrm{ADE} = \angle \mathrm{HDB}$$

由于 CAD 已知且 CAE 为 180° 可得出

$$\angle DAE = 180^{\circ} - \angle CAD$$

∠DAE 与∠ADE 已知，因此可以得出

$$\therefore \angle \mathrm{AED} = 180^{\circ} - \angle DAE - \angle ADE$$

同理可得到

$$\angle DHB = 180^{\circ} - \angle HDB - \angle HBD$$

通过三角函数，可以得知 AC 边与 CD 边的关系即

$$|AC| = \frac{|CD|}{cos\angle ACD}$$

根据三角形 CAD 的内角关系可得出

$$|AD| = |AC| \times \cos\angle CAD$$

$$\therefore |AD| = |DB|$$

由正弦定理可知

$$\because \frac{|ED|}{Sin\angle EAD} = \frac{|\mathrm{AD}|}{\sin\angle AED}$$

$$\because \frac{|DH|}{Sin\angle DBA}=\frac{|\mathrm{DB}|}{\sin\angle DHB}$$

通过上式变换可得

$$\therefore |ED|=\frac{2|CD|2com\times cos\angle CAD\times Sin\angle EAD}{Sin<AED}$$

$$\therefore |DH|=\frac{2|CD|\times cos\angle CAD\times Sin\angle DBH}{Sin\angle DHB}$$

3.1.2 **求解**

由题目可知，当夹角α为0°时，即图2，条带宽度W可以得出以下公式：

$$W=2D\times tan\frac{\theta}{2} \tag{1}$$

$$\eta=1-\frac{d}{W} \tag{2}$$

其中，W为条带宽度，D为海水深度，θ为多波束能器夹角，η代表相邻的两个覆盖宽度的重叠率

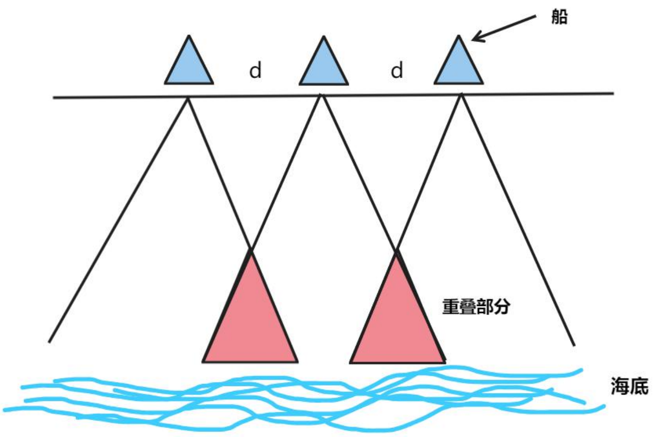


图3（由图可知d为200）

由图4所示，夹角α已知，为1.5°，|CD|（即海水深度）已知

因此，我们可以根据3.1.2的图表分析，条带宽度为|EH|在平面上的投影，即|JE|，我们可以建立W即(JE)的相关公式：

$$W=|EH|\times cos\,\alpha=|JE|$$

$$=\left(\frac{2|CD|\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2|CD|\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times cos\,\alpha \tag{3}$$

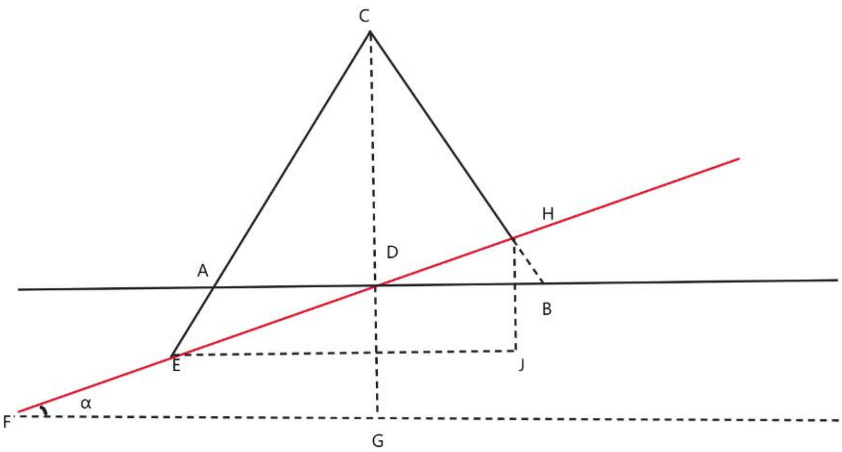


图 4

由于坡度 α，因此对于每一点海底深度不同的情况，我们根据表 1 中测线距中心的距离来计算不同点的深度，相邻的条带宽度如下图 5 所示：

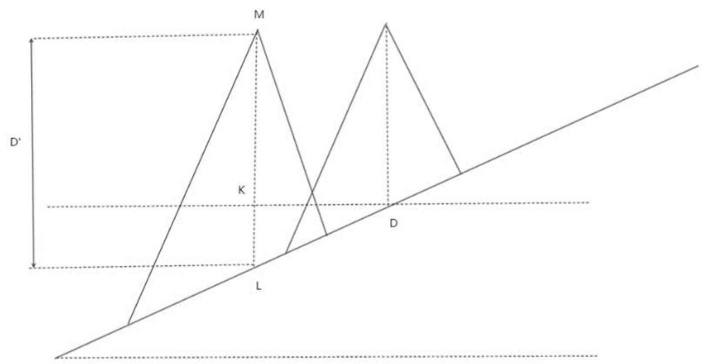


图 5

根据图 5 我们可以得知|ML|长度等于|MK|与|KL|之和，其中|MK|长度已知为海洋中心的深度，KL 可以通过侧线距中心点求出，公式如下所示：

$$D^{'} = D + x \cdot \tan\alpha \tag{4}$$

其中 D`为相邻的三角形的海洋深度，数值如下所示：

表 1 的计算结果

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	64.76	59.53	54.29	49.05
覆盖宽度 /m	314.95	296.81	278.68	260.54	242.4	224.27	206.13	188	169.86
与前一条测线 的重叠率/%	—	34.6	30.49	25.82	20.47	14.29	7.06	-1.49	-11.78

3.2 问题二模型的建立与求解

3.2.1 模型的建立与分析求解

在已给出的条件下，运用三角函数及几何关系建立船的侧线方向与海底法向的水平面的夹角β，建立起多波束测深宽度的数学模型。

我们通过分析和计算，将该模型分为三种情况，根据三种不同的航行角度，对应的深度，宽度也不同，因此我们在计算宽度与深度时，需要根据 β 角的情况进行分析处理

第一种情况当侧线方向的夹角为 0°，180°，即 β 角为 0 时，说明该航线与坡面法向的投影方向相同。如图 6 所示：

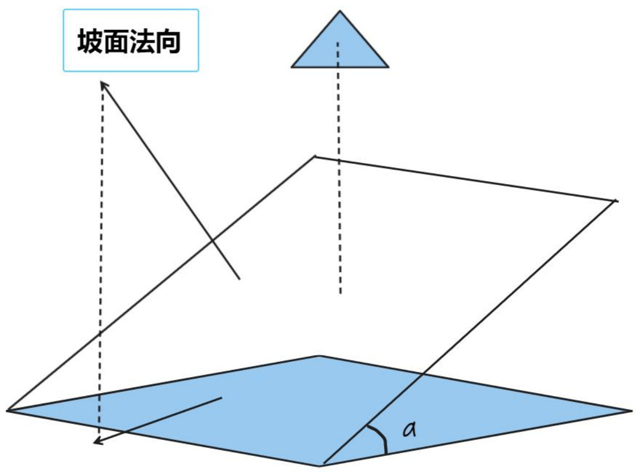


图 6

由此根据图 7 建立的宽度数学模型如下：

$$W=|EJ|=\left(\frac{2D\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2D\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times\cos\alpha \quad (5)$$

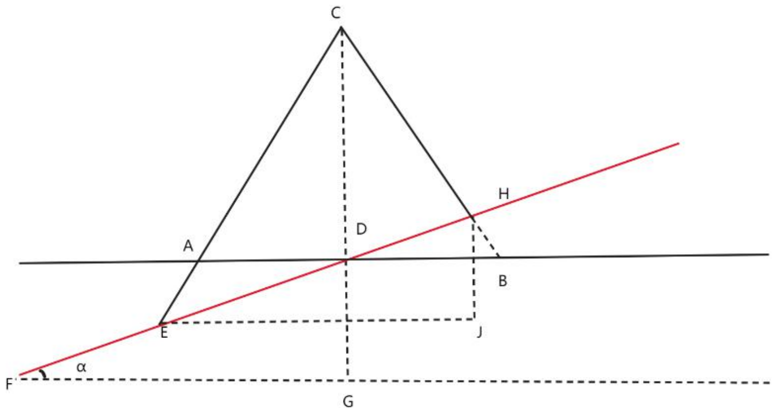


图 7

由于该坡面倾斜，因此对于计算下一个宽度时需要再次计算下一个三角形的深度|C`D`|,由图 8 可得以下方程：

$$D` = D + x \cdot tan\,\alpha \tag{6}$$

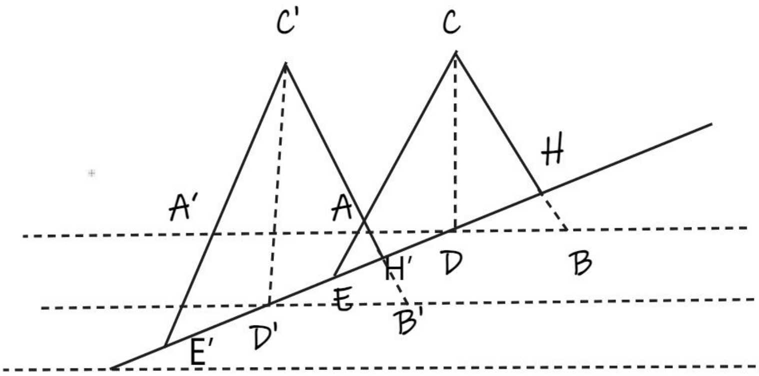


图 8

其中，x 为测量船距海域中心点处的距离

第二种情况，即β为 45°，135°，225°，315°，我们可以根据深度求出宽度如图 9 所示

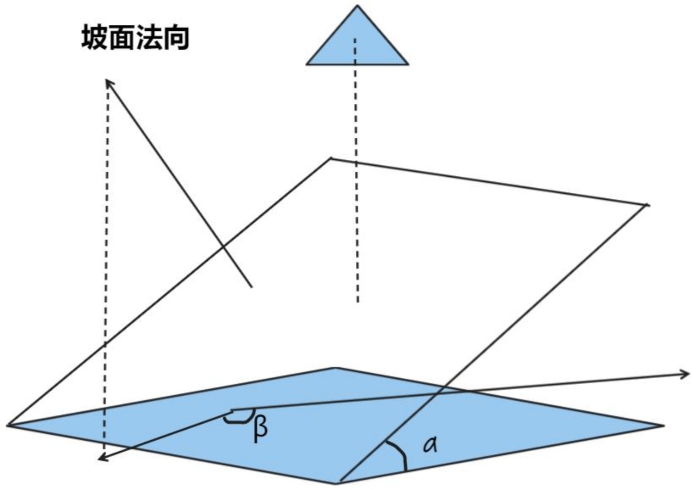


图 9

此时，我们可以画出该运动轨迹的图，即图 10：

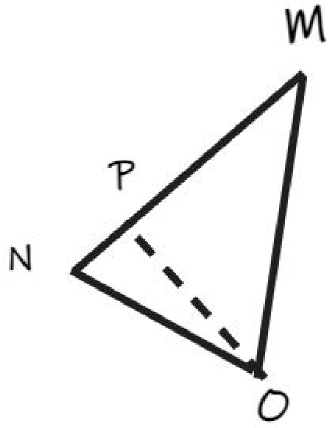


图 10

设 MN 为坡面法向投影，MO 为运动轨迹，角 NMO 为β角

由图 9 得知，O 点与 P 点的深度相同，其深度公式为：

$$D` = D + X \cdot \tan\alpha \times cos\ \angle PMO \tag{7}$$

得知深度，我们可以求得 MP

$$W = \left(\frac{2D\times cos\angle CAD\times sin\ \angle EAD}{sin\ \angle AED} + \frac{2D\times cos\angle CAD\times sin\ \angle DBH}{sin\ \angle DHB}\right)\times cos\,\alpha \tag{8}$$

我们在此以 MP 为底构建 3.1 的三角形，如下图 11 所示

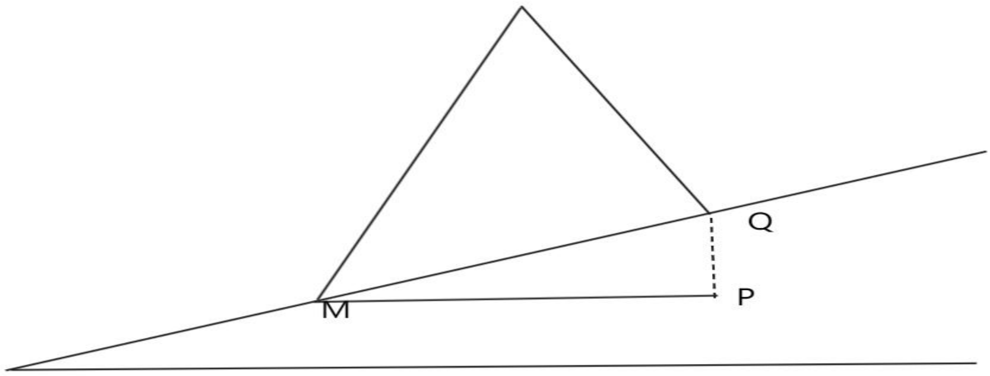


图 11

此时，我们需要求出|MQ|，由于∠QMP 已知，因此可列出方程：

$$\frac{|MP|}{COS\angle QMPQ} = |MQ| \tag{9}$$

跟据图 9，我们得知|MP|与|MO|的斜边的关系如下为：

$$\frac{|MP|}{COS|\angle PMO|} = |MO| \tag{10}$$

得知 MO 方向对应的斜边，则求 MO 方向对应的斜边在水平面的投影便可得出 MO 方向的宽度 W，其对应公式为：

$$W=|MO| \times \cos\alpha \tag{11}$$

第三种情况，即β为 90°，270°，其深度保持不变，因此宽度也不变，又由于 90°和 270°对称所以 90°的宽度与 270°的宽度相对应，根据图 12 可列出以下方程：

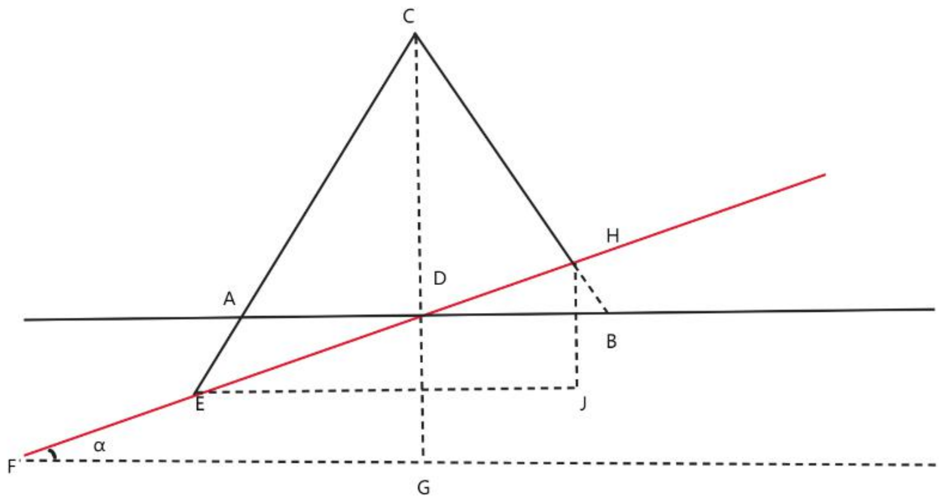


图 12

$$W = \cos\alpha\left(\frac{2D\times cos\angle CAD\times\sin\angle EAD}{\sin\angle AED} + \frac{2D\times cos\angle CAD\times\sin\angle DBH}{\sin\angle DHB}\right) \tag{12}$$

将已知数值代入上述方程，可求得以下计算结果，

表 2 的计算结果

覆盖宽度/m		测量船距海域中心点处的距离/海里							
		0	0.3	0.6	0.9	1.2	1.5	1.8	2.1
测线方向夹角/°	0	416.55	467.05	517.55	568.06	618.56	669.06	719.57	770.07
	45	589.09	639.59	690.09	740.59	791.10	841.60	892.11	942.61
	90	416.55	416.55	416.55	416.55	416.55	416.55	416.55	416.55
	135	589.09	538.59	488.08	437.58	387.08	336.58	286.07	235.57
	180	416.55	366.05	315.54	265.04	214.54	164.04	113.53	63.03
	225	589.09	538.59	488.08	437.58	387.08	336.58	286.07	235.57
	270	416.55	416.55	416.55	416.55	416.55	416.55	416.55	416.5
	315	589.09	639.60	690.09	740.60	791.10	841.60	892.10	942.61

3.3 问题三模型的建立与求解

3.3.1 模型的建立与分析

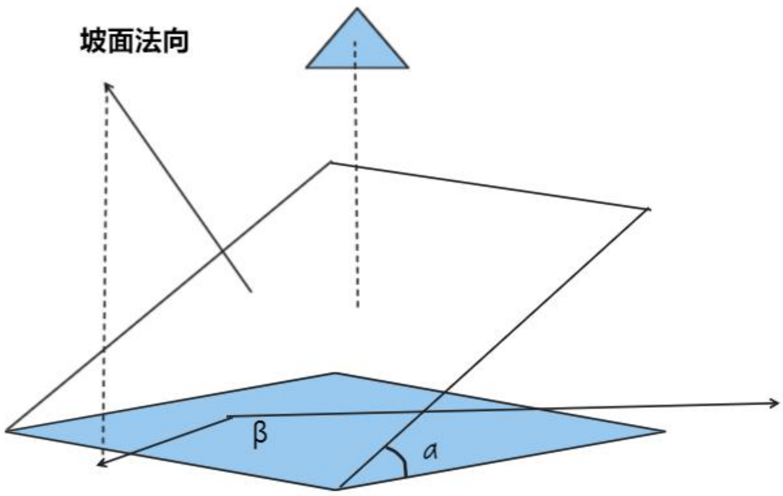


图 13

本题与第二问相似，已知海域南北长 2 海里，东西宽 4 海里（图 12），要求求出最短测线距离，但是坡面法向的投影与航向夹角 β，间距 d 并未给出明确数值。因此，我们需要自己设立相关约束，首先我们求出海域中心的宽度，设测线距离为目标优化函数，以 η，β，d 为约束条件，并求解出单个角度的总测线距离，而后通过不断迭代的思想，求出某个角度的最优总测线距离。

本题我们设总测线距离为 L，L 的模型为

$$L = min\sum L(\beta, \ d)$$

其中，满足条件 β=n,n=(0.1.2…..180°)，d=0.9w

中心海域侧面图如图 14 所示：

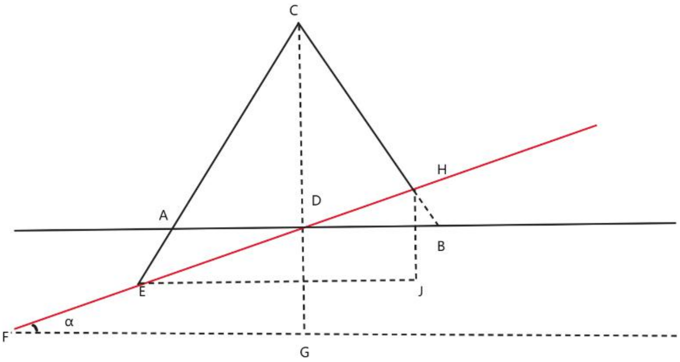


图 14

首先，我们设航船从海域中心点出发，海域中心的宽度如下所示：

$$W=\left(\frac{2D\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2D\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times cos\,\alpha \quad （13）$$

我们假设该船沿β角出发，深度与β角对应的公式为：

$$D`=D+X\times tan\alpha\times cos\beta \ （14） \quad (14)$$

但是，X 距海域中心的距离未知，且此时若按深度 D`计算 W，虽然能计算出对应的数值，但是会导致η不满足题目要求，因此我们需要先满足题目中η的要求，关于η的方程为

$$d=w-w\times\eta \quad (15)$$

从此式中不难看出，当η取值为 10%，d 的取值最大，相对应的间隔也最大，因此在深度随着航船移动不断上升时，η取 0.1，当深度随着航船不断下降时，η取 0.2。本题讨论中，我们设海洋深度随着航船移动不断增大。即η=0.1

通过上式，我们可以计算出测量间距 d，将计算出的 d 带入到开始给出的深度公式中，本题中深度公式为：

$$D`=D+d\times tan\,\alpha\times cos\,\beta \quad (16)$$

此时，我们可以将深度带入到计算下一个宽度的公式中，

$$W`=\left(\frac{2D`\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2D`\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times cos\,\alpha \quad （17）$$

3.3.2 求解

同样的，本题由于未给定β角，因此需要我们自己设立β角并进行检索校验。同时，我们也要跟据第二问得出的结论对第三问进行求解。

由于β角关于海域中心对称，因此我们只需要求出 0-180°的最优解即可。

β角的取值同样分为两种情况。

第一种情况，当β角为 90°时，海洋深度不变，而且此时宽度 w 也不会改变，因此间距可以直接跟据η=1-$\frac{d}{W}$求出，此时η为 0.1

$$W=\left(\frac{2|D|\times cos\angle CAD\times\sin\angle EAD}{\sin\angle AED}+\frac{2|D|\times cos\angle CAD\times\sin\angle DBH}{\sin\angle DHB}\right)\times\cos\alpha \tag{18}$$

$$\eta=1-\frac{d}{W}$$

第二种情况，当β角非 90° 角时，深度和宽度会随着航船的移动而变化，而且此时η的取值也与航船的移动方向有关。对此，我们先求出位于海域中心点的宽度 w，以备用。海域中心的宽度公式为：

$$W=\left(\frac{2|D|\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2|D|\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times cos\,\alpha \tag{19}$$

之后我们用宽度求对应的间距 d，其中η的值取 0.1

$$d=w-w\times\eta \tag{20}$$

得出间距 d，我们可以利用间隔求出下一点的深度：

$$D`=D+d\times tan\,\alpha\times cos\,\beta \tag{21}$$

跟据 D`我们可以求出 W`

$$W`==\left(\frac{2D`\times cos\angle CAD\times sin\angle EAD}{sin\angle AED}+\frac{2D`\times cos\angle CAD\times sin\angle DBH}{sin\angle DHB}\right)\times cos\,\alpha \tag{22}$$

此时，我们可以检验一下η的值是否为 0.1-0.2 之间

$$\eta=1-\frac{d}{\frac{W+W`}{2}} \tag{23}$$

若得出η范围正确，我们将 d 保留，如果η范围不正确，我们将 d 舍去。

之后我们在该β角度的情况下，进行持续迭代，重复上述操作。可是题中给出了海域的宽度，因此当我们保留的 d 的值超过海域的宽度 2 海里时，需要中止迭代，并选取下一个角度继续进行检索校验并寻找 d 总和的最小值，直至 180° 为止。

3.4 问题四模型的建立与求解

3.4.1 模型的建立与分析

根据该题给出的附件，我们可以先利用 matlab 进行绘图，得到的深度图如图 15 和图 16 所示：

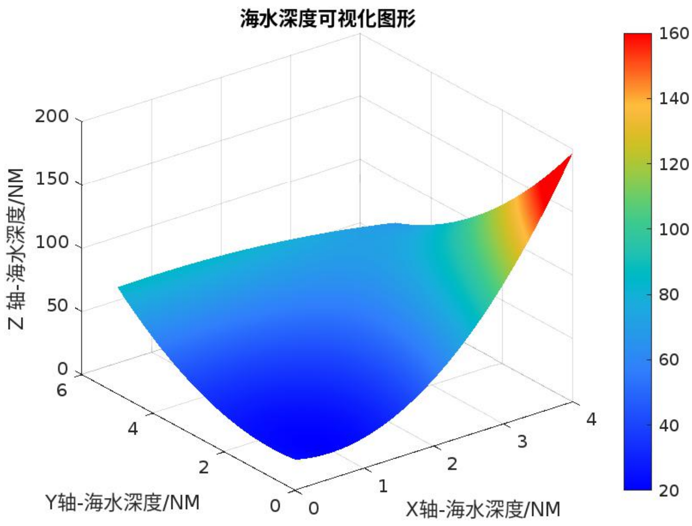


图 15

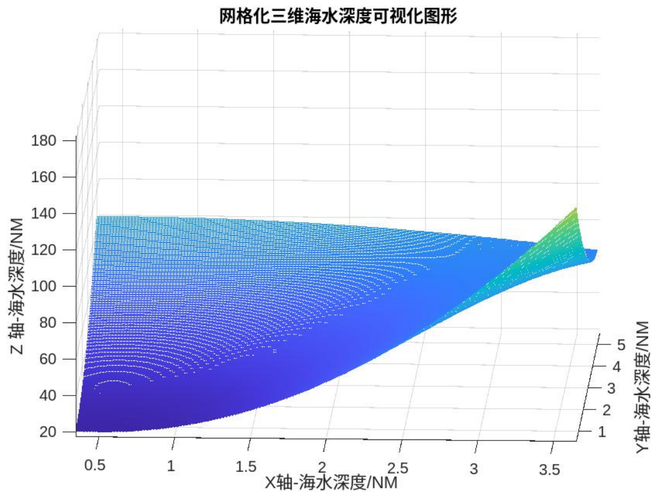


图 16

我们对此建立三角形的测线长度总和数学模型 1

$$L = min\sum_{i=1}^{2}\sum_{j=1}^{n_i} L_{ij}$$

其中n_i第 i 个三角形的海域测线数量，L_{ij}为第 i 个三角形海域的第 j 条测线长度

分析深度图，我们不难发现四个比较特殊的点，其 XY 轴坐标分别为：（0，0），（4，0），（0，5），（4，5）对应的深度分别为 24.4 米，197.2 米，84.4 米，65.2 米，为了方便计算，我们选择建立深度坐标系。如图 17 所示：

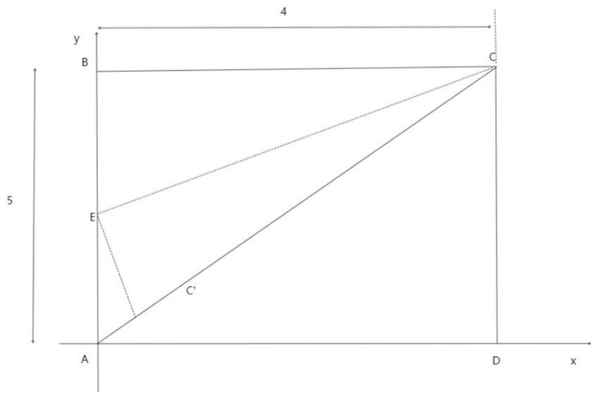
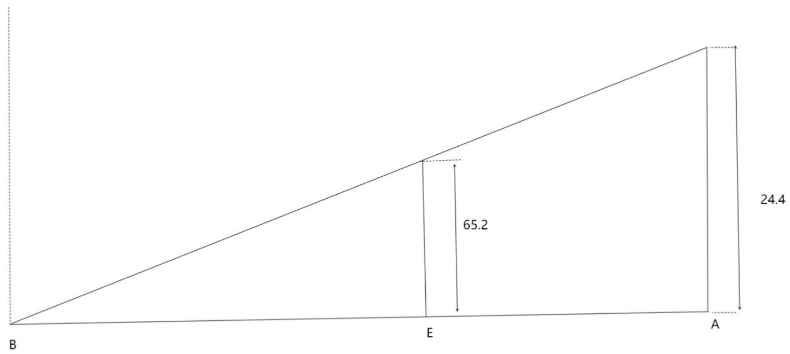


图 17

连接 AC，设 AB 上有一点 E 与 C 点的高度相同，连接 CE。此时我们并不知道 E 的具体位置，因此我们可以利用相似三角形对 E 的具体位置求解，我们构造出以 AB 为底的三角形，如图 18 所示：



由图中可得知以下公式：

$$BE/EA = 19.2/60$$

$$AE = 5\ \times 19.2/60$$

$$BE = 5 - AE$$

得知 BE，EA，可根据勾股定理，得出 CE 边长度:

$$CE = \sqrt{BC^2 + BE^2}$$

此时，我们可以根据三角形 ABC 的边 AB 与 BC，用 arctan 求出角度∠ACB，得知角度为 51.34°；

同理，我们求出∠ECB 为 40.40°；

又得知∠ACE=∠ACB+∠ECB

因此我们可以得出∠ACE 为 10.94°。

此时，我们可以根据三角函数求出 CC`，C`E。如下式所示：

$$CC` = CE/COS\angle ACE$$

$$C`E = CC` \cdot tan\angle ECA$$

我们设 C`的高度为 H;

根据图 19 可得知 H 的关系为：

$$H/40.8 = C`C/AC$$

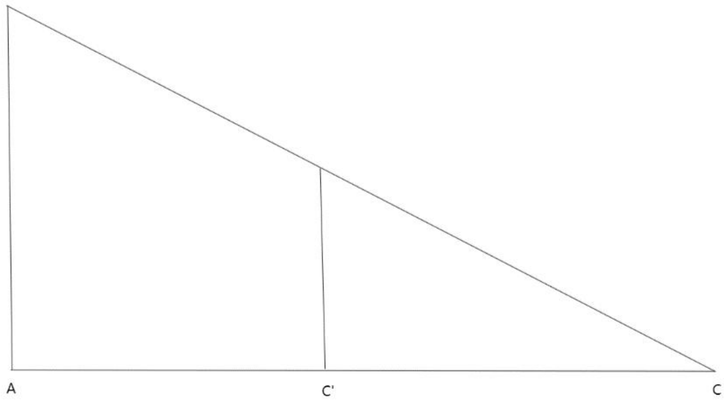


图 19

将 H 求出后，我们可以根据 arctan 求出坡度 α

$$\alpha = arctanH/EC$$

求出 α 后，我们设多波束换能器的开角为 120°；

并对第一问中的三角形重新构造

如图 20 所示:

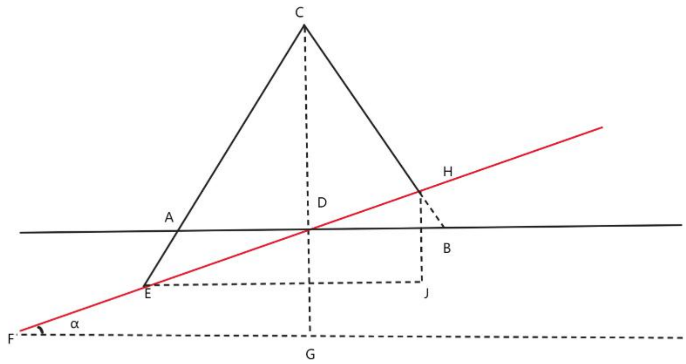


图 20

图中，∠ADE 为 α°，∠AED 为 $30° - \alpha°$；

∠HDB 为 α°,∠DHB 为 $150° - \alpha°$ 其余角度并未改变；

由此，我们可以根据第三问的模型进行求解。但是我们只算出了一侧的数据，还有另一侧数据并未做出改变，因此我们还需计算另一面坡度的数据。

我们再次建立深度图的平面直角坐标系，如图 21 所示

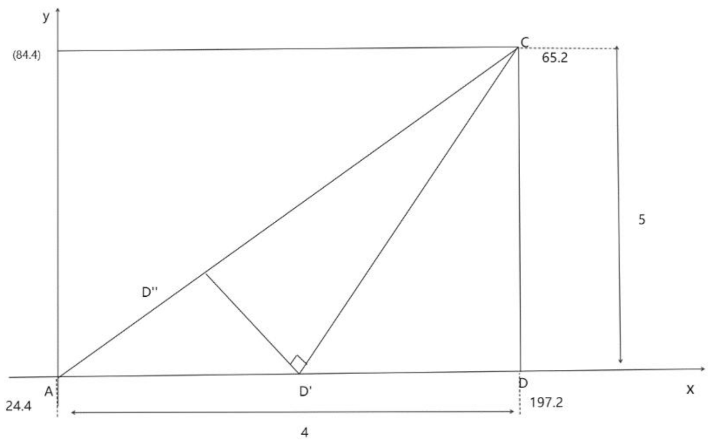


图 21

同样的，我们连接 AC，设 AD 边上存在一点 D`与 C 的高度相同，连接 CD`，并连接 D`D``。

同样的，要求出 D`的位置，我们需要构造相似三角形进行计算，如图 22 所示：

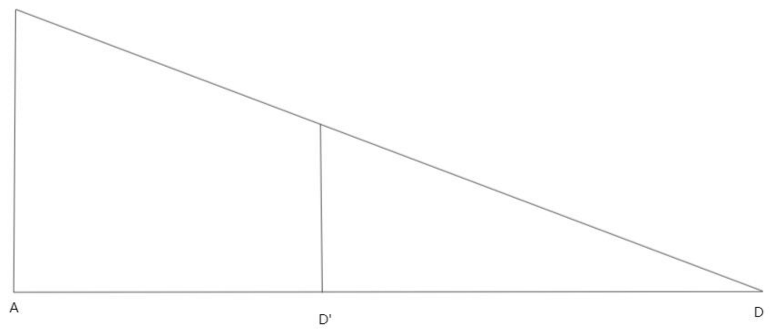


图 22

由图中可得知，$AD`/D`D = 172.8/132$；

因此我们可以得知 AD`值为 4 $\times 132/172.8 = 3.06$；

同理我们可以求出 DD`的值为 0.94；

由于第一次计算中我们已经得出∠BCA 的值；

因此我们只需要 90° $- \angle BCA$；

便可得出∠ACD 的值为 38.66°；

∠D`CD 为 10.67°，∠ACD`为 27.99，根据已知角度，

我们可以求 CD``与 AD``，公式为；

$$CD`/COS\angle ACD` = CD``$$

$$AD`` = AC - CD``$$

CD`与 CD``已知，可求出 D``D`；

$$\mathrm{D``D`} = \sqrt{\mathrm{CD`}^2 + \mathrm{CD``}^2}$$

接着设 D``高度为 H`

AC 边可构造三角形，如图 23 所示：

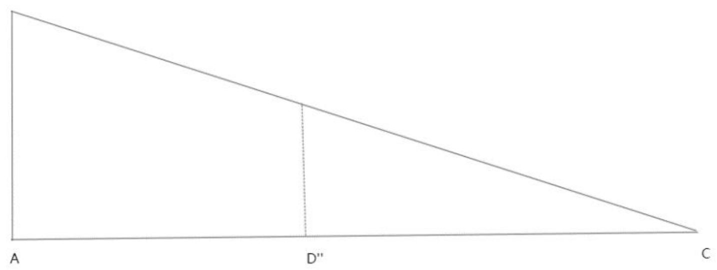


图 23

根据对边成关系，得知

$$H` = 40.08 \times D``C/AD``$$

求出 H`后，我们可以用 arctan 求出角度α

$$ArctanH/D`D`` = \alpha$$

这里我们求出α为 4.42°

同样的，我们求出α后，需要重新构造第一问中的三角形，与上述步骤相同，因此不再赘述。而后再以相同的步骤，求出不断迭代求出α，优化模型。

四、 模型的评价改进与推广

4.1 模型的优缺点

三角函数和几何关系模型优点：基于基本的几何关系，三角函数和几何关系模型简单直观，易于理解和应用，在初等数学基础上进行，通过反复求证，我们的模型经过检验，可以提供较为准确的结果。在同等条件下可以适用不同的测线和海底横截面问题，模型具有一定的实际应用。

缺点：模型建立在特定的假设基础上，且使用了一些近似数据，可能导致随着测线距离中心点的增加，误差逐渐累积，从而降低结果的准确性。

迭代优化的优点：通过多次迭代逐步优化逼近我们的目标函数，根据具体问题进行调整和优化，根据需要选择不同的迭代策略。通过改变迭代策略，控制模型的精度，使结果逼近目标函数。

缺点：该模型通常需要进行多次迭代才能达到所需的精度，需要进行多次试验和调整。相比于一些直接解法，该模型通常需要更多的计算步骤和迭代次数。

4.2 模型的改进与推广

问题一、二主要应用了三角函数和几何关系模型，对于该模型，可以进行实地验证和校正，例如对于复杂的海底地形和水深变化情况，可以引入更好的优化模型，考虑更多的因素，提高模型的适应性，使该模型的精度和可靠性更加准确。通过将这些模型推广应用于更广泛的领域，积累更多的实践经验。

迭代法的改进与推广，通过改变迭代公式、引入加速技术等方法来改进迭代法的效率。利用并行计算技术，将迭代过程中的计算任务分配到多个处理单元上，加快迭代速度，提高计算效率。也可以将迭代法与其他数值方法结合，提高数值模拟的准确性和效率。同时通过与实际问题的对比和验证，对迭代法进行实践验证和校正，进一步改进和完善迭代模型。

可以将迭代法应用于更广泛的领域，如机器学习、数据处理等，积累更多的实践经验，推动迭代法的发展和应用。

五、 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，海洋技术学报，第 35 卷 第 2 期：87-91 页，2016.04，

[2] [2]司守奎，孙玺菁，python«数学建模算法与应用》，北京，国防工业出版社，2022.01

六、　附录

附录 1：
介绍：支持文件的材料列表
1.xlsx First.py Fourth1.py Fourth2.py result1.xlsx result2.xlsx Second.py Third.py TUXING.txt

附录 2
介绍：

问题一代码：

```
import math
import pandas as pd

# 给定一些参数及数据
H_center = 70
theta = math.radians(120)
alpha = math.radians(1.5)
d = 200
distancesCenter = [-800, -600, -400, -200, 0, 200, 400, 600, 800]

# 距中心的距离来计算每个位置深度
depths = {}
for dist in distancesCenter:
    H = H_center - dist * math.tan(alpha)
    depths[dist] = H

# 根据距中心的距离计算每个位置的覆盖宽度 w
coverageWidths = {}
for dist, depth in depths.items():
    w1 = (
        2 * depth * math.cos(math.radians(30)) * math.sin(math.radians(30))
    ) / math.sin(math.radians(148.5))
    w2 = (
        2 * depth * math.cos(math.radians(30)) * math.sin(math.radians(150))
    ) / math.sin(math.radians(28.5))
    wPrime = (w1 + w2) * math.cos(alpha)
    coverageWidths[dist] = wPrime

overlapPercentages = {}
previousWidth = None
for dist, width in sorted(coverageWidths.items()):
    if previousWidth is not None:
        overlap = 1 - (d / ((previousWidth + width) / 2))
        overlapPercentages[dist] = overlap * 100
    previousWidth = width

# 创建一个 DataFrame 来存储数据
dataDict = {
    '距中心米': distancesCenter,
```

```
            '深度米': [depths[dist] for dist in distancesCenter],
            '覆盖宽度米': [coverageWidths[dist] for dist in distancesCenter],
            '重叠率%': [overlapPercentages.get(dist, None) for dist in distancesCenter]
    }

    df = pd.DataFrame(dataDict)


df.to_excel('result1.xlsx', index=False)
```

附录 3
介绍：该代码是用 python 语言编写的，作用是

问题二代码：

```
import math
import pandas as pd

# 一些相关参数的定义说明
H_center = 120   # 海域中心点的深度（米）
wthetaDeg = 120   # 开角（度）
walphaDeg = 1.5   # 坡度角（度）
wangleDeg = [0, 45, 90, 135, 180, 225, 270, 315]   # 测线方向夹角（度）

# 弧度转换
theta = math.radians(wthetaDeg)
alpha = math.radians(walphaDeg)

# 创建一个空的 DataFrame 用于存储数据
data = pd.DataFrame(index=[f"{angle}°" for angle in wangleDeg], columns=["测量船距海域中心点处的距离/海里"] + [f"{dist}海里" for dist in [0, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.1]])

# 遍历每个夹角
for angle in wangleDeg:
    wdeg = math.radians(angle)

    # 计算深度
    wdepths = {}
    distancesCenter = [0, 0.3, 0.6, 0.9, 1.2, 1.5, 1.8, 2.1]
    for wdist in distancesCenter:
        if angle == 90 or angle == 270:
            D = H_center
        else:
            D = H_center + wdist * 1852 * math.tan(alpha) * math.cos(wdeg)
        wdepths[wdist] = D

    # 覆盖宽度
    coverage_widths = {}
    for wdist, depth in wdepths.items():
        w1 = (
            2 * depth * math.cos(math.radians(30)) * math.sin(math.radians(30))
        ) / math.sin(math.radians(148.5))
        w2 = (
            2 * depth * math.cos(math.radians(30)) * math.sin(math.radians(150))
```

```
            ) / math.sin(math.radians(28.5))
            if angle == 90 or angle == 270:
                w_prime = (w1 + w2) * math.cos(alpha)
            else:
                w_prime = ((w1 + w2) * math.cos(alpha)) / abs(math.cos(wdeg))
            coverage_widths[wdist] = w_prime

        # 数据添加至 DataFrame
        data.loc[f"{angle}°", "测量船距海域中心点处的距离/海里"] = angle
        for wdist, width in coverage_widths.items():
            data.loc[f"{angle}°", f"{wdist}海里"] = width

    # 数据写入
    data.to_excel("result2.xlsx", index=False)
```

附录 4
介绍：

问题三代码：

```
import math

# 题目中条件和一些约束
H_Center = 110
theta_deg = 120
alpha_deg = 1.5
angle_range = range(0, 360)
theta = math.radians(theta_deg)
alpha = math.radians(alpha_deg)

# 下边进行一些变量的初始化 满足我们的需求
angle_maxMatching = float("inf")   # zh 用来匹配最小度数
sum_maxMatchingd = float("inf")   # 用于匹配最小间距总和

iterations = 0
w_iteras = 0
for angleDeg in angle_range:
    H_Center = 110
    iterations = 0
    wTotal = 0
    sum_d = 0
    deg = math.radians(angleDeg)
    # 计算航船一侧的覆盖宽度
    w1 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(30))
    ) / math.sin(math.radians(148.5))
    # 另一侧的覆盖宽度
    w2 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(150))
    ) / math.sin(math.radians(28.5))
    # 总的覆盖宽度
    w_prime = (w1 + w2) * math.cos(alpha)
    # 间距取极大
    d = 0.9 * w_prime
    n = d
    d_total = d
    while n < 3670:
        # 当累计间距达到 2 海里时结束
```

```
            H_Center = H_Center + d * math.tan(alpha) * math.cos(deg)
            # 计算下一个航点的深度
            w_prime = (w1 + w2) * math.cos(alpha)

            w1 = (
                2      *      H_Center      *      math.cos(math.radians(30))      *
math.sin(math.radians(30))
            ) / math.sin(math.radians(148.5))
            w2 = (
                2      *      H_Center      *      math.cos(math.radians(30))      *
math.sin(math.radians(150))
            ) / math.sin(math.radians(28.5))
            w_prime1 = (w1 + w2) * math.cos(alpha)
            woverlap = 1 - d / ((w_prime + w_prime1) / 2)
            # 当深度变小/变大时 d 取最小/最大
            if math.cos(deg) < 0:
                d = 0.8 * w_prime1
            else:
                d = 0.9 * w_prime1

            n = n + d
            # 满足重叠率要求进入判断 if
            if 0.1 <= woverlap <= 0.2 or angleDeg == 90:
                iterations += 1
                print(f"Iteration {iterations}:")
                print(f"  测线方向夹角 deg(度): {angleDeg}")
                print(f"  宽度 w_prime = {w_prime}")
                print(f"  间距 d = {d}")
                print(f"  高度 D_center = {H_Center}")
                print(f"  第二个宽度 w_prime1 = {w_prime1}")
                print(f"  重叠率 = {woverlap:.2%}")
                wTotal = wTotal + w_prime
                w_iteras = w_iteras + 1
                d_total = d + d_total
                sum_d = sum_d + d
        if wTotal == 0 and iterations == 0:
            continue
        else:
            w_avg = (wTotal + w_prime1) / (iterations + 1)
            wnum = 3760 / w_avg
        print(f"  测线大小={d_total}")
        print(f"  平均宽度={w_avg}")
        print(f"  该角度下所需条带数={wnum}")
        print(f"  该度的间距总和={sum_d}")
```

```
        if sum_d < sum_maxMatchingd:
            sum_maxMatchingd = sum_d
            angle_maxMatching = angleDeg

    print(f"最小间距总和: {sum_maxMatchingd}")
    print(f"对应的度数: {angle_maxMatching} 度")

    print(f"总共迭代了 {w_iteras} 次")
```

附录 5
介绍：该代码是用某某某语言编写的，作用是什么

```
import math

# 题目中条件和一些约束
H_Center = 45.863
theta_deg = 120
alpha_deg = 1.39
angle_range = range(0, 180)
theta = math.radians(theta_deg)
alpha = math.radians(alpha_deg)

# 初始化变量 这里的变量为下边的总测算
sm1 = 0
sm2 = 0
sm3 = 0

iterations = 0
zongoverlap = 0
witera = 0
chongdiechao = 0
for angleDeg in angle_range:
    n = 0

    w1 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(30))
    ) / math.sin(math.radians(148.61))
    # 另一侧的覆盖宽度
    w2 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(150))
    ) / math.sin(math.radians(28.61))
    # 总的覆盖宽度
    w_prime = (w1 + w2) * math.cos(alpha)
    # 间距取极大
    d = 0.9 * w_prime
    n = d
    d_total = d

    while n < 5560.05:   # 5560.05
        # 当累计间距达到 2 海里时结束
        H_Center = H_Center + d * math.tan(alpha) * math.cos(math.radians(angleDeg))
        # 计算下一个航点的深度
```

```
            w1 = (
                2        *       H_Center        *        math.cos(math.radians(30))        *
math.sin(math.radians(30))
            ) / math.sin(math.radians(148.61))
            #  另一侧的覆盖宽度
            w2 = (
                2        *       H_Center        *        math.cos(math.radians(30))        *
math.sin(math.radians(150))
            ) / math.sin(math.radians(28.61))
            #  总的覆盖宽度
            w_prime1 = (w1 + w2) * math.cos(alpha)
            woverlap = 1 - d / ((w_prime + w2) / 2)
            n += d
            d_total += d

            w_prime1 = (w1 + w2) * math.cos(alpha)
            woverlap = 1 - d / ((w_prime + w2) / 2)
            if 0.1 <= woverlap <= 0.2:
                witera += 1

                print(f"   宽度  w_prime = {w_prime}")
                print(f"   间距  d = {d}")
                print(f"   高度  D_center = {H_Center}")
                print(f"   第二个宽度  w_prime1 = {w_prime1}")
                print(f"   重叠率  = {woverlap:.2%}")
                sm1 = sm1 + d_total
            #  判断条件执行
            if woverlap < 0:
                iterations += 1
                sm2 = sm2 + d_total
            if woverlap > 0.2:
                sm3 = sm3 + d_total
                chongdiechao = d + chongdiechao
d_total = sm1 + sm2 + sm3
print(f"   测线超过 20%大小总长度={chongdiechao}")   #
print(f"   测线大小={d_total}")
# print(f"   0.1 <= woverlap <= 0.2 的次数:={witera}")
zongoverlap = iterations / (iterations + witera)
print(f"   漏测的总覆盖率所占的百分比={zongoverlap:.2%}")
```

```
# 改变α
import math

# 题目中条件和一些约束
H_Center = 45.863
theta_deg = 120
alpha_deg = 4.42
angle_range = range(0, 180)
theta = math.radians(theta_deg)
alpha = math.radians(alpha_deg)

# 初始化变量 这里的变量为下边的总测算
sm1 = 0
sm2 = 0
sm3 = 0

iterations = 0
zongoverlap = 0
witera = 0
chongdiechao = 0
for angleDeg in angle_range:
    n = 0

    w1 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(30))
    ) / math.sin(math.radians(145.58))
    # 另一侧的覆盖宽度
    w2 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(150))
    ) / math.sin(math.radians(25.58))
    # 总的覆盖宽度
    w_prime = (w1 + w2) * math.cos(alpha)
    # 间距取极大
    d = 0.9 * w_prime
    n = d
    d_total = d

    while n < 5174.7:   # 5560.05   2.82*1835
        # 当累计间距达到 2 海里时结束
        H_Center = H_Center + d * math.tan(alpha) * math.cos(math.radians(angleDeg))
        # 计算下一个航点的深度
        w1 = (
```

```
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(30))
    ) / math.sin(math.radians(145.58))
    # 另一侧的覆盖宽度
    w2 = (
        2 * H_Center * math.cos(math.radians(30)) * math.sin(math.radians(150))
    ) / math.sin(math.radians(25.58))
    # 总的覆盖宽度
    w_prime1 = (w1 + w2) * math.cos(alpha)
    woverlap = 1 - d / ((w_prime + w2) / 2)
    n += d
    d_total += d

    w_prime1 = (w1 + w2) * math.cos(alpha)
    woverlap = 1 - d / ((w_prime + w2) / 2)
    if 0.1 <= woverlap <= 0.2:
        witera += 1

        print(f"宽度 w_prime = {w_prime}")
        print(f"间距 d = {d}")
        print(f"高度 D_center = {H_Center}")
        print(f"第二个宽度 w_prime1 = {w_prime1}")
        print(f"重叠率 = {woverlap:.2%}")
        sm1 = sm1 + d_total
    # 判断条件执行
    if woverlap < 0:
        iterations += 1
        sm2 = sm2 + d_total
    if woverlap > 0.2:
        sm3 = sm3 + d_total
        chongdiechao = d + chongdiechao
d_total = sm1 + sm2 + sm3
print(f"测线超过 20%大小总长度={chongdiechao}")  #
print(f"测线大小={d_total}")
# print(f"0.1 <= woverlap <= 0.2 的次数:={witera}")
zongoverlap = iterations / (iterations + witera)
print(f"漏测的总覆盖率所占的百分比={zongoverlap:.2%}")



# 图形绘制
% 从 Excel 文件中读取数据
filename = '1.xlsx';   % 文件名
```

```
sheet = 1;
xlRange = 'A1:GS251';    % 数据范围

data = xlsread(filename, sheet, xlRange);

% 创建横向坐标和纵向坐标
x = 0:0.02:4;
y = 0:0.02:5;

% 创建网格点
[X, Y] = meshgrid(x, y);

% 绘制三维曲面图
figure;
surf(X, Y, data);
xlabel('X 轴-海水深度/NM');
ylabel('Y 轴-海水深度/NM');
zlabel('Z 轴-海水深度/NM');
title('海水深度可视化图形');
```