

南京工业大学
《测量程序设计》
上机指导书

束蝉方 编

南京工业大学测绘学院

二 0 一 二

目 录

上机守则	1
上机实验注意事项	2
实验一：文件数据格式化读写	5
实验二：坐标正反算	8
实验三：大地坐标与空间直角坐标的转换	10
实验四：子午线弧长正反算	13
实验五：高斯投影正反算	16
实验六：高斯投影邻带换算	19
实验七：平面坐标转换	20
实验八：空间直角坐标转换	23
实验九：GPS 水准高程拟合	27

上机守则

1. 学生必须按指导教师安排的上机实验时间进入机房上机，未经许可，不得带外人进入机房。
2. 进入机房时必须穿上鞋套，否则不得进入机房。
3. 认真填写上机情况登记表，若遇计算机有异常情况，应先向老师汇报，不得擅自处理。
4. 遵守计算机操作规程，即开机时先开显示器再开主机；结束时须关闭计算机，关机时先通过 Windows 功能关闭系统，主机电源指示灯灭了以后再关闭显示器。
5. 禁止上机时玩游戏或从事与上机实验无关的内容。
6. 保持机房安静和整洁，不得携带食品、饮料进入机房，严禁随地吐痰、乱扔垃圾或杂物，禁止吸烟、大声喧哗或闲聊。
7. 爱护机房设施，严禁更改设置参数、添加口令或删除非本人文件。对于导致计算机不能正常工作、影响他人上机者，将取消其上机资格。
8. 严禁私自拆卸配件或将室内物品带出室外。一经发现，除要求按价赔偿外，将通报批评和取消其上机资格，情节严重者交有关行政部门和司法部门处理。

上机实验注意事项

1、上机实验前必须阅读有关教材及本上机实验指导书，初步了解上机实验内容要求与步骤。

2、严格遵守上机守则，严禁大声喧哗或玩游戏。

3、认真填上机实验记录，不可潦草，填写内容包括上机实验时间、地点、实验主要内容、步骤及完成成果说明等。

4、独立完成上机实验内容，实事求是，严禁抄袭他人作业。若有弄虚作假现象，该课程成绩按不及格处理。

5、实验结束时，应向指导教师提交上机实验报告和有关成果，符合要求并经允许，方可关机和离开机房。

6、按时上机，不准迟到、早退、旷课，有事须预先履行请假手续，征得指导教师同意后方可离开。

上机实验一、文件数据格式化读写

一、实验目的

通过编辑和运行简单的 matlab 数据读写程序,熟悉和掌握数据格式化输入输出方法。

二、实验内容:

1、文件的打开与关闭

1) 创建标准的对话框并通过交互式操作取得文件名

```
[FileName, PathName, FilterIndex]=uigetfile(FilterSpec, DialogTitle, DefaultName)
```

FileName: 返回的文件名

PathName: 返回的文件的完整路径名

FilterIndex: 选择的文件类型

FilterSpec: 文件类型设置

DialogTitle: 打开对话框的标题

DefaultName: 默认指向的文件名

例:

a. 最简单的形式, 只设置了一种可选择文件类型

```
[FileName, PathName] = uigetfile('*.m', 'Select the M-file');
```

b. 设置多种文件类型

```
[filename, pathname] = uigetfile({'*.txt'; '*.xls'; '*.mat'; '*..*'}, 'File Selector');
```

c. 所有文件类型

```
[FileName, PathName] = uigetfile('*. *');
```

d. 路径文件名合并

```
file = fullfile(pathname, filename)
```

或者 file = [pathname, filename]

2) 打开文件

在读写文件之前，必须先用 `fopen` 函数打开或创建文件，并指定对该文件进行的操作方式。`fopen` 函数的调用格式为：

`fid=fopen(文件名, '打开方式')`

说明：其中 `fid` 用于存储文件句柄值，如果返回的句柄值大于 0，则说明文件打开成功。文件名用字符串形式，表示待打开的数据文件。常见的打开方式如下：

‘r’：只读方式打开文件（默认的方式），该文件必须已存在。

‘r+’：读写方式打开文件，打开后先读后写。该文件必须已存在。

‘w’：打开后写入数据。该文件已存在则更新；不存在则创建。

‘w+’：读写方式打开文件。先读后写。该文件已存在则更新；不存在则创建。

‘a’：在打开的文件末端添加数据。文件不存在则创建。

‘a+’：打开文件后，先读入数据再添加数据。文件不存在则创建。

另外，在这些字符串后添加一个“t”，如‘rt’或‘wt+’，则将该文件以文本方式打开；如果添加的是“b”，则以二进制格式打开，这也是 `fopen` 函数默认的打开方式。

例：

```
fid = fopen(file,'rt');
```

2) 关闭文件

文件在进行完读、写等操作后，应及时关闭，以免数据丢失。关闭文件用 `fclose` 函数，调用格式为：

```
sta = fclose(fid)
```

说明：该函数关闭 `fid` 所表示的文件。`sta` 表示关闭文件操作的返回代码，若关闭成功，返回 0，否则返回-1。如果要关闭所有已打开的文件用 `fclose('all')`。

2、文本文件的读写操作

1) 读文本文件

`fscanf` 函数可以读取文本文件的内容，并按指定格式存入矩阵。其调用格式为：

```
[A, COUNT]=fscanf(fid, format, size)
```

说明：其中 `A` 用来存放读取的数据，`COUNT` 返回所读取的数据元素个数，`fid` 为文件句柄，`format` 用来控制读取的数据格式，由%加上格式符组成，常见的格式符有：`d`（整型）、`f`（浮点型）、`s`（字符串型）、`c`（字符型）等，在%与格式符之间还可以插入附加格式说

明符，如数据宽度说明等。`size` 为可选项，决定矩阵 **A** 中数据的排列形式，它可以取下列值：`N`（读取 `N` 个元素到一个列向量）、`inf`（读取整个文件）、`[M, N]`（读数据到 `M` × `N` 的矩阵中，数据按列存放）。

2) 写文本文件

`fprintf` 函数可以将数据按指定格式写入到文本文件中。其调用格式为：

`fprintf(fid, format, A)`

说明：`fid` 为文件句柄，指定要写入数据的文件，`format` 是用来控制所写数据格式的格式符，与 `fscanf` 函数相同，`A` 是用来存放数据的矩阵。

例 文件数据导入

`test1.txt` 的文件中以下为文件内容：

你好 欢迎来到 南京工业大学 www.njut.edu.cn

1 11 111 1111

2 22 222 2222

3 33 333 3333

4 44 444 4444

5 55 555 5555

方法：

```
[filename, pathname]= uigetfile('*.*');
```

```
file=fullfile(pathname, filename);
```

```
A=importdata(file);
```

```
A. data=
```

```
1 11 111 1111
```

```
2 22 222 2222
```

```
3 33 333 3333
```

```
4 44 444 4444
```

```
5 55 555 5555
```

```
A.textdata=
```

```
'你好' '欢迎来到' '南京工业大学' 'www.njut.edu.cn'
```

`test2.txt` 的文件中以下为文件内容：

你好

1 11 111 1111

欢迎来到

2 22 222 2222

南京工业大学

3 33 333 3333

www.njut.edu.cn

4 44 444 4444

5 55 555 5555

方法:

```
[filename, pathname]= uigetfile('*.*');      %文件查找窗口
file=fullfile(pathname, filename);         %合并路径文件名
fidin=fopen(file,'rt');                    % 打开文件
fidout=fopen('dataout.txt','w');          % 创建输出文件
while ~feof(fidin)                        % 判断是否为文件末尾
    tline=fgetl(fidin);                    % 从文件读行
    if ~isempty(tline)                    % 判断是否空行
        [m,n]=size(tline);
        flag=1;
        for i=1:n                          %判断一行中有没有字符（-.+和空格键除外）
            if ~(tline(i)==' ' | tline(i)=='.' | tline(i)=='+' | tline(i)=='+'
                (double(tline(i))>=48&&double(tline(i))<=57))
                    flag=0;
                    break;
            end
        end
    end
    if flag==1                              % 如果是数字行，把此行数据写入文件
        fprintf(fidout,'%s\n',tline);      % 如果是数字行，把此行数据写入--
                                            %--文件 dataout.txt
    end
    continue                               % 如果是非数字继续下一次循环
```



```
end
end
end
fclose('all');
A=importdata('dataout.txt');      % 将生成的 dataout.txt 文件导入工作--
                                   %-空间，变量名为 A，
```

3、角度格式转换

通常角度存储格式：dd.mmss

例如：22.3511183900 为 22° 35' 11.1839"

转换： 度分秒—>度

```
function degree = dms2degree(jiaodu)
%度分秒(dd.mmss)—>度
degree = fix(jiaodu);
mimute = fix((jiaodu-degree)*100);
second = (jiaodu-degree-mimute/100)*10000;
degree = degree+mimute/60+second/3600;
```

4、任务

1)、读取文件 data.txt 中的数据，并将其中经纬度和大地高数据按照格式存储到 dataout.txt 文件中，

格式为： 纬度 经度 大地高

经纬度格式为度

2)/角度格式转换，度—>度分秒(dd.mmss)

上机实验二、坐标正反算

一、实验目的

编写坐标正反算程序，并对格式化文件数据进行计算，验证程序。

二、实验内容：

1、编写坐标正算程序

要求：

1) 建立以 `xy_direct` 命名的函数，函数输入输出格式为

$$[x2,y2] = xy_direct(x1,y1,distance, azimuth)$$

2) 对文件 `data1.txt` 中数据进行坐标正算，并将已知点和计算点坐标按照格式存贮在文件 `data2.txt` 中，

`data1.txt` 格式为： x1 y1 距离 方位角 (dd.mmss)

`data2.txt` 格式为： x1 y1 x2 y2

2、编写坐标反算程序

要求：

1) 建立以 `xy_inv` 命名的函数，函数输入输出格式为

$$[distance, azimuth] = xy_inv(x1,y1, x2,y2)$$

2) 对文件 `data2.txt` 中数据进行坐标反算，并将计算结果按照格式存贮在文件 `data3.txt` 中，`data3.txt` 格式为： x1 y1 x2 y2 距离 方位角 (dd.mmss)

3、可能用到的函数

`sin(rad)`、`cos(rad)`、`atan2(y,x)`、`find`

例 2、

```
i = find(x>0);
```

```
x(i) = x(i)+1;
```

```
i = find(x<=0);
```

```
x(i) = x(i)-1;
```

上机实验三、大地坐标与空间直角坐标的转换

一、实验目的

编写大地坐标与空间直角坐标相互转换的程序，并对格式化文件数据进行计算，验证程序。

二、实验内容：

1、大地坐标向空间直角坐标转换

转换公式：

$$\begin{aligned}x &= (N + h) \cos B \cos L \\y &= (N + h) \cos B \sin L \\z &= [N(1 - e^2) + h] \sin B\end{aligned}\quad (1)$$

其中：L 为经度，B 为纬度，h 为大地高， $N = \frac{a}{\sqrt{1 - e^2 \sin^2 B}}$ ，为卯酉圈曲率半径， $e = \frac{\sqrt{a^2 - b^2}}{a}$

为第一偏心率，a 为旋转椭球长半轴，b 为短半轴。

WGS84 椭球参数：长半轴 $a = 6378137$

扁率 $f = 1/298.257223563$

根据上式创建以 geo2xyz 命名的函数，函数输入输出格式为

$$[x, y, z] = \text{geo2xyz}(L, B, h)$$

2、空间直角坐标向大地坐标转换

根据式（1）推导大地坐标向空间直角坐标转换公式：

$$\begin{aligned}L &= \arctan(y/x) \\B &= \arctan\left(\frac{z + Ne^2 \sin B}{\sqrt{x^2 + y^2}}\right) \\h &= \frac{\sqrt{x^2 + y^2}}{\cos B} - N\end{aligned}$$

注意计算纬度时需要用到迭代，可用 $B = \arctan\left(\frac{az}{b\sqrt{x^2 + y^2}}\right)$ 作为初始值。

创建以 xyz2geo 命名的函数，函数输入输出格式为

$$[L, B, h] = \text{xyz2geo}(x, y, z)$$

3、实例计算验证

首先将文件 data1.txt 中大地坐标转换为空间直角坐标，并将转换后的数据按照格式存贮在文件 data2.txt 中，

data1.txt 格式为: 经度 (dd.mmss) 纬度 (dd.mmss) 大地高

data2.txt 格式为:

x y z

然后将文件 data2.txt 中空间直角坐标转换为大地坐标, 并将计算结果按照格式存贮在文件 data3.txt 中,

data3.txt 格式为: 经度 (dd.mmss) 纬度 (dd.mmss) 大地高

4、可能用到的函数

sin(rad)、cos(rad)、atan2(y,x)、find, while……end

例

```
x=[20,10,5,2];
```

```
i = find(x>2);
```

```
while length(i)>0
```

```
    x(i) = x(i)/2;
```

```
    i = find(x>2);
```

```
end
```

例

```
B0 =      ;
```

```
i = find (B0+1-B0>10.^-6);
```

```
while length(i)>0
```

```
    B(i) = F(B0);
```

```
    i = find (abs(B0-B)>10.^-6);
```

```
    B0 = B;
```

```
end
```

上机实验四、子午线弧长正反算

一、实验目的

编写子午线弧长正反算的程序，并对格式化文件数据进行计算，验证程序。

二、实验内容：

1、子午线弧长正解公式

已知某点纬度 B ，求对子午线弧长 x ，计算公式：

$$m_0 = a(1 - e^2), \quad m_2 = \frac{3}{2}e^2 m_0, \quad m_4 = \frac{5}{4}e^2 m_2, \quad m_6 = \frac{7}{6}e^2 m_4, \quad m_8 = \frac{9}{8}e^2 m_6$$

$$a_0 = m_0 + \frac{1}{2}m_2 + \frac{3}{8}m_4 + \frac{5}{16}m_6 + \frac{35}{128}m_8$$

$$a_2 = \frac{1}{2}m_2 + \frac{1}{2}m_4 + \frac{15}{32}m_6 + \frac{7}{16}m_8$$

$$a_4 = \frac{1}{8}m_4 + \frac{3}{16}m_6 + \frac{7}{32}m_8$$

$$a_6 = \frac{1}{32}m_6 + \frac{1}{16}m_8$$

$$a_8 = \frac{1}{128}m_8$$

$$x = a_0 B - \frac{a_2}{2} \sin(2B) + \frac{a_4}{4} \sin(4B) - \frac{a_6}{6} \sin(6B) + \frac{a_8}{8} \sin(8B) \quad (1)$$

其中： B 为纬度，单位为弧度， $e = \frac{\sqrt{a^2 - b^2}}{a}$ 为第一偏心率， a 为旋转椭球长半轴， b 为短半轴。

根据上式创建以 `latitude2meridian` 命名的函数，函数输入输出格式为

$$x = \text{latitude2meridian}(B, \text{RefEllipsoid})$$

`RefEllipsoid` 为椭球参数

`RefEllipsoid` = [a, b, c, f, e2, e2_];

WGS84 椭球参数：长半轴 $a = 6378137$

扁率 $f = 1/298.257223563$

$b = a * (1 - f);$

$c = a * a / b;$

$e2 = (a * a - b * b) / (a * a);$

$e2_ = (a * a - b * b) / (b * b);$

2、子午弧长反解公式

如果已知子午线弧长 x ，求对应纬度 B ，称为子午线弧长反解。迭代公式如下：

$$B^{i+1} = (x - F(B^i)) / a_0$$

式中: $B^0 = \frac{x}{a_0}$

$$F(B^i) = -\frac{a_2}{2} \sin(2B^i) + \frac{a_4}{4} \sin(4B^i) - \frac{a_6}{6} \sin(6B^i) + \frac{a_8}{8} \sin(8B^i)$$

创建以 meridian2latitude 命名的函数, 函数输入输出格式为

$$B = \text{meridian2latitude}(x, \text{RefEllipsoid})$$

3、实例计算验证

首先读取文件 data1.txt 中的纬度, 计算相应子午线弧长, 并存贮在文件 data2.txt 中, data1.txt 格式为: 经度 (dd.mmss) 纬度 (dd.mmss) 大地高
data2.txt 格式为:

子午线弧长 x(m)

然后根据文件 data2.txt 中的子午线弧长计算纬度, 并将计算结果按照格式存贮在文件 data3.txt 中, data3.txt 格式为: 纬度 (dd.mmss)

4、可能用到的函数

sin(rad)

上机实验五、高斯投影正反算

一、实验目的

编写高斯投影正反算的程序，并对格式化文件数据进行计算，验证程序。

二、实验内容：

1、高斯投影正算公式

已知大地坐标 (B,L) 及中央子午线精度 L_0 ，计算高斯平面坐标 (x, y)，公式如下：

$$x = X + \frac{N}{2} \sin(B) \cos(B) l^2 + \frac{N}{24} \sin(B) \cos^3(B) (5 - t^2 + 9\eta^2 + 4\eta^4) l^4 \\ + \frac{N}{720} \sin(B) \cos^5(B) (61 - 58t^2 + t^4 + 270\eta^2 - 330\eta^2 t^2) l^6$$

$$y = N \cos(B) l + \frac{N}{6} \cos^3(B) (1 - t^2 + \eta^2) l^3 + \frac{N}{120} \cos^5(B) (5 - 18t^2 + t^4 + 14\eta^2 - 58\eta^2 t^2) l^5$$

其中： B 为纬度， $l = L - L_0$ ，单位为弧度， $N = \frac{a}{\sqrt{1 - e^2 \sin^2 B}}$ ，为卯酉圈曲率半径，

$t = \tan B, \eta^2 = e'^2 \cos^2 B$ ， $e' = \frac{\sqrt{a^2 - b^2}}{b}$ 为第二偏心率， a 为旋转椭球长半轴， b 为短半轴， X

为子午线弧长。

根据上式创建以 GaussianMapDirect 命名的函数，函数输入输出格式为

$$[x,y] = \text{GaussianMapDirect}(B, l, \text{RefEllipsoid})$$

RefEllipsoid 为椭球参数

$$\text{RefEllipsoid} = [a, b, c, f, e2, e2_];$$

WGS84 椭球参数：长半轴 $a = 6378137$

$$\text{扁率 } f = 1/298.257223563$$

$$b = a(1-f)$$

$$c = a*a/b;$$

$$e2 = (a*a-b*b)/(a*a);$$

$$e2_ = (a*a-b*b)/(b*b);$$

2、高斯反算公式

已知高斯平面坐标 (x, y) 及指定中央子午线精度 L_0 ，计算大地坐标 (B,L)：

$$B = B_f - \frac{t_f}{2M_f N_f} y^2 + \frac{t_f}{24M_f N_f^3} (5 + 3t_f^2 + \eta_f^2 - 9t_f^2 \eta_f^2) y^4 \\ + \frac{t_f}{720M_f N_f^5} (61 + 90t_f^2 + 45t_f^4) y^6$$

$$L = L_0 + \frac{1}{N_f \cos B_f} y - \frac{1}{6N_f^3 \cos B_f} (1 + 2t_f^2 + \eta_f^2) y^3$$

$$+ \frac{1}{120N_f^5 \cos B_f} (5 + 28t_f^2 + 24t_f^4 + 6\eta_f^2 + 8t_f^2\eta_f^2) y^5$$

式中, $N_f = \frac{a}{\sqrt{1 - e^2 \sin^2 B_f}}$, $M_f = \frac{a(1 - e^2)}{\sqrt{(1 - e^2 \sin^2 B_f)^3}}$, $\eta_f = e'^2 \cos^2 B_f$,

$t_f = \tan B_f$, B_f 为根据子午线弧长 x 反算的底点纬度。

创建以 GaussianMapInverse 命名的函数, 函数输入输出格式为

[B,l] = GaussianMapInverse(x, y, RefEllipsoid)

3、实例计算验证

首先读取文件 data1.txt 中的数据, 计算其在相应六度带高斯投影带内的高斯平面直角坐标, 并存贮在文件 data2.txt 中,

data1.txt 格式为: 经度 (dd.mmss) 纬度 (dd.mmss) 大地高

data2.txt 格式为:

x(m) y(m) 中央子午线经度 (dd.mmss)

然后根据文件 data2.txt 中的高斯平面直角坐标及其中央子午线经度, 计算其经纬度, 并将计算结果按照格式存贮在文件 data3.txt 中,

data3.txt 格式为: 经度 (dd.mmss) 纬度 (dd.mmss)

4、可能用到的函数

sin(rad), cos(rad), tan(rad)

上机实验六、高斯投影邻带换算

一、实验目的

编写高斯投影邻带换算的程序，并对格式化文件数据进行计算，验证程序。

二、实验内容：

1、高斯投影邻带换算方法

已知高斯平面坐标 (x_0, y_0) 及其中央子午线经度 L_0 ，计算其在相邻投影带（3 度或 6 度或任意，中央子午线经度为 L_1 ）内的高斯平面直角坐标 (x_1, y_1) ，转换步骤：先利用高斯投影反算将高斯平面坐标 (x_0, y_0) 转换为经纬度，再利用高斯投影正算转换为相邻投影带内的高斯平面直角坐标 (x_1, y_1) 。

创建以 GaussianZoneTtrans 命名的函数，函数输入输出格式为

$$[x_1, y_1] = \text{GaussianZoneTtrans}(x_0, y_0, L_0, L_1, \text{RefEllipsoid})$$

2、实例计算验证

首先读取文件 data1.txt 中的数据，计算其在相应六度带高斯投影带内的高斯平面直角坐标，并存贮在文件 data2.txt 中，

data1.txt 格式为： $x_0(\text{m})$ $y_0(\text{m})$ 中央子午线经度 (dd.mmss)

data2.txt 格式为：

$$x_1(\text{m}) \quad y_1(\text{m}) \quad \text{中央子午线经度 (dd.mmss)}$$

然后根据文件 data2.txt 中的高斯平面直角坐标及其中央子午线经度，计算其经纬度，并将计算结果按照格式存贮在文件 data3.txt 中。

上机实验七、平面直角坐标转换

一、实验目的

不同坐标系下平面直角坐标转换是测量工程中经常会遇到的问题，本实验通过编写平面直角坐标四参数转换的程序，并对格式化文件数据进行计算，验证程序，从而掌握平面直角转换的基本原理和方法。

二、实验内容：

1、平面直角坐标转换四参数模型

假设某点坐标转换前后的平面坐标分别为 (x_1, y_1) 和 (x_2, y_2) ，则二维坐标的坐标转换公式为：

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} + (1+m) \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \quad (1)$$

式中 m ：坐标尺度参数，表示坐标转换前后坐标系的尺度变化情况；

θ ：坐标旋转角，表示转换后的坐标系绕前一个坐标系逆时针旋转的角度；

$\Delta x, \Delta y$ ：平移参数，表示坐标原点的平移量。

2、坐标转换参数的计算

如已知测区两个或两个以上公共点分别在西安 80 坐标和局部独立坐标系下的平面坐标 (x_i^{80}, y_i^{80}) 和 $(x_i^{local}, y_i^{local})$ ，可计算测区的平面坐标转换参数。

$$\begin{aligned} x_i^{local} &= \Delta x + (1+m) \cos \theta x_i^{80} + (1+m) \sin \theta y_i^{80} \\ y_i^{local} &= \Delta y - (1+m) \sin \theta x_i^{80} + (1+m) \cos \theta y_i^{80} \end{aligned}$$

令 $(1+m) \cos \theta = \mu$ ， $(1+m) \sin \theta = \delta$ ，则可得方程：

$$\begin{bmatrix} x_i^{local} \\ y_i^{local} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_i^{80} & y_i^{80} \\ 0 & 1 & y_i^{80} & -x_i^{80} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \mu \\ \delta \end{bmatrix}$$

根据上式可解得转换参数 $\Delta x, \Delta y, \mu, \delta$ 。

进一步可得：
$$m = \sqrt{\mu^2 + \delta^2} - 1$$

$$\theta = \arctan(\delta / \mu)$$

创建以 TtransParam_Comp 命名的函数，用于计算转换参数，函数输入输出格式为

$$[\Delta x, \Delta y, m, \theta] = \text{TtransParam_Comp}(x_1, y_1, x_2, y_2)$$

创建以 CoordTtrans2D 命名的函数,用于二维平面坐标转换,函数输入输出格式为:

$$[x2,y2] = \text{CoordTtrans2D}(x1, y1, \text{Parameter4})$$

3、实例计算验证

首先读取文件 data1.txt 中的公共点数据,计算转换参数,然后根据计算的转换参数对 data2.txt 中坐标进行转换,并将计算结果按照格式存贮在文件 data3.txt 中。

data1.txt 格式为: 点号 x1(m) y1(m) x2(m) y2(m)

data2.txt 格式为: 点号 x1(m) y1(m)

data3.txt 格式为: 点号 x1(m) y1(m) x2(m) y2(m)

上机实验八、空间直角坐标转换

一、实验目的

不同坐标系下空间直角坐标转换是测量工程中经常会遇到的问题，本实验通过编写空间直角坐标七参数转换的程序，并对格式化文件数据进行计算，验证程序，从而掌握空间直角转换的基本原理和方法。

二、实验内容：

1、空间直角坐标转换七参数模型（布尔沙模型）

布尔沙-沃尔夫模型(在我国通常被简称为布尔沙模型)又被称为七参数转换(7-Parameter Transformation)或七参数赫尔墨特变换(7-Parameter Helmert Transformation)，其数学公式为：

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} dX_0 \\ dY_0 \\ dZ_0 \end{pmatrix} + (1+m)R_3(\omega_Z)R_2(\omega_Y)R_1(\omega_X) \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (1)$$

其中， X 、 Y 、 Z 和 X' 、 Y' 、 Z' 为某点分别在 $O-XYZ$ 和 $O'-X'Y'Z'$ 下的坐标； ω_X 、 ω_Y 、 ω_Z 为由 $O-XYZ$ 转换到 $O'-X'Y'Z'$ 的旋转参数； m 为由 $O-XYZ$ 转换到 $O'-X'Y'Z'$ 的尺度参数； dX_0 、 dY_0 、 dZ_0 由 $O-XYZ$ 转换到 $O'-X'Y'Z'$ 的平移参数；

$$R_1(\omega_X) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \omega_X & \sin \omega_X \\ 0 & -\sin \omega_X & \cos \omega_X \end{pmatrix} \quad R_2(\omega_Y) = \begin{pmatrix} \cos \omega_Y & 0 & -\sin \omega_Y \\ 0 & 1 & 0 \\ \sin \omega_Y & 0 & \cos \omega_Y \end{pmatrix}$$
$$R_3(\omega_Z) = \begin{pmatrix} \cos \omega_Z & \sin \omega_Z & 0 \\ -\sin \omega_Z & \cos \omega_Z & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{分别为三个旋转矩阵。}$$

通常情况下，涉及两个不同大地基准间旋转的3个欧拉角 ω_X 、 ω_Y 、 ω_Z 都非常小，式1可近似简化为：

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} dX_0 \\ dY_0 \\ dZ_0 \end{pmatrix} + (1+m) \begin{pmatrix} 1 & \omega_Z & -\omega_Y \\ -\omega_Z & 1 & \omega_X \\ \omega_Y & -\omega_X & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (2)$$

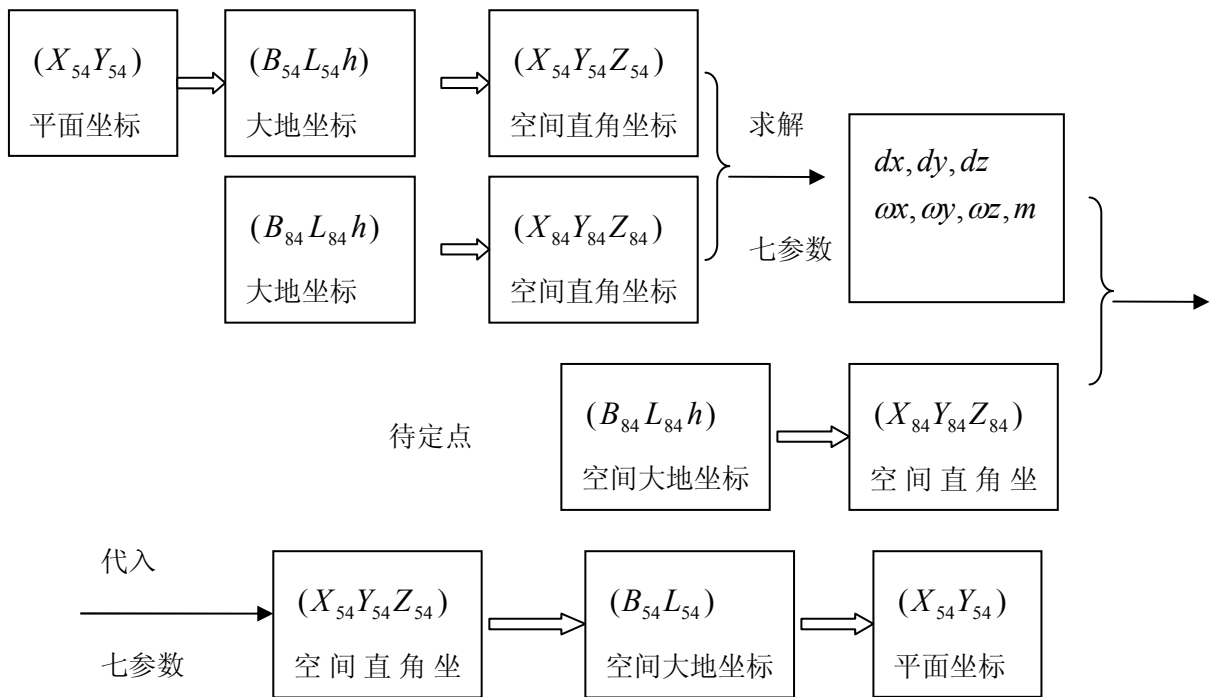
上式可进一步写为：

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 & -Z & Y & X \\ 0 & 1 & 0 & Z & 0 & -X & Y \\ 0 & 0 & 1 & -Y & X & 0 & Z \end{pmatrix} \begin{pmatrix} dX_0 \\ dY_0 \\ dZ_0 \\ \omega_X \\ \omega_Y \\ \omega_Z \\ m \end{pmatrix} \quad (3)$$

上式即为适用于任意两个空间直角坐标系统相互转换的布尔沙七参数公式。

2、坐标转换流程

如已知测区三个以上公共点的北京 54 高斯直角坐标和 WGS84 坐标,可按照下列流程来计算坐标转换参数和进行坐标转换。



创建以 TtransParam7_Comp 命名的函数,用于计算转换参数,函数输入输出格式为

$$\text{Param7} = \text{TtransParam7_Comp}(x1, y1, z1, x2, y2, z2)$$

创建以 CoordTtrans3D 命名的函数,用于空间直角坐标转换,函数输入输出格式为:

$$[x2, y2, z2] = \text{CoordTtrans3D}(x1, y1, z1, \text{Param7})$$

北京 54 坐标系的椭球参数 (克拉索夫斯基椭球):

$$a = 6378245; \quad f = 1/298.3;$$

3、实例计算验证

首先读取文件 data1.txt 中的公共点数据,计算转换参数,然后根据计算的转换参数对 data2.txt 中坐标进行转换,并将计算结果按照格式存贮在文件 data3.txt 中。

data1.txt 格式为: 点号 x1(m) y1(m) x2(m) y2(m)

data2.txt 格式为: 点号 x1(m) y1(m)

data3.txt 格式为: 点号 x1(m) y1(m) x2(m) y2(m)

上机实验九、GPS 水准高程拟合

一、实验目的

通过对已知 GPS 水准数据进行拟合确定区域范围内任意一点高程异常，从而确定该点的正常高是 GPS 应用中经常遇到的问题，本实验通过编写 GPS 水准高程拟合程序，实现 GPS 高向正常高的转换，从而掌握 GPS 水准高程测量的基本原理和方法。

二、实验内容：

1、多项式拟合模型

假设某地区有 n 个已知 GPS 水准点，测站点的坐标是 (x_i, y_i) ，已知高程异常为 ζ ，其中 $i=1,2,3,L,n$ 。多项式拟合模型的一般表达式可写为：

$$\zeta_i = f(x_i, y_i) + \varepsilon_i \quad (1)$$

式中： $f(x_i, y_i)$ 为拟合的似大地水准面， ε_i 为误差。

当对 $f(x_i, y_i)$ 只取一次项时，即将似大地水准面表示为一个平面，有

$$f(x_i, y_i) = a_0 + a_1 x_i + a_2 y_i \quad (2)$$

同理，表示为二次和三次多项式的表达式分别为

$$f(x_i, y_i) = a_0 + a_1 x_i + a_2 y_i + a_3 x_i y_i + a_4 x_i^2 + a_5 y_i^2 \quad (3)$$

$$f(x_i, y_i) = a_0 + a_1 x_i + a_2 y_i + a_3 x_i y_i + a_4 x_i^2 + a_5 y_i^2 + a_6 x_i^3 + a_7 x_i^2 y_i + a_8 x_i y_i^2 + a_9 y_i^3 \quad (4)$$

式中， (a_0, a_1, a_2, L, a_9) 为多项式的待定系数，可称它为参数。

多项式拟合的数学模型表示为矩阵的形式，则令

$$L = \begin{Bmatrix} \zeta_1 \\ \zeta_2 \\ \Lambda \\ \zeta_n \end{Bmatrix}, \quad A = \begin{Bmatrix} a_1 \\ a_2 \\ \Lambda \\ a_9 \end{Bmatrix}, \quad V = \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \Lambda \\ \varepsilon_n \end{Bmatrix}$$

$$B = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 & x_1^3 & x_1^2 y_1 & x_1 y_1^2 & y_1^3 \\ 1 & x_2 & y_2 & x_2^2 & x_2 y_2 & y_2^2 & x_2^3 & x_2^2 y_2 & x_2 y_2^2 & y_2^3 \\ L & L & L & L & L & L & L & L & L & L \\ 1 & x_n & y_n & x_n^2 & x_n y_n & y_n^2 & x_n^3 & x_n^2 y_n & x_n y_n^2 & y_n^3 \end{bmatrix}$$

多项式的矩阵形式为：

$$L = BA + V \quad (5)$$

当采用一次多项式对该区域高程异常进行拟合时，由于只有 3 个待定参数，需要至少 3 个已知的 GPS 测点，用最小二乘法 ($[\epsilon_i^2] = \min$) 求拟合系数，得到

$$A = (B^T B)^{-1} B^T L \quad (6)$$

然后可以利用拟合出来的多项式对该区域内已知位置（即已知坐标）的点进行内插，求得其高程异常，最后得到正常高。对于二次和三次多项式，则要知道至少 6 和 10 个已知点按上面所提供的模型进行拟合来确定模型中的待定参数。

3、实例计算验证

创建以 Fitting_Param 命名的函数，用于求多项式拟合系数，函数输入输出格式为：

$$\text{Polyn_Par} = \text{Fitting_Param}(x1, y1, h, n)$$

其中 Polyn_Par 为 n 次多项式系数 ($a_0, a_1, a_2, \dots, a_n$)。

创建以 Fitting_Polyn 命名的函数，用于多项式拟合，函数输入输出格式为：

$$\text{Value_fitting} = \text{Fitting_Polyn}(x1, y1, \text{Polyn_Par})$$

首先读取文件 GPSLeveling.txt 中高程异常数据，数据格式说明文件，求得多项式系数后，计算文件 GPSL_check.txt 中检核点的拟合高程异常，将之与检核文件中已知的高程异常进行比较。