

*U.S. Department of Agriculture
Forest Service General Technical Report SO-9*

***Combining-Ability Determinations
for
Incomplete Mating Designs***

E. B. Snyder

*Southern Forest Experiment Station
Forest Service
U.S. Department of Agriculture*

1975

COMBINING-ABILITY DETERMINATIONS FOR INCOMPLETE MATING DESIGNS

E. B. Snyder¹

It is shown how general combining ability values (GCA's) from cross-, open-, and self-pollinated progeny can be derived in a single analysis. Breeding values are employed to facilitate explaining genetic models of the expected family means and the derivation of the GCA's. A FORTRAN computer program also includes computation of specific combining ability values and several options.

Additional keywords: Diallel cross, reciprocal cross, specific combining ability, panmixia.

The plant breeder ranks parents according to their general combining abilities. The calculation of mean effects is complicated when all parents of a particular set have not been crossed systematically, but theories and procedures for analyzing even severely unbalanced data are nevertheless available (Bohren et al. 1965, Gilbert 1967).

The mathematics of such theories may present difficulties to the practical breeder. Here, I advance a simplified explanation by showing how breeding values ($2 \text{ GCA} + \bar{X}$ terms) can be manipulated for analyzing data from cross, self, and open pollinations in a single analysis. I will also discuss the usefulness of selfing, a means of testing the randomness of open-pollination, a technique for estimating the population mean, and a method for combining data from several experiments.

As computer programs for such unbalanced data have generally not been published, a versatile, efficient FORTRAN program is described along with substitute procedures requiring no more than a desk calculator.

¹The author is Principal Plant Geneticist, Southern Forest Experiment Station, Forest Service—USDA, Gulfport, Mississippi. He is indebted to W. L. Nance, Southern Forest Experiment Station, for much guidance, and to N. E. Gilbert, University of British Columbia, and N. G. Alvey, Rothamsted Experiment Station, England, for test data and comments.

THEORY AND APPLICATIONS

When reciprocal crosses are pooled, a model for determining general combining ability effects (GCA's) from intercrossed parents is:

$$y_{ijk} = \mu + gca_i + gca_j + \varepsilon_{ijk}$$

where y_{ijk} = k^{th} individual in the i, j^{th} cross

gca_i = GCA effect of the i^{th} parent

gca_j = GCA effect of the j^{th} parent

with the assumptions that

$$\sum n_i gca_i = 0$$

$$\{\varepsilon_{ijk}\} \sim \text{NID}(0, \sigma^2)$$

where n_i = the number of plants per family (given a value of 1 in unweighted analyses).

The GCA estimates are applicable only to the restricted set of parents tested, i.e., the model is fixed.

Breeding values ($2 \text{ GCA} + \bar{X}$) are easily visualized and hence facilitate explaining the genetic model. A value of the i^{th} parent is ($2 \text{ GCA}_i + \bar{X}$), or twice the parental contribution to an individual progeny. The models for the expected family means (mid-parental values) of individual progeny of various types are:

$$\begin{aligned} \text{Cross-pollinated } & \frac{(2 \text{ GCA}_i + \bar{X}) + (2 \text{ GCA}_j + \bar{X})}{2} \\ & = \text{GCA}_i + \text{GCA}_j + \bar{X} \end{aligned}$$

$$\begin{aligned} \text{Self-pollinated } & \frac{(2 \text{ GCA}_i + \bar{X}) + (2 \text{ GCA}_i + \bar{X})}{2} \\ & = 2 \text{ GCA}_i + \bar{X} \end{aligned}$$

$$\begin{aligned} \text{Open-pollinated } & \frac{(2 \text{ GCA}_i + \bar{X}) + (2 \text{ GCA}_{op} + \bar{X})}{2} \\ & = (\text{GCA}_i + \text{GCA}_{op} + \bar{X}) \end{aligned}$$

or ($\text{GCA}_i + \bar{X}$), because $\text{GCA}_{op} = 0$.

Since formulae for all three family types are built up from breeding values, observed means can be used to calculate breeding values for each family type. Furthermore, breeding values for each type can be entered independently or together in a single diallel analysis, thereby giving more replication and confidence to estimates.

Selves should be incorporated only if there is no inbreeding depression. As the structural model shows, an extreme breeding value is doubled among selfed progeny, whereas among crosses it is diluted by being paired with a less extreme value. Thus, the extremeness is a "selfing" effect only in the sense that a single extreme breeding value (variant) would not be so conspicuous in cross-pollinated progeny. If there were no inbreeding effect or if it could be compensated for, selfing would efficiently and unambiguously identify extreme parents.

Provided there are some parents in common, populations of various types of material from different experiments can be used in a single diallel table by adjusting breeding values and, from these, the phenotypes to be integrated. Combining populations thus requires the preliminary analysis of each and subsequently calculating the adjusting ratios from the average of breeding values in common. Similarly, data from separate plantings of the same families may be combined after appropriate adjustments.

Data from open-pollinated (OP) families can be included in the same analysis with those from cross-pollinated families. This is possible when wind-borne pollen is treated as that from a single male parent representing the population. Two values of interest to the breeder can be deduced from the OP model. Deviation of the GCA_{op} value from zero measures the deviation of selected females from randomness. Also, the mean of the population is estimated by the OP breeding value.

The computer calculations are based on the usual least-squares methods suggested by Gilbert (1967) and illustrated by England (1974). If a computer is not available, iteration with a calculating machine will serve. Thus, for unweighted analysis with reciprocals absent or pooled and no selfs (Yates 1947), parental mean of i^{th} parent, $\bar{P}_i \cong$

Breeding value of i^{th} parent + $\overline{BV}$ of other parents
crossed with P_i

2

For example, the breeding value for parent 3 of a 13-parent half-diallel with no selfs becomes:

$$(2 GCA_3 + \bar{X}) = 2\bar{P}_3 - 1/12(2GCA_1 + \bar{X} + 2 GCA_2 + \bar{X} + 2GCA_4 + \bar{X} \dots + 2GCA_{13} + \bar{X})$$

The $\bar{P}_3$ value is the mean of all families for which P_3 is a parent. When the initial equations are set up, $\bar{P}_i$ values must be substituted for the $(2GCA_i + \bar{X})$ values within the righthand parentheses. As the analysis continues, $(2GCA_i + \bar{X})$ values are entered as soon as determined. Usually, satisfactory convergence will be achieved with five or fewer iterations.

Once the breeding values are found, specific combining abilities may be computed directly in the usual way:

$$SCA = \text{observed mean of } i^{th}, j^{th} \text{ family} \\ - \frac{(2 GCA_i + \bar{X}) + (2GCA_j + \bar{X})}{2}$$

By definition, SCA's are not appropriate for selfs.

Because the mid-parental expectation of a cross is the last term in the formula, definition and use of $(2 GCA + \bar{X})$ breeding values is easy. They are simply parental values which, when averaged, predict performance of the cross. The convenience and ease of visualization for instruction and breeding are lacking with GCA or 2 GCA deviations alone.

THE PROGRAM

The mathematics of the program is similar to that used by most statisticians—least-squares equations are formed and GCA values are solved for by inverting the matrix according to the elimination method. For some limitations to the program, Gilbert (1967) should be consulted. Solutions may differ slightly depending on the degrees of design balance and order in which data are fitted. Also, if it is desired to estimate values separately within male or female sets, Milliken et al. (1970) should be consulted.

The usefulness of the program lies with its supplementary options and listings:

Sum or mean data with or without the number of plants per family may be entered for as many as 50 parents. Options are offered for weighting by number of plants, not weighting, or both. If no reciprocal crosses are present, data are entered as upper-triangle matrix elements. Reciprocal data are entered in the lower triangle and will automatically be accumulated or averaged into the upper triangle.

Input or data errors will be signalled, after which the analysis will be terminated and the next one started. Warnings of some types of singular or

near-singular solutions are given. The program is generously supplied with comment cards that should facilitate modification by others.

Listed in addition to combining ability and breeding values are: the data entered, family means, number of parents, effective number of families per parent, mean breeding value, and experimental mean.

The inverse is printed for possible use in obtaining confidence intervals for the GCA values (Milliken et al. 1970). An error mean square must be available from a separate variance-analyzing program. The program of Schaffer and Usanis (1969) is adequate and so versatile that we do not supply variance terms except for a GCA sum of squares for comparison with it and other programs.

LITERATURE CITED

- Bohren, B. B., H. E. McKean, and G. W. Friars.
1965. The expected mean squares in genetic experiments when only one parent is identified. *Biometrics* 21: 436-446.
- England, F.
1974. A general approximate method of fitting additive and specific combining abilities to the diallel cross with unequal numbers of observations in the cells. *Theor. and Appl. Genet.* 44: 378-380.
- Gilbert, N. E.
1967. Additive combining abilities fitted to plant breeding data. *Biometrics* 23: 45-49.
- Milliken, G. A., H. L. Bush, A. W. Ericksen, and A. Suzuki.
1970. Estimating general combining ability from an incomplete crossing system. *J. Am. Soc. Sugar Beet Technol.* 16: 264-274.
- Schaffer, H. E., and R. A. Usanis.
1969. General least squares analysis of diallel experiments, a computer program—DIALL. N.C. State Univ. Genet. Dep. Res. Rep. 1, 61 p.
- Yates, F.
1947. Analysis of data from all possible reciprocal crosses between a set of parental lines. *Heredity* 1: 287-301.

```

1* C E.B.SNYDER-W.NANCE.SFES THIS PROGRAM IS FOR ANY TYPE OF DIALLEL
2* C BUT ASSUMES NO SELFING EFFECTS.SUMS WITH PLANT COUNTS/FAMILY,AND
3* C MEANS WITH OR WITHOUT COUNTS MAY BE ENTERED TO OBTAIN WEIGHTED OR
4* C UNWEIGHTED VALUES.ALL TYPES OF DATA ARE ACCUMULATED IN UPPER TRI-
5* C ANGLE OF MATRICES FOR ANALYSIS
6* C
7* C *****
8* C * CARD ORDER *
9* C *****
10* C
11* C CONTROL CARD **
12* C INPUT FORMAT CARD * REPEATED FOR EACH EXPERIMENT
13* C TITLE CARD *
14* C DATA CARDS **
15* C
16* C TRAILER CARD ** CONSISTS OF A BLANK CARD USED ONLY AFTER FINAL
17* C DATA SET
18* C
19* C *****
20* C * CONTROL CARD *
21* C *****
22* C
23* C COLS.1-7 CONTROL
24* C COLS.8-10 PROBLEM NO.
25* C COLS.11-13 NUMBER OF PARENTS
26* C COLS.14-17 NO. OF FAMILIES WITH DATA--LEAVE BLANK IF COL.19 IS 0
27* C
28* C * 1 IF ID IS PROVIDED WITH DATA
29* C COLS.18-19 PUNCH*
30* C * 0 IF NO ID IS PROVIDED. IN THIS CASE ZEROS
31* C MUST BE ENTERED WHERE MISSING DATA
32* C
33* C *1 IF DATA ENTERED ARE SUMS AND PLANTS/FAMILY
34* C COL. 21 PUNCH * 2 IF DATA ENTERED ARE MEANS AND PLANTS/FAMILY
35* C *3 IF DATA ENTERED ARE MEANS ONLY
36* C
37* C *1 IF ANALYSIS WEIGHTED BY PLANTS/FAMILY
38* C COL. 23 PUNCH * 2 IF WEIGHTED AND UNWEIGHTED ANALYSIS
39* C *3 IF UNWEIGHTED ANALYSIS
40* C
41* C *****
42* C * INPUT FORMAT CARD *
43* C *****
44* C
45* C PLANTS/FAMILY,IF USED,ARE ENTERED IN F FORMAT
46* C
47* C *****
48* C * TITLE CARD *
49* C *****
50* C
51* C COLS.1-5 TITLE
52* C COLS.8-72 TEXTUAL IDENTIFICATION OF THE VARIABLE OR CHARACTER
53* C
54* C *****
55* C * DATA CARDS *
56* C *****
57* C
58* C ONE TRAIT AT A TIME IS ANALYZED. MEASUREMENT DATUM FOR A FAMILY
59* C IS PREFERABLY PRECEDED BY PARENTAL COMPOSITION ID,WHERE FEMALE
60* C (CODED SEQUENTIALLY 1 TO N) IS IMMEDIATELY FOLLOWED BY A SIMIL-
61* C ARLY CODED MALE,E.G.0102 GIVES BOTH PARENTAGE AND MATRIX INDIC-
62* C ES. THE MEASUREMENT DATA MAY BE FOLLOWED IN THE SAME FIELD BY
63* C NUMBER OF PLANTS/FAMILY. SEE OPTIONS LISTED FOR CONTROL CARDS
64* C ABOVE.THE DATA FROM MORE THAN ONE PLANT CAN BE PLACED ON ONE

```

```

65*      C      CARD ONLY IF ID'S ARE NOT SUPPLIED BUT HERE THERE ARE NO INPUT
66*      C      ERROR CHECKS. ALL DATA ARE ENTERED BY ROWS AND, IF THERE ARE NO
67*      C      RECIPROCAL CROSSES, IN THE UPPER TRIANGLE. WHERE THE TRUE VALUE
68*      C      OF DATUM IS ZERO, A VERY SMALL NUMBER MUST BE ENTERED INSTEAD.
69*      C
70*      C*****
71*      C
72*      C
73*      REAL NN,MM,KK
74*      INTEGER P1,DE,WT,W,EMPTY,CONTRO
75*      DIMENSION U(50,50), X(50,50), IX(50,50), KK(50,50), MM(50,50), NN(
76*      150,50), Q(50,50), Y(50,1), N(50), P(50), B(50), G(50), FMT(12), TI
77*      2TLE(11), XPY(50,1)
78*      C      FOR SOME COMPUTERS THE FOLLOWING MAY BE NEEDED
79*      C      DIMENSION RAY(50,50), BB(50,1), JJ(50)
80*      EQUIVALENCE (IX,X)
81*      DATA CONTRO /6HCONTRO/
82*      DATA IOU,IIU /6,5/
83*      DATA EMPTY /5HEMPTY/
84*      WRITE (IOU,530)
85*      5 READ (IIU,570,END=520) IA
86*      IF (IA.NE.CONTRO) GO TO 60
87*      10 READ (Q,540) P1,NF,ID,DE,WT
88*      20 IF (P1.EQ.0) GO TO 520
89*      READ (IIU,550) (FMT(I1),I1=1,12)
90*      READ (IIU,590) (TITLE(I3),I3=1,11)
91*      DO 30 IJK=1,P1
92*          DO 30 IKL=1,P1
93*              NN(IJK,IKL)=0.
94*              IX(IJK,IKL)=EMPTY
95*      30 CONTINUE
96*      IF (ID.EQ.1) GO TO 40
97*      IF (ID.EQ.0.AND.DE.LT.3) READ (IIU,FMT) ((X(I,J),NN(I,J),J=1,P1),I
98*      1=1,P1)
99*      IF (ID.EQ.0.AND.DE.EQ.3) READ (IIU,FMT) ((X(I,J),J=1,P1),I=1,P1)
100*      GO TO 80
101*      40 DO 50 IJK=1,NF
102*          IF (DE.LT.3) READ (IIU,FMT) IFEM,IMAL,XIN,XNNIN
103*          IF (DE.EQ.3) READ (IIU,FMT) IFEM,IMAL,XIN
104*          IF (IFEM.LE.0.OR.IFEM.GT.P1) GO TO 60
105*          IF (IMAL.LE.0.OR.IMAL.GT.P1) GO TO 60
106*          IF (IX(IFEM,IMAL).NE.EMPTY) GO TO 60
107*          X(IFEM,IMAL)=XIN
108*          IF (DE.LT.3) NN(IFEM,IMAL)=XNNIN
109*      50 CONTINUE
110*      GO TO 80
111*      60 WRITE (IOU,560)
112*      C      SEARCH FOR OTHER PROBLEMS SUBMITTED BUT AN ERROR CAN RUIN THE
113*      C      READING IN OF IMMEDIATELY FOLLOWING PROBLEM(S)
114*      70 READ (IIU,570,END=520) IA
115*      IF (IA.NE.CONTRO) GO TO 70
116*      READ (Q,540) P1,NF,ID,DE,WT
117*      GO TO 20
118*      80 WRITE (IOU,580)
119*      WRITE (IOU,590) (TITLE(I3),I3=1,11)
120*      WRITE (IOU,600) P1
121*      DO 90 I1=1,P1
122*          DO 90 I2=1,P1
123*              IF (IX(I1,I2).EQ.EMPTY) X(I1,I2)=0.
124*      90 CONTINUE
125*      KOUNT=0
126*      DO 100 I=1,P1
127*          DO 100 J=1,P1
128*              U(I,J)=X(I,J)
129*      100 CONTINUE

```

```

130*         IF (DE.NE.1) GO TO 120
131*         DO 110 I=1,P1
132*             DO 110 J=1,P1
133*                 IF (NN(I,J).EQ.0.) GO TO 110
134*                 X(I,J)=X(I,J)/NN(I,J)
135*         110     CONTINUE
136*     C         TEST FOR RECIPROCAL IN LOWER TRIANGLE
137*     120 DO 130 I=2,P1
138*         L=I-1
139*         DO 130 J=1,L
140*     130     IF (X(I,J).GT.0.) GO TO 420
141*         W=0.
142*     C         GIVES NATURE OF DATA ENTERED AND WEIGHTING DESIRED
143*     140 IF (DE.NE.1) GO TO 150
144*         WRITE (10U,610)
145*     150 IF (DE.NE.3) GO TO 190
146*         WRITE (10U,620)
147*         IF (DE.EQ.3.AND.WT.EQ.3) GO TO 160
148*         WRITE (10U,630)
149*     160 WRITE (10U,640)
150*     C         ENTER 1'S IN NN MATRIX IF UNWEIGHTED ANALYSIS
151*         DO 170 IJK=1,P1
152*             DO 170 IKL=1,P1
153*     170     NN(IJK,IKL)=0.
154*             DO 180 I=1,P1
155*                 DO 180 J=1,P1
156*                     IF (X(I,J).NE.0.) NN(I,J)=1.
157*     180     CONTINUE
158*         GO TO 210
159*     190 IF (DE.EQ.1) GO TO 200
160*         WRITE (10U,650)
161*     200 IF (WT.EQ.3) GO TO 160
162*         WRITE (10U,660)
163*     210 DO 220 I=1,P1
164*         DO 220 J=1,P1
165*     220     KK(I,J)=NN(I,J)
166*     C         ADJUST FREQ OF SELFS IN DIAG OF NN
167*         DO 230 I=1,P1
168*             NN(I,I)=4.*NN(I,I)
169*     230     CONTINUE
170*         KOUNT=KOUNT+1
171*         DO 240 I=1,P1
172*             N(I)=0
173*             P(I)=0.
174*     240     B(I)=0.
175*         N2=0
176*     C         START MAIN PROGRAM BY SETTING SELF DATA IN ACCUMULATORS
177*         DO 250 I=1,P1
178*             N(I)=0.5*NN(I,I)
179*             P(I)=0.5*NN(I,I)*X(I,I)
180*     250     CONTINUE
181*     C         ACCUMULATE YIELDS AND COUNTS FOR EACH PARENT
182*         DO 260 J=2,P1
183*             L=J-1
184*             DO 260 I=1,L
185*                 IF (X(I,J).EQ.0.) GO TO 260
186*                 N(I)=N(I)+1
187*                 N(J)=N(J)+1
188*                 P(I)=P(I)+NN(I,J)*X(I,J)
189*                 P(J)=P(J)+NN(I,J)*X(I,J)
190*     260     CONTINUE
191*         DO 270 I=1,P1
192*             N2=N2+N(I)
193*     270     CONTINUE
194*     C         ARRANGE P(J) IN MATRIX AND PRINT

```

```

195* DO 280 J=1,P1
196*   Y(J,1)=2.*P(J)
197*   CONTINUE
198* DO 290 I=1,P1
199*   XPY(I,1)=Y(I,1)
200*   CONTINUE
201* C DUPLICATE UPPER NN ELEMENTS IN LOWER TRIANGLE
202* DO 300 I=1,L
203*   DO 300 J=2,P1
204*     NN(J,I)=NN(I,J)
205* C ACCUMULATE TOTAL ADJUSTED COUNTS IN DIAGONALS
206* DO 320 J=1,P1
207*   N(J)=0.
208*   DO 310 I=1,P1
209*     N(J)=N(J)+NN(J,I)
210*   CONTINUE
211*   NN(J,J)=N(J)
212*   CONTINUE
213*   WRITE (100,670)
214*   WRITE (100,680) (NN(I,I),I=1,P1)
215*   CALL MATINV (NN,P1,Y,1,DETERM,NDEP,1)
216*   WRITE (100,690) NDEP
217*   S=0.
218*   DO 330 I=1,P1
219*     B(I)=Y(I,1)
220*     S=S+B(I)
221*   CONTINUE
222*   XP1=P1-NDEP
223*   S=S/XP1
224*   IF (NDEP.EQ.0) GO TO 335
225*   S=S/2.0
226*   JXP1 = XP1
227*   DO 332 I = 1,JXP1
228*     B(I)=B(I)-S
229*   CONTINUE
230* C332
231* C335 EAN AND SS TERMS LINK WITH ANV
232*   SUM1=0.
233*   SUM2=0.
234*   SUM3=0.
235*   DO 340 I=1,P1
236*     DO 340 J=1,P1
237*       SUM1=SUM1+X(I,J)*KK(I,J)
238*       SUM2=SUM2+KK(I,J)
239*     CONTINUE
240*   DO 350 I=1,P1
241*     SUM3=SUM3+XPY(I,1)*Y(I,1)
242*   CONTINUE
243*   Y1=SUM1/SUM2
244*   SUM2=((SUM1)**2)*SUM2
245*   SUM3=(SUM3/4.0)-SUM2
246*   WRITE (100,710)
247*   WRITE (100,720) SUM2
248*   WRITE (100,730) SUM3
249*   WRITE (100,740) SUM1
250*   DO 360 J=1,P1
251*     IF (B(J).EQ.0.) B(J)=S
252*     G(J)=((B(J)-S)/2.0)
253*   CONTINUE
254*   DO 370 I=1,P1
255*     L=I+1
256*     DO 370 J=L,P1
257*       IF (X(I,J).EQ.0.) Q(I,J)=0.
258*       IF (X(I,J).NE.0.) Q(I,J)=X(I,J)-((B(I)+B(J))/2.0)
259*     CONTINUE

```

```

260*      WRITE (IOU,750)
261*      CALL MPRINT (NN,P1,1,6HNN MAT)
262*      IF (DE.GT.1.AND.W.EQ.0) GO TO 390
263*      IF (DE.EQ.1.AND.W.EQ.0) GO TO 380
264*      WRITE (IOU,760)
265*      CALL MPRINT (MM,P1,1,6HMM MAT)
266*      WRITE (IOU,770)
267*      CALL MPRINT (U,P1,1,6H U MAT)
268*      GO TO 390
269*      380 WRITE (IOU,780)
270*      CALL MPRINT (U,P1,1,6H U MAT)
271*      390 WRITE (IOU,790)
272*      WRITE (IOU,800) (G(I),I=1,P1)
273*      WRITE (IOU,810)
274*      WRITE (IOU,820) (B(I),I=1,P1)
275*      WRITE (IOU,830)
276*      LINES=0
277*      DO 410 I=1,P1
278*          DO 410 J=I,P1
279*              IF (KK(I,J).EQ.0.) GO TO 410
280*              LINES=LINES+1
281*              IF (LINES.NE.51) GO TO 400
282*              LINES=0
283*              WRITE (IOU,830)
284*      400 WRITE (IOU,840) I,J,KK(I,J),X(I,J),Q(I,J)
285*      410 CONTINUE
286*      IF (WT.EQ.2.AND.KOUNT.LT.2) GO TO 160
287*      GO TO 5
288*      C THIS ENDS ANALYSIS
289*      420 W=1.
290*      C THE FOLLOWING PUTS RECIPS IN HALF DIALLEL FORM
291*      IF (DE.LT.3) GO TO 450
292*      DO 430 IJK=1,P1
293*          DO 430 IKL=1,P1
294*      430 NN(IJK,IKL)=0
295*      DO 440 I=1,P1
296*          DO 440 J=1,P1
297*      440 IF (X(I,J).NE.0.) NN(I,J)=1.
298*      450 DO 460 I=1,P1
299*          DO 460 J=1,P1
300*              X(I,J)=NN(I,J)*X(I,J)
301*      460 CONTINUE
302*      DO 470 I=1,P1
303*          DO 470 J=1,P1
304*      470 MM(I,J)=NN(I,J)
305*      DO 480 I=1,P1
306*          DO 480 J=1,P1
307*              NN(I,J)=NN(I,J)+NN(J,I)
308*              X(I,J)=X(I,J)+X(J,I)
309*      480 CONTINUE
310*      DO 490 I=1,P1
311*          NN(I,I)=0.5*NN(I,I)
312*          X(I,I)=0.5*X(I,I)
313*      490 CONTINUE
314*      DO 500 I=2,P1
315*          L=I-1
316*          DO 500 J=1,L
317*              NN(I,J)=0.
318*              X(I,J)=0.
319*      500 CONTINUE
320*      DO 510 I=1,P1
321*          DO 510 J=I,P1
322*              IF (X(I,J).EQ.0.) GO TO 510
323*              X(I,J)=X(I,J)/NN(I,J)

```

```

324*      510      CONTINUE
325*      GO TO 140
326*      520 STOP
327*
328*      C
329*      530 FORMAT (1H1,61H          GCA AND SCA VALUES FOR COMPLETE OR INCOMPLET
330*      1E DIALLELS)
331*      540 FORMAT (10X,I3,I4,3I2)
332*      550 FORMAT (12A6)
333*      560 FORMAT (1H0,65H END OF ANALYSIS OR TERMINATION DUE TO INPUT FORMAT
334*      1 OR DATA ERROR)
335*      570 FORMAT (A6)
336*      580 FORMAT (1H0,51H * * * * * * * * * * * * * * * * * * * * * * * * * *
337*      1)
338*      590 FORMAT (1H0,6X,11A6)
339*      600 FORMAT (1H0,25H THE NUMBER OF PARENTS IS,I4)
340*      610 FORMAT(1H0,41H DATA ENTERED WERE SUMS AND PLANTS/FAMILY)
341*      620 FORMAT (1H0,29H DATA ENTERED WERE MEANS ONLY)
342*      630 FORMAT (1H0,31H WEIGHTED ANALYSIS NOT POSSIBLE)
343*      640 FORMAT (1H0,20H UNWEIGHTED ANALYSIS)
344*      650 FORMAT (1H0,46H DATA ENTERED WERE MEANS AND PLANTS PER FAMILY)
345*      660 FORMAT (1H0,19H WEIGHTED ANALYSIS)
346*      670 FORMAT (1H0,82H THE EFFECTIVE NUMBER OF FAMILIES OR PLANTS PER PAR
347*      1ENT(RECIPS ABSENT OR POOLED)ARE)
348*      680 FORMAT (1H ,10F10.1)
349*      690 FORMAT (1H0,107H WHENEVER A DEPENDENCY(SINGULAR MATRIX) OCCURS THE
350*      1. GCA EFFECT IS SET TO ZERO. THE NUMBER OF DEPENDENCIES IS ,I3)
351*      700 FORMAT (1H0,28H THE MEAN(2GCA+XBAR)VALUE IS,E16.9)
352*      710 FORMAT (1H0,117H IF CORRECT DATA WERE ENTERED THE FOLLOWING WILL C
353*      1ORRESPOND TO TERMS FOUND BY INDEPENDENTLY DERIVED LEAST SQUARES AN
354*      2V)
355*      720 FORMAT (1H0,23H THE CORRECTION TERM IS,E16.9)
356*      730 FORMAT (1H0,31H THE SS TERM FOR GCA EFFECTS IS,E16.9)
357*      740 FORMAT (1H0,25H THE EXPERIMENTAL MEAN IS,E16.9)
358*      750 FORMAT (1H1,51HMATRIX INVERSE FOR DETERMINING CONFIDENCE INTERVALS
359*      1)
360*      760 FORMAT (1H1,64H THERE WERE RECIPROCAL CROSSES.THE ORIGINAL PLANTS
361*      1 /FAMILY WERE)
362*      770 FORMAT (1H1,71H THERE WERE RECIPROCAL CROSSES.THE DATA ENTERED--
363*      1 MEANS OR SUMS--WERE)
364*      780 FORMAT (1H1,39H THE ORIGINAL DATA ENTERED AS SUMS WERE)
365*      790 FORMAT (1H1,24H THE MEAN GCA VALUES ARE)
366*      800 FORMAT (1H ,5E20.9)
367*      810 FORMAT (1H1,26H THE (2GCA+XBAR)VALUES ARE)
368*      820 FORMAT (1H ,5E20.9)
369*      830 FORMAT (1H1,20X,1HI,5X,1HJ,7X,1HN,14X,4HXBAR,14X,3HSCA//)
370*      840 FORMAT (1H ,19X,I2,4X,I2,3E16.9)
371*      C
      END

```

```

1*      C      SUBROUTINE MPRINT
2*      C
3*      C      THIS SUBROUTINE WILL PRINTOUT SQUARE MATRICES UP TO 50X50
4*      C      ALL ROWS AND COLUMNS ARE APPROPRIATELY LABELED.
5*      C      DEFINITIONS OF SUBROUTINES ARGUMENTS
6*      C      RAY(I,J) = MATRIX TO BE PRINTED
7*      C      M=ORDER(MAXIMUM OF 50)
8*      C      L=1 FOR 6E20.9 OUTPUT OR 2 FOR 10F12.2
9*      C      TI IS A SIX-CHARACTER DESIGNATE OF THE MATRIX
10*     C
11*         SUBROUTINE MPRINT (RAY,M,L,TI)
12*         DIMENSION RAY(50,50),JJ(50)
13*         IF(L-1)2,2,4
14*     2   L1=5
15*         GO TO 5
16*     4   L1 = 9
17*     5   J1 = 0
18*         J2 = 0
19*         JSEC = 0
20*         DO 8 I= 1,M
21*     8   JJ(I)=I
22*     9   J1 = J2+1
23*         J2 = J1+L1
24*         IF(J2-M)13,13,12
25*     12  J2=M
26*     13  JSEC = JSEC + 1
27*         IF(JSEC - 1) 18,18,19
28*     18  WRITE(6 ,17) TI,JSEC
29*     17  FORMAT(1H0,A6,9H SECTION ,I3/)
30*         GO TO 201
31*     19  WRITE(6 ,20) TI,JSEC
32*     20  FORMAT(1H1,A6,9H SECTION ,I3)
33*     201 IF(L-1)21,21,26
34*     21  WRITE(6 ,22) (JJ(I), I=J1,J2)
35*     22  FORMAT(6H0 POW ,3X,I12,5I20)
36*         DO 23 I=1,M
37*     23  WRITE(6,24) I, (RAY(I,K), K=J1,J2)
38*     24  FORMAT(I6,4X,6E20.9)
39*         GO TO 31
40*     26  WRITE(6 ,27) (JJ(I),I=J1,J2)
41*     27  FORMAT(6H0 ROW ,3X,10I11)
42*         DO 29 I=1,M
43*     29  WRITE(6,30) I, (RAY(I,K),K=J1,J2)
44*     30  FORMAT(I6,4X,10F11.4)
45*     31  IF(J2-M)9,32,32
46*     32  RETURN
47*         END

```

```

1*      C      SUBROUTINE MATINV MODIFIED TO HANDLE DEPENDENCIES
2*      C
3*      C      PROGRAMMED BY BURTON S. GARBOW, ARGONNE NATIONAL LABORATORY,
4*      C      AND REPORTED IN IBM 704-709 SHARE LIBRARY AS AN F402.
5*      C
6*      C      THIS SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF MATRIX A,
7*      C      OF ORDER N, BY THE GAUSS-JORDAN METHOD. A-INVERSE REPLACES A,
8*      C      AND THE DETERMINANT OF A IS PLACED IN DETERM. IF M = 1 THE
9*      C      VECTOR BB CONTAINS THE CONSTANT VECTOR WHEN MATINV IS CALLED,
10*     C      AND THIS IS REPLACED WITH THE SOLUTION VECTOR. IF M = 0, NO
11*     C      SIMULTANEOUS EQUATION SOLUTIONS ARE CALLED FOR, AND BB IS NOT
12*     C      PERTINENT. NN IS NOT TO EXCEED 50. A, NN, BB, M, AND DETERM IN
13*     C      THE ARGUMENT LIST ARE DUMMY VARIABLES.
14*     C      IORDER=1  ROWS OR COLUMNS NOT REORDERED
15*     C      =0 REORDERED
16*     C
17*     C
18*     C      SUBROUTINE MATINV (A,NN,BB,M,DETERM,NDEP,IORDER)
19*     C      SUBROUTINE MATINV DIMENSION
20*     C      DIMENSION IPIVOT(50),A(50,50),BB(50,1),INDEX(50,2),PIVOT(50)
21*     C      EQUIVALENCE (IROW,JROW), (ICOLUMN,JCOLUMN), (AMAX, TT, SWAP)
22*     C      INITIALIZATION
23*     C      NDEP= 0
24*     C      EPS= 1.0E-06
25*     C      10 DETERM=1.0
26*     C      15 DO 20 J=1,NN
27*     C      20 IPIVOT(J)=0
28*     C      30 DO 555 I = 1, NN
29*     C      SEARCH FOR PIVOT ELEMENT
30*     C      FOLLOWING ALLOWS FOR NO REORDERING
31*     C      IF(IORDER.EQ.0) GO TO 40
32*     C      IROW=I
33*     C      ICOLUMN=I
34*     C      GO TO 110
35*     C      40 AMAX=0.0
36*     C      45 DO 105 J=1,NN
37*     C      50 IF(IPIVOT(J)-1) 60, 105, 60
38*     C      60 DO 100 K=1,NN
39*     C      70 IF(IPIVOT(K)-1) 80, 100, 740
40*     C      80 IF(ABS(AMAX)-ABS(A(J,K))) 85, 100, 100
41*     C      85 IROW=J
42*     C      90 ICOLUMN=K
43*     C      95 AMAX=A(J,K)
44*     C      100 CONTINUE
45*     C      105 CONTINUE
46*     C      110 IPIVOT(ICOLUMN)=IPIVOT(ICOLUMN)+1
47*     C      INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
48*     C      130 IF(IROW-ICOLUMN) 140, 260, 140
49*     C      140 DETERM=-DETERM
50*     C      150 DO 200 L=1,NN
51*     C      160 SWAP=A(IROW,L)
52*     C      170 A(IROW,L)=A(ICOLUMN,L)
53*     C      200 A(ICOLUMN,L)=SWAP
54*     C      205 IF(M) 260, 260, 210
55*     C      210 DO 250 L=1,M
56*     C      220 SWAP=BB(IROW,L)
57*     C      230 BB(IROW,L)=BB(ICOLUMN,L)
58*     C      250 BB(ICOLUMN,L)=SWAP
59*     C      260 INDEX(I,1)=IROW
60*     C      270 INDEX(I,2)=ICOLUMN
61*     C      310 PIVOT(I)=A(ICOLUMN,ICOLUMN)
62*     C      IF(ABS(PIVOT(I)).GT.EPS) GO TO 320
63*     C      IF(M.NE.1) GO TO 600
64*     C      DO 3700 L=1,M

```

```

65*      3700 BB(ICOLUM,L)=0.
66*      DETERM=0
67*      NDEP = NDEP +1
68*      GO TO 555
69*      320 DETERM=DETERM*PIVOT(I)
70*      C   DIVIDE PIVOT ROW BY PIVOT ELEMENT
71*      330 A(ICOLUM,ICOLUM)=1.0
72*      340 DO 350 L=1,NN
73*      350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)
74*      355 IF(M) 380, 380, 360
75*      360 DO 370 L=1,M
76*      370 BB(ICOLUM,L)=BB(ICOLUM,L)/PIVOT(I)
77*      C   REDUCE NON-PIVOT ROWS
78*      380 DO 550 L1=1,NN
79*      390 IF(L1-ICOLUM) 400, 550, 400
80*      400 TT=A(L1,ICOLUM)
81*      420 A(L1,ICOLUM)=0.0
82*      430 DO 450 L=1,NN
83*      450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*TT
84*      455 IF(M) 550, 550, 460
85*      460 DO 500 L=1,M
86*      500 BB(L1,L)=BB(L1,L)-BB(ICOLUM,L)*TT
87*      550 CONTINUE
88*      555 CONTINUE
89*      C   INTERCHANGE COLUMNS
90*      600 DO 710 I=1,NN
91*      610 L=NN+1-I
92*      620 IF(INDEX(L,1)-INDEX(L,2)) 630, 710, 630
93*      630 JROW=INDEX(L,1)
94*      640 JCOLUM=INDEX(L,2)
95*      650 DO 705 K=1,NN
96*      660 SWAP=A(K,JROW)
97*      670 A(K,JROW)=A(K,JCOLUM)
98*      700 A(K,JCOLUM)=SWAP
99*      705 CONTINUE
100*     710 CONTINUE
101*     740 RETURN
102*     END

```

Snyder, E. B.

1975. Combining-ability determinations for incomplete mating designs. South. For. Exp. Stn., New Orleans, La. 12 p. (USDA For. Serv. Gen. Tech. Rep. SO-9)

It is shown how general combining ability values (GCA's) from cross-, open-, and self-pollinated progeny can be derived in a single analysis. Breeding values are employed to facilitate explaining genetic models of the expected family means and the derivation of the GCA's. A FORTRAN computer program also includes computation of specific combining ability values and several options.

Additional keywords: Diallel cross, reciprocal cross, specific combining ability, *panmixia*.

Snyder, E. B.

1975. Combining-ability determinations for incomplete mating designs. South. For. Exp. Stn., New Orleans, La. 12 p. (USDA For. Serv. Gen. Tech. Rep. SO-9)

It is shown how general combining ability values (GCA's) from cross-, open-, and self-pollinated progeny can be derived in a single analysis. Breeding values are employed to facilitate explaining genetic models of the expected family means and the derivation of the GCA's. A FORTRAN computer program also includes computation of specific combining ability values and several options.

Additional keywords: Diallel cross, reciprocal cross, specific combining ability, *panmixia*.