```
      CALL NUMBER(1.7, 9.7,.18,ANIM,0,4HF4.0)
      CALL SYMBOL(2.5, 9.7,.18,4HFROM,0,4)
      CALL NUMBER(3.2, 9.7,.18,AM1,0,4HF3.0)
      CALL SYMBOL(3.7, 9.7,.18,1H/,0,1)
      CALL NUMBER(3.8, 9.7,.18,DA1,0,4HF3.0)
      CALL SYMBOL(4.2, 9.7,.18,1H/,0,1)
      CALL NUMBER(4.4, 9.7,.18,T1,0,4HF5.0)
      CALL SYMBOL(5.3, 9.7,.18,2HTO,0,2)
      CALL NUMBER(5.6, 9.7,.18,AME,0,4HF3.0)
      CALL SYMBOL(6.1, 9.7,.18,1H/,0,1)
      CALL NUMBER(6.2, 9.7,.18,DAE,0,4HF3.0)
      CALL SYMBOL(6.6, 9.7,.18,1H/,0,1)
      CALL NUMBER(6.7, 9.7,.18,TE,0,4HF5.0)
      CALL SYMBOL(7.7, 9.7,.18,2H19,0,2)
      CALL NUMBER(7.9, 9.7,.18,YEAR,0,4HF3.0)
      CALL PLOT(12.0,0.0,-3)
      GO TO 1
   95 CALL PLOTS(-58,1)
      STOP $ END $ END
```


This program uses the simulation proceedure outlined in Fig. 2 and then draws the graphic display generated by the given set of distributions. Examples of the output are given in Figs. 9, 10 and 11. The input to this program consists of one parameter card followed by the distributions for angle between successive positions, distance traveled, length of move and length of rest. These are read under the format numbers 1,2,3,300, and 414 respectively. The parameter card requires the following information in the apropriate columns.

Columns 1-5 The beginning x-coordinate for the simulation plot.

Columns 6-10 The beginning y-coordinate for the simulation plot.

Columns 21-25 The number of positions that a given simulation is to run.

Columns 26-30 If this number is greater than zero the program will continue to the next set of distribution cards, if not it will be directed to stop.

Columns 31-35 The constant that determines the scale of the distance traveled. At this time this number should equal 25.0 or 50.0, and if the value is 50.0 then the smallest distance is 50 feet and every other distance moved will be some multiple of 50. The same pattern will be true if the value 25.0 is used.

Donald Blair Siniff, "A Simulation Model of Animal Movement Patterns," Ph.D. thesis, University of Minnesota, 1967.

Columns 41-50 Determinesthe type of map presentation given. If a -10 is placed in columns 48,49 and 50 the map produced will have a small cross at each position. If a -20 is placed in columns 48, 49 and 50 the resulting map will have a small cross at each position and the positions will be connected by a continuous line. If a -30 is placed in these columns only a continuous line will connect the simulated positions(Fig. 9)

Columns 51-55 This code controls the size of the scale of the map. If a -10 is placed in columns 53,54 and 55 the smallest possible scale will be 400 feet equals one inch; however if the given plot will not fit on this scale the program will search upward through the scales 0.25 mile equals one inch, 0.50 mile equals one inch and 1.0 mile equals one inch. If the plot will not fit on any of these a printed output will indicate that it is out of the range of the plot program. The values -20, -30, and -40 will yield lower limits on the scale of 0.25 mile equals one inch, 0.50 mile equals one inch and 1.0 mile equals one inch, respectively.

Columns 56-60 This code controls whether the first choice in the simulation procedure is move or rest. If 10 is placed in columns 59 and 60 the first choice will be move. If any other value is used the first choice will be rest.

Columns 61-65 This code determines the time between successive positions. For example if one is sumulating 10 minutes between positions then a 10 should be placed in columns 64 and 65.

This program is written to be run on the University of Minnesota's 1604 with the associated Cal Comp plotter. Thus the functions used are appropriate for that particular complier and it is unlikely that this program will work on any other computer complex without modification.

```
      PROGRAM SIMPLOT
      DIMENSION REST(21),AMOVE(21)
      DIMENSION U(11),V(11),S(11),D(11),W(3005),Z(3005),E(12)
      DIMENSION XA(3005),YA(3005),ANG(40),DIST(60)
      CALL PLOTS(58,1) $ CALL PLOT(3.0,0.45,-3)
    2 FORMAT(13F6.5)
 5005 FORMAT(11F8.2,I10)
    3 FORMAT(16F5.4)
    1 FORMAT(11F5.1,I5)
   33 FORMAT(10X,7F10.5)
   35 FORMAT(10X,7F10.4)
  300 FORMAT(16F5.4)
  303 FORMAT(10F10.4)
  414 FORMAT(16F5.4)
  304 FORMAT(10F10.5)
 4607 FORMAT(10X,5F20.5)
 5500 FORMAT(1H1,10X,32HOUT OF RANGE OF PLOTTING PROGRAM)
 5501 FORMAT(3F20.3)
  500 READ 1,A,B,DA,EA,F,AA,BB,FF,HA,FA,P,MN
      PRINT 5005,A,B,DA,EA,F,AA,BB,FF,HA,FA,P,MN
      IF(AA.EQ.0.0)503,502
  502 READ 2,(ANG(I),I=1,36)
      READ 3,(DIST(I),I=1,55)
      PRINT 33,(ANG(I),I=1,36)
      PRINT 35,(DIST(I),I=1,55)
      READ 300,(AMOVE(I),I=1,20)
      PRINT 303,(AMOVE(I),I=1,20)
```



```
      READ 414,(REST(I),I=1,20)
      PRINT 304,(REST(I),I=1,20)
   37 CC=F $ K=1 $ A1=0.0 $ L=0
  407 IF(P.EQ.1.0)402,403
  402 RN=RANF(-1) $ M=1
  406 RN=RN-AMOVE(M) $ IF(RN)404,404,405
  405 M=M+1 $ GO TO 406
  404 NUM=M $ MIN=(NUM*30)+15 $ GO TO 23
  403 RN=RANF(-1) $ N=1
  410 RN=RN-REST(N) $ IF(RN)408,408,409
  409 N=N+1 $ GO TO 410
  408 NU=N $ MI=(NU*30)+15
  411 XA(K)=A $ YA(K)=B
      K=K+1
      MI=MI-MN $ L=F $ IF(K.GT.L)40,412
  412 IF(MI)402,402,411
   23 RN=RANF(-1) $ I=1 $ J=1
    7 RN=RN-ANG(I) $ IF(RN)6,6,5
    5 I=I+1 $ GO TO 7
    6 AN=I*5+((I-1)*5)
      IF(AN.LE.180.0)208,201
  201 IF(AN.LE.360.0)202,503
  208 AN=180.0-AN+A1 $ GO TO 203
  202 AN=540.0-AN+A1
  203 IF(AN.GE.360.0)204,205
  204 AN=AN-360.0
  205 A1=AN $ AN=AN/57.29578 $ RN=RANF(-1)
```

```
 10 RN=RN-DIST(J) $ IF(RN)9,9,8
  8 J=J+1 $ GO TO 10
  9 IF(BB.EQ.50.0)11,12
 11 R=J*50+((J-1)*50)  $ R=R/5280.0 $ GO TO 13
 12 R=J*25+((J-1)*25)  $ R=R/5280.0
 13 A=A+R*COSF(AN) $ B=B+R*SINF(AN) $ XA(K)=A $ YA(K)=B
    K=K+1
    L=F $ MIN=MIN-MN $ IF(K.GT.L)40,413
413 IF(MIN)403,403,23
 40 DO 44 I=1,L $ TEMP=XA(I) $ W(I)=TEMP $ TEMP=YA(I)
    Z(I)=TEMP
 44 CONTINUE
    DO 53 I=1,L   $ IP=I+1 $ DO 53 J=IP,L
    IF(W(I).GT.W(J))54,53
 54 TEMP=W(I) $ W(I)=W(J) $ W(J)=TEMP
 53 CONTINUE
    DO 55 I=1,L   $ IP=I+1 $ DO 55 J=IP,L
    IF(Z(I).GT.Z(J))56,55
 56 TEMP=Z(I) $ Z(I)=Z(J) $ Z(J)=TEMP
 55 CONTINUE
    DIFX=W(L)-W(1) $ DIFY=Z(L)-Z(1) $ IF(DIFX-DIFY)60,61,61
 61 ALENG=DIFX $ GO TO 62
 60 ALENG=DIFY
 62 XMIN=W(1) $ YMIN=Z(1) $ IXMIN=XMIN*10.0
    IYMIN=YMIN*10.0 $ XMIN=IXMIN $ YMIN=IYMIN
    XMIN=XMIN/10.0 $ YMIN=YMIN/10.0
    IF(ALENG/0.076-9.0)102,103,103
```



```
103 IF(ALENG/0.25-9.2)105,106,106
106 IF(ALENG/0.50-9.0)107,108,108
108 IF(ALENG/1.0-9.0)109,501,501
102 SCALE=0.076 $ ASC=0.10 $ GO TO 355
105 SCALE=0.25 $ ASC=0.2 $ GO TO 355
107 SCALE=0.50 $ ASC=0.4 $ GO TO 355
109 SCALE =1.0 $ ASC=0.6 $ GO TO 355
355 IF(FA.EQ.-1.0.AND.SCALE.EQ.0.076)356,450
450 IF(FA.EQ.-2.0.AND.SCALE.EQ.0.076)451,452
452 IF(FA.EQ.-3.0.AND.SCALE.EQ.0.076)453,454
454 IF(FA.EQ.-4.0.AND.SCALE.EQ.0.076)455,456
456 IF(FA.EQ.-1.0.AND.SCALE.EQ.0.25)356,457
457 IF(FA.EQ.-2.0.AND.SCALE.EQ.0.25)356,458
458 IF(FA.EQ.-3.0.AND.SCALE.EQ.0.25)459,460
460 IF(FA.EQ.-4.0.AND.SCALE.EQ.0.25)461,462
462 IF(FA.EQ.-1.0.AND.SCALE.EQ.0.50)356,463
463 IF(FA.EQ.-2.0.AND.SCALE.EQ.0.50)356,464
464 IF(FA.EQ.-3.0.AND.SCALE.EQ.0.50)356,465
465 IF(FA.EQ.-4.0.AND.SCALE.EQ.0.50)466,356
451 SCALE=0.25 $ GO TO 356
453 SCALE=0.50 $ GO TO 356
455 SCALE=1.0 $ GO TO 356
459 SCALE=0.50 $ GO TO 356
461 SCALE=1.0 $ GO TO 356
466 SCALE=1.0 $ GO TO 356
356 IF(SCALE.LT.0.15)4600,4601
4600 JX=XMIN*10.0 $ JY=YMIN*10.0 $ KX=XMIN
```

```
      KY=YMIN $ KX=KX*10 $ KY=KY*10 $ QX=JX-KX $ QY=JY-KY
      IF(QX.LT.5.0)4602,4607
 4602 XPLOT=KX/10 $XPLOT=XPLOT+0.5 $ GO TO 4603
 4607 XPLOT=KX/10 $ GO TO 4603
 4603 IF(QY.LT.5.0)4604,4605
 4604 YPLOT=KY/10 $ YPLOT=YPLOT+0.5 $ GO TO 4606
 4605 YPLOT=KY/10 $ GO TO 4606
 4601 JX=XMIN $ JY=YMIN $ XPLOT=JX+1 $ YPLOT=JY+1
 4606 PRINT 4607,XPLOT,YPLOT,XMIN,YMIN,SCALE
      XP=XPLOT $ YP=YPLOT
      XMIN=XMIN/1000.0 $ YMIN=YMIN/1000.0 $ XMIN=XMIN-ASC
      IX=1000.0*XMIN $ IY=1000.0*YMIN $ XMIN=IX $ YMIN=IY
      YMIN=YMIN-ASC $ XTEMP=XMIN $ YTEMP=YMIN
      DO 119 J=1,11 $ U(J)=XTEMP $ V(J)=YTEMP
      XTEMP=XTEMP+SCALE $ YTEMP=YTEMP+SCALE
  119 CONTINUE
C     DRAWING THE SCALES FOR THE MAP
      SCS=-0.2 $ DO 120 J=1,11 $ D(J)=SCS $ SCS=SCS+1.0
  120 CONTINUE
      SCY=0.0 $ DO 121 J=1,11
      E(J)=SCY $ SCY=SCY+1.0
  121 CONTINUE
      DO 80 I=1,L $ XA(I)=(XA(I)-XMIN)/SCALE
      YA(I)=(YA(I)-YMIN)/SCALE
   80 CONTINUE
      XPLOT=(XPLOT-XMIN)/SCALE
      YPLOT=(YPLOT-YMIN)/SCALE $ AA=1.0
```



```
      DO 123 J=1,10 $ S(J)=AA $ AA=AA+1.0
  123 CONTINUE
      CALL PLOT(0.0,0.0,3) $ CALL PLOT(0.0,0.0,2)
      DO 124 J=1,10
      CALL PLOT(S(J),0.0,2) $ CALL PLOT(S(J),0.1,2)
      CALL PLOT(S(J),0.0,2)
  124 CONTINUE
      CALL PLOT(0.0,0.0,3) $ DO 125 J=1,10
      CALL PLOT(0.0,S(J),2) $ CALL PLOT(0.1,S(J),2)
      CALL PLOT(0.0,S(J),2)
  125 CONTINUE
      CALL PLOT(0.0,0.0,3) $ DO 126 J=1,11
  126 CALL NUMBER(D(J),-0.2,0.1,U(J),0.4HF5.2)
      CALL PLOT(0.0,0.0,3) $ CALL PLOT(0.0,1.0,1)
      DO 127 J=1,11
  127 CALL NUMBER(-0.5,E(J),0.1,V(J),0.4HF5.2)
      IF(HA.EQ.-1.0)81,401
  401 IF(HA.EQ.-2.0)92,82
   82 IF(HA.EQ.-3.0)83,501
   83 CALL PLOT(XA(1),YA(1),3) $ DO 84 I=1,L
      CALL PLOT(XA(I),YA(I),2)
   84 CONTINUE
      GO TO 21
   81 DO 88 I=1,L
   88 CALL SYMBOL(XA(I),YA(I),0.1,3,0.,-1)
      GO TO 21
   92 CALL PLOT(XA(1),YA(1),3) $ CALL PLOT(XA(1),YA(1),2)
```

```
      DO 99 I=1,L
   99 CALL SYMBOL(XA(I),YA(I),0.1,3,0.,-2)
      GO TO 21
   21 CALL PLOT(12.0,0.0,-3)
      GO TO 500
  501 PRINT 5500
      PRINT 5501,XMIN,YMIN,ALENG
  503 STOP $ END $ END
```


This program uses the simulation procedure outlined in Fig. 2 and divides the area covered by the simulation plot into squares and tallies the number of squares which contain 1,2,3, etc. positions. The parameter card and the input data cards are the same as for the previous program(SIMPLOT) with the exception or the following changes on the parameter card.

Columns 36-40 This code chooses the random number sequence for the given simulation according to the description presented in the mimeo G5 UOFM RANGEN published Aug. 1965, by the Numerical Analysis Center, Univ. of Minn.

Columns 41-45 This code give the starting place for the grid of squares which is placed over the simulated movement data.

Columns 46-50 This code gives the size of the square in miles. e.g. a 100 in columns 48,49 and 50 would yield square sizes of 528 feet on a side or one-tenth of a mile.