多波束测深系统测线排布方案

摘　要

多波束测深系统利用声波在水中传播并反射的特性，通过记录声波发射到接受的时间可求出海水的深度，在海洋测绘中应用广泛。面对不同海域地形如何进行测线设置是实际操作的关键问题，合理的测线设置不仅可以减少工作量，还能提高检测的精度。本文通过对不同复杂海域情况的分析，利用海底坡面与测线的几何关系，给出了不同情况下覆盖宽度和覆盖率的计算公式，并对复杂的海域情况提出相应的布线策略。

针对问题一 由中心海域海底深度D_0和海底坡度α的值求出不同测线对应的海底深度D_i，由于不同测线之间相互平行，测线之间呈现**线性关系**$D_i = D_0 - d_i \tan\alpha$，随后即可通过正弦定理求出每条测线的覆盖宽度$W_i$。由平坦海底的重叠率定义进一步推广得海底坡面上当相邻覆盖宽度不同时的重叠度定义，进而求出结果。

针对问题二 在引入新变量测线角度β后，可通过求出两个关键夹角将问题转化为问题一求解，即求出测线在斜坡上的投影与水平面的夹角γ，测波覆盖线与水平面的夹角φ。求得为了便于表示图中的几何关系，在所给图中建立**空间直角坐标系** O-xyz，再通过向量的**叉乘**求解，$\gamma = \arctan(\tan\alpha\sin\beta)$，$\varphi = \arctan(\tan\alpha\tan\beta)$，由问题一的求解公式推广即可问题二求得覆盖宽度问题

针对问题三 在该问中，可以通过定义与问题二相同的β，即测线方向与海底坡面法向量在水平面上的投影的夹角，将问题二的模型运用到问题三的求解中。在测线平行的情况下，分析不同的 beta，发现当测线与等深线平行时能满足题目要求。因此考虑测线与等深线平行的情况，因为这种情况下所有的测线长度相同，所以只须求最小的测线数量。显然，当重叠率η取最小即 10%时，测线数量取最小。然后我们根据η表达式，可以得到测线之间的间隔距离 d，并据此求得最少的测线数量 34。

针对问题四 可通过将问题简化成第三问中的模型来求解。我们根据数据绘制等深线图，根据等深线密集程度，将其划分为三个部分。再使用最小二乘法将各个分割面拟合为有一定倾斜角的平面。接下来通过第三问的最短测线排布方式对各个坡面求解测线排布位置。由此可以求解出最短测线总长度为 371481.81m。对于实际的测量面积和重叠情况的求解，先进行坐标变换，将网格数据放入 x 轴与测线平行的坐标系内，通过测线的位置及步长迭代判断，确定测线两端覆盖点位置，得到近似漏测海区占总待测海域面积的百分比为 7.68%及重叠率超过 20%的总测线长度为 12407.49m。

关键词：多波束测线　最小二乘拟合　步长迭代　　测线排布方案

一、问题重述

1.1 问题背景

单波束测深技术利用声波在水中的传播特性测量水体深度，利用声波在介质中沿直线传播并遇不同界面反射的特性，测量船向水底发射声波并记录声波从发射到信号接收的时间，由声波在海水中的传播速度和传播时间计算得海水深度。但是由于单波束测量采用单点连续的测量方法，其数据沿测线密集而在测线之间则没有数据。

由此基础出现了多波束测深系统，其工作原理为在与测线垂直的平面中发射大量波束，进而得到以测线为轴的全覆盖水深条带。

所得到多波束测深条带的覆盖宽度 W 随着换能器开角 θ 和水的深度 D 的改变而改变。在平坦的海底处，我们将海面 $\eta=1-\dfrac{d}{W}$ ，d 表示两相邻测线的距离，W 表示条带的覆盖宽度。$\eta<0$ 表示测量有遗漏，为了保证测量便利和确保数据的完整性，应使相邻条带之间有 10%~20%的重叠率。

由于实际地形起伏变化大，采用海面平均水深设计测线间距时，尽管保证了重叠率的要求，但是会导致水深较浅处漏测，影响测量准确性。而直接选用浅水处水深设计测线间隔，则会在水深处出现大量重叠情况，导致大量多余数据的测量，降低了效率。

1.2 问题提出

（1） 问题 1 与测线方向垂直的平面与海底面相交得到一条与水平面夹角为 α 的直线。以题中所给示意图建立出多波束测深条带覆盖宽度与相邻条带之间重叠率的数学模型，由题给条件，对于测线距中心点不同距离处计算海水深度，覆盖宽度及与前一条测线的重叠率并填入表 1。

（2） 问题 2 在一片矩形待测海域中，测线方向与海底坡面法向在水平面上投影的夹角为 β，以此构造多波束测深覆盖宽度的模型，由图给条件求出不同位置的覆盖宽度并填入表 2。

（3） 在一片与水平面成一定夹角的矩形海域中，由题设条件设计一组测线，需满足条件“测量长度最短”，“测量条带覆盖整片海域”且“相邻条带之间重叠率满足 10%~20%”。

（4） 以附件所给对某海域进行单波束测量的数据参考，设计多波束测量的测量布线并需满足以下需求：所得条带应覆盖整片海域，且将相邻条带重叠率需控制在 20%以下，同时使测线长度最短。并计算“测线总长度”，“漏测海域所站百分比”，“重叠部分超过 20%的总长度”

二、 问题分析

2.1 问题一的分析

问题 1 要求建立测深的覆盖宽度和相邻条带重叠率。以便于后续第二问转化后的数值计算与求解。建模思路流程如下：

角 B,C 可由运算得到，利用不同轨迹相互平行的特性，可由中心点的海水深度D_0和海域的坡度α求出距中心点不同距离处点的海水深度D_i，再利用正弦定理求出不同点所测得的覆盖宽度W_i。

计算相邻条带重叠度则由题目中相邻条带重叠率定义$\eta=1-\dfrac{d}{W}$推广得到在坡面上条带重叠率的公式：$\eta_i=\dfrac{2\times(W_{ib}+W_{(i+1)a}-d)}{W_i+W_{i+1}}$,(其中 a，b 分指覆盖宽度的左右)其中$W_i$与$W_{i+1}$为相邻测线的覆盖宽度。

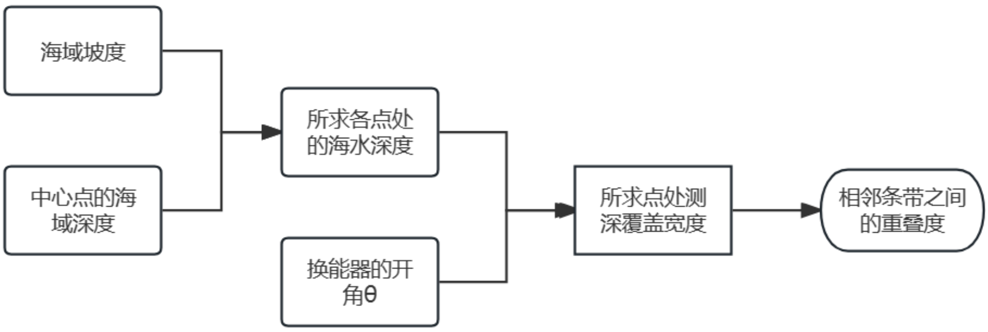


图 1 问题一问题求解流程

根据该模型即可求得所需数据，并将结果填入表中

2.2 问题二的分析

分析问题二时，我们发现问题一是问题二的一个特殊情况((即$\beta=90°$)，且问题二的模型可经过转化后使用问题一求解，所以只需考虑对模型二进行变化。

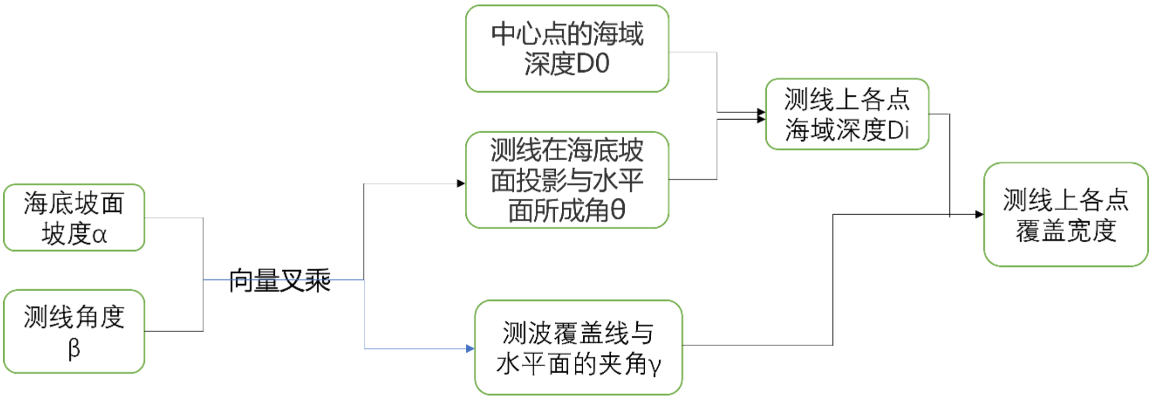
问题二在问题一的基础上进行了两个变化：

增加了变量测线角度β，且β的取值不再是固定的$90°$，固此时测线上各点处海域深度D_i与中心点海域深度D_0不再成简单的线性关系；此时需求出测线在海底坡面上

的投影与水平面所成角γ，计算可通过建立坐标系后对向量进行叉乘求得。

另一个变化测线上不同点处在海底坡面上的测深覆盖线与水平面的夹角γ不再是固定的α值，同理也可通过向量叉乘求出γ。

由此我们将模型二与模型一联系起来，得到新的W_i公式，其中γ对应于模型一中的α.



图表 2 问题 2 求解流程

2.3 问题三的分析

在问题三中，我们能建立与问题二相同的模型，在测线平行的情况下，分析不同的β，发现当测线与等深线平行时能满足题目要求。而当测线与等深线不平行时，由于同一条测线上各点覆盖宽度的不同，想要完全覆盖整个海域不仅难以保证测线长度最短，而且重叠率很大，难以控制在 20%以下。所查阅的资料中的内容也能支撑这个观点[1]。因此我们考虑测线与等深线平行的情况，因为这种情况下所有的测线长度相同，所以只须求最小的测线数量。显然，当重叠率 eta 取最小即 10%时，测线数量取最小。然后我们根据 eta 表达式，可以得到测线之间的间隔距离 d，并据此求得最少的测线数量。

2.4 问题四的分析

问题四是在已有数据上进行处理，可通过将问题简化成第三问中的模型来求解。由于等深线的密集程度相似时，区域中具有相似的坡度。因此我们进行数据的可视化，根据等深线密集程度，将其划分为三个平面。再使用最小二乘法的将各个分割面拟合为有一定倾斜角的平面。接下来通过第三问的最短测线排布方式对各个坡面求解测线排布位置。对于实际的测量面积和重叠情况，先进行坐标变换，将网格数据放入 x 轴与测线平行的坐标系内，通过测线的位置及步长迭代判断，确定测线两端覆盖点位置，得到近似覆盖面积及重叠面积。

三、 模型假设

1.不考虑声波传输时间过程中测量船的位移[2]。

2.不考虑测量船信号接收器在移动过程中由于海水起伏，风浪等不可控因素而导致的上下位移

3.实际复杂水域环境对声速的影响忽略不计

4.声波在传播过程中遇声速界面(跃层) 发生折射导致的信号接受异常问题

步长的影响

四、 符号说明

符号	说明	单位
D_0	海域中心点处的海水深度	m
α	海底坡面的坡度	
η	相邻条带的重叠率	1
β	测线与海底坡面法向量水平投影的夹角	
D_i	测线上 i 点处的海底深度	m
γ	测线在斜坡上的投影与水平面的夹角	
φ	测波覆盖线与水平面的夹角	
W_i	测线上第 i 点的覆盖宽度	m
W_{ia}	第 i 条测线的覆盖宽度的左部分（以竖直线分隔）	m
W_{ib}	第 i 条测线的覆盖宽度的右部分（以竖直线分隔）	m
$l_{投}$	测线在海底坡面投影线	
D_k	点在散点坐标网络的深度	m
A_{ij}	表示原坐标系的点	
B_{ij}	表示坡面坐标系的点	

五、 模型的建立与求解

5. 1 问题一模型的建立与求解

根据题目所问将所求分为两部分，先由题给条件构造多波束测深覆盖宽度的模型，再由此求出相邻测量条带的重叠度。

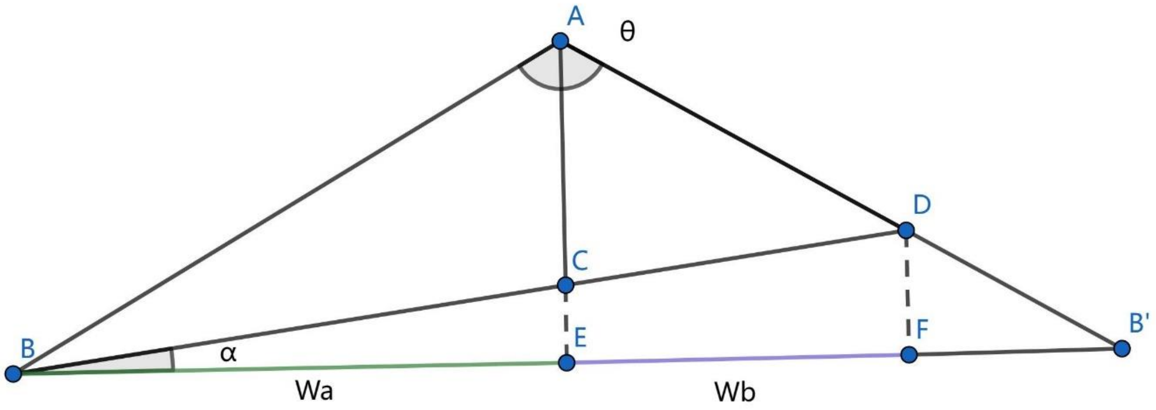
5. 1. 1 求解各点海底深度及其测深覆盖宽度

1.求解各点处海底深度：

由于相邻测线之间相互平行，则由中心点的海底深度和海底坡面角α可求得测线上各点处的海底深度

$$D_i = D_0 - d_i \tan\alpha \tag{1}$$

2.求解各点处测深覆盖宽度



图表 1 正弦定理图示

覆盖宽度指多波束在海底探测边界的**水平距离**，由图 1 知 $\angle B=\frac{\pi}{2}-\frac{\theta}{2}-\alpha, \angle C=\frac{\pi}{2}-\frac{\theta}{2}+\alpha$，对图中三角形使用正弦定理得

$$\begin{aligned}\frac{AC}{\sin B}&=\frac{BC}{\sin\frac{\theta}{2}}\\ \frac{AC}{\sin D}&=\frac{CD}{\sin\frac{\theta}{2}}\end{aligned} \tag{2}$$

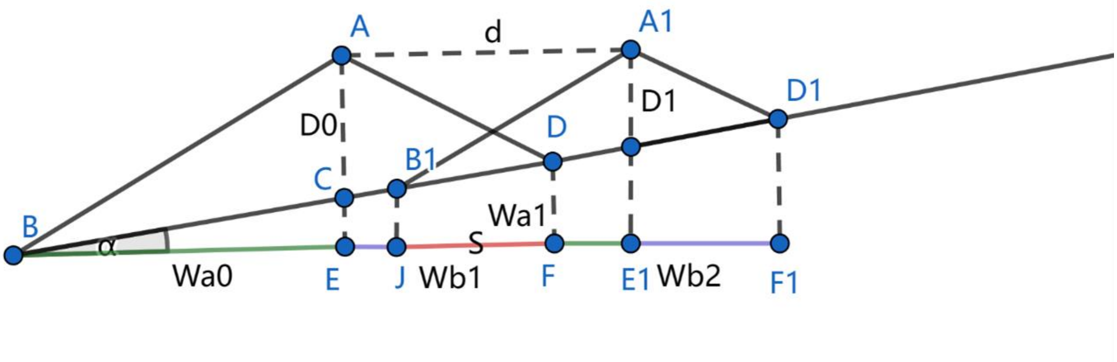
则由几何关系得覆盖宽度为

$$W_i=(BC+CD)\cdot\cos\alpha \tag{3}$$

再由（2），（3）式联立解得覆盖宽度为

$$W_i=D_i\sin\frac{\theta}{2}\cdot\left(\frac{1}{\sin\left(\pi-\frac{\theta}{2}-\alpha\right)}+\frac{1}{\sin\left(\pi-\frac{\theta}{2}+\alpha\right)}\right)\cdot\cos\alpha \tag{4}$$

5.1.2 求解相邻条带的重叠率



图表 2 重叠度求解示意图

由于题目中将覆盖率定义为$\eta = 1 - \dfrac{d}{W}$，而在具有一定坡度的海底坡面中相邻条带的覆盖宽度W并不相等，此时覆盖率的计算公式与平坦时并不完全相同。由原公式推出海底坡面上的重叠率为相邻条带重叠部分宽度的两倍与相邻条带覆盖宽度之和的比值

$$\eta_i = \frac{2\times(W_{ib}+W_{(i+1)a}-d)}{W_i+W_{i+1}} \quad (i\in(1,2,\ldots 9))\ i=5\text{时为中心点} \tag{5}$$

再用此公式推导平坦海底的覆盖率公式，发现与题中所给公式相同，因此可认为该公式是合理的。由此公式即可计算得相邻条带的重叠率。

5.1.3 计算结果即结果解释：

表格 1 问题 1 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	70.00	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/%	——	34.64	30.52	25.84	20.5	14.32	7.1	-1.45	-11.74

结果说明：所求得的覆盖宽度特指多波束在海底探测边界的**水平距离，**而在计算重叠度时，当离中心点较远处，其海水深度浅，导致其覆盖宽度小，相邻条带覆盖宽度和的一半小于测量宽度，使得所求出覆盖度$\eta < 0$，表示出现了漏测现象。

5.2 问题二模型的建立与求解

1 坐标系的建立：

为了便利地表示出图中各直线所成角的大小，对模型建立直角坐标系如图 3 所示，以测量船 A 在海底的投影点为坐标原点，以坡面法向量在水平面的投影方向作为 X 轴，依据右手系原则建立右手坐标系 O-xyz。

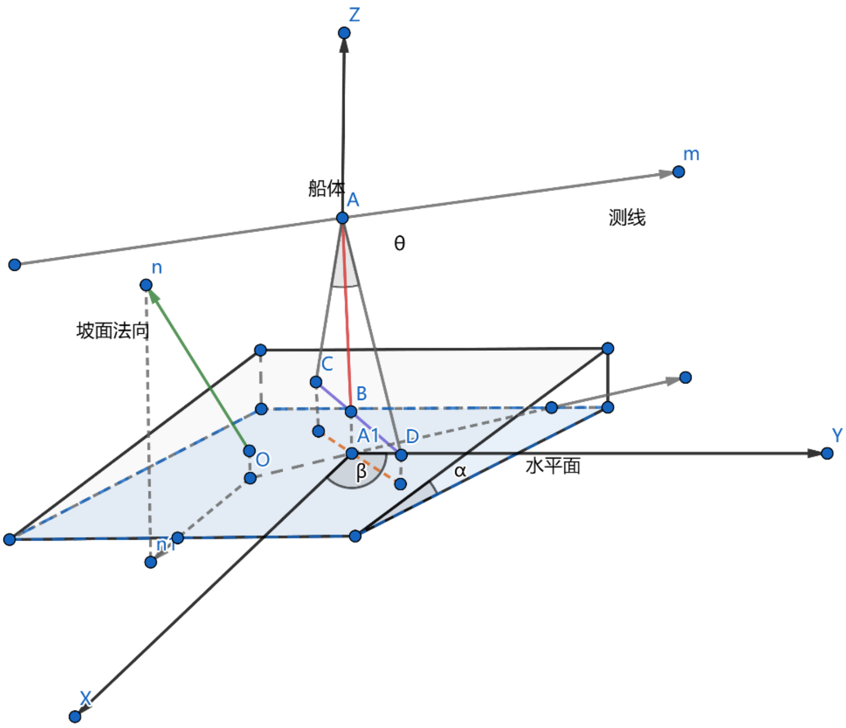


图 3 问题 2 模型示意图

5.2.1 测线在斜坡上的投影与水平面的夹角γ求解

由图 3 可得到海底坡面法向量$\vec{n}$，测线方向$\vec{m}$为

$$\vec{n}=(\sin\alpha,0,\cos\alpha) \qquad \vec{m}=(\cos\beta,\sin\beta,0) \tag{6}$$

由于多波束测面 ACD 与测线$\vec{m}$垂直，又由于$\overrightarrow{CD}$在坡平面上，可得

$$\overrightarrow{CD}\perp\vec{m}\,,\ \overrightarrow{CD}\perp\vec{n} \tag{7}$$

再由向量的叉乘原理可得

$$\overrightarrow{CD}=\vec{n}\times\vec{m} \tag{8}$$

即

$$\overrightarrow{CD}=\begin{vmatrix}\vec{i} & \vec{j} & \vec{k}\\ \sin\alpha & 0 & \cos\alpha\\ \cos\beta & \sin\beta & 0\end{vmatrix}=(-\cos\alpha\sin\beta,\cos\alpha\cos\beta,\sin\alpha\sin\beta) \tag{9}$$

由$\overrightarrow{CD}$即可求出测线在斜坡上的投影与水平面的夹角γ的正切值$\tan\gamma$

$$\tan\gamma=\frac{|CD_z|}{\sqrt{CD^2{}_x+CD^2{}_y}} \tag{10}$$

即

$$\gamma=\arctan(\tan\alpha\sin\beta) \tag{11}$$

5.2.2 测波覆盖线与水平面的夹角φ的求解

由图中易知坡平面的法向量为

$$\vec{k}=(\sin\beta,-\cos\beta,0) \tag{12}$$

由于测线在海底坡面投影线$l_{投}$，在测线所在的垂直于水平面的面上，且位于坡平面上，则可得几何关系

$$\vec{l}_{投}\times\vec{n}=0,\quad \vec{l}_{投}\times\vec{k}=0 \tag{13}$$

由向量的叉乘原理得

$$\overrightarrow{l_{投}}=\vec{n}\times\vec{k} \tag{14}$$

即

$$l_{投}=\vec{n}\times\vec{k}=\begin{vmatrix}\vec{i} & \vec{j} & k\\ \sin\alpha & 0 & \cos\alpha\\ \sin\beta & -\cos\beta & 0\end{vmatrix}=(\cos\alpha\cos\beta,\sin\beta\cos\alpha,-\sin\alpha\cos\beta) \tag{15}$$

由$\vec{l}_{投}$即可求出测线在斜坡上的投影与水平面的夹角φ的正切值$\tan\varphi$

$$\tan\varphi=\frac{|l_{投z}|}{\sqrt{l^2{}_x+l^2{}_y}} \tag{16}$$

即可求得

$$\varphi=\arctan(\tan\alpha\tan\beta) \tag{17}$$

5.2.3 覆盖宽度求解

在求得夹角γ之后即可求得测线上各点处测线深度，由问题一所分析几何关系得

$$D_i=D_0+d_i\tan\varphi=D_0+d_i\cos\beta\tan\alpha \tag{18}$$

得到测线深度D_i且已知测波覆盖线与水平面的夹角φ后，即可类比问题一利用正

弦定理的过程得到覆盖宽度为

$$W_i = D_0 \sin\frac{\varphi}{2}\cos\gamma \cdot \left(\frac{1}{\sin\left(\pi-\frac{\theta}{2}-\gamma\right)}+\frac{1}{\sin\left(\pi-\frac{\theta}{2}+\gamma\right)}\right) \tag{19}$$

5.2.4 求解结果

对题中所给数据进行运算得

表格 2 问题二所求数据

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	415.69	466.09	516.49	566.89	617.29	667.69	718.09	768.48
	45	416.12	451.79	487.47	523.14	558.82	594.49	630.16	665.84
	90	416.55	416.55	416.55	416.55	416.55	416.55	416.55	416.55
	135	416.12	380.45	344.77	309.1	273.42	237.75	202.08	166.4
	180	415.69	365.29	314.89	264.5	214.1	163.7	113.3	62.9
	225	416.12	380.45	344.77	309.1	273.42	237.75	202.08	166.4
	270	416.55	416.55	416.55	416.55	416.55	416.55	416.55	416.55
	315	416.12	451.79	487.47	523.14	558.82	594.49	630.16	665.84

5.3 问题三模型的建立和求解

5.3.1 测线平行于等深线的情况

首先我们从最简单的情况进行讨论，即测线平行于等深线时。我们考虑由于测线之间重叠率对整体覆盖率的影响。在满足重叠率在 10%-20%之间的情况下，对于测线平行于等深线的情况作了分析，我们发现在使之两两重叠率为 10%的三条相邻的测线中，三条测线所覆盖的宽度能达到最大。在三条测线无论如何移动，其重叠率都不再满足大于 10%的情况，故我们得知两两相邻重叠率为 10%的时候其组合的测线能达到最大的覆盖率。

由此，我们考虑前后测线的覆盖率为 10%的情况，通过第一第二问的建模，我们可知覆盖宽度和深度的关系，又由深度和相邻测线的间距之间的关系，可以确立相邻测线间距与覆盖宽度的关系，由η的定义式，我们可以将相邻的测线间距和覆盖宽度及间距联立起来，最终联立得到与η，深度，覆盖宽度和相邻测线间距的表达式。我们只取深度和相邻测线间距关系η，在确定的情况下，我们可以通过已知的测线深度得到最优的测线位置。

第 i 条测线的覆盖宽度以竖直线分隔的左右两部分计算公式分别为：

$$W_{ia}=\frac{D_i\sin\frac{\theta}{2}\cdot\cos\alpha}{\sin B}，W_{ib}=\frac{D_i\sin\frac{\theta}{2}\cdot\cos\alpha}{\sin C} \tag{20}$$

其中 B，C 角度由问题一分析所得$\angle B=\frac{\pi}{2}-\frac{\theta}{2}-\alpha,\angle C=\frac{\pi}{2}-\frac{\theta}{2}+\alpha$

第 i+1 条测线处的深度为：

$$D_{i+1}=D_i-d_{i+1}\tan\alpha \tag{21}$$

其中 Di 为第 i 条测线处的深度，di+1 为第 i+1 条测线与第 i 条测线之间的距离。由此可表示第 i 条测线与第 i+1 条测线覆盖宽度左边部分的比值为：

$$\frac{W_{ia}}{W_{(i+1)a}}=\frac{D_i}{D_{i+1}}=\frac{D_i}{D_i-d_{i+1}\tan\alpha} \tag{22}$$

变化得：

$$W_{(i+1)a}=\frac{D_i-d_{i+1}\tan\alpha}{D_i}\cdot W_{ia}=\left(1-\frac{d_{i+1}\tan\alpha}{D_i}\right)W_{ia} \tag{23}$$

同理，相邻的覆盖宽度具有以下递推关系：

$$W_{i+1}=\left(\frac{D_i-d_{i+1}\tan\alpha}{D_i}\right)W_i \tag{24}$$

故将上述公式代入第（2）问中η的表达式中：

$$\eta=\frac{2\times(W_{ib}+W_{(i+1)a}-d_{i+1})}{W_i+W_{i+1}}=\frac{2\times\left[W_{ib}+\left(1-\frac{d_{i+1}}{D_i}\cdot\tan\alpha\right)\cdot W_{ia}-d_{i+1}\right]}{\left(1-\frac{d_{i+1}}{D_i}\cdot\tan\alpha\right)W_i+W_i} \tag{25}$$

整理得

$$\eta=\frac{\sin\frac{\theta}{2}\cos\alpha\left(\frac{1}{\sin B}+\frac{1}{\sin C}\right)-\left(\tan\alpha\cdot\frac{\sin\frac{\theta}{2}\cdot\cos\alpha}{\sin B}+1\right)\cdot\frac{d_{i+1}}{D_i}}{\sin\frac{\theta}{2}\cos\alpha\cdot\left(\frac{1}{\sin B}+\frac{1}{\sin C}\right)-\frac{\tan\alpha\cdot\sin\frac{\theta}{2}\cdot\cos\alpha\left(\frac{1}{\sin B}+\frac{1}{\sin C}\right)\cdot\frac{d_{i+1}}{D_i}}{2}} \tag{26}$$

将表达式中的参数部分用C_1,C_2,C_3,C_4表示，将我们要求得的比值$\frac{d_{i+1}}{D_i}$用 X 表示

$$C_1=C_3=\sin\frac{\theta}{2}\cos\alpha\left(\frac{1}{\sin B}+\frac{1}{\sin C}\right),\ C_2=-\left(1+\tan\alpha\cdot\frac{\sin\frac{\theta}{2}\cdot\cos\alpha}{\sin B}\right) \tag{27}$$

$$C_4=-\frac{1}{2}\tan\alpha\sin\frac{\theta}{2}\cos\alpha\left(\frac{1}{\sin B}+\frac{1}{\sin C}\right),\ X=\frac{d_{i+1}}{D_i}$$

得到η更简洁的表达式

$$\eta=\frac{C_1+C_2X}{C_3+C_4X} \tag{28}$$

再进一步整理可得到

$$\eta=\frac{C_1}{C_3}+\left(C_2-\frac{C_1\cdot C_4}{C_3}\right)\cdot\left(\frac{C_3}{X}+C_4\right)^{-1} \tag{29}$$

根据上式，我们可以知道η是关于 X 的减函数，由于$X=\frac{d_{i+1}}{D_i}$，此时可以清晰地知道，在确定η的值的情况，我们可以通过某一测线的深度情况，推出下一侧线的位置和深度。

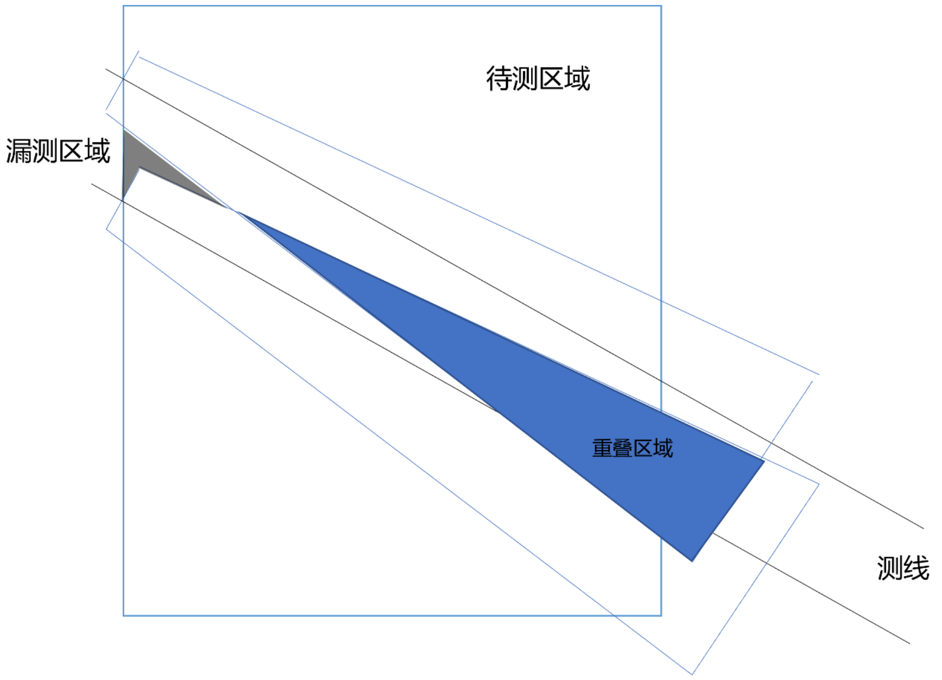
间距 X 为

$$X=\frac{C_3\eta-C1}{C_2-C_4\eta} \tag{30}$$

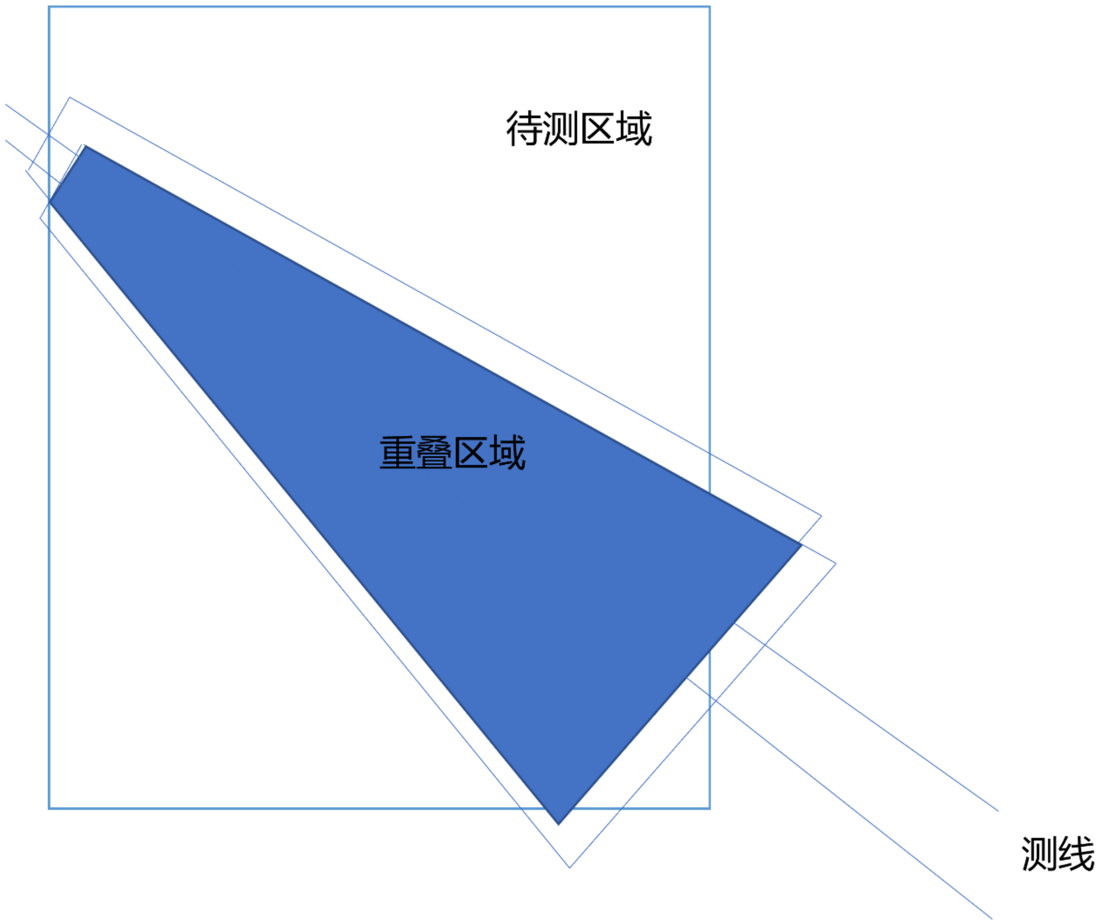
5.3.2 对于测线与等深线不平行的情况：

由文献查阅可得在实际操作中测线要求平行布设。区域内的测线应为彼此平行关系，且走向也要符合航道底部等深线走向这样布设可尽量扩大测线范围，提高覆盖率[3]。

我们注意到在测线方向上的深度分布不均，在水深较浅的位置的覆盖率较低，在水深较深的位置覆盖率较大，在考虑重叠率时，我们要同时保证两个极大极小位置的重叠率满足条件。第二，我们注意到船体航行到矩形海域边界时，由于多波束平面与测线垂直，海域外部的深度未知，船体的一侧无法得到测量数据，造成漏测情况，如图 5 所示。为了补救漏测情况，我们希望用下一测线进行补救，但这一补救方式将导致在浅海处的测线位置过近，导致浅海处的重叠部分过高，如图 6 所示，无法达到重叠率低于 20%的情况。第三，我们注意到在海域的角度位置，若排线不在顶点位置，必然会出现顶点处的小三角漏测的情况。第四，我们计算出在相同中心深度时，平行于等深线排布的方式的平均单位面积覆盖率要高于不平行排布的情况，因此我们认为在限制条件，操作难易程度，侧线排布的优良对比，我们得出了平行于等深线排布的方式更优。



图表 4 当测线距离较远时导致漏测区域过大



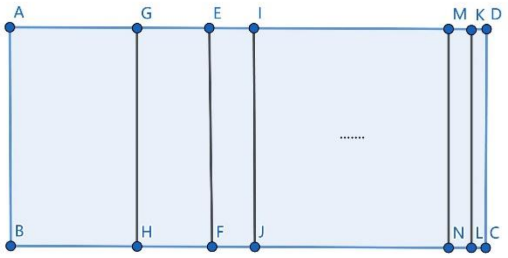
图表 5 当测线距离过近时导致重叠过大情况

5.3.3 结果求解

由式(30)通过编程求得即可得到 34 条平行测线的间距，如表(2)所示,测线布设如图(7)所示

表格 3 测线间距(自左向右，由上到下为顺序)

358.52	591.91	545.48	502.69	463.26	426.93	393.44	362.58
334.14	307.93	283.78	261.52	241	222.1	204.68	188.63
173.83	160.19	147.63	136.05	125.38	115.54	106.48	98.13
90.43	83.34	76.8	70.78	65.23	60.11	55.39	51.05
47.05	43.36						

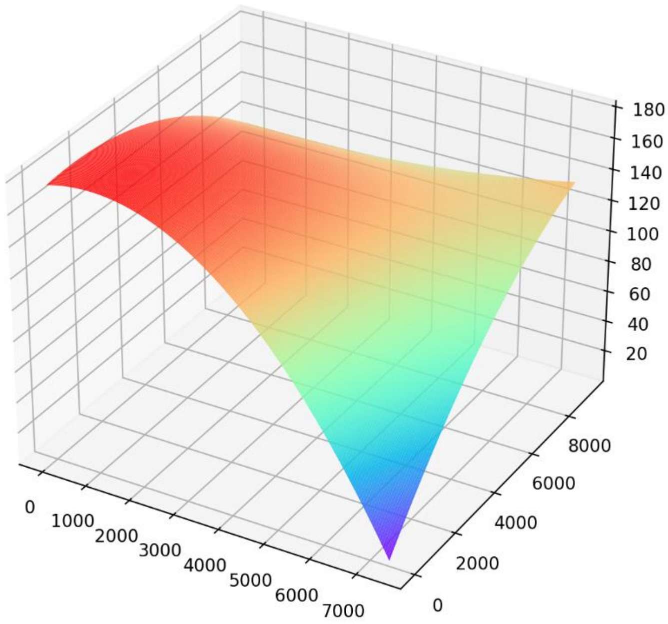


图表 6 测线布设示意图

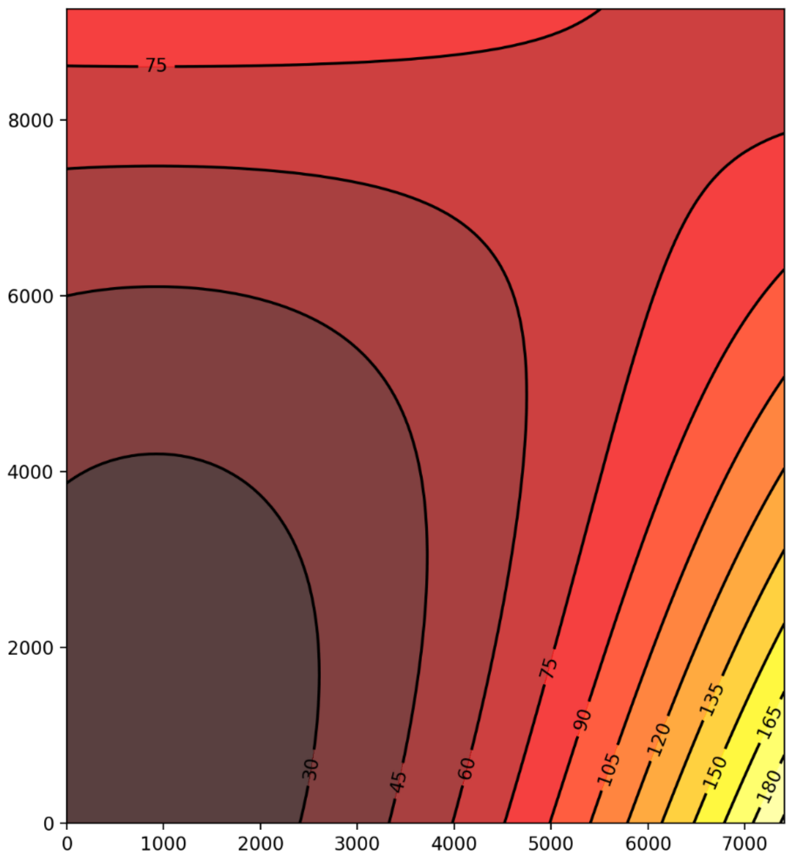
5.4 模型四的建立

5.4.1 数据的预处理

数据的预处理，我们先对网格图可视化，得到海域的等高线位置，根据等高线的密集程度，我们可以将其划分为三个矩形区域。



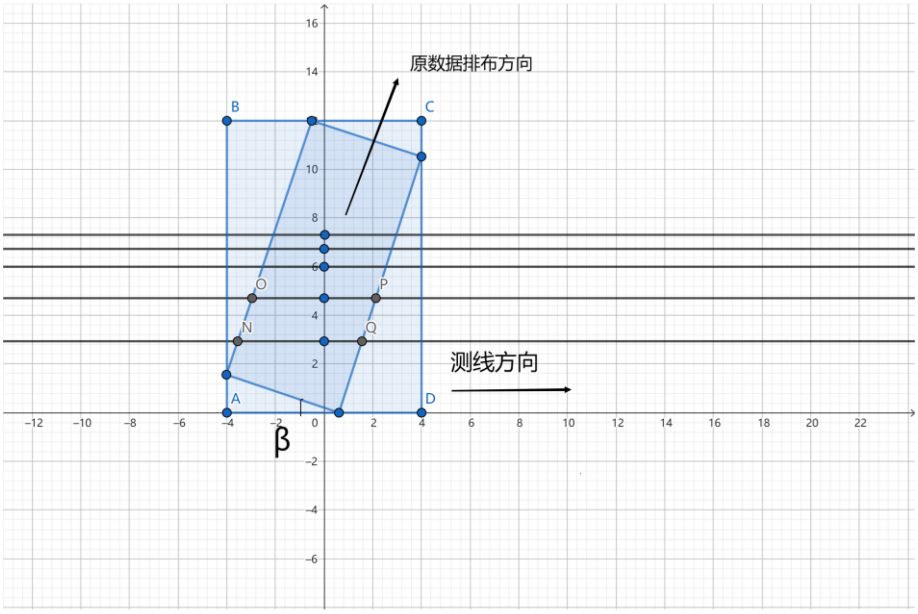
图表 **7** 单波束测深数据三维可视化图



图表 **8** 单波束测深数据等深线图

5.4.2 最短总测线长度

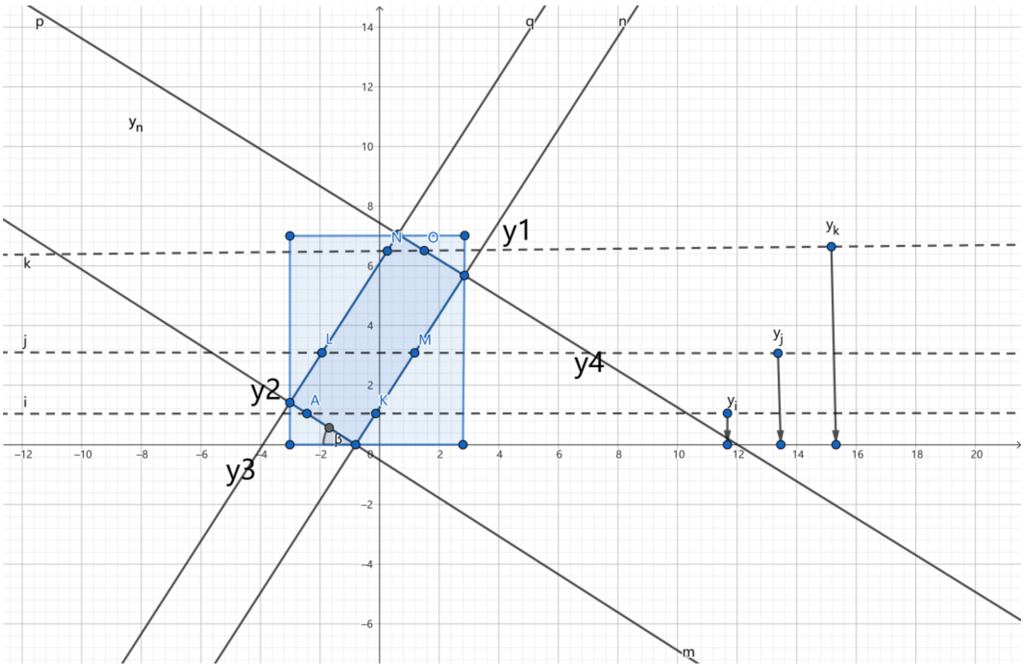
在矩形区域中，我们进行最小二乘法函数拟合，分别得到一个具有坡度的平面，我们坡面最深位置及坡度角使用第三问模型进行计算，我们建立直角坐标系，将测线平行排布在 y 轴正方向上



图表 8 测线平行分布情况

得到最佳位置后，我们要求解其测线布置长度，设y_n为第 n 条测线在 y 轴上距离最深位置的长度，如图所示y_i, y_j, y_k，测线长度为C_n

如图，通过几何关系



图表 9 测线与最深点距离位置图

由几何关系可知，测线长度分作三部分进行计算

$$y_n < L_1 \tan\beta,\ c_n = y_n(\cot\beta + \tan\beta) \tag{31}$$

$$L_1 \tan\beta < y_n < L_2 \sin\beta \quad ,c_n = L_1 \tan\beta \tag{32}$$

$$L_2 \sin\beta < y_n < L_1 \sin\beta + L_2 \cos\beta, \tag{33}$$

$$c_n = (L_1 \sin\beta + L_2 \cos\beta - y_n) \cdot (\cot\beta + \tan\beta) \tag{34}$$

将三部分求和得到一图的测线长度，最终三个分图结合可得到最后的测线长度和为 371481.81m。

5.4.3 坐标变换和深度求解

我们注意到题目所给出的深度数据是按照矩形区域边长平行的方向以相同步进排布的，在考虑转化后的坡度平面时，我们发现其最佳的排布方向和原矩形成一定角度，我们不妨设为β，

我们考虑将原矩形数据进行坐标变换，得到与测线方向平行和垂直方向有相同步进长度的深度数据。

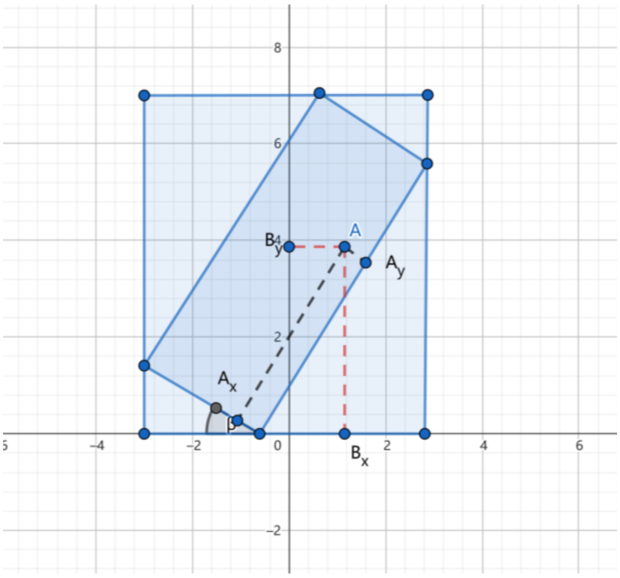
我们做如下的字母定义：将 a 记作新坐标系下的步进长度，将点所在位置的深度记作D_{ij}，将原坐标系的点记作 A，将变换后的点记作 B。定义矩形区域的长为L_1,宽为L_2。

做如下数据的坐标变换

$$A(x,y,D) \rightarrow B(x',y',D) \tag{35}$$

考虑原坐标系的深度数据，我们用以下方式计算原坐标和新坐标的关系。

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



图表 10 坐标点转换可视化图

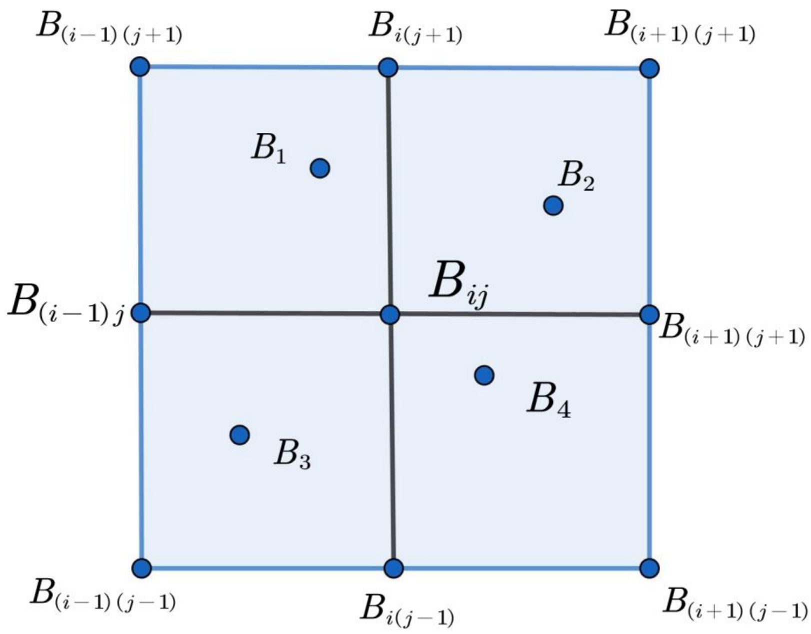
我们为了反应新坐标系下的深度情况，我们考虑点附近的深度数据求取平均值，

$$D_{ij}=\frac{1}{n_{ij}}\sum_{k=1}^{n_{ij}}D_k=\frac{1}{n}\sum_{k=1}^{n}D_k\ , \tag{36}$$

D_k 为在方格 $(i-1)a\le x\le(i+1)a,(j-1)a\le y\le(j+1)a$ 中点的深度

为在方格中的点的深度，n_{ij} 为方格中点的个数，则得到坡面深度分布数据 $A_{ij}=(ia,ja,D_{ij})$

方格 $(i-1)a\le x\le(i+1)a,(j-1)a\le y\le(j+1)a$ 中的点的深度，n_{ij} 为方格中点的个数，则得到坡面深度分布数据 $A_{ij}=(ia,ja,D_{ij})$



图表 11 深度的计算可视化图

通过上述方法，我们能得到关于新坐标系下的深度数据，通过对于测线的求解，我们容易知第 n 道测线的方程为：

$$y_n = \sum_{k=1}^{n} d_k \tag{37}$$

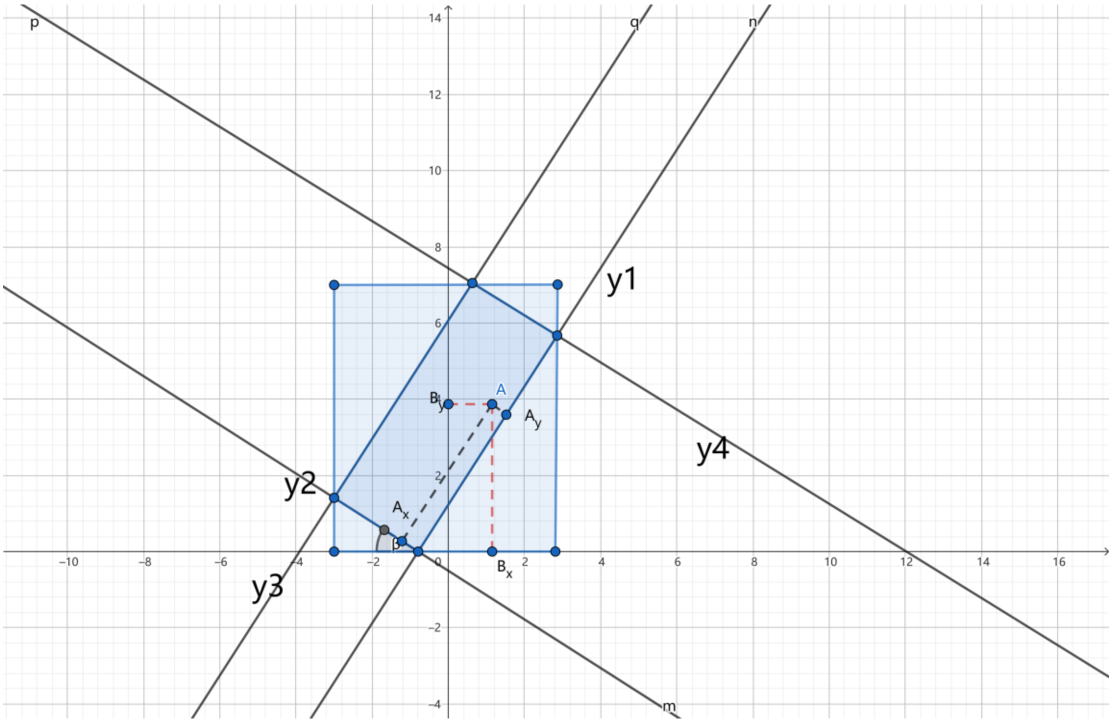
同时我们能求得原坡面的在新坐标下的边界为

$$y'_1 = \tan\left(\frac{\pi}{2} - \beta\right) \cdot x' = \cot\beta \cdot x' \tag{38}$$

$$y'_2 = -\tan\beta \cdot x' \tag{39}$$

$$y'_3 = \cot\beta \cdot x' + \frac{L_1}{\sin\beta} \tag{40}$$

$$y_4{}' = -\tan\beta \cdot x' + \frac{L_2}{\cos\beta} \tag{41}$$



图表 9 边界位置图

5.4.4 相邻测线覆盖情况及漏测面积求解

接下来我们着手于计算覆盖及漏测情况，由于我们所得到的数据是离散的点，我们考虑测线的步进位置与相邻测线的覆盖情况，我们可以求出新坐标系下关于步进长度的深度数据。

尝试去求解测线的覆盖的宽度，通过联立边界和测线方程，我们能得到测线的限制范围

在实际的深度数据上，我们可以通过几何关系，得到在步进范围上大致的覆盖范围，可得

$$B_{ik}=\sum_{k=1}^{n}d_k-k_a=y_i-k_a \tag{42}$$

$$if\ \frac{D_{i(k+1)}}{OB_{i(k+1)}}>\tan 30^{\circ} \tag{43}$$

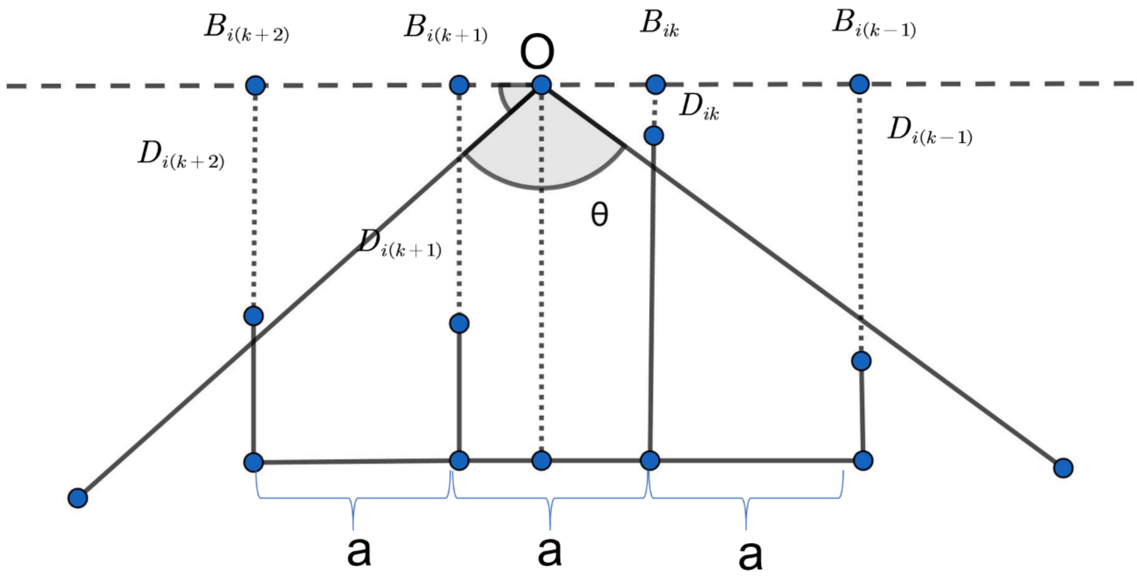
则说明波束没有接触海底，继续通过步进递推得

$$\frac{D_{i(k+2)}}{OB_{i(k+1)}+a}\to\frac{D_{i(k+3)}}{OB_{i(k+1)}+2a} \tag{44}$$

直到

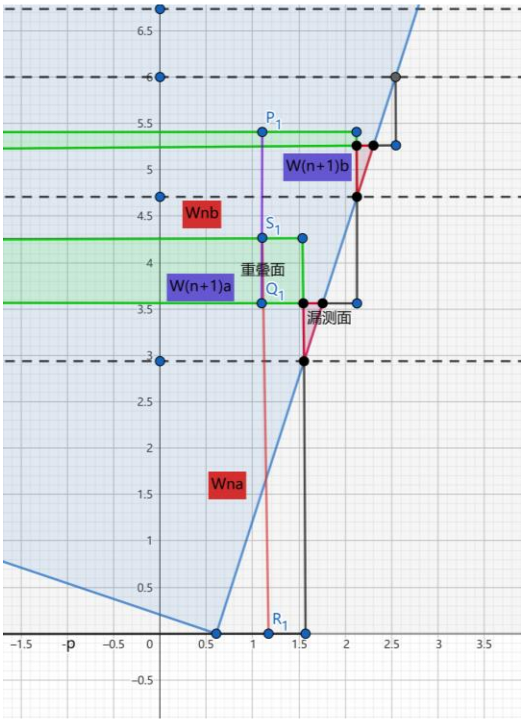
$$if\ \frac{D_{i(k+j)}}{OB_{i(k+j)}} \leq \tan 30° \tag{45}$$

说明波束到达底面，覆盖宽度即为 $OB_{i(k+j)}$



图表 10

我们可以计算出每一条测线在步进上的点的覆盖范围和最终覆盖点的位置，我们将其保留为每一条测线的数据。得到所有测线以步进为长度的各个测量位置，将第 n 条测线的两端的覆盖最远端的点分别为Wa_n，Wb_n



图表 11

在相邻的两条测线中，我们可以以步进为间距，将两条测线以步进为间距的覆盖长度进行计算

可以分别计算出其覆盖率为：

由于我们仅得到离散的覆盖率为 20%的点，由于测线长度是根据步进长度进行划分，因此覆盖率为 20%的长度近似为步进和覆盖率超过 20%的点的数量的乘积。

即：计算出一点 η >20%时,超过 20%的测线长度增加 a。

计算得重叠率超过 20%部分的总长度为：L=12407.49m

同理可得，漏测面积可以通过计算相邻测线相同步进位置的点之间，其覆盖率为 0 时，计算两点之间未覆盖的长度即是两测线之间间距减去两测线在该点的覆盖长度 R 为：$R=d_{n+1}-Wa_{ni+1}-Wb_{ni}$

由编程计算得漏测面积的百分比为r=7.68%

六、 模型的评价、改进与推广

6.1 模型的优点

1.在分析复杂空间解构的几何关系时，通过空间直角坐标系的建立和向量叉乘准确且便捷地表示出关键的角度关系。

2．在解决第三问布线设置时，关于测线是否平行于等深线的问题，在查阅文献所得结论之外，模型中还对该情况对题目所要求的覆盖度范围和漏测情况进行比对，更

科学地论证得所得到的模型结论

6.2 模型的缺点

一：第三问对于非平行于等高线情况的考虑，没有严格证明该情况要劣于平行等高线的情况。

二：对于第四问的模型建立，处理数据是依据等高线的密集程度进行切割，具有一定主观性。会导致所求解可能不是最优解。使用平面拟合的模型，不能保证其测线的长度最短。

6.3 模型的改进

问题三中可根据海域中各点处覆盖宽度与 beta 和位置的关系建立更加全面的模型对结果进行严格的证明。问题四中可以应用微元法将曲面分割成若干个极小的曲面，这些曲面可类似为问题三中的平面，利用这些平面建立模型求解可以得到更加精确的答案。但由于时间限制，我们未能实现这些更加复杂的模型。

6.4 模型的推广

我们可以通过该模型解决类似的测绘地理问题，应用该算法可以更好的拟合不同地理条件的坡度模型，再结合实际情况多波束的照射方向和角度，可以确定更灵活的测量方法，从而得到更加精准的测线排布方案。

七、 参考文献

[1]焦安龙,邹永刚.多波束测量海区技术设计内容研究[J].海洋测绘,2015,35(03):43-45+49.

[2] 曲萌. 多波束测深质量后评估方法研究[D]. 山东科技大学,2021.DOI:10.27275/d.cnki.gsdku.2019.000453.

[3] 张旭,叶小心,洪德玫.多波束系统在长江航道测量中的测线布设方法研究[J].中国水运.航道科技,2017(01):52-55.DOI:10.19412/j.cnki.42-1395/u.2017.01.011.

八、 第一问代码

```
import numpy as np
import pandas as pd
theta = 2*np.pi/3
alpha = np.pi/120

#计算覆盖宽度W
C = np.pi/2 - theta/2 + alpha
B = np.pi/2 - theta/2 - alpha
dis = np.linspace(-800, 800, 9)      #距离中心点不同的距离
D = 70 - dis*np.tan(alpha)       #距离中心点不同的距离对应的深度
W = (np.sin(theta/2)/np.sin(C) + np.sin(theta/2)/np.sin(B))*D*np.cos(alpha)
rate = np.sin(C)/np.sin(B)       #W 左比右的比例
W_left = W*(rate/(1 + rate))
W_right = W*(1/(1 + rate))

#计算重叠率eta
d = 200      #相邻侧线的间距
eta = 2*(W_right[:8] + W_left[1:9] - d)/(W[0:8] + W[1:9])*100  #前n-1 条线分别与各自后一条线的重叠率(百分比数)
print("海水深度：", D)
print("覆盖宽度：", W)
print("重叠率：", eta)

#将数据导入到excel 中
D = pd.DataFrame(D)
W = pd.DataFrame(W)
eta = pd.DataFrame(eta)
eta.to_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\中转.xlsx", float_format='%.2f', sheet_name='Sheet2')
```

8.1 第二问代码

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

theta = 2*np.pi/3
alpha = np.pi/120

"""计算不同 β 和距离下的覆盖宽度 W"""
dis = np.linspace(0, 2.1, 8)        #测量船距海域中心的距离（单位：海里）
dis = dis*1852      #将单位海里转化为米
beta = np.linspace(0, np.pi*7/4, 8)
tan_gama = np.tan(alpha)*np.cos(beta)       #测线在斜坡上的投影与水平面的夹角
phi = np.arctan(abs(np.tan(alpha)*np.sin(beta)))     #波的方向在斜坡上的投影与水平面的夹角
data = np.zeros((8, 8))
#计算各个 beta 下不同距离的 W
for i in range(len(tan_gama)):
    D = 120 + dis*tan_gama[i]
    C = np.pi / 2 - theta / 2 + phi[i]
    B = np.pi / 2 - theta / 2 - phi[i]
    W = (np.sin(theta / 2) / np.sin(C) + np.sin(theta / 2) / np.sin(B)) * D * np.cos(fai[i])
    data[i] = W
    print(W)

#将数据保存到 excel 中
data = pd.DataFrame(data)
data.to_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\中转.xlsx", float_format='%.2f', sheet_name='Sheet2')
```

8.2 第三问代码

```
import numpy as np
import pandas as pd
alpha = np.pi/120        #坡度
theta = np.pi*2/3
C = np.pi/2 - theta/2 + alpha
B = np.pi/2 - theta/2 - alpha
length = 4*1852
width = 2*1852

D_max = 110 + 2*1852*np.tan(alpha)      #海域最西边界处的深度
print("D_max:", D_max)
d1 = D_max*np.sin(theta/2)*np.cos(alpha)/(np.sin(B) + np.sin(theta/2)*np.sin(alpha))    #第一条测线距离海域最西边界的距离
D1 = D_max - d1*np.tan(alpha)       #第一条测线处的深度
D = [D1]        #创建存放各条测线处的深度的列表
d = [d1]     #创建存放各条测线与上一条测线的距离的列表

C1 = np.sin(theta/2)*np.cos(alpha)*(1/np.sin(B) + 1/np.sin(C))
C2 = -(np.tan(alpha)*np.sin(theta/2)*np.cos(alpha)/np.sin(B) + 1)
C3 = C1
C4 = -1/2*np.tan(alpha)*np.sin(theta/2)*np.cos(alpha)*(1/np.sin(B) + 1/np.sin(C))
eta = 0.1
k = (C1 - C3*eta)/(C4*eta - C2)      #d_i+1 与 D_i 的比值

#计算每一条测线对应的深度和与上一条测线的距离
d_sum = d1       #存储目前东西方向的长度，如果超出 length，则退出循环
for i in range(100):
    d.append(k*D[i])
    D.append(D[i] - d[i+1]*np.tan(alpha))
    d_sum += d[i+1]
    if d_sum >= length:
        break
D.pop()
d.pop()

#计算每一条测线处的覆盖宽度
D = np.array(D)
W = (np.sin(theta/2)/np.sin(C) + np.sin(theta/2)/np.sin(B))*D*np.cos(alpha)
cum_d = np.cumsum(d)
print(len(d))
```

```
#计算重叠率 eta
rate = np.sin(C)/np.sin(B)        #W 左比右的比例
W_left = W*(rate/(1 + rate))
W_right = W*(1/(1 + rate))
n = len(W)
eta = 2*(W_right[:n-1] + W_left[1:n] - d[1:n])/(W[0:n-1] + W[1:n])*100      #前 n-1 条线分别与各自后一条线的重叠率(百分比数)
line_sum = len(d)*width
print("d:", d)
print(line_sum)
d = pd.DataFrame(d)
d.to_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\中转.xlsx", float_format='%.2f')
```

8. 3 第四问代码

```
#将附件中的测量结果可视化
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import pandas as pd
import numpy as np


x = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", header=None, sheet_name="Sheet1")
x = np.array(x).flatten()*1852
y = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", sheet_name="Sheet2", header=None)
y = np.array(y).flatten()*1852
z = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", sheet_name="Sheet3", header=None)
Z = 200 - np.array(z)

fig = plt.figure(figsize=(7, 8))
X, Y = np.meshgrid(x, y)

#plt.contourf(X, Y, Z, 16, alpha=0.75, cmap=plt.cm.hot)
#C = plt.contour(X, Y, Z, 16, colors='black', linewidth=8)
#plt.clabel(C, inline=True, fontsize=10)

ax = plt.axes(projection="3d")
ax.plot_surface(X, Y, Z, alpha=0.9, cstride=1, rstride=1, cmap='rainbow')
```

```python

plt.savefig(fname='3d', dpi=200)
plt.show()
```

计算测线长度

```python
#计算总测线长度
import pandas as pd
import numpy as np
from scipy import optimize


x = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", header=None, sheet_name="Sheet1")
x = np.array(x).flatten()*1852
y = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", sheet_name="Sheet2", header=None)
y = np.array(y).flatten()*1852
z = pd.read_excel(r"C:\Users\pj\Desktop\CUMCM2023Problems\CUMCM2023Problems\B 题\坐标.xlsx", sheet_name="Sheet3", header=None)
z = np.array(z)

#将原海域划分为三个部分
z1 = z[:215, :123]
x1 = x[:123]
y1 = y[:215]

z2 = z[216:, :]
x2 = x
y2 = y[216:]

z3 = z[:215, 124:]
x3 = x[124:]
y3 = y[:215]

#拟合三个平面
"""传入 x，y，z，返回组合后的点"""
def get_points(x, y, z):
    Z = z.flatten()
    Y = y
    m, n = z.shape
    for i in range(n - 1):
        Y = np.vstack([Y, y])
    Y = Y.T.flatten()
    X = x
    for i in range(m - 1):
```

```python
        X = np.vstack([X, x])
    X = X.flatten()
    points = np.vstack([X, Y, Z]).T
    return points

points1 = get_points(x1, y1, z1)
points2 = get_points(x2, y2, z2)
points3 = get_points(x3, y3, z3)

"""定义误差函数"""
def err(para, points):
    a0, a1, a2 = para
    return a0*points[:, 0] + a1 * points[:, 1] + a2 - points[:, 2]


plsq1 = optimize.leastsq(err, [1, 1, 1], points1)
para1 = plsq1[0]
plsq2 = optimize.leastsq(err, [1, 1, 1], points2)
para2 = plsq2[0]
plsq3 = optimize.leastsq(err, [1, 1, 1], points3)
para3 = plsq3[0]
print("para1:", para1)
print("para2:", para2)
print("para3:", para3)

#计算三个平面的坡度
cos_alpha1 = 1/((para1[0]**2 + para1[1]**2 + 1)**0.5)
alpha1 = np.arccos(cos_alpha1)
cos_alpha2 = 1/((para2[0]**2 + para2[1]**2 + 1)**0.5)
alpha2 = np.arccos(cos_alpha2)
cos_alpha3 = 1/((para3[0]**2 + para3[1]**2 + 1)**0.5)
alpha3 = np.arccos(cos_alpha3)

def compute_line(alpha, length, D_max):
    theta = np.pi * 2 / 3
    C = np.pi / 2 - theta / 2 + alpha
    B = np.pi / 2 - theta / 2 - alpha

    d1 = D_max * np.sin(theta / 2) * np.cos(alpha) / (np.sin(B) + np.sin(theta / 2) * np.sin(alpha))
    D1 = D_max - d1 * np.tan(alpha)
    D = [D1]  # 创建存放各条测线处的深度的列表
    d = [d1]  # 创建存放各条测线与上一条测线的距离的列表
```

```python

    C1 = np.sin(theta / 2) * np.cos(alpha) * (1 / np.sin(B) + 1 / np.sin(C))
    C2 = -(np.tan(alpha) * np.sin(theta / 2) * np.cos(alpha) / np.sin(B) + 1)
    C3 = C1
    C4 = -1 / 2 * np.tan(alpha) * np.sin(theta / 2) * np.cos(alpha) * (1 / np.sin(B) + 1 / np.sin(C))
    eta = 0.1
    k = (C1 - C3 * eta) / (C4 * eta - C2)  # d_i+1 与 D_i 的比值

    # 计算每一条测线对应的深度和与上一条测线的距离
    d_sum = d1   #存储目前垂直于测线方向的长度，如果超过 length，则退出循环
    for i in range(100):
        d.append(k * D[i])
        D.append(D[i] - d[i + 1] * np.tan(alpha))
        d_sum += d[i + 1]
        if d_sum >= length:
            break
    D.pop()
    D = np.array(D)
    d.pop()
    d = np.array(d)
    return D, d

cos_x1 = abs(para1[0]/((para1[0]**2 + para1[1]**2)**0.5))
beta1 = np.pi/2 - np.arccos(cos_x1)     #测线方向与 x 轴的夹角
cos_x2 = abs(para2[0]/((para2[0]**2 + para2[1]**2)**0.5))
beta2 = np.pi/2 - np.arccos(cos_x2)     #测线方向与 x 轴的夹角
cos_x3 = abs(para3[0]/((para3[0]**2 + para3[1]**2)**0.5))
beta3 = np.pi/2 - np.arccos(cos_x3)     #测线方向与 x 轴的夹角

length1 = (x1[-1] - x1[0])*np.sin(beta1) + (y1[-1] - y1[0])*np.cos(beta1)
length2 = (x2[-1] - x2[0])*np.sin(beta2) + (y2[-1] - y2[0])*np.cos(beta2)
length3 = (x3[-1] - x3[0])*np.sin(beta3) + (y3[-1] - y3[0])*np.cos(beta3)

D1, d1 = compute_line(alpha1, length1, 75)
D2, d2 = compute_line(alpha2, length2, 84.4)
D3, d3 = compute_line(alpha3, length3, 197.2)
```

```python

d_1min = min([(x1[-1] - x1[0])*np.sin(beta1), (y1[-1] - y1[0])*np.cos(beta1)])
d_1max = max([(x1[-1] - x1[0])*np.sin(beta1), (y1[-1] - y1[0])*np.cos(beta1)])
d_2min = min([(x2[-1] - x2[0])*np.sin(beta2), (y2[-1] - y2[0])*np.cos(beta2)])
d_2max = max([(x2[-1] - x2[0])*np.sin(beta2), (y2[-1] - y2[0])*np.cos(beta2)])
d_3min = min([(x3[-1] - x3[0])*np.sin(beta3), (y3[-1] - y3[0])*np.cos(beta3)])
d_3max = max([(x3[-1] - x3[0])*np.sin(beta3), (y3[-1] - y3[0])*np.cos(beta3)])

#计算三个平面内各自的测线总长
def compute_line_sum(d, d_min, d_max, beta, x, y, length):
    cum_d = np.cumsum(d)
    line_sum = 0
    for i in range(len(cum_d)):
        if cum_d[i] < d_min:
            line_sum += cum_d[i]/np.tan(beta) + cum_d[i]*np.tan(beta)
        elif d_min <= cum_d[i] <= d_max:
            if x[-1] - x[0] >= y[-1] - y[0]:
                line_sum += (y[-1] - y[0])/np.sin(beta)
            else:
                line_sum += (x[-1] - x[0])/np.cos(beta)
        else:
            line_sum += (length - cum_d[i])*(1/np.tan(beta) + np.tan(beta))
    return line_sum

line_sum1 = compute_line_sum(d1, d_1min, d_1max, beta1, x1, y1, length1)
line_sum2 = compute_line_sum(d2, d_2min, d_2max, beta2, x2, y2, length2)
line_sum3 = compute_line_sum(d3, d_3min, d_3max, beta3, x3, y3, length3)
print("line_sum1:", line_sum1)
print("line_sum2:", line_sum2)
print("line_sum3:", line_sum3)

```