CONSTRUCTION OF WORKING PROBABILITIES AND JOINT SELECTION PROBABILITIES FOR FELLEGI'S PPS SAMPLING SCHEME

G.H. Choudhry[1]

A FORTRAN Subroutine to obtain the "working probabilities" for Fellegi's (1963) method of unequal probability sampling is given. The solution is obtained by an iterative procedure where the starting values for the (k+1)th draw "working probabilities" are the solutions for the kth draw "working probabilities" and the iterative procedure is terminated when a prespecified accuracy is achieved. The limitation is that the Subroutine can only be used to obtain upto and including the 5th draw "working probabilities". It was observed that the convergence occurs very fast in double precision. Therefore all real variables have been declared as double precision. The joint selection probabilities Π_{ij}'s i.e. the probability that both the ith and jth units are in the sample are obtained by summing the probabilities of selecting those samples that contain both the ith and jth units. The joint selection probabilities are required for the variance estimation of the Horvitz-Thompson estimator of population total of the characteristic of interest.

1. DESCRIPTION

Fellegi (1963) has proposed a method for selecting a sample of n (≥ 2) units draw by draw and without replacement out of N units in such a way that the probability for the i-th unit to be selected is equal to p_i at each of the n successive draws ($\sum_{i=1}^{N} p_i = 1$). This is achieved by determining (n-1) sets of selection probabilities referred to as "working probabilities". Let the (n-1) sets of "working probabilities" be

[1] G.H. Choudhry, Census and Household Survey Methods Division, Statistics Canada.

$$p_i(k) \geq 0, \; i = 1, 2, \ldots, N; \; k=2, 3, \ldots, n$$

$$\sum_{i=1}^{N} p_i(k) = 1, \; k = 2, 3, \ldots, n.$$

The $p_i(k)$, $i = 2, 3, \ldots, N$ are the "working probabilities" for selecting a unit at the k-th draw. The selection probabilities at the first draw $p_i(1)$ are given by

$$p_i(1) = p_i, \quad i = 1, 2, \ldots, N.$$

Then the overall (unconditional) probability $\delta_i(k)$ of selecting i-th unit at the k-th draw is given by

$$\delta_i(k) = \sum_{(k-1;\; i)} [p_{i_1}(1) \times \frac{p_{i_2}(2)}{1-p_{i_1}(2)} \times \ldots \times \frac{p_{i_{k-1}}(k-1)}{1-p_{i_1}(k-1)-p_{i_2}(k-1)\ldots -p_{i_{k-2}}(k-1)}$$

$$\times \frac{p_i(k)}{1-p_{i_1}(k) - p_{i_2}(k) \ldots -p_{i_{k-1}}(k)}]$$

$$i = 1, 2, \ldots, N;$$

$$k = 1, 2, \ldots, n$$

where $\sum_{(k-1;\; i)}$ denotes the summation over all possible ordered (k-1) - tuples of $(i_1, i_2, \ldots, i_{k-1})$ such that $i_1, i_2, \ldots, i_{k-1}$ are different integers between 1 and N, and none of them is equal to i. The condition that the i-th unit be selected with probability p_i at each of the n successive draws is satisfied by setting

$$\delta_i(k) = p_i, \; i = 1, 2, \ldots, N; \; k = 1, 2, \ldots, n.$$

We have $p_i(1) = p_i$, i=1, 2, ..., N. Given that $p_i(2), \ldots, p_i(k-1)$ have already been found, then approximate $p_i^{(0)}(k)$ by $p_i(k-1)$ and obtain $p_i^{(1)}(k)$ from the following formula

$$p_i^{(m)}(k) = p_i \times \left\{ \sum_{(k-1;\; i)} [p_{i_1}(1) \times \frac{p_{i_2}(2)}{1-p_{i_1}(2)} \times \ldots \times \frac{p_{i_{k-1}}(k-1)}{1-p_{i_1}(k-1)-p_{i_2}(k-1)\ldots-p_{i_k}(k-1)} \times \frac{1}{1-p_{i_1}^{(m-1)}(k)-p_{i_2}^{(m-1)}(k) \ldots - p_{i_k}^{(m-1)}(k)}] \right\}^{-1}$$

by setting m = 1 for i = 1, 2, ... N. Repeat for m = 2, 3, ..., etc. until $p_i^{(m)}(k) = p_i^{(m-1)}(k)$ for all i up to the required number of decimal places. The procedure is carried out for k = 2, 3, ...n, thus obtaining the (n-1) sets of "working probabilities" $p_i(2)$, $p_i(3)$, ..., $p_i(n)$. Since i-th unit is selected with probability equal to p_i at each of the n successive draw, this property of the scheme makes it very attractive for rotating sample designs.

Bayless and Rao (1977) excluded Felleig's (1963) method from their study for n=4 due to covergence problems with the routine they used for obtaining the "working probabilities". They were not getting satisfactory answers even after a large number of iterations especially when c.v.(x)* was not small, where x-values are the sizes of the units in the population.

$$^* \quad \text{C.V.}(x) = \frac{\sqrt{\left(\sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i\right)^2 /N\right)/ (N-1)}}{\left(\sum_{i=1}^{N} x_i /N\right)}$$

We have used Fellegi's (1963) example for which C.V. (x) is small and two populations [Cochran (1978) and Kish (1965)] with larger values for C.V.(x) to obtain the "working probabilities" for selecting upto 4 units. The iterative procedure was terminated when the change between two successive iterations was less than 10^{-6} for each element of the solution vector. The description of the populations and the number of iterations require to obtain the "working probabilities" at each of the draws is given below:

Pop. No.	Source	N	C.V.(X)	No. of iterations at draw		
				2	3	4
1	Fellegi [1963, p. 198]	6	0.25	5	7	12
2	Cochran [1978, p. 152]	20	1.03	4	5	7
3	Kish [1965, p. 42]	20	1.19	4	6	8

It is noticed that for the three populations we have used, the convergence at each of the draws is obtained in a very few number of iterations although the number of iterations required at each successive draw increases. It should be remarked that the values of "working probabilities" obtained for Fellegi's (1963) example agree with his values.

The joint selection probabilities are required for estimating the variance of the Horvitz-Thompson estimator

$$\hat{Y} = \frac{1}{n} \sum_{i \varepsilon s} \frac{y_i}{p_i}$$

of the total $Y = \sum_{i=1}^{N} y_i$ of y - variate of interest, where y_i is the value of y - variate pertaining to the i-th unit. Let π_{ij} denote the probability that both the i-th and j-th units are included in the sample, then π_{ij}, $i=1, 2, \ldots N-1$; $j= i+1, i+2, \ldots, N$ can be obtained as follows:

Let $\delta_{ij}(k, \ell)$ denote the probability that the i-th unit was selected at the k-th draw and the j-th unit was selected at the ℓth draw ($\ell > k$). The probability $\delta_{ij}(k, \ell)$ is given by:

$$\delta_{ij}(k,\ell) = \sum_{(\ell-2;i,j)} [p_{i_1}(1) \times \frac{p_{i_2}(2)}{1-p_{i_1}(2)} \times \ldots \times \frac{p_{i_{k-1}}(k-1)}{1-p_{i_1}(k-1)-p_{i_2}(k-1)\ldots -p_{i_{k-1}}(k-1)} \times$$

$$\frac{p_i(k)}{1-p_{i_1}(k)-p_{i_2}(k) \ldots - p_{i_{k-1}}(k)} \times \frac{p_{i_{k+1}}(k+1)}{1-p_{i_1}(k+1)-p_{i_2}(k+1) \ldots -p_{i_{k-1}}(k+1)-p_i(k+1)} \times$$

$$\ldots \times \frac{p_{i_{\ell-1}}(\ell-1)}{1-p_{i_1}(\ell-1)-p_{i_2}(\ell-1) \ldots -p_{i_{k-1}}(\ell-1)-p_i(\ell-1)-p_{i_{k+1}}(\ell-1) \ldots p_{i_{\ell-2}}(\ell-1)} \times$$

$$\frac{p_j(\ell)}{1-p_{i_1}(\ell) - p_{i_2}(\ell) \ldots - p_{i_{k-1}}(\ell) - p_i(\ell) - p_{i_{k+1}}(\ell) \ldots - p_{i_{\ell-1}}(\ell)}],$$

$$i \neq j = 1, 2, \ldots, N;$$

$$k = 1, 2, \ldots, n-1;$$

$$\ell = k+1, k+2, \ldots, n$$

where $\sum_{(\ell-2;i,j)}$ denotes the summation over all possible ordered $(\ell-2)$-tuples of $(i_1, i_2, \ldots, i_{k-1}, i_{k+1}, \ldots i_{\ell-1})$ such that $i_1, i_2, \ldots, i_{k-1}, i_{k+1}, \ldots, i_{\ell-1}$ are different integers between 1 and N, and none of them is equal to i or j. Then π_{ij}, the probability that both i-th and j-th units are included in the sample, is given by

$$\pi_{ij} = \sum_{k=1}^{n-1} \sum_{\ell=k+1}^{n} [\delta_{ij}(k,\ell) + \delta_{ji}(k,\ell)],$$

$$i = 1, 2, \ldots, N-1;$$

$$j = i+1, i+2, \ldots, N.$$

Structure

SUBROUTINE WKPROB (N, NS, MA, P, P1, P2, P3, P4, Q1, Q2, DEL, MAX, ACC, PI, TOL, IFAULT)

Formal parameters - all real parameters in double precision.

N	Integer	Input:	number of units in the population
NS	Integer	Input:	sample size, $2 \le NS \le 5$
MA	Integer	Input:	dimension of PI in the calling program
P	Real Array(N)	Input:	contains the relative measure of sizes of units in the sequence $p_1, p_2, \ldots, p_N$; $\sum_{i=1}^{N} p_i = 1$
P1	Real Array (N)	Output:	working probabilities for selecting a unit at the 2nd draw
P2	Real Array (N)	Output:	working probabilities for selecting a unit at the 3rd draw
P3	Real Array (N)	Output:	working probabilities for selecting a unit at the 4th draw

P4	Real Array(N)	Output:	working probabilities for selecting a unit at the 5th draw.
Q1	Real Array(N)	Workspace	
Q2	Real Array(N)	Workspace	
DEL	Real Array (MA,NS)	Workspace	
MAX	Integer	Input:	maximum number of interations allowed for obtaining each set of working probabilities
ACS	Real	Input:	desired accuracy of the working probabilities
P1	Real Array (MA,MA)	Output:	matrix returning the joint selection probabilities π_{ij}, $i = 1, 2, \ldots, N-1$ $j = i + 1, i + 2, \ldots N$
TOL	Real	Input:	maximum allowed value for the absolute difference between $\sum_{i=1}^{N} p_i$ and the number one
IFAULT	Integer	Output:	failure indicator

Failure Indications

IFAULT = 0 normal termination

= 1 one or more of $p_i > (1/NS)$

= 2 $\text{DABS}(\sum_{i=1}^{N} p_i - 1.0) > \text{TOL}$

= 3 both conditions 1 and 2 occur

= 4 sample size greater than 5

= 5 desired accuracy was not obtained in maximum allowed number of interations

ACKNOWLEDGEMENT

The author wishes to thank the referee for some helpful comments.

RESUME

L'auteur expose un sous-programme FORTRAN visant à obtenir les "probabilités de travail" à l'aide de la méthode d'échantillonnage à probabilités inégales de Fellegi (1963). On obtient la solution par une méthode itérative dans laquelle les valeurs de départ des "probabilités de travail" du (k 1)-ième tirage sont la solution du k-ième tirage des "probabilités de travail"; ce calcul prend fin lorsque l'on atteint un niveau de précision déterminé à l'avance. Le sous-programme est limité car son utilisation ne peut dépasser le 5e tirage des "probabilités de travail". On a observé que la convergence se produit très rapidement en double précision. Par conséquent, toutes les variables réelles ont été déclarées en double précision. Les probabilités conjointes de sélection, c.-à-d. la probabilité que les i-ième et j-ième unités fassent toutes deux partie de l'échantillon, s'obtiennent par sommation des probabilités de sélection des échantillons contenant les deux unités en cause. Les probabilités conjointes de sélection sont nécessaires à l'estimation de la variance de l'estimateur Horvitz-Thompson du total de la caractéristique à l'étude dans la population.

REFERENCES

[1] Bayless, D.L. and Rao, J.N.K. (1970), "An empirical study of Stabilities of Estimators and Variance Estimators in Unequal Probability Sampling (n=3 or 4)", Journal of the American Statistical Association, 65, 1645-1667.

[2] Cochran, W.G. (1977), Sampling Techniques, 3rd Ed., New York: John Wiley and Sons.

[3] Fellegi, I.P. (1963), "Sampling with and without Replacement: Rotating and Non-Rotating Samples", Journal of the American Statistical Association, 58, 183-201.

[4] Kish, L. (1965), Survey Sampling, New York: John Wiley and Sons.

```
C.....SUBROUTINE TO OBTAIN WORKING PROBABILITIES AND
C.....JOINT SELECTION PROBABILITIES FOR FELLEGI'S PPS
C.....SAMPLING SCHEME. REF: 1963 JASA 58 , PP 183-201 .
C.....
      SUBROUTINE WKPROB(N,NS,MA,P,P1,P2,P3,P4,Q1,Q2,DEL,MAX,ACC,
     1           PI,TOL,IFAULT)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION P(N),P1(N),P2(N),P3(N),P4(N),Q1(N),Q2(N),DEL(MA,NS),
     1          PI(MA,MA)
C.....N      IS POPULATION SIZE.
C.....NS     IS SAMPLE SIZE AND CAN HAVE VALUES 2 , 3 , 4 , 5 .
C.....MA     IS MAXIMUM DIMENSION  OF  PI IN THE MAIN PROGRAM.
C.....P      IS THE VECTOR OF  GIVEN PROBABILITIES . SUM P = 1.0
C.....P , P1 , P2 , P3 , P4 AND P5 ARE SELECTION PROBABILITIES AT
C.....1ST , 2ND , 3RD , 4-TH AND 5-TH DRAWS RESPECTIVELY .
C.....Q1 , Q2 , DEL .... WORK SPACE .
      CONTINUE
C.....MAX IS THE MAXIMUM NUMBER OF ITERATIONS ALLOWED TO OBTAIN
C.....THE WORKING PROBABILITIES.
C.....ACC IS THE DESIRED ACCURACY OF THE WORKING PROBABILITIES.
C.....PI IS THE OUTPUT RETURNING THE JOINT SELECTION PROBABLITIES .
C.....TOL IS THE PARAMETER SO THAT SUM P CANNOT DEVIATE FROM 1.0 BY
C.....MORE THAN THE VALUE OF TOL.
C.....IFAULT IS  FAILURE INDICATOR TAKING THE FOLLOWING VALUES:
C.....
C..... 0           IF PI COMPUTED, NORMAL TERMINATION.
C..... 1           IF NS*P .GE.1.0   FOR ONE OR MORE P VALUES .
C..... 2           IF DABS(SUM P - 1.0) IS GREATER THAN TOL.
C..... 3           IF BOTH OF THE ABOVE TWO CONDITIONS.
C..... 4           IF NS , THE SAMPLE SIZE, IS GREATER THAN 5 .
C..... 5           IF DESIRED ACCURACY NOT OBTAINED IN MAXIMUM
C.....             ALLOWED NUMBER OF ITERATIONS.
C.....
      IFAULT=4
      IF(NS.GT.5) RETURN
      IFAUL1=0
      IFAUL2=0
      IDRAW=1
      XNS=NS
      SUMP=0.0
      DO 1 I=1,N
      SUMP=SUMP+P(I)
      Q1(I)=P(I)
      DEL(I,1)=P(I)
      IF(XNS*P(I).GT.1.0) IFAUL1=1
    1 CONTINUE
      IF(DABS(SUMP-1.0).GT.TOL) IFAUL2=2
      IFAULT=IFAUL1+IFAUL2
      IF(IFAULT.NE.0) RETURN
C.....
```

```
C.....SELECTING UNIT 2 .
C.....
      IDRAW=IDRAW+1
      A=0.0
      DO 20 J=1,N
   20 A=A+F0(N,J,P,Q1)
      ICOUNT=0
   21 ICOUNT=ICOUNT+1
      IF(ICOUNT.GT.MAX) GO TO 999
      DMAX=0.0
      DO 22 I=1,N
      DEN=A-F0(N,I,P,Q1)
      Q2(I)=P(I)/DEN
      DIFF=DABS(Q2(I)-Q1(I))
      IF(DIFF.GT.DMAX) DMAX=DIFF
      Q1(I)=Q2(I)
      A=DEN+F0(N,I,P,Q1)
   22 CONTINUE
      IF(DMAX.GT.ACC) GO TO 21
      WRITE(3,24) IDRAW,ICOUNT
   24 FORMAT(1H1,////,20X,'WORKING PROBABILITIES AT DRAW        : ',I5,
     1           ////,20X,'NUMBER OF ITERATIONS FOR CONVERGENCE = ',I6,
     2           ////)
      DO 25 I=1,N
      P1(I)=Q1(I)
      DEL(I,2)=P1(I)
      WRITE(3,26) I,P1(I)
   25 CONTINUE
   26 FORMAT(1H0,20X,' PROB ( ',I2,' ) = ',D14.6)
      IF(IDRAW.EQ.NS) GO TO 550
C.....
C.....SELECTING UNIT 3.
C.....
      IDRAW=IDRAW+1
      A=0.0
      DO 30 J=1,N
      DO 30 K=1,N
   30 A=A+F1(N,J,K,P,P1,Q1)
      ICOUNT=0
   37 ICOUNT=ICOUNT+1
      IF(ICOUNT.GT.MAX) GO TO 999
      DMAX=0.0
      DO 31 I=1,N
      S1=0.0
      DO 32 J=1,N
      S1=S1+F1(N,I,J,P,P1,Q1)+F1(N,J,J,P,P1,Q1)+F1(N,J,I,P,P1,Q1)
   32 CONTINUE
      DEN=A-S1+2.0*F1(N,I,I,P,P1,Q1)
      Q2(I)=P(I)/DEN
      DIFF=DABS(Q2(I)-Q1(I))
```

```
      IF(DIFF.GT.DMAX) DMAX=DIFF
      Q1(I)=Q2(I)
      S1=0.0
      DO 33 J=1,N
      S1=S1+F1(N,I,J,P,P1,Q1)+F1(N,J,J,P,P1,Q1)+F1(N,J,I,P,P1,Q1)
   33 CONTINUE
      A=DEN+S1-2.0*F1(N,I,I,P,P1,Q1)
   31 CONTINUE
      IF(DMAX.GT.ACC) GO TO 37
      WRITE(3,24) IDRAW,ICOUNT
      DO 36 I=1,N
      P2(I)=Q1(I)
      DEL(I,3)=P2(I)
      WRITE(3,26) I,P2(I)
   36 CONTINUE
      IF(IDRAW.EQ.NS) GO TO 550
C.....
C.....SELECTING UNIT 4.
C.....
      IDRAW=IDRAW+1
      A=0.0
      DO 40 J=1,N
      DO 40 K=1,N
      DO 40 L=1,N
   40 A=A+F2(N,J,K,L,P,P1,P2,Q1)
      ICOUNT=0
   49 ICOUNT=ICOUNT+1
      IF(ICOUNT.GT.MAX) GO TO 999
      DMAX=0.0
      DO 41 I=1,N
      S1=0.0
      S2=0.0
      DO 42 J=1,N
      DO 43 K=1,N
      S1=S1+F2(N,I,J,K,P,P1,P2,Q1)+F2(N,J,J,K,P,P1,P2,Q1)
     1     +F2(N,J,I,K,P,P1,P2,Q1)+F2(N,J,K,I,P,P1,P2,Q1)
     2     +F2(N,J,K,J,P,P1,P2,Q1)+F2(N,J,K,K,P,P1,P2,Q1)
   43 CONTINUE
      S2=S2+2.0*F2(N,I,I,J,P,P1,P2,Q1)+F2(N,J,J,I,P,P1,P2,Q1)
     1     +2.0*F2(N,I,J,I,P,P1,P2,Q1)+F2(N,J,I,J,P,P1,P2,Q1)
     2     +2.0*F2(N,J,I,I,P,P1,P2,Q1)+F2(N,I,J,J,P,P1,P2,Q1)
     3     +2.0*F2(N,J,J,J,P,P1,P2,Q1)
   42 CONTINUE
      DEN=A-S1+S2-6.0*F2(N,I,I,I,P,P1,P2,Q1)
      Q2(I)=P(I)/DEN
      DIFF=DABS(Q2(I)-Q1(I))
      IF(DIFF.GT.DMAX) DMAX=DIFF
      Q1(I)=Q2(I)
      S1=0.0
      S2=0.0
```

```
      DO 44 J=1,N
      DO 45 K=1,N
      S1=S1+F2(N,I,J,K,P,P1,P2,Q1)+F2(N,J,J,K,P,P1,P2,Q1)
     1     +F2(N,J,I,K,P,P1,P2,Q1)+F2(N,J,K,I,P,P1,P2,Q1)
     2     +F2(N,J,K,J,P,P1,P2,Q1)+F2(N,J,K,K,P,P1,P2,Q1)
   45 CONTINUE
      S2=S2+2.0*F2(N,I,I,J,P,P1,P2,Q1)+F2(N,J,J,I,P,P1,P2,Q1)
     1     +2.0*F2(N,I,J,I,P,P1,P2,Q1)+F2(N,J,I,J,P,P1,P2,Q1)
     2     +2.0*F2(N,J,I,I,P,P1,P2,Q1)+F2(N,I,J,J,P,P1,P2,Q1)
     3     +2.0*F2(N,J,J,J,P,P1,P2,Q1)
   44 CONTINUE
      A=DEN+S1-S2+6.0*F2(N,I,I,I,P,P1,P2,Q1)
   41 CONTINUE
      IF(DMAX.GT.ACC) GO TO 49
      WRITE(3,24) IDRAW,ICOUNT
      DO 47 I=1,N
      P3(I)=Q1(I)
      DEL(I,4)=P3(I)
      WRITE(3,26) I,P3(I)
   47 CONTINUE
      IF(IDRAW.EQ.NS) GO TO 550
C.....
C.....SELECTING UNIT 5.
C.....
      IDRAW=IDRAW+1
      A=0.0
      DO 50 J=1,N
      DO 50 K=1,N
      DO 50 L=1,N
      DO 50 M=1,N
   50 A=A+F3(N,J,K,L,M,P,P1,P2,P3,Q1)
      ICOUNT=0
   59 ICOUNT=ICOUNT+1
      IF(ICOUNT.GT.MAX) GO TO 999
      DMAX=0.0
      DO 51 I=1,N
      S1=0.0
      S2=0.0
      S3=0.0
      DO 52 J=1,N
      DO 53 K=1,N
      DO 54 L=1,N
      S1=S1+F3(N,J,K,I,L,P,P1,P2,P3,Q1)+F3(N,J,K,J,L,P,P1,P2,P3,Q1)
     1     +F3(N,J,K,K,L,P,P1,P2,P3,Q1)+F3(N,I,J,K,L,P,P1,P2,P3,Q1)
     2     +F3(N,J,J,K,L,P,P1,P2,P3,Q1)+F3(N,J,I,K,L,P,P1,P2,P3,Q1)
     3     +F3(N,J,K,L,I,P,P1,P2,P3,Q1)+F3(N,J,K,L,J,P,P1,P2,P3,Q1)
     4     +F3(N,J,K,L,K,P,P1,P2,P3,Q1)+F3(N,J,K,L,L,P,P1,P2,P3,Q1)
   54 CONTINUE
      S2=S2+2.0*F3(N,I,J,I,K,P,P1,P2,P3,Q1)+F3(N,J,J,I,K,P,P1,P2,P3,Q1)
     1     +2.0*F3(N,J,I,I,K,P,P1,P2,P3,Q1)+F3(N,J,I,J,K,P,P1,P2,P3,Q1)
```

```
     2      +2.0*F3(N,J,J,J,K,P,P1,P2,P3,Q1)+F3(N,I,J,J,K,P,P1,P2,P3,Q1)
     3      +2.0*F3(N,I,I,J,K,P,P1,P2,P3,Q1)+F3(N,J,K,J,I,P,P1,P2,P3,Q1)
     4      +2.0*F3(N,J,K,I,I,P,P1,P2,P3,Q1)+F3(N,J,K,K,I,P,P1,P2,P3,Q1)
     5      +2.0*F3(N,I,J,K,I,P,P1,P2,P3,Q1)+F3(N,J,J,K,I,P,P1,P2,P3,Q1)
     6      +2.0*F3(N,J,I,K,I,P,P1,P2,P3,Q1)+F3(N,J,K,I,J,P,P1,P2,P3,Q1)
     7      +2.0*F3(N,J,K,J,J,P,P1,P2,P3,Q1)+F3(N,J,K,K,J,P,P1,P2,P3,Q1)
     8      +2.0*F3(N,J,J,K,J,P,P1,P2,P3,Q1)+F3(N,J,I,K,J,P,P1,P2,P3,Q1)
     9      +    F3(N,J,K,I,K,P,P1,P2,P3,Q1)+F3(N,J,K,J,K,P,P1,P2,P3,Q1)
     A      +2.0*F3(N,J,K,K,K,P,P1,P2,P3,Q1)+F3(N,I,J,K,J,P,P1,P2,P3,Q1)
     B      +    F3(N,I,J,K,K,P,P1,P2,P3,Q1)+F3(N,J,J,K,K,P,P1,P2,P3,Q1)
     C      +    F3(N,J,I,K,K,P,P1,P2,P3,Q1)
  53 CONTINUE
      S3=S3+6.0*F3(N,I,I,I,J,P,P1,P2,P3,Q1)
     1      +6.0*F3(N,I,J,I,I,P,P1,P2,P3,Q1)
     2      +2.0*F3(N,J,J,I,I,P,P1,P2,P3,Q1)
     3      +6.0*F3(N,J,I,I,I,P,P1,P2,P3,Q1)
     4      +2.0*F3(N,J,J,J,I,P,P1,P2,P3,Q1)
     5      +2.0*F3(N,J,I,J,I,P,P1,P2,P3,Q1)
     6      +2.0*F3(N,I,J,J,I,P,P1,P2,P3,Q1)
     7      +6.0*F3(N,I,I,J,I,P,P1,P2,P3,Q1)
     8      +2.0*F3(N,J,J,I,J,P,P1,P2,P3,Q1)
     9      +2.0*F3(N,J,I,I,J,P,P1,P2,P3,Q1)
     A      +6.0*F3(N,J,J,J,J,P,P1,P2,P3,Q1)
     B      +2.0*F3(N,J,I,J,J,P,P1,P2,P3,Q1)
     C      +2.0*F3(N,I,J,I,J,P,P1,P2,P3,Q1)
     D      +2.0*F3(N,I,J,J,J,P,P1,P2,P3,Q1)
     E      +2.0*F3(N,I,I,J,J,P,P1,P2,P3,Q1)
  52 CONTINUE
      DEN=A-S1+S2-S3+24.0*F3(N,I,I,I,I,P,P1,P2,P3,Q1)
      Q2(I)=P(I)/DEN
      DIFF=DABS(Q2(I)-Q1(I))
      IF(DIFF.GT.DMAX) DMAX=DIFF
      Q1(I)=Q2(I)
      S1=0.0
      S2=0.0
      S3=0.0
      DO 55 J=1,N
      DO 56 K=1,N
      DO 57 L=1,N
      S1=S1+F3(N,J,K,I,L,P,P1,P2,P3,Q1)+F3(N,J,K,J,L,P,P1,P2,P3,Q1)
     1      +F3(N,J,K,K,L,P,P1,P2,P3,Q1)+F3(N,I,J,K,L,P,P1,P2,P3,Q1)
     2      +F3(N,J,J,K,L,P,P1,P2,P3,Q1)+F3(N,J,I,K,L,P,P1,P2,P3,Q1)
     3      +F3(N,J,K,L,I,P,P1,P2,P3,Q1)+F3(N,J,K,L,J,P,P1,P2,P3,Q1)
     4      +F3(N,J,K,L,K,P,P1,P2,P3,Q1)+F3(N,J,K,L,L,P,P1,P2,P3,Q1)
  57 CONTINUE
      S2=S2+2.0*F3(N,I,J,I,K,P,P1,P2,P3,Q1)+F3(N,J,J,I,K,P,P1,P2,P3,Q1)
     1      +2.0*F3(N,J,I,I,K,P,P1,P2,P3,Q1)+F3(N,J,I,J,K,P,P1,P2,P3,Q1)
     2      +2.0*F3(N,J,J,J,K,P,P1,P2,P3,Q1)+F3(N,I,J,J,K,P,P1,P2,P3,Q1)
     3      +2.0*F3(N,I,I,J,K,P,P1,P2,P3,Q1)+F3(N,J,K,J,I,P,P1,P2,P3,Q1)
     4      +2.0*F3(N,J,K,I,I,P,P1,P2,P3,Q1)+F3(N,J,K,K,I,P,P1,P2,P3,Q1)
```

```
     5     +2.0*F3(N,I,J,K,I,P,P1,P2,P3,Q1)+F3(N,J,J,K,I,P,P1,P2,P3,Q1)
     6     +2.0*F3(N,J,I,K,I,P,P1,P2,P3,Q1)+F3(N,J,K,I,J,P,P1,P2,P3,Q1)
     7     +2.0*F3(N,J,K,J,J,P,P1,P2,P3,Q1)+F3(N,J,K,K,J,P,P1,P2,P3,Q1)
     8     +2.0*F3(N,J,J,K,J,P,P1,P2,P3,Q1)+F3(N,J,I,K,J,P,P1,P2,P3,Q1)
     9     +    F3(N,J,K,I,K,P,P1,P2,P3,Q1)+F3(N,J,K,J,K,P,P1,P2,P3,Q1)
     A     +2.0*F3(N,J,K,K,K,P,P1,P2,P3,Q1)+F3(N,I,J,K,J,P,P1,P2,P3,Q1)
     B     +    F3(N,I,J,K,K,P,P1,P2,P3,Q1)+F3(N,J,J,K,K,P,P1,P2,P3,Q1)
     C     +    F3(N,J,I,K,K,P,P1,P2,P3,Q1)
   56 CONTINUE
      S3=S3+6.0*F3(N,I,I,I,J,P,P1,P2,P3,Q1)
     1     +6.0*F3(N,I,J,I,I,P,P1,P2,P3,Q1)
     2     +2.0*F3(N,J,J,I,I,P,P1,P2,P3,Q1)
     3     +6.0*F3(N,J,I,I,I,P,P1,P2,P3,Q1)
     4     +2.0*F3(N,J,J,J,I,P,P1,P2,P3,Q1)
     5     +2.0*F3(N,J,I,J,I,P,P1,P2,P3,Q1)
     6     +2.0*F3(N,I,J,J,I,P,P1,P2,P3,Q1)
     7     +6.0*F3(N,I,I,J,I,P,P1,P2,P3,Q1)
     8     +2.0*F3(N,J,J,I,J,P,P1,P2,P3,Q1)
     9     +2.0*F3(N,J,I,I,J,P,P1,P2,P3,Q1)
     A     +6.0*F3(N,J,J,J,J,P,P1,P2,P3,Q1)
     B     +2.0*F3(N,J,I,J,J,P,P1,P2,P3,Q1)
     C     +2.0*F3(N,I,J,I,J,P,P1,P2,P3,Q1)
     D     +2.0*F3(N,I,J,J,J,P,P1,P2,P3,Q1)
     E     +2.0*F3(N,I,I,J,J,P,P1,P2,P3,Q1)
   55 CONTINUE
      A=DEN+S1-S2+S3-24.0*F3(N,I,I,I,I,P,P1,P2,P3,Q1)
   51 CONTINUE
      IF(DMAX.GT.ACC) GO TO 59
      WRITE(3,24) IDRAW,ICOUNT
      DO 60 I=1,N
      P4(I)=Q1(I)
      DEL(I,5)=P4(I)
      WRITE(3,26) I,P4(I)
   60 CONTINUE
  550 CONTINUE
C.....CALCULATE THE JOINT SELECTION PROBABILITIES .
      DO 551 I=1,N
      DO 552 J=1,N
      IF(J.EQ.I) GO TO 552
      S1=0.0
      S2=0.0
      S3=0.0
      T1=DEL(I,1)*DEL(J,2)/(1.0-DEL(I,2))
      IF(NS.EQ.2) GO TO 590
      DO 553 K=1,N
      IF(K.EQ.I.OR.K.EQ.J) GO TO 553
      SN=DEL(I,1)*DEL(K,2)*DEL(J,3)
      SD=(1.0-DEL(I,2))*(1.0-DEL(I,3)-DEL(K,3))
      T2=SN/SD
      SN=DEL(K,1)*DEL(I,2)*DEL(J,3)
```

```
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(I,3))
      T3=SN/SD
      S1=S1+T2+T3
      IF(NS.EQ.3) GO TO 553
      DO 554 L=1,N
      IF(L.EQ.I.OR.L.EQ.J.OR.L.EQ.K) GO TO 554
      SN=DEL(I,1)*DEL(K,2)*DEL(L,3)*DEL(J,4)
      SD=(1.0-DEL(I,2))*(1.0-DEL(I,3)-DEL(K,3))
     1 *(1.0-DEL(I,4)-DEL(K,4)-DEL(L,4))
      T4=SN/SD
      SN=DEL(K,1)*DEL(I,2)*DEL(L,3)*DEL(J,4)
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(I,3))
     1 *(1.0-DEL(K,4)-DEL(I,4)-DEL(L,4))
      T5=SN/SD
      SN=DEL(K,1)*DEL(L,2)*DEL(I,3)*DEL(J,4)
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(L,3))
     1 *(1.0-DEL(K,4)-DEL(L,4)-DEL(I,4))
      T6=SN/SD
      S2=S2+T4+T5+T6
      IF(NS.EQ.4) GO TO 554
      DO 555 M=1,N
      IF(M.EQ.I.OR.M.EQ.J.OR.M.EQ.K.OR.M.EQ.L) GO TO 555
      SN=DEL(I,1)*DEL(K,2)*DEL(L,3)*DEL(M,4)*DEL(J,5)
      SD=(1.0-DEL(I,2))*(1.0-DEL(I,3)-DEL(K,3))
     1 *(1.0-DEL(I,4)-DEL(K,4)-DEL(L,4))
     2 *(1.0-DEL(I,5)-DEL(K,5)-DEL(L,5)-DEL(M,5))
      T7=SN/SD
      SN=DEL(K,1)*DEL(I,2)*DEL(L,3)*DEL(M,4)*DEL(J,5)
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(I,3))
     1 *(1.0-DEL(K,4)-DEL(I,4)-DEL(L,4))
     2 *(1.0-DEL(K,5)-DEL(I,5)-DEL(L,5)-DEL(M,5))
      T8=SN/SD
      SN=DEL(K,1)*DEL(L,2)*DEL(I,3)*DEL(M,4)*DEL(J,5)
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(L,3))
     1 *(1.0-DEL(K,4)-DEL(L,4)-DEL(I,4))
     2 *(1.0-DEL(K,5)-DEL(L,5)-DEL(I,5)-DEL(M,5))
      T9=SN/SD
      SN=DEL(K,1)*DEL(L,2)*DEL(M,3)*DEL(I,4)*DEL(J,5)
      SD=(1.0-DEL(K,2))*(1.0-DEL(K,3)-DEL(L,3))
     1 *(1.0-DEL(K,4)-DEL(L,4)-DEL(M,4))
     2 *(1.0-DEL(K,5)-DEL(L,5)-DEL(M,5)-DEL(I,5))
      TA=SN/SD
      S3=S3+T7+T8+T9+TA
  555 CONTINUE
  554 CONTINUE
  553 CONTINUE
  590 PI(I,J)=T1+S1+S2+S3
  552 CONTINUE
  551 CONTINUE
      N1=N-1
```

```
      DO 556 I=1,N1
      J1=I+1
      DO 557 J=J1,N
      PI(I,J)=PI(I,J)+PI(J,I)
  557 CONTINUE
  556 CONTINUE
      RETURN
  999 IFAULT=5
      WRITE(3,1000) IDRAW,MAX
 1000 FORMAT(1H1,////,20X,'DRAW  ',I2,'  DID NOT CONVERGE IN  ',
     1       I4,'  ITERATIONS .')
      RETURN
      END
      DOUBLE PRECISION FUNCTION F0(N,J,P,Q1)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION P(N),Q1(N)
      F0=P(J)/(1.0-Q1(J))
      RETURN
      END
      DOUBLE PRECISION FUNCTION F1(N,J,K,P,P1,Q1)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION P(N),P1(N),Q1(N)
      F1=P(J)*P1(K)/((1.0-P1(J))*(1.0-Q1(J)-Q1(K)))
      RETURN
      END
      DOUBLE PRECISION FUNCTION F2(N,J,K,L,P,P1,P2,Q1)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION P(N),P1(N),P2(N),Q1(N)
      F2=P(J)*P1(K)*P2(L)/((1.0-P1(J))*(1.0-P2(J)-P2(K))
     1                    *(1.0-Q1(J)-Q1(K)-Q1(L)))
      RETURN
      END
      DOUBLE PRECISION FUNCTION F3(N,J,K,L,M,P,P1,P2,P3,Q1)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION P(N),P1(N),P2(N),P3(N),Q1(N)
      F3=P(J)*P1(K)*P2(L)*P3(M)/((1.0-P1(J))*(1.0-P2(J)-P2(K))
     1                          *(1.0-P3(J)-P3(K)-P3(L))
     2                          *(1.0-Q1(J)-Q1(K)-Q1(L)-Q1(M)))
      RETURN
      END
```