

放 点 方 法 二 则

中国航空工业勘察设计研究院 杨 杰

【提要】 本文提出在施工放点测量过程中,经常遇到的两已知控制点,相互间既不通视又不能架设经纬仪的一种扩展法。改进了传统测量方法的一种快速、准确的补充手段,为在工作中提供了方便。

【Abstract】 An extended method for setting out the two controlling points which are not in sight with each other during the setting out surveying is proposed. This method is a quick and precise complementary method. It provides convenient for surveying.

一、概述

在施工放点过程中,目前普遍使用的极坐标法、直角坐标法、交会法等传统方法都是要在控制点上架设经纬仪,这就必然会遇到不通视,控制点保存不完善等问题,因而在使用过程中就存在着一定的局限性。随着电磁波测距仪和微型电子计算机在我们测量工作中的普及使用,使我们可以充分有效地发挥其快速准确的优越性,为我们的工作提供方便。本文提出的方法可以做为一种方法,也可做为传统方法的一种补充手段,解决放点过程中的一些问题,同时精度也能满足要求。

二、任意点为测站, 两点后方交会放点

1. 原理

如图1所示,在任意点P架设电磁波测距仪(P点为任意选定、通视条件良好,便于用来放点的点)。1、2分别是实地上的两个点(其坐标 x_1, y_1, x_2, y_2 均已知。它们可以是控制点,也可以是已放过的点)。使用测距仪测得P点至1、2两点的距离 S_1, S_2 。为了避免 α 角的多值问题,在 $\triangle 1P2$ 中利用余弦定理得

$$\cos\alpha = \frac{S_1^2 + S_2^2 - S^2}{2S_1S_2} \quad (1)$$

$$\text{而 } \alpha_{1P} = \alpha_{12} \pm \alpha \quad (2)$$

(2)式中当P点在路线1, 2前进方向右边时取 $\alpha_{1P} = \alpha_{12} + \alpha$, 反之则 $\alpha_{1P} = \alpha_{12} - \alpha$ 。

由此可得P点坐标

$$\begin{cases} x_p = x_1 + S_1 \cos\alpha_{1P} \\ y_p = y_1 + S_1 \sin\alpha_{1P} \end{cases} \quad (3)$$

算出P点坐标后即可在P点上按传统方法

放点。

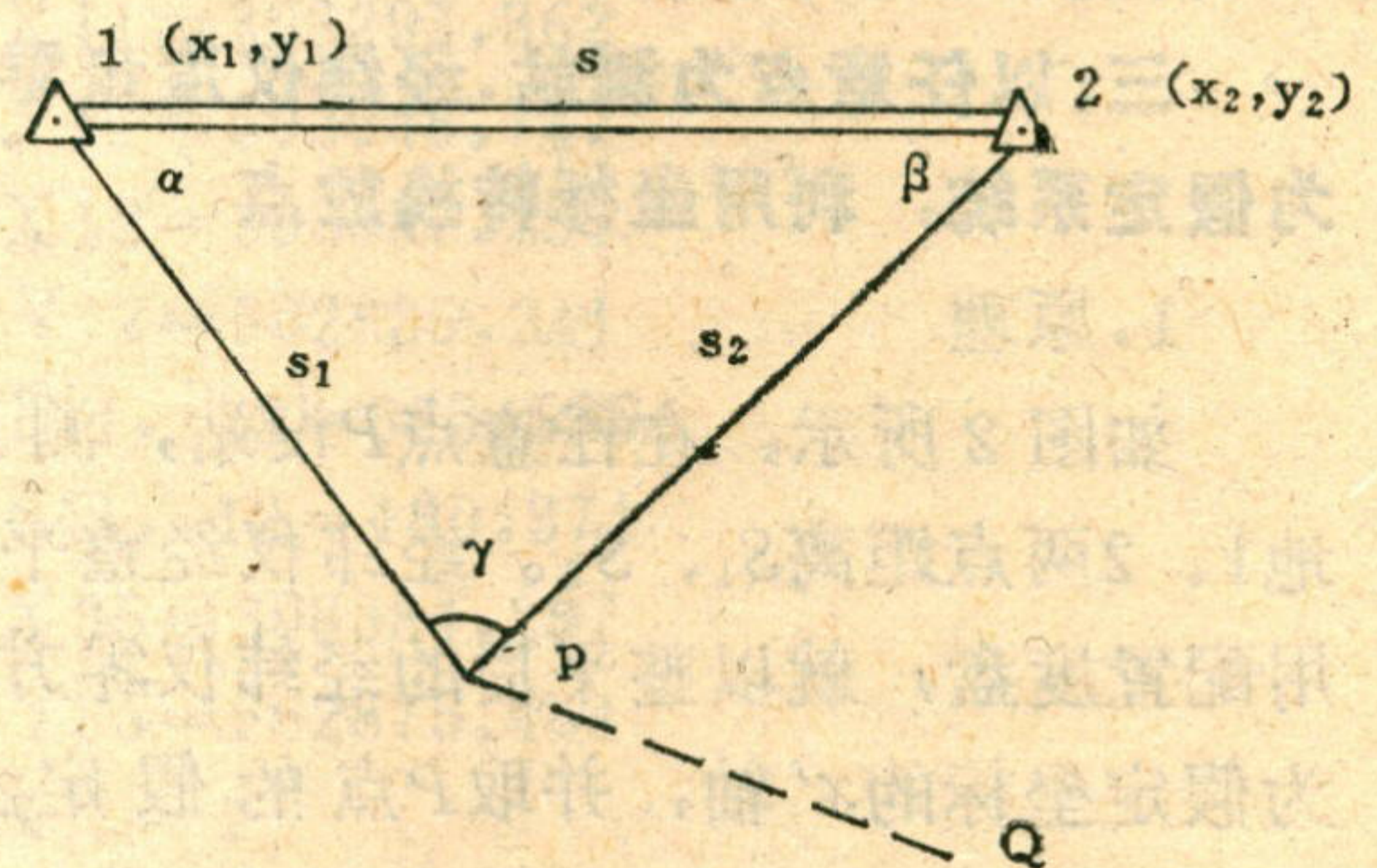


图 1

2. 任意点P点位精度分析

P点坐标是已知数据 x_1, y_1, x_2, y_2 及观测值 S_1, S_2 的函数

$$\begin{cases} (x_p - x_1)^2 + (y_p - y_1)^2 = S_1^2 \\ (x_p - x_2)^2 + (y_p - y_2)^2 = S_2^2 \end{cases} \quad (4)$$

不考虑起算数据误差的影响,对(4)式进行全微分并进行变换整理可得

P点的点位中误差公式:

$$m_p = \frac{\sqrt{2}}{\sin\gamma} m_s \quad (5)$$

式中 m_s 为边长测量中误差, 设 $m_s = m_{s1} = m_{s2}$, γ 为交会角。

从(5)式可看出, P点的点位中误差与 m_s 的大小成正比, 而与 γ 角的正弦函数成反比, m_s 愈大 m_p 愈大, 并且成 $\sqrt{2}$ 倍地增长, $\sin\gamma$ 愈小, m_p 愈大, $\sin\gamma$ 愈大, m_p 愈小, 当 $\sin\gamma = 1$, 即 $\gamma = 90^\circ$ 时, m_p 最好。由此可知, 要保证P点的精度, 首先要保证测距的精度, 其次要使 γ 角限制在一定的范围之内, 一般取 $30^\circ \leq \gamma \leq 120^\circ$ 。

3. 操作过程

①选择一通视良好的点为测站，整平仪器。

②测出P点至1、2点距离 S_1 、 S_2 ，按(1)、(2)、(3)式即可计算出P点坐标 x_p 、 y_p 。

③经纬仪后视1点，度盘配制为 $\alpha_{1p} + 180^\circ$ 或为 $\alpha_{1p} - 180^\circ$ 。

④计算P点至放样点的距离S和方位 α 。

⑤经纬仪方位拨至方位 α ，测距仪按S可放出我们要放的Q点。

三、以任意点为测站，经纬仪度盘零方位为假定系统，利用坐标转换放点

1. 原理

如图2所示，在任意点P设站，测至实地1、2两点距离 S_1 、 S_2 。经纬仪经整平后不用配置度盘，就以整平后的经纬仪零方位做为假定坐标的 x' 轴，并取P点的假定坐标 $x' = 0$ ， $y' = 0$ 。经纬仪瞄准1、2两点，测得1、2两点的方向值，即可得假定方位 β_1 、 β_2 。则可知1、2两点在假设坐标系内的坐标为：

$$\begin{cases} x'_1 = S_1 \cos \beta_1 \\ y'_1 = S_1 \sin \beta_1 \end{cases} \quad \begin{cases} x'_2 = S_2 \cos \beta_2 \\ y'_2 = S_2 \sin \beta_2 \end{cases} \quad (6)$$

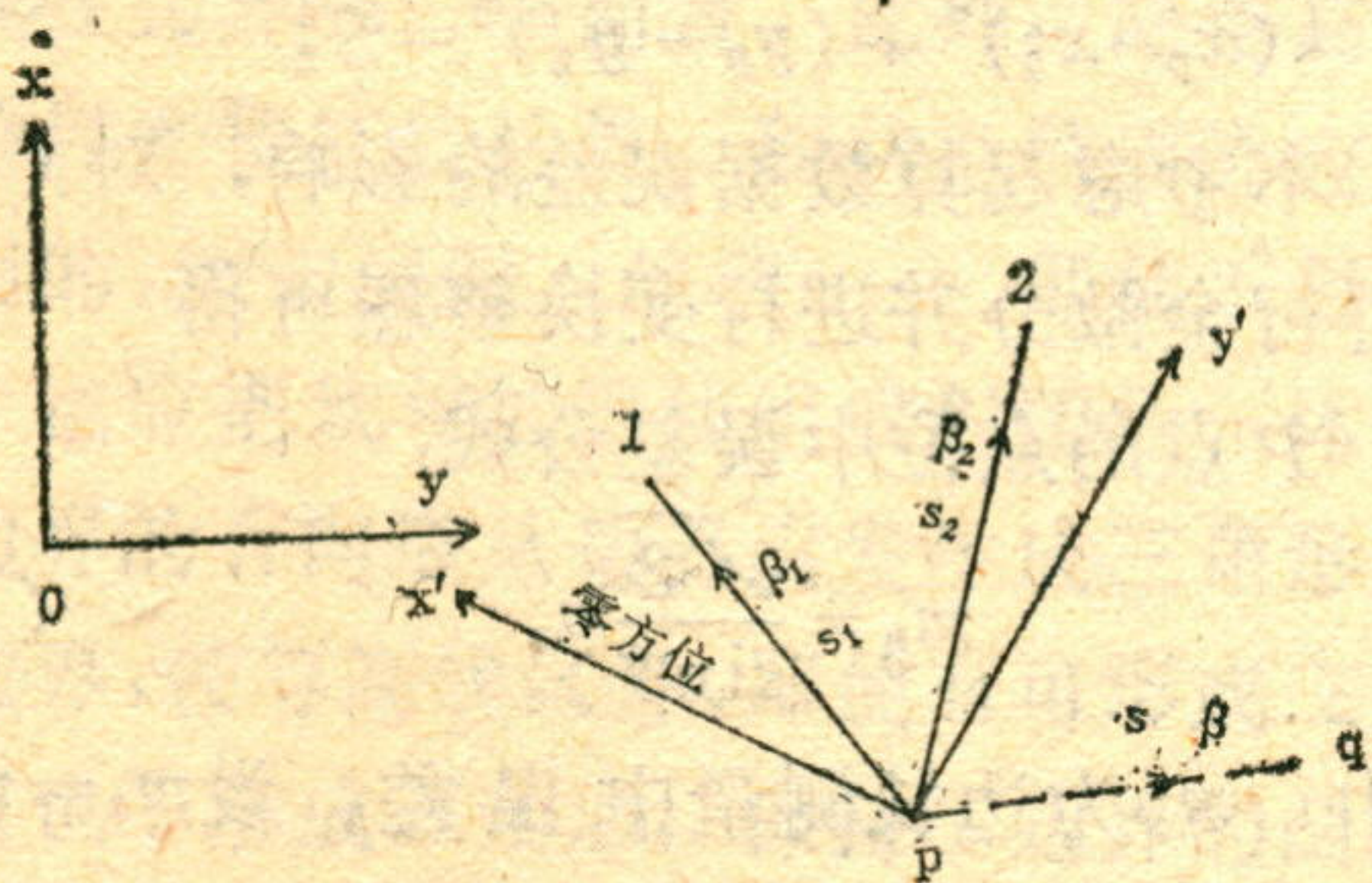


图 2

设任意两点*i*，*j*在两套坐标系内的坐标分别为 (x_i, y_i) (x'_i, y'_i) (x_j, y_j) (x'_j, y'_j) 则下列的坐标变换公式适用于任意两点：

$$\begin{cases} x'_j - x'_i = (y_j - y_i)E + (x_j - x_i)F \\ y'_j - y'_i = -(x_j - x_i)E + (y_j - y_i)F \end{cases} \quad (7)$$

其中 $E = \sin \theta$ ， $F = \cos \theta$ ， θ 为旋转角。

将已知点1、2两点在两套坐标系统的坐

标代入(7)式，并解出方程组中未知数E、F：

$$\begin{cases} E = \frac{(x'_2 - x'_1)(y_2 - y_1) - (y'_2 - y'_1)(x_2 - x_1)}{l_{12}^2} \\ F = \frac{(x'_2 - x'_1)(x_2 - x_1) + (y'_2 - y'_1)(y_2 - y_1)}{l_{12}^2} \end{cases} \quad (8)$$

式中 l_{12} 为1、2两点间距离。

放样点q的城市坐标 (x_a, y_a) 是给定的，而其假定坐标 (x'_a, y'_a) 为待求值。将E、F之值及 (x_1, y_1) 、 (x'_1, y'_1) (x_a, y_a) 代入(7)式可得：

$$\begin{cases} x'_a = x'_1 + (y_a - y_1)E + (x_a - x_1)F \\ y'_a = y'_1 - (x_a - x_1)E + (y_a - y_1)F \end{cases} \quad (9)$$

同理可得：

$$\begin{cases} x'_a = x'_2 + (y_a - y_2)E + (x_a - x_2)F \\ y'_a = y'_2 - (x_a - x_2)E + (y_a - y_2)F \end{cases} \quad (10)$$

取中数 $\frac{(9)+(10)}{2}$ 即可得放样点q在假定坐标系内坐标 (x'_a, y'_a) 。这样我们就可以按下面的式子求得测站P至放样点q之间的距离S值及经纬仪所要拨至的水平度盘方向值 β 。

$$\begin{aligned} S &= \sqrt{x'^2_a + y'^2_a} \\ \beta &= \arctg \frac{y'_a}{x'_a} \end{aligned} \quad (11)$$

这样，我们就可省去了对测站P点的坐标计算，由(11)式计算得S、 β 。直接在任意点P测站上放出我们所要放的放样点q。

2. 操作步骤

(1)选择好测站点P，整平仪器。

(2)测出P至已知点1、2间的距离 S_1 、 S_2 。经纬仪整平后不用配置度盘，瞄准1、2两点即可得假定方位 β_1 、 β_2 。

(3)按(6)、(8)、(9)、(10)、(11)式计算出放样元素S、 β 。

(4)经纬仪水平方向拨至 β 方位，测距仪按距离S放出放样点q。

四、任意点距离后方交会放点计算程序

上述两种方法会给我们的野外工作提供

一些方便,但由于计算较复杂,故特将这两种方法用BASIC语言在PC-1500计算机上编制程序,这样就可以使野外工作和计算过程同步进行,使野外工作灵活方便。下面就将计算程序见五、源程序“SLCS-1”以及运算过程介绍如下:

1. “SLCS-1” 程序运行说明

步聚	显 示	操 作
I	R=?	DEFA或RUN启动 键入1为测站距离两点后方交会,键入2为零方位假定坐标系统放点。
	NA=?	键入第一个已知点点号。
	NB=?	键入第二个已知点点号。
	NC=?	键入待放点点号。
	XA=?	键入A点坐标X、Y、
	YA=?	
	XB=? YB=?	键入B点坐标X、Y 打印反算已知点A、B间距离S,方位T。
II	XC=? YC=? RN=?	当输入R=1后程序进入II 输入待放点坐标X、Y。 RN=1为测站P点在已知点A、B方向右侧,RN=2表示在左侧
	S:P,NA=? S:P,NB=?	键入P点测至NA点距离。 键入P点测至NB点距离。
		打印放点元素T、P,NA=为测站P点至第一个已知后视方位;T:P,NC=为P点至待放点NC的放点方位 S:P,NC=为P点至待放点距离
IV	XC=? YC=? S(1)=? B(1)=? S(2)=? B(2)=?	当R键入2以后,程序执行完I过程,便转入步骤IV。 键入C点坐标X、Y。 键入P点至NA点距离。 键入P点至NA点度盘方位。 键入P点至NB点距离。 键入P点至NB点度盘方位。
		打印放点元素 T:P,NC=为经纬仪将要拨至的水平度盘方位。 S:P,NC=为P点至待放点,NC间的距离。

2.算例:

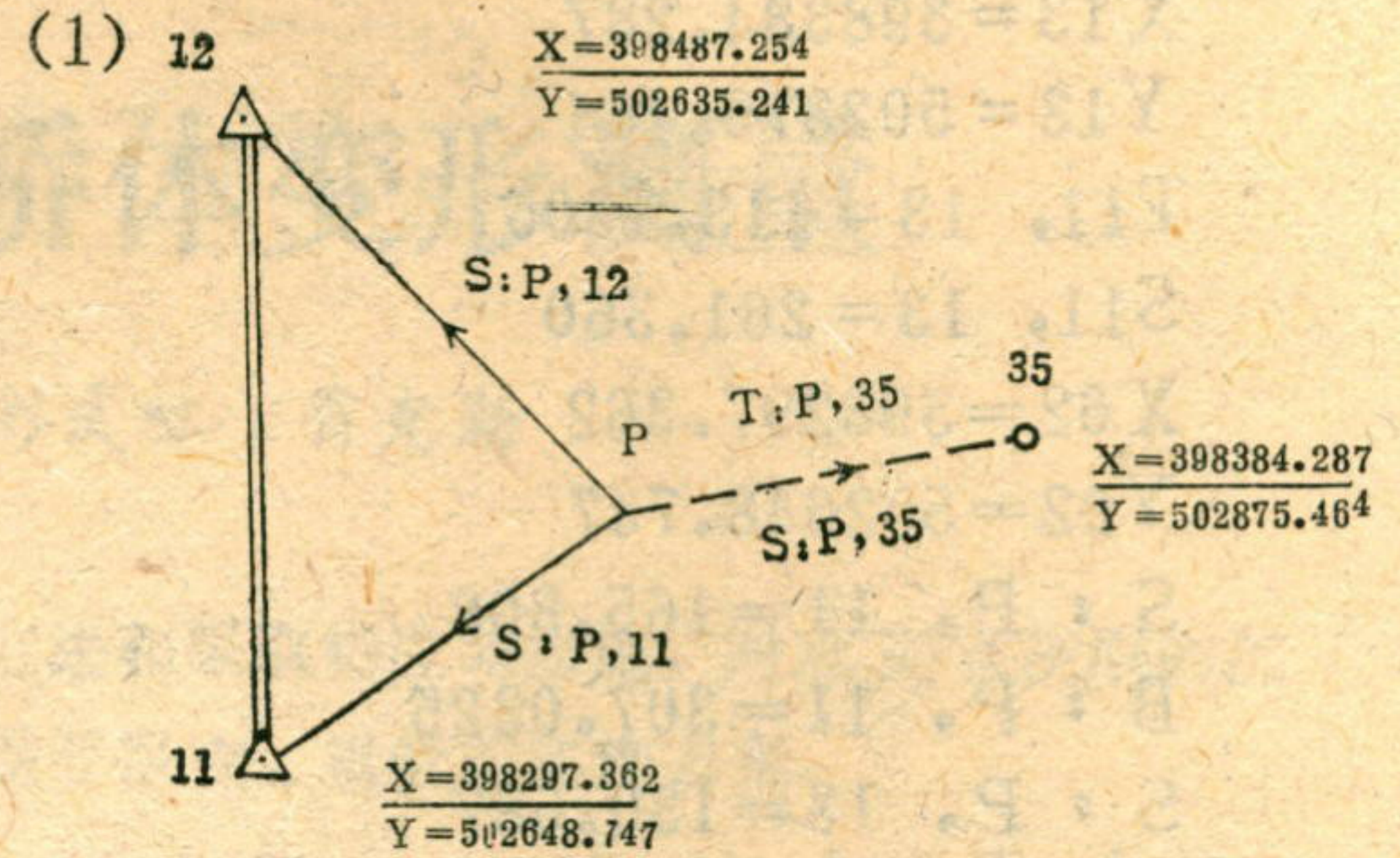


图 3

R = 1
 X11 = 398297.362
 Y11 = 502648.747
 X12 = 398487.254
 Y12 = 502635.241
 T11, 12 = 355.5554
 S11, 12 = 190.371
 X35 = 398384.287
 Y35 = 502875.464
 RN = 1
 S : P, 11 = 118.382
 S : P, 12 = 165.808
 XP = 398364.2469
 YP = 502746.4235
 T : P, 11 = 235.3553
 T : P, 35 = 81.1020
 S : P, 35 = 130.587

如图3所示,实地有已知点11, 12。已知11, 12, 35三点坐标,现将仪器置于任意便于放点的P点上。按本文二中的方法,实地测量S:P, 11及S:P, 12距离,这一方法计算机在运行过程中,取R=1,由于我们取NA=11, NB=12,则P点位于11, 12前进方向右侧,故取RN=1。计算方法按程序运行说明,最后计算机将输出P点坐标 X_P 、 Y_P ; T:P, 11为P点至11方位,当得出结果后,经纬仪后视11点,水平度盘配置为T:P, 11 = 235°35'53",然后将经纬仪拨到T:P, 35 = 81°10'20",按距离S·P·35 = 130.587m将35号点放于实地。

R = 2
 X11 = 398487.254

Y11 = 502635.241
 X13 = 398384.287
 Y13 = 502875.464
 T11, 13 = 113.1205
 S11, 13 = 261.360
 X62 = 398297.362
 Y62 = 502648.747
 S : P, 11 = 165.808
 B : P, 11 = 307.0325
 S : P, 13 = 130.587
 B : P, 13 = 70.202
 T : P, 62 = 224.4553
 S : P, 62 = 118.381

(2)

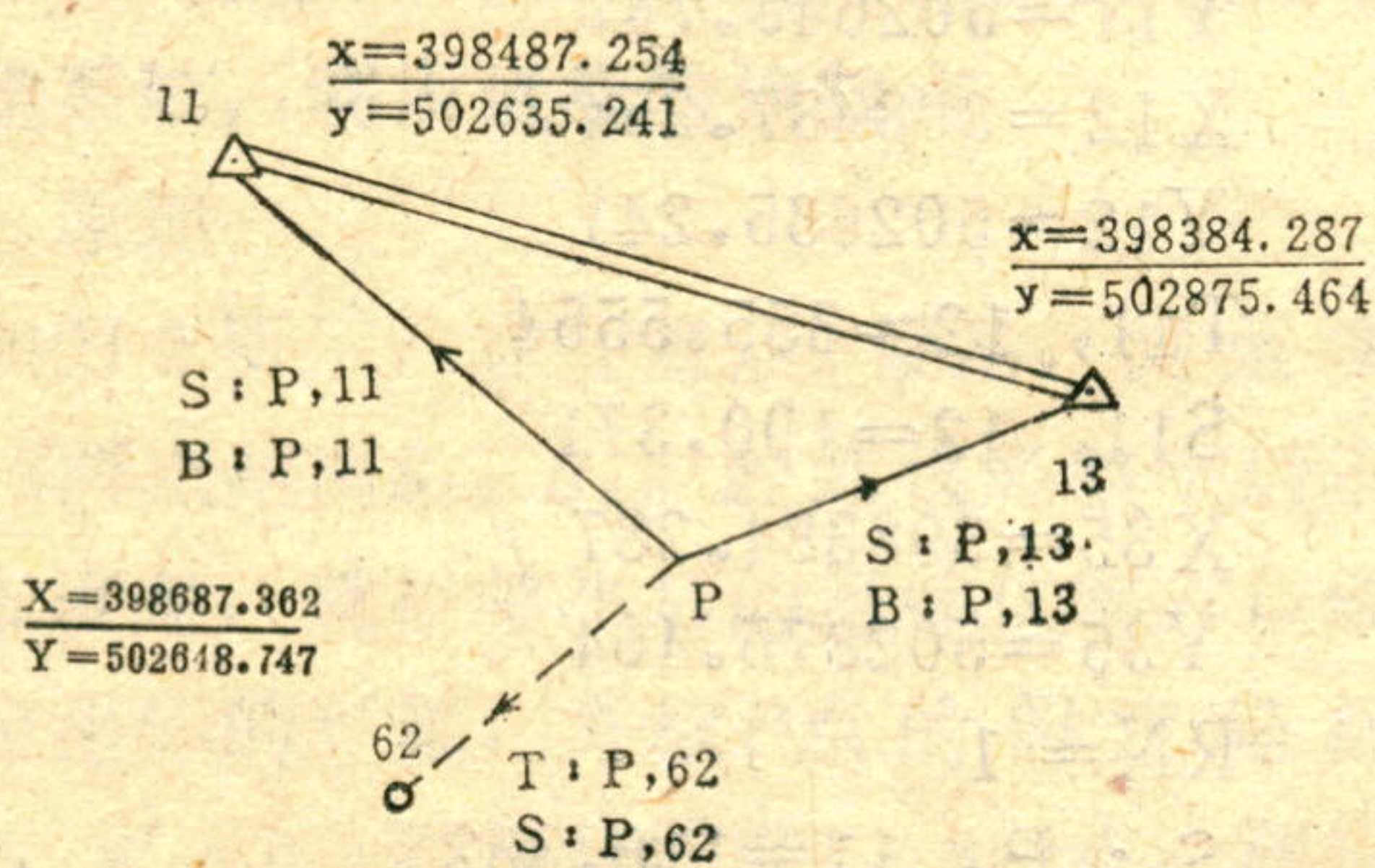


图 4

图 5 为本文三所述方法, 按运行过程计算所得 T : P, 62 方位即为经纬仪所要拨至方位, 这一方位与实地方位不一致, T : P, 62 方位为经纬仪水平度盘零方位假定坐标系方位。在 P 测站我们就按经纬仪上这一方位及距离 S : P, 62 进行放点。

五、源程序 "SLCS-1"

```
5: "A" CLEAR : WAIT 0 : USING
7: A=0 : T=0
10: DIM X(10), Y(10), S(10), B(10)
12: INPUT "R=" ; R, "NA=" ; NA,
    "NB=" ; NB, "NC=" ; NC
14: WAIT 0 : CLS : PRINT "X" ; NA,
    "=" ; INPUT X1 : CLS : PRINT "Y",
    NA ; "=" ; INPUT Y1
16: CLS : PRINT "X" ; NB ; "=" ; INPUT X
2 : CLS : PRINT "Y" ; NB ; "=" ;
    INPUT Y2
18: C = X2 - X1 ; D = Y2 - Y1 + 1E - 10 ;
    GOSUB 80
20: LPRINT "R=" ; R ; LPRINT "X" ; N
```

```
A ; "=" ; X1 ; LPRINT "Y" ; NA ;
    "=" ; Y1
22: LPRINT "X" ; NB ; "=" ; X2 ; LPR
    INT "Y" ; NB ; "=" ; Y2
24: T$ = STR$ NA + "," + STR$ NB +
    "=" ; LPRINT USING "###.#
    ###" ; "T" ; T$ ; DMS T
26: LPRINT USING "###.#" ;
    "S" ; T$ ; S ; USING
27: A$ = " : P, " + STR$ NA + "=" :
    B$ = " : P, " + STR$ NB + "="
28: CLS : PRINT "X" ; NC ; "=" ; : IN
    PUT X3 : CLS : PRINT "Y" ; NC ;
    "=" ; : INPUT Y3 : CLS
30: LPRINT "X" ; NC ; "=" ; X3 : LPR
    INT "Y" ; NC ; "=" ; Y3
32: IF R=2 THEN GOTO 50
33: CLS : PRINT "RN=" ; : INPUT
    RN : CLS : PRINT "S" ; A$ ; : INP
    UT S1 : CLS : PRINT "S" ; B$ ; :
    INPUT S2 ; LPRINT "RN=" ; RN
35: LPRINT "S" ; A$ ; S1 : LPRINT "S
    "B$ ; S2 : LPRINT
36: A = ACS ( (S1 * S1 + S * S - S2 * S2)
    / (2 * S1 * S) )
38: IF RN=1 THEN LET A=T+A
39: IF RN=2 THEN LET A=T-A
40: XP = X1 + S1 * COS A : YP = Y1 + S1
    * SIN A : C = X3 - XP : D = Y3 + YP
    + 1E - 10 : GOSUB 80
41: LPRINT "XP=" ; XP : LPRINT "Y
    P=" ; YP : IFA >= 180 THEN LET
    B = A - 180
42: IFA < 180 THEN LET B = A + 180
43: LPRINT USING "###.#" ;
    "#" ; "T" ; A$ ; DMS B
44: GOSUB 82 : GOTO 70
50: FOR I=1 TO 2 : CLS : PRINT "S(" ;
    STR$ I ; ")=" ; : INPUT S(I) ; CLS :
    PRINT "B(" ; STR$ I ; "(" ; : INP
    UT B(1) : NEXT I
51: LPRINT "S" ; $ ; S(1) : LPRINT
    "E" ; A$ ; B(1) : LPRINT "S" ; B
    $ ; S(2) : LPRINT "B" ; B$ ; B(2)
52: FOR I=1 TO 2 : B(I) = DEGB(I) :
    (下转第55页)
```