

郭九训 编著

控制网平差程序设计

KONG ZHI WANG PING CHA CHENG XU SHE JI

原子能出版社

图书在版编目(CIP)数据

控制网平差程序设计/郭九训编著. —北京:原子能出版社, 2004. 8

ISBN 7-5022-3269-9

I. 控… II. 郭… III. 控制网平差-应用软件-程序设计 IV. P207

中国版本图书馆 CIP 数据核字(2004)第 090343 号

控制网平差程序设计

出版发行 原子能出版社(北京市海淀区阜成路 43 号 100037)

责任编辑 谭 俊

责任校对 李建慧

责任印制 丁怀兰

印 刷 保定市印刷厂

开 本 850mm×1168mm 1/32

印 张 5.75

字 数 155 千字

版 次 2004 年 9 月第 1 版 2004 年 9 月第 1 次印刷

书 号 ISBN 7-5022-3269-9/P·1

经 销 全国新华书店

印 数 1—800

定 价 18.00 元

版权所有 侵权必究

网址: <http://www.aep.com.cn/>

前 言

本书是《测量平差学》、《控制测量学》的后续教材,主要介绍常规的测量控制网平差程序设计的基本方法、编程时技术处理的某些办法、一些常用的过程以及几个有代表性的又有较大实用价值的程序。其目的是让读者在已有 VB 语言的基础上,增加控制网平差计算程序的数据结构、编程方法和技巧等方面的知识,以期日后在程序使用、编制和开发等方面有所帮助。

20 世纪 80 年代电子计算机逐步开始在测量上普及使用,经历了 PC-1500 机和 PC-E500 机的 BASIC 语言、QB 语言和现在的 VB 语言等几个主要阶段,广大测量工作者开发了许多实用的程序,为减轻测量工作的繁重劳动,提高工作效率做出了巨大贡献。本书介绍的程序是在东华理工学院使用多年讲义的基础上,参考有关资料全部用 VB6.0 语言改编而成。在此,还要感谢东华理工学院教材委员会对本教材的肯定,并对该教材公开出版给予资助。

由于作者水平有限,错误及不当之处在所难免。在程

序设计语言迅猛发展的当今,尽管作者力求使用语言简洁合理,便于阅读,但也未必尽如人意。恳请专家、学者不吝赐教。

编著者
2004 年 3 月

目 录

第一章 控制网程序设计中的几个问题	1
1.1 加法器与计数器	1
1.2 角度的度分秒制与度制的相互化算	3
1.3 关于 For 循环	4
1.4 由坐标计算坐标方位角	9
1.5 近似坐标的计算.....	13
1.6 测角网中极条件的检验.....	21
1.7 测边网中圆周角(组合角)条件的检验.....	30
第二章 法方程式的解算及系数阵的求逆	37
2.1 高斯约化法.....	37
2.2 平方根法求法方程式系数阵的逆.....	43
2.3 改正的平方根法求法方程式系数阵的逆.....	46
2.4 加边求逆.....	49
2.5 法方程式系数阵的一维存放、准二维存放方法	52
第三章 控制网平差程序设计的主要方法	60
3.1 平差方法的选择.....	60
3.2 附有条件的间接观测平差误差方程式、条件方程式逐次 组成法方程式.....	61
3.3 精度评定.....	65
第四章 几个应用程序	70
4.1 高斯投影计算.....	70
4.2 单一附(闭)合导线严密平差程序.....	79
4.3 测边大地四边形平差程序.....	90

4.4	测边中点多边形平差程序	101
4.5	水准网平差程序	115
4.6	平面控制网平差程序	129
4.7	三角高程网平差程序	165
主要参考文献		178

第一章 控制网程序设计中的几个问题

1.1 加法器与计数器

1.1.1 加法程序与加法器

加法程序是程序设计中最简单,也是最常用的一个程序单元。其功能是利用两个工作变量来计算一系列数据之和。其中一个变量(设为 t)依次存放待累加的每个数据,另一个变量(设为 s)存放每次累加的结果,当程序运行结束时,其值即为所求一系列数据之和。通常计算的都是有限个数据之和,因而多用 for-next 循环语句进行累加。

例[1-1]现列出一个加法程序如下,其中 n 存放累加的数据的个数, t 和 s 的含义如上所述。

```
Input n
s=0
For i=1 To n
Input t
s=s+t
Next i
```

程序相应的框图如图 1-1 所示。

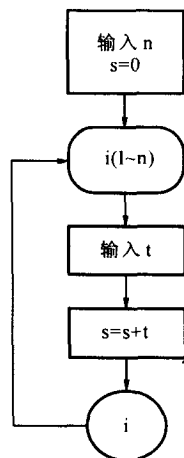


图 1-1

习惯称上述这种程序为“加法程序”,称求和变量(即上述程序中的 s)为“加法器”。加法程序经常同其他功能的语句配合使用,共用一个 for-next 循环语句。在测量平差中,计算一组等权直接观测值的算术平均值,计算带权平均值都用到了加法程序。

1.1.2 计数器

计数程序是加法程序的特殊情况,即当例[1-1]的程序中变量 t 取值恒为 1 的时候。在这种情况下, s 的值与循环变量 i 的值相同,因而没有必要使用变量 s ,而仅取循环变量 i 即可。变量 i 起到了计数作用,也是计数器。这时计数程序即化简为单纯的 for-next 循环语句:

```
For i=1 to n
Next i
```

变量 i 从 1 开始每循环一次加 1,直到 n 为止,起到计数器的作用。

在 for 循环中也有取另外的变量作为计数器的情况,可称为“条件计数器”。下例就是一个应用“条件计数器”的例子。

例[1-2]计算测方向网中各三角点起始方向的序号并存放在数组 q 中。

将测方向网中的观测方向按照三角点顺序和各三角点上的观测顺序编号并存放于一维数组 r 中。以图 1-2 为例,测站 1 的起始方向为 1,测站 2 的起始方向为 4,测站 3 的起始方向为 8,测站 4 的起始方向为 12,测站 5 的起始方向为 16。由于各测站的起始方向的方向值都为 0,根据这一特性,可以很方便地求出各测站起始方向的序号并存放在数组 q 中。

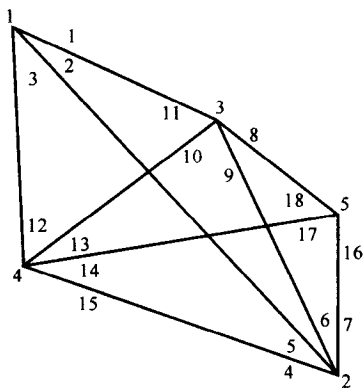


图 1-2

```
q(1)=1
j=1
For i=2 to m2
If r(i)=0 Then j=j+1:q(j)=i
```

Next i

$q(z+1)=m2+1$

上段程序中 $m2$ 为方向的最大编号, z 为三角点的最大编号。上段程序中只有当 $\text{If } r(i)=0$ 时,才执行 $j=j+1; q(j)=i$, 因而称为条件计数器。有了上段程序的最后一句 $q(z+1)=m2+1$, 可以很方便地求得 k 号三角点的观测方向序号, 它们是 $q(k) \sim q(k+1)-1$ 。因为凡是涉及有关方向值的计算都必须知道该方向的序号, 才能准确无误的找到, 所以该数组在平差程序设计中是个很重要的图形信息。

1.2 角度的度分秒制与度制的相互化算

方向和角度习惯采用全圆周为 360° 和度以下的 60 等分制。对 60 等分制数值进行四则运算是不方便的。PC-1500 机有 DEG 和 DMS 标准函数进行角度的度分秒制与度制的相互化算, 在 PC-E500 机中把 DEG 和 DMS 称为命令, 其功能也是进行角度的度分秒制与度制的相互化算。QB 和 VB 语言都没有这两个函数(或命令), 因此, 一般须编两个函数过程(也可以是子过程)来实现角度的度分秒制与度制的相互化算。

度分秒制化算为度制的函数过程:

```
Public Function deg(xx)
xx=xx+0.0000000001
deg1=Fix(xx)
deg2=(xx - deg1) * 100
deg3=Fix(deg2)
deg4=(deg2 - deg3) * 100
deg=deg1 + deg3/60 + deg4 /3600
End Function
```

度制化算为度分秒制的函数过程:

```

Public Function dms(xx)
xx=xx+0.0000000001
dms1=Fix(xx)
dms2=(xx-dms1)*60
dms3=Fix(dms2)
dms4=(dms2-dms3)*60
dms=dms1+dms3/100+dms4/10000
End Function

```

QB 和 VB 语言在进行三角函数和反三角函数计算时采用弧度制,为此还须进行度制与弧度制相互化算。一般设立固定常数 $\rho^\circ = 57.2957795130823$ 。将度制的角值除以 ρ° ,就变为以弧度为单位的角值;将弧度制的角值乘以 ρ° ,就变为以度为单位的角值。这样在程序的整个流程中,角度的单位是度制或度分秒制,符合人们的习惯,便于阅读。上两段程序段中,Fix 是取整(又叫截取)函数,用 Fix 函数可避免负角转换中的不必要的麻烦,比用 Int 函数要方便简单。

1.3 关于 For 循环

循环是程序设计中常用的基本语句之一。PC-1500 和 PC-E500 的 BASIC 语言,以及 QB 和 VB 中都有 For—Next 循环,在使用时,除应遵守一般 BASIC 书籍中提到的一些规定之外,还应注意下面讨论的问题。20 世纪 80 年代以来,广大测量工作者开发了许多适用于 PC-1500 机的测量程序,在将这些程序改写成 VB 语言时也要注意这一问题。

1.3.1 两类 For 循环

For-Next 循环有两类:

第一类:语句形式为

For i=a To b Step c

循环体

Next i

其框图如图 1-3 所示。由图知,它的特点是先执行循环体,后判断是否满足结束循环的条件,可简称为“先执行,后判断”。

第二类:语句形式同上,其框图如图 1-4 所示。它的特点是先判断是否满足结束循环的条件,后执行循环体,可简称为“先判断,后执行”。

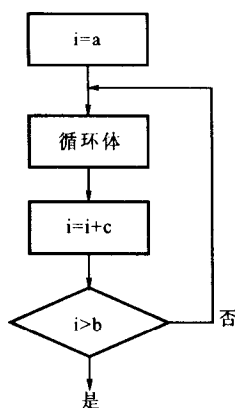


图 1-3

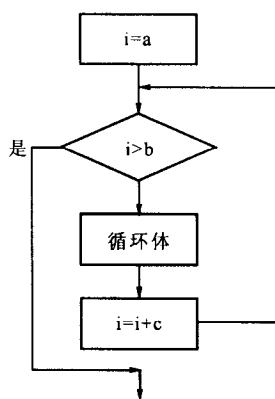


图 1-4

在一般情况下用这两种循环所得的结果是一样的,只是在循环的初值一开始就超过终值的时候(即 a 超过 b 时),第一类循环仍会执行一次循环体;而第二类循环则自动跳过循环体,一次也不执行。在这种情况下,第一类循环可能会产生程序设计者意想不到的错误结果。请看下例。

例[1-3]以下是一个简化了的输入平面控制网观测值的程序段:

```

100 input"观测方向数";n
110 input"观测边数";ns
120 dim l(n),s(ns)
130 for i=1 to n
140 print"第"; i ; "个方向观测值";
150 input l(i):cls
160 next i
170 for i=1 to ns
180 print "第"; i ; "条边观测值";
190 input s(i):cls
200 next i
210

```

先说明一下,行号对于 PC-1500 机和 PC-E500 机的 BASIC 语言是必需的,而对于 QB 和 VB 语言则可有可无,我们为解释方便打上行号。PC-1500 机和 PC-E500 机的 BASIC 语言以及 QB 语言关键字都是大写,而 VB 语言关键字只是第一个字是大写,其余都是小写。程序的前两行是根据网的实际情况输入观测方向数 n 和观测边数 ns ,如果网中没有观测方向(完全测边网)则 $n=0$;如果网中没有观测边(完全测角网)则 $ns=0$ 。第 130 行至 160 行为输入方向观测值的程序段,第 170 行至 200 行为输入边观测值的程序段。

如将程序段用于完全测角网,则有 $ns=0$,在执行到 170 行时,for 循环的初值 1 大于终值 0,对于第二类 for 循环,程序会自动跳过循环体,执行 210 行。但对于第一类 for 循环,程序仍会执行一次循环体,即要求输入第一条边的观测值,因而与网的实际情况发生矛盾。同样,如将程序段用于完全测边网,则有 $n=0$,在执行到 130 行时,for 循环的初值 1 大于终值 0,对于第二类 for 循环,程序会自动跳过循环体,执行 170 行。但对于第一类 for 循环,程序仍会执行一次循环体,即要求输入第一个方向的观测值,因而与网的实际情况发生矛盾。

为解决第一类 for 循环的这一问题,可在 for 语句前加上条件转移语句,来控制是否执行循环体。对于上例,可在 120 和 130 行之间加入 125 if n=0 goto 170,在 160 和 170 行之间加入 165 if ns=0 goto 210 即可。由于较多的使用 goto 语句,编出的程序结构不清晰,可读性差。结构化程序设计中要求尽可能少用或不用 goto 语句。如果将上面程序段作如下的修改会显得完善些。

```
input"观测方向数";n
input"观测边数";ns
dim l(n),s(ns)
if n>0 then
    for i=1 to n
        print"第";i;"个方向观测值";
        input l(i):cls
    next i
end if
if ns>0 then
    for i=1 to ns
        print "第";i;"条边观测值";
        input s(i):cls
    next i
end if
.....
```

1.3.2 多重循环中内循环与外循环的互换

例[1-4]变量 a 取值 1 至 10,间隔 1,变量 b 取值 11 至 13,间隔 1,对 a,b 的所有组合计算 $\sin(a)$ 、 $\cos(b)$ 的值。

程序的主要段如下:

```
20 for a=1 to 10
30 w=sin(a)
40 for b=11 to 13
```

```
50 y=w * cos(b)
```

```
60 print y
```

```
70 next b
```

```
80 next a
```

执行上面程序的过程,对 a 赋初值 1 次,加步长 10 次,检测 10 次,计算 $\sin(a)$ 10 次;对 b 赋初值 $10 * 1$ 次,加步长 $10 * 3$ 次,检测 $10 * 3$ 次,计算 $\cos(b)$ $10 * 3$ 次。总计给循环变量赋初值 11 次,加步长 40 次,检测 40 次,计算正弦和余弦共 40 次。若将程序改写成:

```
20 for b=11 to 13
```

```
30 w=cos(b)
```

```
40 for a=1 to 10
```

```
50 y=w * sin(a)
```

```
60 print y
```

```
70 next a
```

```
80 next b
```

则运算过程中总计给循环变量赋初值 4 次,加步长 33 次,检测 33 次,计算正弦和余弦共 33 次,可见修改后的程序执行效率较高。因此,对嵌套的 for 循环,若可能的话,应将重复次数少的循环作为外循环,这样可提高循环语句的效率。再看矩阵乘积的例子。

例[1-5]计算矩阵乘积 $T = A \cdot S$
 $2 \times 3 \quad 2 \times 2 \quad 2 \times 3$

程序的主要部分如下:

```
110 for i=1 to 2
```

```
120 for j=1 to 3
```

```
130 for k=1 to 2
```

```
140 t(i,j)=t(i,j)+a(i,k) * s(k,j)
```

```
150 next k
```

```
160 next j
```

```
170 next i
```

这是大家习惯的按矩阵 T 的行、列顺序计算其中的元素。也可

将该程序的 i, j 循环互换, 即 j 循环在外, i 循环在内, 这是按矩阵 T 的列、行顺序计算其中的元素(注意, 相应的 next i, next j 语句也应互换, 下同)。除此之外, 还可将 k 循环作为最外层循环, 而将 i(或 j) 循环放在第二层, 将 j(或 i) 循环放在最内层, 则程序变为:

```

110 for k=1 to 2
120 for i=1 to 2
130 for j=1 to 3
140 t(i,j)=t(i,j)+a(i,k)*s(k,j)
150 next j
160 next i
170 next k

```

前面已经讲过, 研究多重循环中内循环与外循环互换的问题是为了提高循环语句的效率。现在计算机的计算速度已很快, 对于测量平差计算, 这个问题已不十分重要。在测量控制网平差程序设计中, 利用间接平差中法方程具有可加性的特征, 由 1 个误差方程式及其权即可组成与其相应的部分法方程式系数和常数项, 把部分的法方程式系数和常数项逐个累加起来就是总体法方程。这时用到的方法与例[1-5]类同。

1.4 由坐标计算坐标方位角

控制网的概算与平差经常要计算边长和坐标方位角, 为计算边长和坐标方位角可根据不同的情况采用不同的方法。控制网程序设计一般采用由坐标计算边长和坐标方位角的方法, 将它编制成一个子过程便于调用。

由两点的坐标计算边长的公式

$$S_{12} = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (1-1)$$

由坐标计算坐标方位角可分别采用反正弦反余弦和反正切两种方法。

1.4.1 采用反正弦和反余弦

由坐标计算坐标方位角

如坐标方位角值在第一象限, 如图 1-5, 对于 P_1 点, 由于 $\Delta x > \Delta y$, 用反余弦

$$A = \cos^{-1} \left(\frac{\Delta x}{S} \right) = \cos^{-1} \left(\frac{|\Delta x|}{S} \right) \quad (1-2)$$

对于 P_2 点, 由于 $\Delta x < \Delta y$, 用反正弦

$$A = \sin^{-1} \left(\frac{\Delta y}{S} \right) \quad (1-3)$$

将(1-2)和(1-3)两式合写成

$$A = \begin{cases} \cos^{-1} \left(\frac{|\Delta x|}{S} \right) * \text{Sgn}(\Delta y) \\ \sin^{-1} \left(\frac{\Delta y}{S} \right) \end{cases} \quad (1-4)$$

如坐标方位角值在第二象限, 如图 1-6, 作 P 点的对称点 P' , 可得

$$A' = \begin{cases} \cos^{-1} \left(\frac{|\Delta x|}{S} \right) * \text{Sgn}(\Delta y) \\ \sin^{-1} \left(\frac{\Delta y}{S} \right) \end{cases} \quad (1-5)$$

$$A = 180^\circ - A' \quad (1-6)$$

如坐标方位角值在第三象限, 如图 1-7, 作 P 点的对称点 P' , 可得:

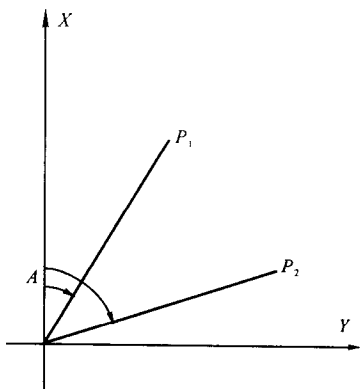


图 1-5

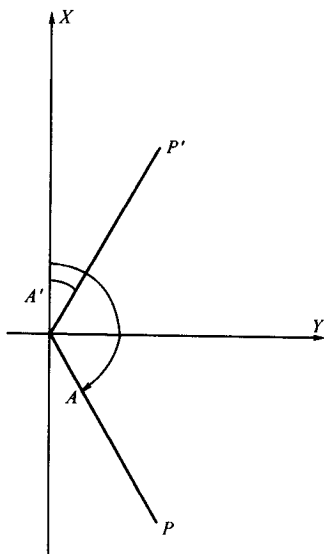


图 1-6

$$A' = \begin{cases} \cos^{-1} \left(\frac{|\Delta x|}{S} \right) * \text{Sgn}(\Delta y) \\ \sin^{-1} \left(\frac{\Delta y}{S} \right) \end{cases} \quad (1-7)$$

$$A = 180^\circ - A' \quad (1-8)$$

坐标方位角值在第二象限与在第三象限的计算公式完全一样,只是由(1-5)式计算所得的 A' 角是正数,而由(1-7)式计算所得的 A' 角是负数。

如坐标方位角值在第四象限,如图 1-8,

$$A' = \begin{cases} \cos^{-1} \left(\frac{|\Delta x|}{S} \right) * \text{Sgn}(\Delta y) \\ \sin^{-1} \left(\frac{\Delta y}{S} \right) \end{cases} \quad (1-9)$$

$$A = 360^\circ + A' \quad (1-10)$$

归纳(1-4)至(1-10)式,采用反正弦和反余弦由坐标计算边长和坐标方位角适用于 PC-1500 机的子程序如下:

```

1000 REM 求边长和坐标方位角子程序
1002 DX=X(J)-X(I)
1004 DY=Y(J)-Y(I)
1006 S=SQR(DX^2+DY^2)
1008 A=ABS(DX)
1010 T=ACS(A/S)*SGN(DY): IF A <= ABS

```

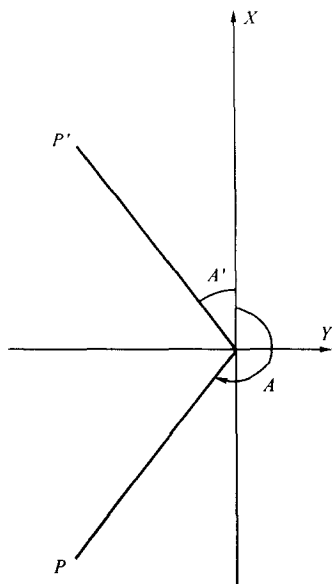


图 1-7

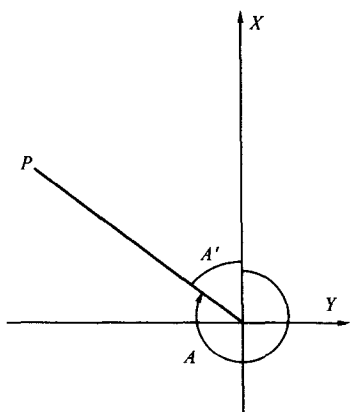


图 1-8