粗差探测与抗差估计作业

姓名：白腾飞

专业：测绘科学与技术

学号：29620209220

# **GPS向量网抗差估计**

## 理论分析

我们可以推导出的抗差解式为：



的权逆阵（协方差阵）为



为等价权矩阵，矩阵元素为



上式是双因子等价权模型，适用于观测值相关和不相关。，分别是观测值，对应的权因子（自适应降权因子、收缩因子）。采用IGG3作为等价权函数时，权因子函数为



实际计算中，一般取3.0-5.0，取8.0-10.0。注意，上式中的是标准化残差

，是单位权方差因子，是的协方差（权倒数）。

等价权需要计算权因子，权因子是残差的函数，残差又是参数平差值的函数，故抗差估计需选权迭代。主要过程：首先以最小二乘平差值作为参数平差值的初始值，计算残差和标准化残差，再计算每个观测值的权因子，然后进行抗差估计计算；这样可生成新的参数估值，又可以计算残差、权因子、参数估值，迭代，直到连续两次迭代的坐标改正数小于正数。

### 残差的协方差阵的求取



之前认识不足，认为上式可以采用协方差传播率，但是却忘记了协方差传播率的先决条件：需要构造一个量与观测值的函数关系，且涉及到的观测值应该是独立的。

例:







可以得到



应用于协方差传播率，







从1.10和1.11可以得到



所以残差的权倒数为：



是观测值的权倒数，可以从读取到的观测值的方差-协方差矩阵中获得；

对于来说，1.11也有的方差-协方差阵，但是在程序设计中，没必要每一次把整个方差-协方差阵计算出来，可以通过（注意观测值是基线向量，参数是站坐标）



假设这个基线向量起点是k1，结点是k2，则



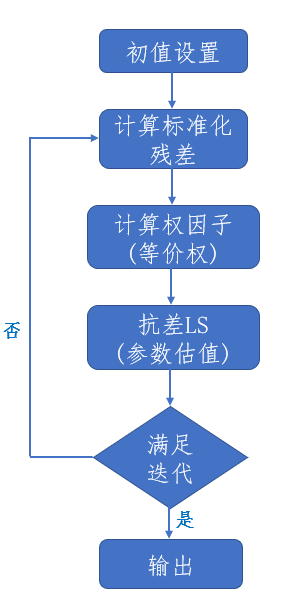
### 标准化残差的计算

一般先计算，这两个值上文都给了求取过程。之后通过中位数计算方差因子，即



是的数组，之后便可计算标准化残差。

## 流程图



## 程序设计

void CCoGPS::Robust(int fname,double eps)

{

int n=3\*Vnumber; //观测值总数

QV=new double [n]; //残差的权倒数（协方差阵）

W =new double [n]; //权因子

//------------------------------------

// 计算观测值的权矩阵，从文件中读取的是协方差阵

int ii=0; // 计数器

for(int i=0; i<Anumber; i++)//Anumber 向量组总数

{

int ni = 3\*(dir0[i+1]-dir0[i]); //向量组观测值总数

for(int j=0;j<ni;j++)

{

QV[ii]=P[i][ij(j,j)];//把协方差阵存到QV中

ii++;

}

inverse(P[i],ni);//权矩阵

}

//--------------------------------------

// 计算残差的权倒数

CaATPA(); //组成法方程

if(!inverse(ATPA,3\*Pnumber)) //法方程系数矩阵求逆

{

//MyBreak("法方程系数降秩降秩！");

exit(0);

}

caQV(); //计算残差的权倒数，并且把残差的权倒数（协方差阵）存到QV数组中

//--------------------------------------

// 将观测值的权备份到数组PP

double \*\*PP= new double \*[Anumber]; //等价权数组

for(int k=0; k<Anumber; k++)

{

int nk=3\*(dir0[k+1]-dir0[k]);

PP[k]=new double[nk\*(nk+1)/2];

for(int i=0;i<nk\*(nk+1)/2;i++) PP[k][i]=P[k][i];

}

//---------------------------------------

// 抗差估计迭代

for( int i=0;i<n;i++) W[i]=1.0;//权因子赋初值，每个观测值的权因子都赋为1，

int No=0; //迭代次数

while(1)

{

No++;

// 计算等价权

for(int k=0; k<Anumber; k++) // k－向量组编号

{

int nk = 3\*(dir0[k+1]-dir0[k]); //观测值个数

double \*Wk=W+3\*dir0[k]; //权因子的数组地址

int index = 0;

for(int i=0;i<nk;i++)

{

for(int j=0;j<=i;j++)

{

if(Wk[i]<1.0 || Wk[j]<1.0)

P[k][index] = PP[k][index]\*sqrt(Wk[i]\*Wk[j]);

else P[k][index] = PP[k][index];

index++;

}

}

}

CaATPA(); //组成法方程

Known(); // 已知点处理

double maxX=Ca\_dX(0);

CaLV(V); //计算残差

fprintf(resultfp,"\n抗差估计，第%d次迭代...\nmax(dX)=%e",No,maxX);

PrintLV("抗差估计残差",V);

if(fabs(maxX)<eps && No>1)//迭代结束的判断

{

break;

}

//-----------------------------------------------

//权因子计算

Ca\_UintV(); //标准化残差计算(放在W里)

for(int i=0;i<3\*Vnumber;i++)

{

W[i]=Wi(fname,W[i]/MEDIAN,5.0 , 8.0);

}

}

int dd=0; //降权观测值的个数

for(int i=0;i<3\*Vnumber;i++)

{

if(W[i]<1.0)dd++;

}

if(dd==0)

{

fprintf(resultfp,"\n 所有观测值均未降权!\n");

}

else

{ //输出降权因子

fprintf(resultfp,"\n 共有%d个观测值被降权!\n",dd);

fprintf(resultfp,"\n\n权因子: \n");

for(int i=0;i<3\*Vnumber;i++)

{

if(W[i]<1.0)

{

int ni=AreaNumber[i/3];

fprintf(resultfp,"\n%8d",ni+1);

fprintf(resultfp,"%10s %10s",

Pname[dir1[i/3]],Pname[dir2[i/3]]);

fprintf(resultfp," d%c %10.4lf",'x'+i%3,W[i]);

}

}

}

double VTPV=CaVTPV();

Sigma=sqrt(VTPV/(n-dd-3\*unPnumber));

PrintXYZ();

PrintLV("V",V);

fprintf(resultfp,"\n\n抗差估计:\n[pvv]=%e",VTPV);

fprintf(resultfp,"\n单位权中误差μ=±%e",Sigma);

delete []PP;

PrintM(resultfp,W,3\*Vnumber,3,"%6.3lf ","迭代权因子"); }

## 算例分析

抗差估计，第1次迭代...

max(dX)=4.271008e-01

===== 抗差估计残差 =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0199 0.1167 -0.3655

1 E B 0.0065 -0.0378 -0.0188

1 E F 0.4356 0.4094 0.3143

2 B F -0.5827 0.4293 0.3119

2 B E -0.0195 0.0278 0.0168

2 B D -0.0152 0.1773 0.1481

3 C A -0.0471 -0.0527 -0.0141

3 C B 0.0114 -0.0668 -0.0499

3 C D -0.0052 0.1009 0.1036

4 D E -0.0108 -0.1277 -0.1365

4 D B 0.0038 -0.1753 -0.1501

4 D C -0.0016 -0.1209 -0.1066

5 F E -0.4271 -0.3948 -0.2997

5 F D -0.4035 -0.2759 -0.1367

5 F B -0.4071 -0.4562 -0.3249

6 A B 0.0400 -0.0614 -0.0252

6 A C 0.0402 0.0563 0.0413

抗差估计，第2次迭代...

max(dX)=-4.478696e-01

===== 抗差估计残差 =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0256 0.0777 -0.3687

1 E B -0.0074 0.0052 0.0019

1 E F -0.0123 -0.0217 -0.0261

2 B F -1.0166 -0.0448 -0.0492

2 B E -0.0056 -0.0152 -0.0039

2 B D 0.0044 0.0952 0.1243

3 C A -0.0402 -0.0520 -0.0182

3 C B 0.0044 -0.0230 -0.0334

3 C D 0.0074 0.0626 0.0962

4 D E -0.0165 -0.0887 -0.1333

4 D B -0.0158 -0.0932 -0.1263

4 D C -0.0142 -0.0826 -0.0992

5 F E 0.0208 0.0363 0.0407

5 F D 0.0501 0.1162 0.2006

5 F B 0.0268 0.0179 0.0362

6 A B 0.0261 -0.0184 -0.0045

6 A C 0.0333 0.0556 0.0454

抗差估计，第3次迭代...

max(dX)=-2.274615e-01

===== 抗差估计残差 =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0066 -0.1498 -0.5818

1 E B -0.0128 -0.0733 -0.0623

1 E F 0.0060 -0.0768 -0.0658

2 B F -0.9930 -0.0214 -0.0247

2 B E -0.0002 0.0633 0.0603

2 B D -0.0093 -0.0537 -0.0246

3 C A -0.0326 0.0343 0.0597

3 C B 0.0067 -0.0152 -0.0198

3 C D -0.0040 -0.0785 -0.0390

4 D E 0.0025 0.1388 0.0798

4 D B -0.0021 0.0557 0.0226

4 D C -0.0028 0.0585 0.0360

5 F E 0.0025 0.0914 0.0804

5 F D 0.0127 -0.0562 0.0272

5 F B 0.0032 -0.0055 0.0117

6 A B 0.0207 -0.0969 -0.0687

6 A C 0.0257 -0.0307 -0.0325

抗差估计，第4次迭代...

max(dX)=1.376257e-01

===== 抗差估计残差 =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0106 -0.0122 -0.4945

1 E B -0.0099 0.0079 0.0048

1 E F -0.0018 0.0178 0.0210

2 B F -1.0037 -0.0080 -0.0050

2 B E -0.0031 -0.0179 -0.0068

2 B D -0.0082 0.0027 -0.0044

3 C A -0.0358 -0.0342 0.0102

3 C B 0.0064 -0.0025 -0.0021

3 C D -0.0032 -0.0095 -0.0011

4 D E -0.0015 0.0012 -0.0075

4 D B -0.0032 -0.0007 0.0024

4 D C -0.0036 -0.0105 -0.0019

5 F E 0.0103 -0.0032 -0.0064

5 F D 0.0246 -0.0132 0.0276

5 F B 0.0139 -0.0189 -0.0080

6 A B 0.0236 -0.0157 -0.0016

6 A C 0.0289 0.0378 0.0170

抗差估计，第5次迭代...

max(dX)=3.826495e-10

===== 抗差估计残差 =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0106 -0.0122 -0.4945

1 E B -0.0099 0.0079 0.0048

1 E F -0.0018 0.0178 0.0210

2 B F -1.0037 -0.0080 -0.0050

2 B E -0.0031 -0.0179 -0.0068

2 B D -0.0082 0.0027 -0.0044

3 C A -0.0358 -0.0342 0.0102

3 C B 0.0064 -0.0025 -0.0021

3 C D -0.0032 -0.0095 -0.0011

4 D E -0.0015 0.0012 -0.0075

4 D B -0.0032 -0.0007 0.0024

4 D C -0.0036 -0.0105 -0.0019

5 F E 0.0103 -0.0032 -0.0064

5 F D 0.0246 -0.0132 0.0276

5 F B 0.0139 -0.0189 -0.0080

6 A B 0.0236 -0.0157 -0.0016

6 A C 0.0289 0.0378 0.0170

共有2个观测值被降权!

权因子:

1 E D dz 0.0000

2 B F dx 0.0000

STATION X Y Z RMS\_X RMS\_Y RMS\_Z

(m) (m) (m) (cm) (cm) (cm)

----------------------------------------------------------------------------

1 A -2703200.7813 4678971.0170 3376976.8039 0.00 0.00 0.00

2 B -2700177.6199 4681527.9438 3375839.6019 0.33 1.40 1.05

3 C -2700962.6503 4677908.5816 3380169.5992 0.48 1.88 1.39

4 D -2699182.6825 4680027.0469 3378669.1672 0.36 1.13 0.90

5 E -2696225.5305 4683062.6388 3376832.8181 0.00 0.00 0.00

6 F -2697018.6637 4684079.9613 3374834.1779 0.65 1.98 1.42

Max(RMS\_X)= 0.65

Max(RMS\_Y)= 1.98

Max(RMS\_Z)= 1.42

===== V =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0106 -0.0122 -0.4945

1 E B -0.0099 0.0079 0.0048

1 E F -0.0018 0.0178 0.0210

2 B F -1.0037 -0.0080 -0.0050

2 B E -0.0031 -0.0179 -0.0068

2 B D -0.0082 0.0027 -0.0044

3 C A -0.0358 -0.0342 0.0102

3 C B 0.0064 -0.0025 -0.0021

3 C D -0.0032 -0.0095 -0.0011

4 D E -0.0015 0.0012 -0.0075

4 D B -0.0032 -0.0007 0.0024

4 D C -0.0036 -0.0105 -0.0019

5 F E 0.0103 -0.0032 -0.0064

5 F D 0.0246 -0.0132 0.0276

5 F B 0.0139 -0.0189 -0.0080

6 A B 0.0236 -0.0157 -0.0016

6 A C 0.0289 0.0378 0.0170

抗差估计:

[pvv]=5.059688e+01

单位权中误差μ=±1.169394e+00

迭代权因子:

1 1.000 1.000 0.000

2 1.000 1.000 1.000

3 1.000 1.000 1.000

4 0.000 1.000 1.000

5 1.000 1.000 1.000

6 1.000 1.000 1.000

7 1.000 1.000 1.000

8 1.000 1.000 1.000

9 1.000 1.000 1.000

10 1.000 1.000 1.000

11 1.000 1.000 1.000

12 1.000 1.000 1.000

13 1.000 1.000 1.000

14 1.000 1.000 1.000

15 1.000 1.000 1.000

16 1.000 1.000 1.000

17 1.000 1.000 1.000

# GPS向量网粗差探测

## 理论分析

依据均值平移模型粗差检验原理，设粗差参数以g表示，均值平移模型的误差方程为

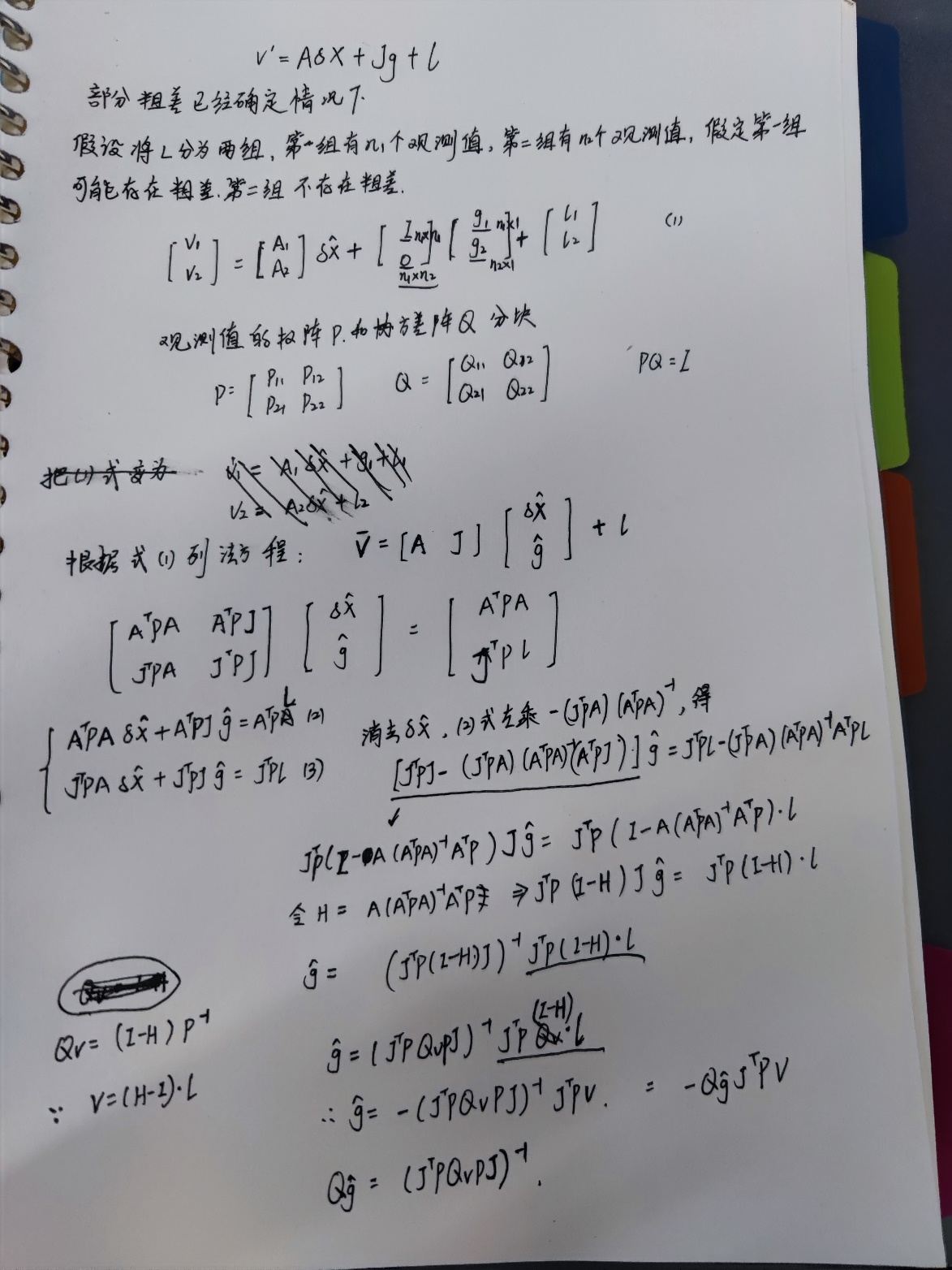


式中，表示残差向量。g的维数是n1，表示有粗差的观测值的个数。

假设n=6,n1=2,在第1、第4个观测值有粗差，那么J可表示为：

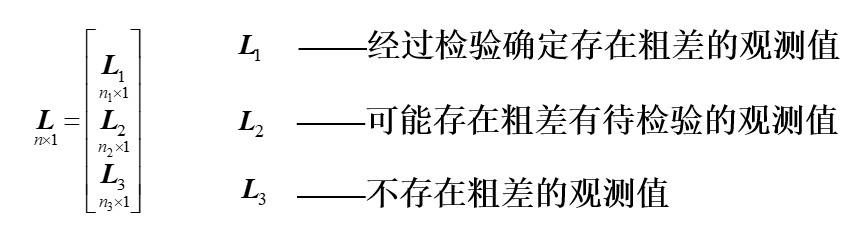


可以证明，式1.18进行最小二乘平差得到的参数估值，相当于将可疑观测值剔除之后进行最小二乘平差得到的参数估值，即两者等价。



对于的推导要注意的是，，H是幂等阵，那么也是幂等阵。

### 部分粗差已经确定情况下的粗差检验



可列误差方程式：



设



可以得到：



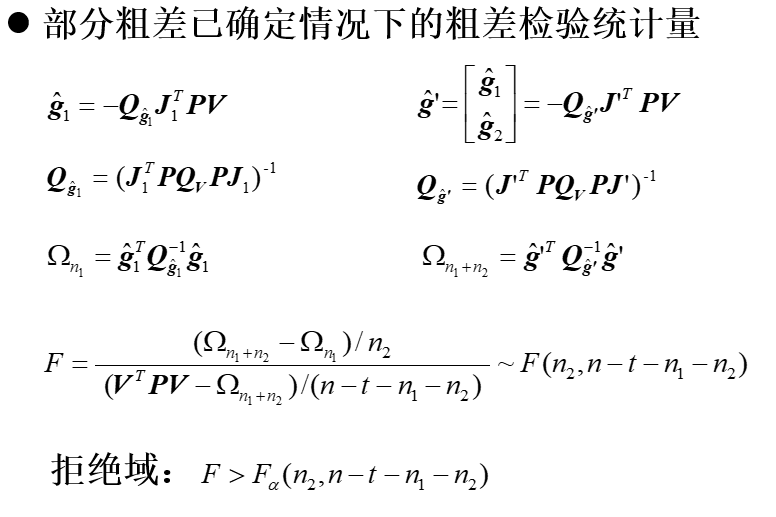
根据上面的推导可以得到





注意，上式中的是根据最小二乘计算出来的残差，也是。

### 粗差检验统计量的构造（为什么要这样选统计量？）



成立，则判定存在粗差。

这个统计量是服从F分布的。

### 粗差搜索流程

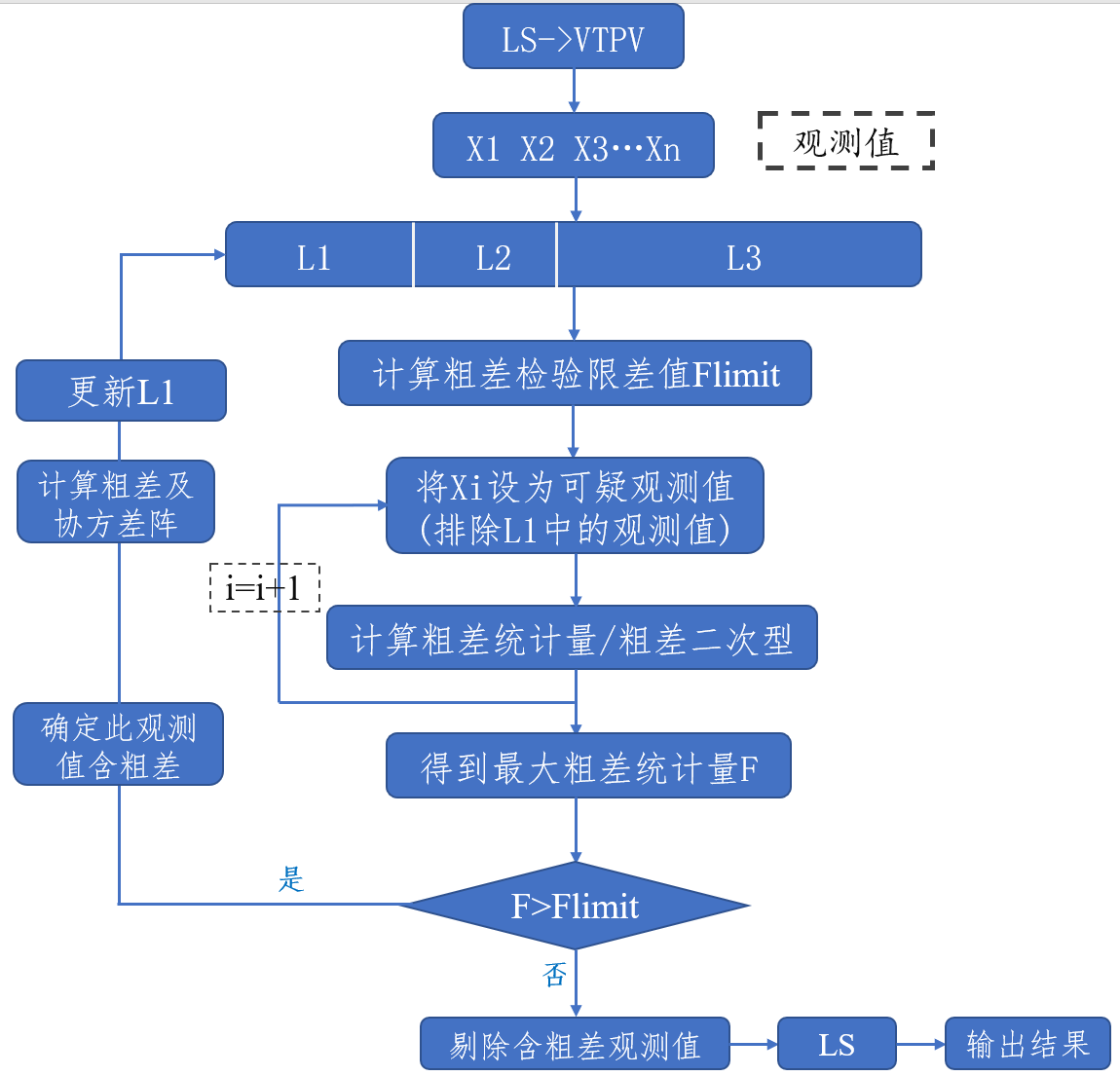
1）假定观测值中仅有一个粗差()，但不知是哪一个观测值。这时，首先将第1个观测值当作可疑观测值，即L2中的观测值，将其余的个观测值作为无粗差的观测值，即L3 ，先求粗差估值再求权逆阵，之后求得统计量的值并记下统计量的值；接着，再将第2个观测值当作可疑观测值，即L2中的观测值，将其余的个观测值作为无粗差的观测值，也求出粗差估值和统计量的值；……；直至最后1个观测值，可以得到*n*个统计量。在计算统计量的过程中记下绝对值最大的统计量及最大统计量相应的观测值编号，统计量计算结束后，对绝对值最大的统计量判断是否超限，如果统计量不超限，则认为观测值中没有粗差，结束搜索，转至4）；否则，判定绝对值最大的统计量相应的观测值存在粗差。此即第一次搜索；

2）将前次搜索出的含粗差观测值归入L1 ；

3）将L中不属于L1的观测值逐个假定为可疑观测值，这时观测值就分为了三类：经检验确定存在粗差的观测值、可疑观测值、无粗差的观测值，求得部分粗差已确定情况下的粗差估值并记下统计量的值，总共可计算出n-n1个统计量，仍然挑选绝对值最大的统计量进行统计检验，若绝对值最大的统计量不超限，认为观测值中不存在新的粗差，结束搜索，转至4）；否则，判定绝对值最大的统计量计算中最新加入的观测值存在粗差，转至2），继续下次搜索；

4）输出计算成果。

## 流程图



## 程序设计

//Arfa是给定的显著水平，在粗差搜索的过程中，函数内将根据显著水平的值调用概率计算类计算粗差检验的限差值

void CCoGPS::DataSnooping(double Arfa,int Max) // Max-最大搜索次数

{

// 为数组申请内存

double \*JTPA=new double [3\*Pnumber\*Max];

double \*JTPV=new double [Max];

double \*Qg= new double [Max\*(Max+1)/2]; // 粗差权逆阵

double \*gross= new double [Max]; // 粗差估值

int \*Jint= new int [Max]; // 可疑观测值序号

CProbability MyProb; //概率计算类

//-------------------------------------------------

// 最小二乘平差

for(int i=0; i<Anumber; i++)

{

int ni = 3\*(dir0[i+1]-dir0[i]); //向量组观测值总数

inverse(P[i],ni);

}

CaATPA();

Known();

Ca\_dX(false);

CaLV(V); //计算最小二乘残差

double VTPV=CaVTPV();//计算VTPV二次型，在之后计算统计量的时候用的到

//---------------------------------------------------

//粗差探测

fprintf(resultfp,"\n 逐个搜索法粗差探测:\n");

int maxp; //当前搜索中，统计量最大的观测值序号

int Gn=0; //当前搜索次数

double new\_Omg; // 本次搜索最大粗差二次型;

double last\_Omg=0.0; // 前次搜索最大粗差二次型

int r=3\*Vnumber-3\*unPnumber; // 多余观测数

bool \*W=new bool[3\*Vnumber]; //粗差标志数组

for( int i=0; i<3\*Vnumber; i++)W[i]=0;//用来标定此观测值是不是含有粗差，若有，赋值为1；若没有，则为0.

while(1)

{

if(Gn+1>Max)//最多能搜索到的粗差的个数为Max

{

//MyBreak("给定的搜索次数过小，请增大搜索次数，重新开始搜索。");

fclose(resultfp);

exit(0);

}

fprintf(resultfp,"\n第%d次搜索:",Gn+1);

double Flimit=MyProb.re\_F(1,r-(Gn+1),1.0-Arfa);//计算粗差检验的限差值,r 多余观测数， Gn在一定程度上代表了经检验确定含粗差的观测值的个数（L1） 1 代表L2

new\_Omg=0.0;

for(int i=0; i<3\*Vnumber; i++)//3\*Vnumber 观测值总数

{

Jint[Gn]=i;//将每个观测值依次定为可疑观测值（L2）

if(W[i]) continue; // 检查号观测值是否为已知粗差观测值

double Omg = CaQg(Gn+1,Jint,gross,Qg,JTPA,JTPV); //粗差二次型

if(Omg<0.0)continue;

if(Omg>new\_Omg && Omg<VTPV)

{

new\_Omg=Omg;

maxp=i;

}

}//循环之后可以得到最大的粗差二次型，其实也就可以计算最大的粗差统计量

double F=(new\_Omg-last\_Omg)/(VTPV-new\_Omg)\*(r-(Gn+1));//第一次搜索结束后得到最大的粗差统计量此时last\_omg=0.0

//第二次、第三次搜索结束后最大的粗差统计量 （此时last\_Omg不为0.）

if(F>Flimit)fprintf(resultfp,"\n F=%6.3lf(>%6.3lf)",F,Flimit); //最大的粗差统计量超限

else fprintf(resultfp,"\n F=%6.3lf(<%6.3lf)",F,Flimit);

Jint[Gn]=maxp;//更新，代表第几次搜索得到的最大统计量的观测值所在的位置

double Omg=CaQg(Gn+1,Jint,gross,Qg,JTPA,JTPV);//更新，计算粗差的大小和Qg

for(int i=0;i<Gn+1;i++)

{

int j=Jint[i];

fprintf(resultfp,"\n No.%-2d",j);

fprintf(resultfp," 向量组:%-3d ",AreaNumber[j/3]+1);

fprintf(resultfp,"%5s ",Pname[dir1[j/3]]);

fprintf(resultfp,"%5s ",Pname[dir2[j/3]]);

fprintf(resultfp," (%c)",'X'+j%3);

fprintf(resultfp,"%12.4lf",gross[i]);

}

if( F<=Flimit || fabs(gross[Gn])<0.01)break;//粗差搜索结束条件

Gn++;

last\_Omg=new\_Omg;

W[maxp]=1;//将这个观测值定为粗差，标识为1.

}

if(Gn==0)

{

fprintf(resultfp,"\n\n粗差探测结果：未发现粗差");

}

else

{

for(int j=1;j<=Gn;j++)

{

new\_Omg=CaQg(j,Jint,gross,Qg,JTPA,JTPV);

}

fprintf(resultfp,"\n\n 逐个搜索法最后定位:(粗差改正)\n");

fprintf(resultfp," 粗差总数: %d\n",Gn);

for(int i=0;i<Gn;i++)

{

int j=Jint[i];

fprintf(resultfp,"\n No.%-2d",j);

fprintf(resultfp," 向量组:%-3d ",AreaNumber[j/3]+1);

fprintf(resultfp,"%5s ",Pname[dir1[j/3]]);

fprintf(resultfp,"%5s ",Pname[dir2[j/3]]);

fprintf(resultfp," (%c)",'X'+j%3);

fprintf(resultfp,"%12.4lf",gross[i]);

}

fprintf(resultfp,"\n\n剔除粗差后最小二乘平差结果");

VTPV=VTPV-new\_Omg;

Sigma=sqrt(VTPV/(r-Gn));

Ca\_X\_Qx(gross,Qg,JTPA,Gn);

PrintXYZ();

CaLV(V);//计算残差

PrintLV("V",V);

fprintf(resultfp,"\n\n粗差探测:\n[pvv]=%e",VTPV);

fprintf(resultfp,"\n单位权中误差μ=±%e",Sigma);

}

delete []JTPA;

delete []JTPV;

delete []Qg;

delete []gross;

delete []Jint;

delete []W;

}

## 算例分析

数据采用程序中自带的数据，在向量组4 D E (Y) 加入2m的粗差；在向量组2 B F (X) 加入1m的粗差；在向量组1 E D (Z) 加入0.5m的粗差。

生成结果如下：

逐个搜索法粗差探测:

第1次搜索:

F=144.504(>12.714)

No.28 向量组:4 D E (Y) 1.9410

第2次搜索:

F=183.684(>12.771)

No.28 向量组:4 D E (Y) 1.9281

No.9 向量组:2 B F (X) 0.9964

第3次搜索:

F=2426.765(>12.832)

No.28 向量组:4 D E (Y) 1.9779

No.9 向量组:2 B F (X) 0.9991

No.2 向量组:1 E D (Z) 0.4862

第4次搜索:

F=13.212(>12.896)

No.28 向量组:4 D E (Y) 1.9754

No.9 向量组:2 B F (X) 0.9996

No.2 向量组:1 E D (Z) 0.4817

No.41 向量组:5 F D (Z) -0.0402

第5次搜索:

F= 8.627(<12.965)

No.28 向量组:4 D E (Y) 1.9765

No.9 向量组:2 B F (X) 0.9934

No.2 向量组:1 E D (Z) 0.4816

No.41 向量组:5 F D (Z) -0.0407

No.3 向量组:1 E B (X) 0.0151

逐个搜索法最后定位:(粗差改正)

粗差总数: 4 共设置3个粗差，但探测出来4个。

No.28 向量组:4 D E (Y) 1.9754

No.9 向量组:2 B F (X) 0.9996

No.2 向量组:1 E D (Z) 0.4817

No.41 向量组:5 F D (Z) -0.0402

剔除粗差后最小二乘平差结果

STATION X Y Z RMS\_X RMS\_Y RMS\_Z

(m) (m) (m) (cm) (cm) (cm)

----------------------------------------------------------------------------

1 A -2703200.7813 4678971.0170 3376976.8039 0.00 0.00 0.00

2 B -2700177.6193 4681527.9308 3375839.5977 0.26 1.20 0.90

3 C -2700962.6500 4677908.5662 3380169.5958 0.38 1.57 1.14

4 D -2699182.6828 4680027.0304 3378669.1654 0.29 1.28 0.97

5 E -2696225.5305 4683062.6388 3376832.8181 0.00 0.00 0.00

6 F -2697018.6628 4684079.9521 3374834.1728 0.51 1.64 1.20

Max(RMS\_X)= 0.51

Max(RMS\_Y)= 1.64

Max(RMS\_Z)= 1.20

===== V =====

向量组 点 名 点 名 Vx Vy Vz

1 E D 0.0103 -0.0287 -0.4963

1 E B -0.0093 -0.0051 0.0006

1 E F -0.0009 0.0086 0.0159

2 B F -1.0034 -0.0041 -0.0059

2 B E -0.0037 -0.0049 -0.0026

2 B D -0.0091 -0.0008 -0.0021

3 C A -0.0361 -0.0188 0.0136

3 C B 0.0067 -0.0003 -0.0030

3 C D -0.0038 -0.0107 0.0003

4 D E -0.0012 -1.9823 -0.0057

4 D B -0.0023 0.0028 0.0001

4 D C -0.0030 -0.0093 -0.0033

5 F E 0.0094 0.0060 -0.0013

5 F D 0.0234 -0.0206 0.0308

5 F B 0.0136 -0.0228 -0.0071

6 A B 0.0242 -0.0287 -0.0058

6 A C 0.0292 0.0224 0.0136

粗差探测:

[pvv]=2.965990e+01

单位权中误差μ=±9.205573e-01

与直接采用最小二乘的精度进行比较明显提高。（直接采用最小二乘[pvv]=8.006343e+04单位权中误差μ=±4.530903e+01）

测试1：

向量组1 E D (Y) 2m

向量组3 C B (X) -1m

向量组 5 F B(Z) 0.5m

*逐个搜索法最后定位:(粗差改正)*

*粗差总数: 3*

*No.1 向量组:1 E D (Y) 2.0194*

*No.21 向量组:3 C B (X) -1.0128*

*No.44 向量组:5 F B (Z) 0.5069*

测试2：

向量组1 E D (Y) 0.5m

向量组3 C B (X) -1m

向量组 5 F B(Z) 2m

*逐个搜索法最后定位:(粗差改正)*

*粗差总数: 3*

*No.44 向量组:5 F B (Z) 2.0069*

*No.21 向量组:3 C B (X) -1.0128*

*No.1 向量组:1 E D (Y) 0.5194*

# 总结与分析：

1. 设置了3个粗差，探测出来4个？

与这行代码设置的值0.01有关。

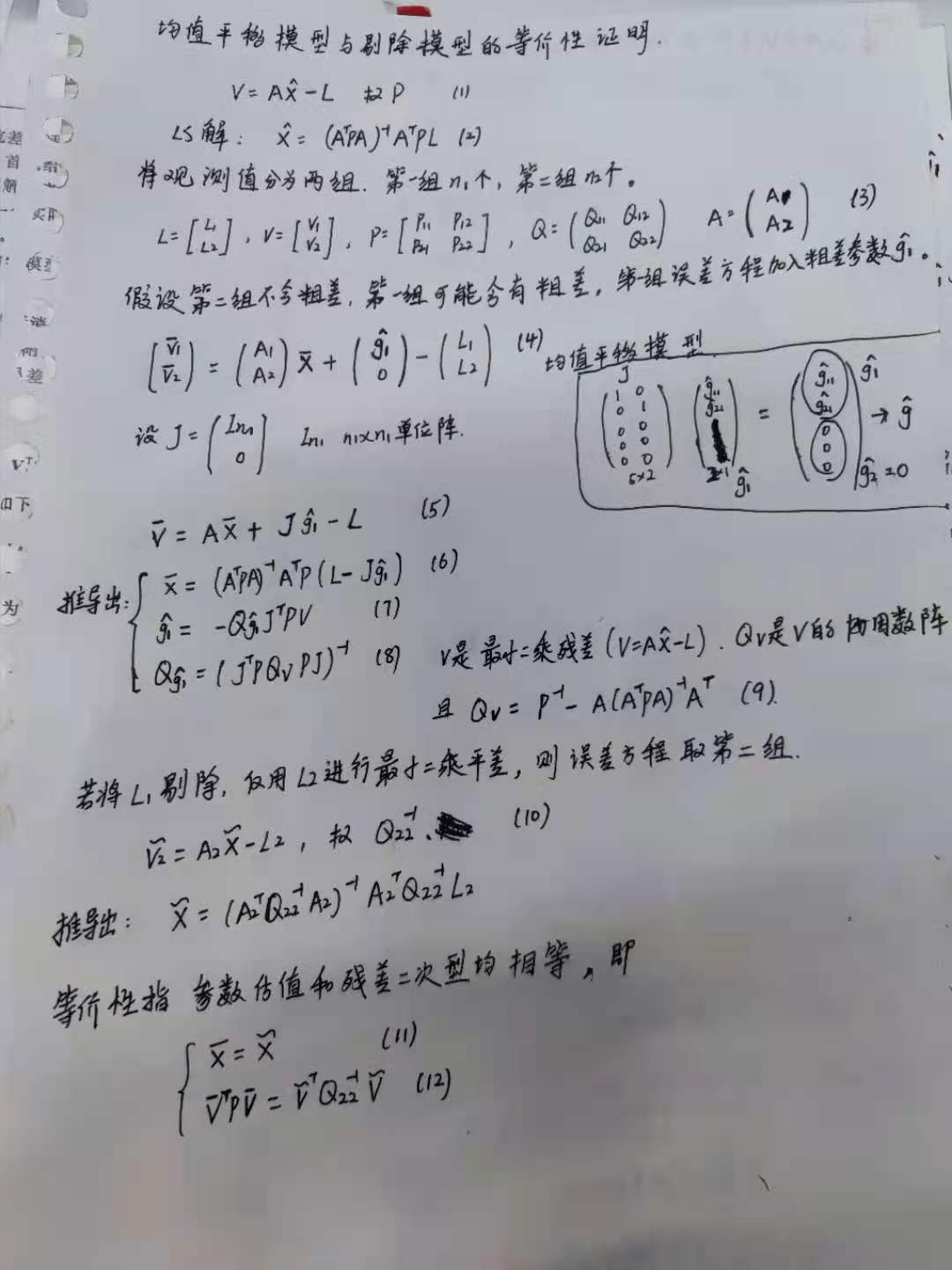
if( F<=Flimit || fabs(gross[Gn])<0.01)break;//粗差搜索结束条件

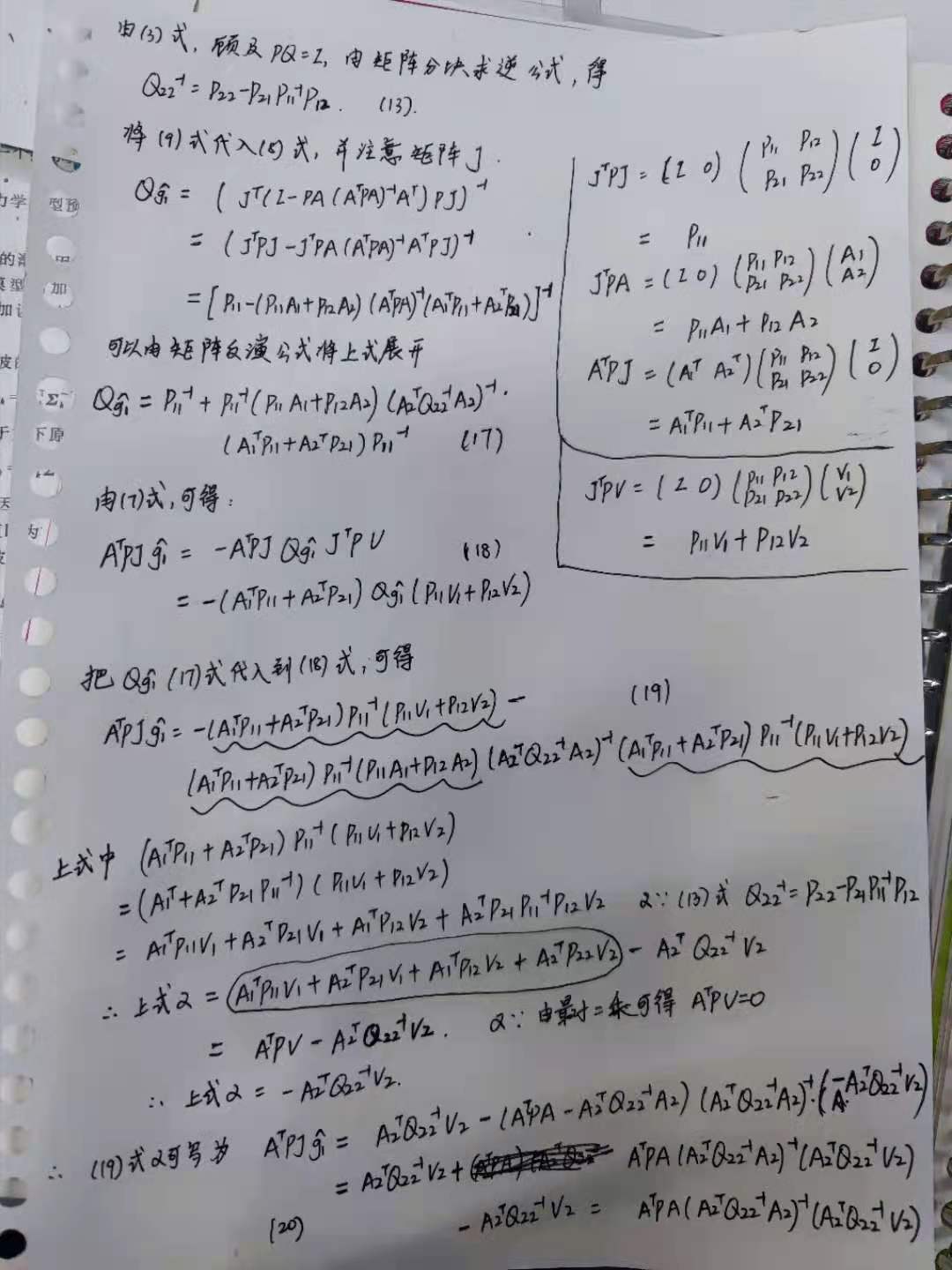
可是能不能从理论上分析一下呢？

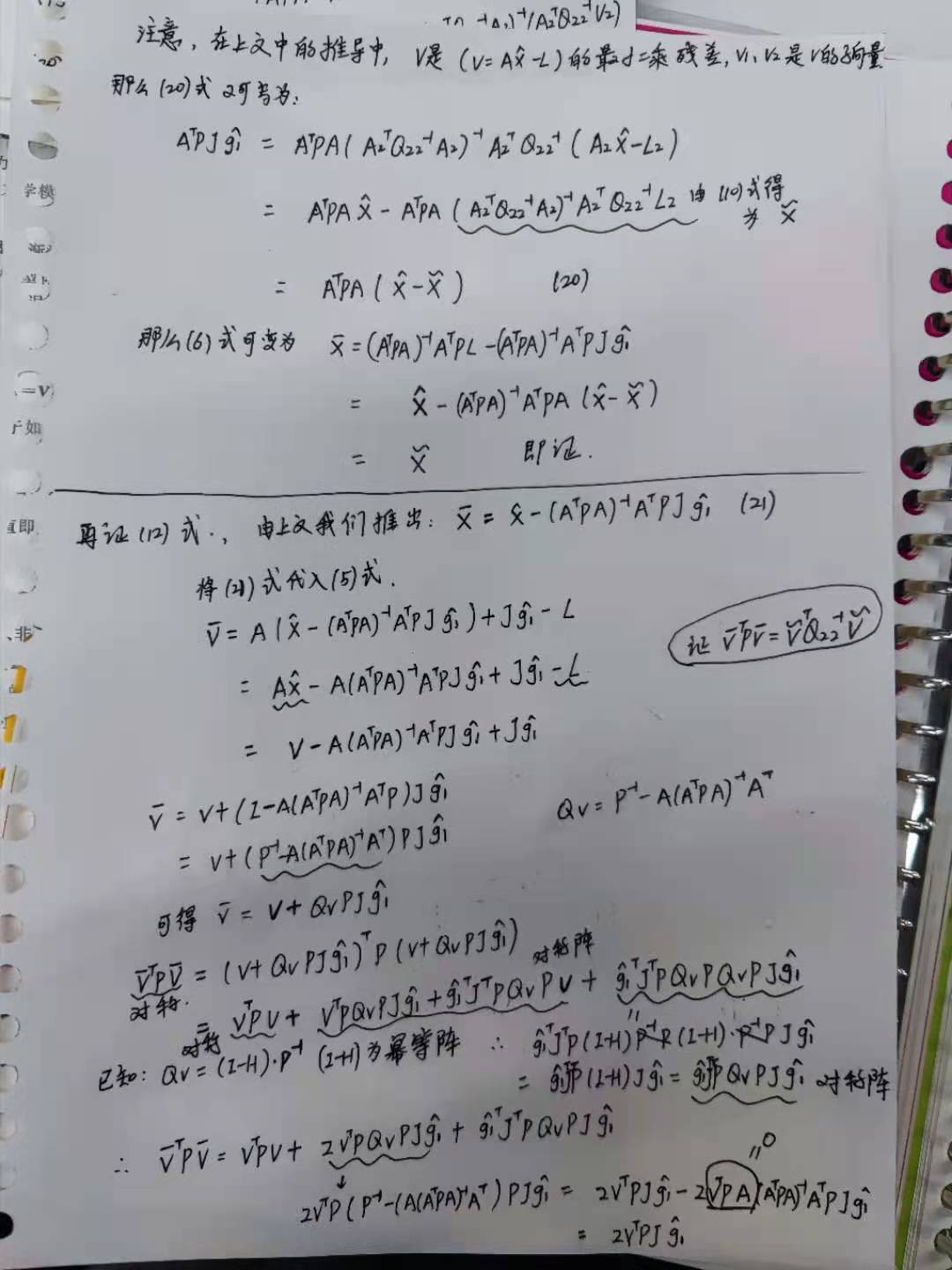
2. 探测到的粗差怎么才能更精确？

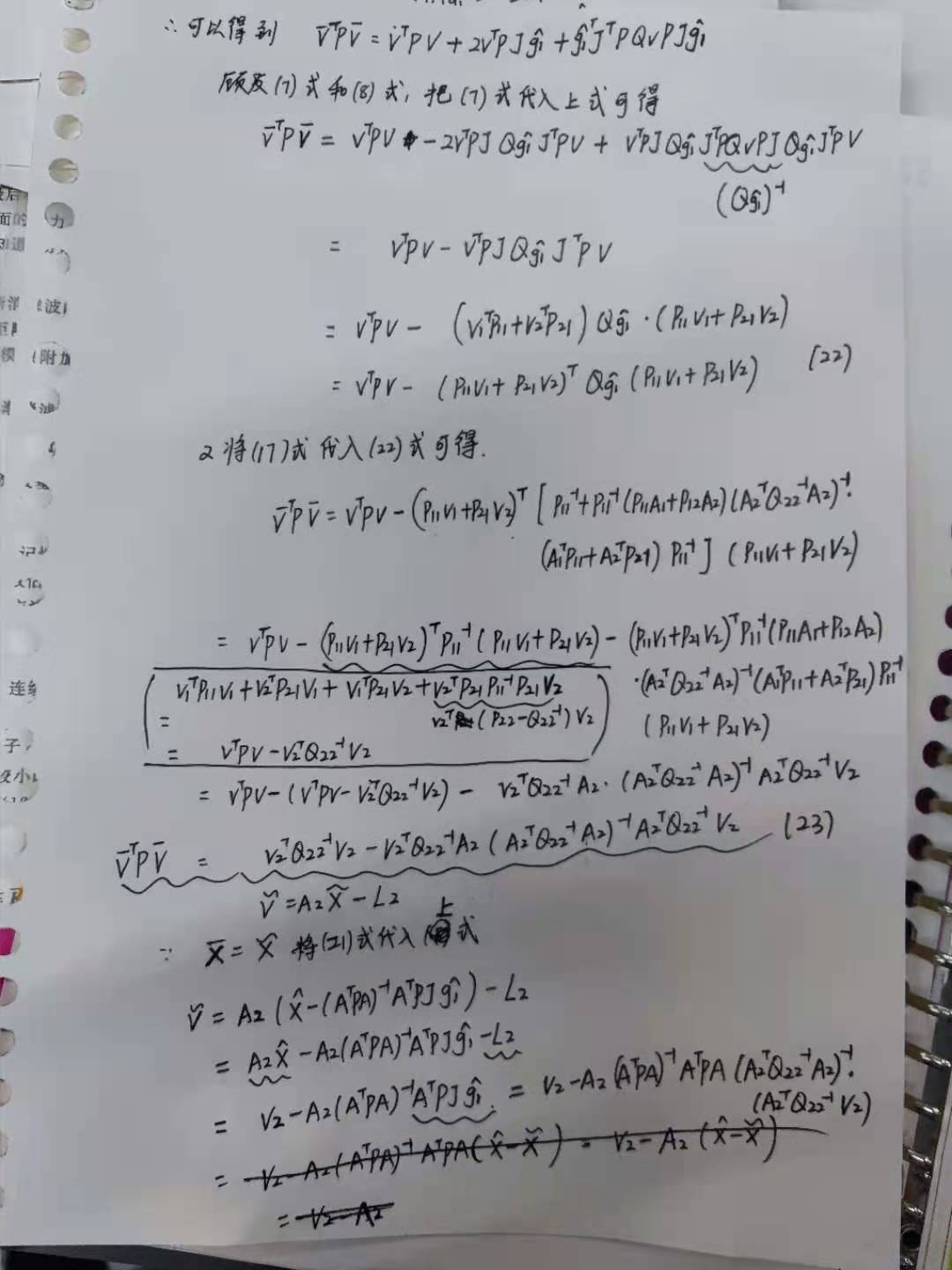
3. 剔除粗差与均值平移模型是等价的，还是不怎么理解。

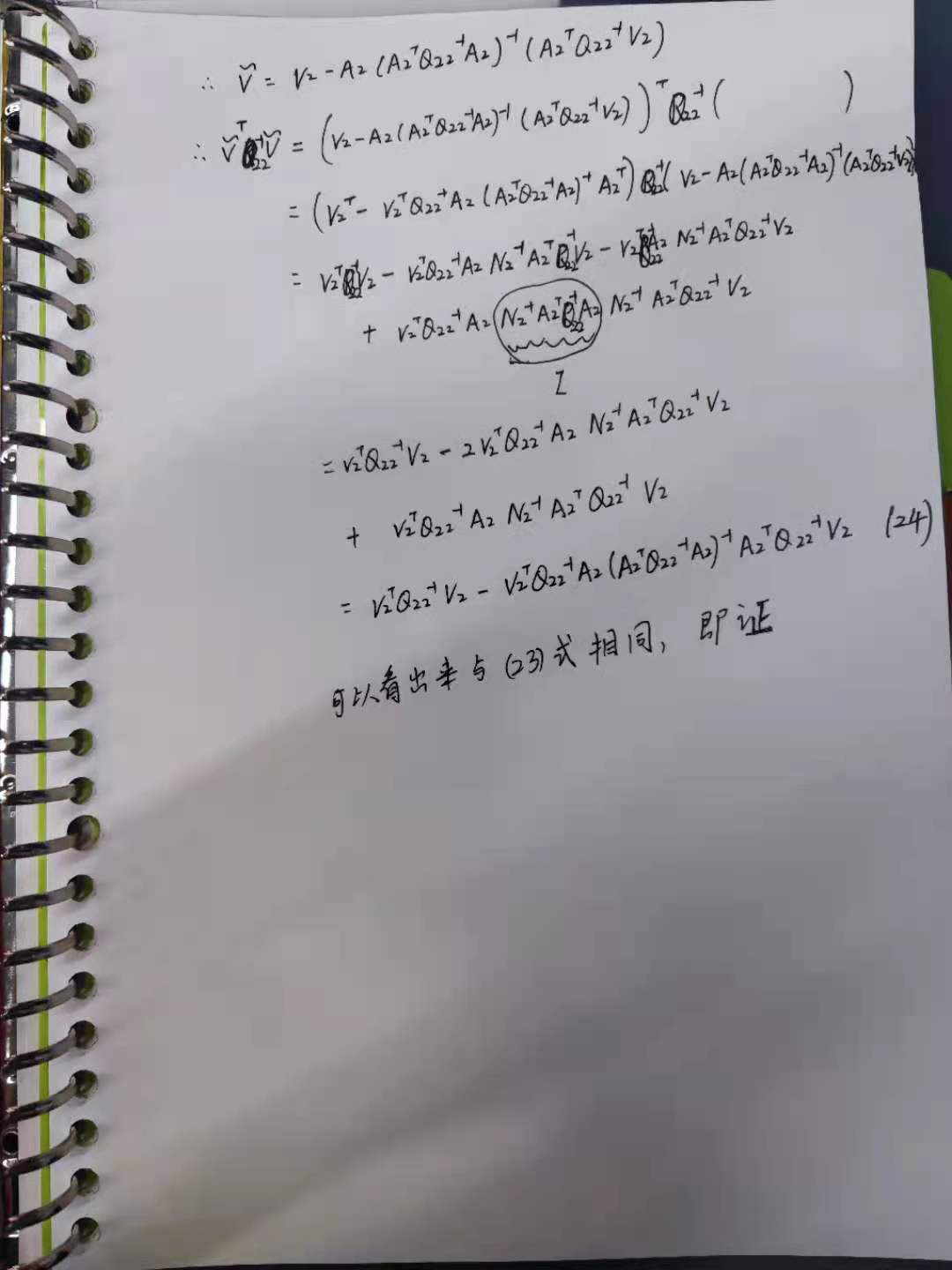
读宋力杰教授《均值平移模型的理论研究及分析》有感











1. 经过推导，我们可以得到均值平移模型和剔除粗差模型的参数解、二次型都相同，那么在采用均值平移模型进行粗差探测时，我们可以直接采用均值平移模型的参数解进行求解，不需要将粗差剔除观测值剔除后在重新平差。

2. 均值平移模型和剔除模型的验后单位权方差估值都可表示为：

