

## COSIM: A FORTRAN IV PROGRAM FOR COCONDITIONAL SIMULATION

JAMES R CARR

Department of Geological Engineering, University of Missouri, Rolla, MO 65401, U S A

and

DONALD E MYERS

Department of Mathematics, University of Arizona, Tucson, AZ 85721, U S A

(Received 9 November 1984, accepted 12 July 1985)

**Abstract**—Recently, a computer algorithm was presented for joint estimation of random functions, this cokriging technique demonstrated the utility and increased accuracy obtained through best linear unbiased estimation based on auto- and cross-correlation. A worthwhile extension of cokriging is coconditional simulation, a technique whereby several nonconditionally simulated random functions are conditioned using cokriging. Although coconditional simulation can be performed, one random function at a time, using kriging, this can result in an incorrect portrayal of the cross-correlation between the simulated random functions. This is particularly true if one, or several variables is sampled sparsely. Coconditional simulation based on cokriging correctly reproduces variable cross-correlation independent of variable sampling density.

*Key Words* Conditional Simulation, Cokriging

### INTRODUCTION

Conditional simulation (Journel, 1974a) is a regionalized variable technique for the simulation of the spatial distribution of a single random function. This technique is conditional in that the simulation, if desired, is forced to assume actual data values at spatial locations where the random function was observed. With the development of generalized cokriging (Carr, Myers, and Glass, 1985), the feasibility of coconditional simulation was assured. This advanced technique allows several random functions to be simulated, more-over conditioned, at once.

Succinctly restated, coconditional simulation is the simulation of spatial coregionalization (Journel and Huijbregts, 1978, p 516). Its purpose is to model spatial situations comprising several variables of interest. One example involves bentonite deposits, here, pH, silt content, and dry compressive strength may be important (Myers and Carr, 1984). Another example involves the development of tripartite earthquake response spectra, each a portrait of peak values of acceleration, velocity, and displacement as a function of natural frequency of vibration during earthquakes (Carr and McCallister, 1985). To model these situations correctly, all variables of importance must be considered.

At present, the simulation of spatial coregionalization involves the preparation of several conditional simulations, one per random function of interest. This process suffers from two major drawbacks:

- (1) If  $N$  random functions are important, it is tedious to execute a conditional simulation process  $N$  times, especially if  $N$  is large.
- (2) Except at the data locations used for condition-

ing, the cross-correlation between two random functions can break down. This is particularly true if any of the random functions is under-sampled.

These drawbacks may be tolerated, however, for the sake of simplicity.

Developing a procedure for coconditional simulation has the benefit of achieving correct cross-correlation between simulated random functions. This procedure, because it relies on cokriging for conditioning, is more expensive than single variable conditional simulation. This expense is offset by the convenience of multiple simulation in one program run and by the increased accuracy obtained in the reproduction of cross-correlation. Moreover, from a quantitative standpoint, the CPU required for coconditional simulation is roughly twice that required for conditional simulation for the same number of random functions, hence the increased cost is not exorbitant.

### A REVIEW OF CONDITIONAL SIMULATION

Each conditional simulation begins by forming a nonconditional simulation. Such a simulation comprises a spatially correlated random function having a desired histogram and variogram. A nonconditional simulation, however, except by chance, lacks certain desirable spatial attributes. Conditioning provides these spatial attributes.

Because the preliminary stage of conditional simulation begins by forming a nonconditional simulation, this latter procedure is emphasized initially. It is acknowledged that nonconditional simulation is the most mathematically rigorous and fundamental aspect of

```

FILE: EXAM      DATA      *
VM/SP CONVERSATIONAL MONITOR SYSTEM

2
50,50,5.00,5.00
250.0,0.0
666666663.DD
1
5.81,1.001,-.2043,.0373,-.0388,-.007,.00172,-.0022,-.0002,.00016,80.0
5.954,2.978,.243,-.0309,-.0425,.00499,-.0104,-.0029,.00053,.00025,80.
1
0.20,1.30,80.0,0.0,1.0
1.60,10.0,80.0,0.0,1.0
1.80,11.30,80.0,0.0,1.0
0
106.51,112.18,11.0,18.1
91.17,132.36,7.000,10.2
102.28,133.21,7.000,15.6
63.37,132.55,6.00,6.1
135.64,1.0,5.00,1.7
57.81,248.49,5.00,2.3
182.89,71.85,5.00,1.0
104.17,155.39,6.00,4.5
94.83,173.98,5.00,2.1
141.20,97.86,6.00,6.2
141.20,164.32,5.00,2.3
108.81,163.43,6.00,11.7
152.31,143.47,6.00,7.6
20.0,220.0,5.00,1.5
71.71,167.24,6.00,2.3
100.0,220.0,5.00,2.0
98.95,44.41,6.00,3.2
90.05,131.44,7.00,9.9
145.00,220.0,5.0,2.0
225.00,220.0,5.0,2.0
145.00,180.0,6.00,3.0
225.00,180.0,5.00,2.0
185.00,140.0,5.0,2.0
225.00,100.0,5.0,2.0
225.00,90.0,2.0,2.0
185.00,20.0,2.0,2.0
185.0,220.0,3.0,2.0
225.00,140.0,3.0,2.0
0.0,0.0,0.0,0.0
1
169.67,58.92
189.82,130.08
76.49,173.44
131.43,92.28
123.81,97.17
141.59,94.50
155.66,71.15
120.68,93.39
120.86,80.05
143.47,100.39
0.0,0.0
    
```

Figure 1 Example data set for two variable coconditional simulation, Hermitian transformation

a conditional simulation model Conditioning simply involves linear estimation and then differencing

*Nonconditional simulation in one dimension*

Let  $T$  represent a collection of random numbers, and let  $f$  be a weighting function Then, a one-dimensional random function (RF),  $Y(u)$ , is defined as (Journal and Huybrechts, 1978, p 505)

$$Y(u) = \int_{-\infty}^{+\infty} f(u+r)T(r)dr = T * f \tag{1}$$

$Y$  has a covariance in this single dimension, expressed as

$$C_1(s) = \sigma^2 \cdot f * f, \tag{2}$$

where  $\sigma^2$  is the variance of  $Y$  Because Equation 1 describes a convolution, a method of moving averages can be employed Hence, Equation 1 becomes (Journal and Huybrechts, 1978, p 505)

$$y_i = \sum_{k=-\infty}^{+\infty} t_{i+k} \cdot f(kb) \tag{3}$$

This infinite sum can be clipped, and it is sufficient to use only  $(2R + 1)$  values,  $t_{i+k}$ , to form  $y_i$  Therefore, Equation 3 becomes

$$y_i = \sum_{k=-R}^{+R} t_{i+k} \cdot f(kb), \tag{4}$$

with,  $f(kb) = k$ , for a spherical covariance, and  $t_{i+k}$  = a random number with mean = 0 Thus,  $y_i$  has a mean of zero and a finite variance

*Extension to three dimensions the method of turning bands*

A nonconditional simulation is needed in three-dimensional space The preceding development, however, was for a one-dimensional, stationary RF An effective, cost-efficient technique is desired for extending the one-dimensional simulation to three space

In three space, the simulation of a RF can be envisioned as the sum of an infinite number of single-dimension simulations along lines passing through the center of a sphere This center is the point at which the simulation will be obtained in three space This



CLASS	GAMMA
1	0.194
2	0.392
3	0.616
4	0.759
5	0.924
6	1.069
7	1.149
8	1.204
9	1.184
10	1.179
11	1.131
12	1.118
13	1.129
14	1.149
15	1.145
16	1.080
17	1.000
18	0.912
19	0.871
20	0.831

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = -0.118E+00  
 VARIANCE = 0.113E+01

Figure 3A Variogram for Figure 2A

$$[\text{ICOS}_1] = \theta \begin{bmatrix} k & 1 & (1+k) \\ 1 & -(1+k) & k \\ (1+k) & k & -1 \end{bmatrix}, \quad (6)$$

moreover, let  $k = 0.618033989$ , one of the roots of the equation  $k^2 + k - 1 = 0$  (Journal and Huybregts, 1978, p 503)

The remaining groups of vectors are obtained from Equation 6 by applying a rotation provided by the following matrix

$$[\mathbf{R}] = \frac{1}{2} \begin{bmatrix} 1 & -(1+k) & k \\ (1+k) & k & -1 \\ k & 1 & (k+1) \end{bmatrix}, \quad (7)$$

where  $k$  is the same as for equation 6. The remaining vectors are

$$\begin{aligned} [\text{ICOS}_2] &= [\text{ICOS}_1][\mathbf{R}] \\ [\text{ICOS}_3] &= [\text{ICOS}_2][\mathbf{R}] \\ [\text{ICOS}_4] &= [\text{ICOS}_3][\mathbf{R}] \\ [\text{ICOS}_5] &= [\text{ICOS}_4][\mathbf{R}], \end{aligned} \quad (8)$$

and the order of multiplication is as shown. Throughout the process, the constant,  $\theta$ , given in Equation 6 is either 1 or 1/2, depending on personal preference. The significance of  $\theta$  will be acknowledged subsequently.

Once the fifteen vectors are defined, the three-dimensional simulation can be developed from fifteen one-dimensional simulations. As a first step, correlated random numbers are generated along each of the fifteen vectors using the following expression

$$y_i(J) = \psi \sum_{m=-NR}^{NR} f(m) T(J+m), \quad (9)$$

an algorithm similar to Equation 4. For the simulation of correlated random numbers having a spherical spatial law, the following holds

$$\begin{aligned} NR &= \frac{\text{RANGE}}{2(\text{Grid Spacing})} \\ \psi &= \sqrt{\frac{36}{NR(NR+1)(2NR+1)}} \\ f(m) &= m, \quad -NR \leq m \leq NR \end{aligned} \quad (10)$$

The constant,  $\psi$ , is introduced to give  $v$  a variance of one. Before, it was acknowledged that Equation 4 yielded a RF having a mean of zero and finite variance, yet the variance was not defined explicitly.

Further with respect to Equation 9,  $J$  correlated random numbers are developed with  $J$  computed as

$$J = [\text{NROW}^2 + \text{NCOL}^2 + \text{NELEV}^2]^{1/2} + 5, \quad (11)$$

where NROW, and NCOL, and NELEV are the number of rows, columns, and levels, respectively, in the model grid, the value, 5, is added to insure against exceeding the number of values contained in the array,  $T$ . Furthermore,  $J$  is the variable, NGMAX, reported in Journal and Huybregts (1978, p 527).

To complete the unconditional simulation, each realization of the simulated random function,  $Z$ , is obtained as

$$Z_0(x) = \frac{1}{\sqrt{15}} \sum_{i=1}^{15} v_i(k_i), \quad (12)$$

where

$$\begin{aligned} k_i &= \text{ICOS}[1, i]u_0 + \text{ICOS}[2, i]v_0 \\ &+ \text{ICOS}[3, i]w_0 + J, \end{aligned} \quad (13)$$

where  $(u_0, v_0, w_0)$  is the coordinate location of point, 0 in three space. In Equation 13, the subscripts on ICOS are discarded and, instead, are described by  $i$ . Furthermore, with respect to Equation 6, if  $\theta = 1$ , each  $k_i$  must be halved. The column, row, and elevation location  $(u_0, v_0, w_0)$  of a grid intersection each has the range

CLASS	GAMMA
1	0.723
2	0.451
3	0.710
4	0.874
5	1.063
6	1.231
7	1.322
8	1.385
9	1.367
10	1.358
11	1.302
12	1.286
13	1.300
14	1.322
15	1.318
16	1.243
17	1.152
18	1.049
19	1.007
20	0.956

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.580E+01  
 VARIANCE = 0.130E+01

Figure 3B Variogram for Figure 2B

OBS	REL	CUML	LOW	
129	0.093	0.093	-0.5	+ *****
162	0.117	0.211	-0.4	+ *****
143	0.104	0.314	-0.3	+ *****
144	0.104	0.419	-0.2	+ *****
120	0.087	0.506	-0.1	+ *****
137	0.099	0.605	-0.0	+ *****
125	0.091	0.696	0.1	+ *****
141	0.102	0.798	0.2	+ *****
150	0.109	0.907	0.3	+ *****
129	0.093	1.000	0.4	+ *****

1380  
LOCATION INDEX RANGE = 4 TO 70

Figure 4A Histogram of Random Numbers used for simulation which yielded Figure 2

OBS	REL	CUML	LOW	
28	0.011	0.011	-3.7	+ **
54	0.022	0.033	-3.0	+ ****
96	0.038	0.071	-2.4	+ *****
237	0.095	0.166	-1.7	+ *****
543	0.217	0.383	-1.1	+ *****
603	0.241	0.625	-0.4	+ *****
538	0.215	0.840	0.3	+ *****
306	0.122	0.962	0.9	+ *****
89	0.036	0.998	1.6	+ *****
5	0.002	1.000	2.3	+ *****

2499

Figure 4B Histogram for Figure 2A

OBS	REL	CUML	LOW	
28	0.011	0.011	2.0	+ **
54	0.022	0.033	2.7	+ ****
96	0.038	0.071	3.4	+ *****
237	0.095	0.166	4.1	+ *****
543	0.217	0.383	4.8	+ *****
603	0.241	0.625	5.5	+ *****
538	0.215	0.840	6.2	+ *****
306	0.122	0.962	6.9	+ *****
89	0.036	0.998	7.6	+ *****
5	0.002	1.000	8.3	+ *****

2499

Figure 4C Histogram for Figure 2B

$$\begin{aligned}
 & -\text{NCOL} \leq u_0 \leq \text{NCOL} \\
 & -\text{NROW} \leq v_0 \leq \text{NROW} \\
 & -\text{NELEV} \leq w_0 \leq \text{NELEV}
 \end{aligned}
 \tag{14}$$

*Transformation of the nonconditional simulation*

Prior to conditioning, the nonconditional simulation must be transformed to a desired distribution. As defined, Equation 12 yields a RF having a mean of zero and a variance of one with a normal distribution. Because most physical situations encompass RF's with other than normal distributions with mean values other than zero and differing variances, a transformation must be applied to the RF defined by equation 12 to yield a physically realistic model.

Before any transformation is applied, the simulated RF must be corrected to yield a perfect, (0, 1) Gaussian distribution. Theoretically, Equation 12 yields this distribution, practically, however, depending on the random numbers used to fill the array,  $T$ , of Equation 9, the mean of the simulated RF may not be exactly zero (practice has shown that  $-0.3 \leq \text{mean} \leq 0.3$ ), and the variance may not be precisely 1.

If the mean of the simulated RF is designated  $M_{er}$  and the standard deviation defined as  $S_{er}$ , then the corrected RF is defined as

$$Z_c = (Z_l - M_{er})/S_{er}, \tag{15}$$

where  $Z_l$  is the RF yielded by Equation 12

Once this correction has been applied, two techniques are employable for transformation. The first is a simple linear transformation. This type of transformation is useful for obtaining the desired mean and variance, yet it maintains a Gaussian distribution. In equation form, the linear transformation is obtained as

$$Z_T = (Z_c)(S_T) + M_T, \tag{16}$$

where  $Z_T$  is the transformed RF,  $Z_c$  is the corrected RF with distribution (0, 1),  $S_T$  is the desired standard deviation, and  $M_T$  is the desired mean.

As an alternative, a transformation can be effected using a Hermite polynomial expansion (Journal and Huijbregts, 1978, p 472). For this transformation, the procedure is

$$Z_T = \sum_{i=0}^N \frac{\Psi}{i!} H_i(u), \tag{17}$$

where  $H_i(u)$  encompasses  $Z_c$ ,  $\Psi$ , are coefficients of the Hermite polynomial expansion, and  $i$ , for the computer

SIMULATION RESULTS

IF NCOL.GT.70, THE FOLLOWING IS EVERY OTHER COLUMN BY EVERY OTHER ROW.

57667655666556665666566644443323221101000010233344
6667766555666676665655654543233220000002223345
6776766555666776665554555433332311000013233333
6866655666776655545554332211223333333333333333
5786655577655455566654443223333321122112332323
5666676556676554445676656443233433322221233
555666665555433456665543322332233232211133
45666665544444333445554332323333333333321121
5655666553344333444455443234333333333334343112
566665544443443334445555444432334332334543212
66655543334454333344554444323334445444234
66655444343335523334445445554433232345454444
545544433323433333333333333333333333333333333
554444454332233333333333333333333333333333333
444443454442211233222245555543334334333333333
33444445442112233312344456665434333433444555
2444444432112223322344556665443342324555555457
24444433211212233233333345665443321214544554567
23454443321113323333334566544332123444555666778
334444332201243333233456543323332433445576778
33444433222234332234566543343333334446778888
443444322312323222345553334432233333356677889
2434443222221122223344343443122224345567789
3446543333212112222233443323232312344455677899
3344554322210011211223443321210322233355567789
33445654452321111131664433213321223224465667788
235656665321112222244433212233234334567789
346667665432221101433433332333333334456677999
456676544422000133333321233322232233345677998
5566555544443201223333222232222222323234587877
65665445654433210123332222223222222323457776777
675555554433343211133333222232222223466677777
665556654444452322443322223222232223345676678
6666565544445432232443322223222222322244656678
56655544543244321235543222112223212221123455556
56654546665433333222133223232121233556556
675445666554323344455532122333211111234555555
666566655544443433554455442123332211111134554555
76676656655555554445556654343332210123234545555
78888665555555445566654454332211213444445555
777775554455444433321133211012232333333333333
776776555544555665445666433331111131233344334
7767666664555546554455444333332112233333333333
666667666555345455444544433222210122232222223
77666765554454554444444432232223222222222112
77867775565555555444443542232222233423211222
78886665445555544433344311112233234221111112
6788776545446666343422334444311023333343210011
6778776544567654433333422221234434311010
6677778765446665644443354444322123334434342000

LEGEND

Table with 3 columns: MAP VALUE, LOW RANGE, HIGH RANGE. Values range from 0 to 9 and -0.27E+01 to 0.34E+01.

Figure 5A Untransformed, nonconditional simulation, variable 2

algorithm COSIM to be introduced subsequently, ranges from 0-9 Moreover, Psi\_0 = M\_T The coefficients, Psi\_i/l can be computed using a variety of available software (e.g., Kim, Myers, and Knudsen, 1977)

In Equation 17, H\_i(u) can be determined as

H\_i(u) = e^{u^2/2} \* d^i / du\_i \* e^{-(u^2/2)} (18)

This can be expressed as the recurrence relationship (Jornel and Huylbregts, 1978, p 476)

H\_{i+1}(u) = uH\_i(u) - iH\_{i-1}(u) (19)

In this series of equations, u is any realization of Z\_c

Conditioning the simulation

Once the nonconditioned simulation is developed by solving either Equation 15 or Equation 17 at each intersection of the model grid, the simulation, if desired, can be conditioned. The objective of conditioning

SIMULATION RESULTS

IF NCOL.GT.70, THE FOLLOWING IS EVERY OTHER COLUMN BY EVERY OTHER ROW.

57667655666556665666566644443323221101000010233344
6667766555666676665655654543233220000002223345
6776766555666776665554555433332311000013233333
6866655666776655545554332211223333333333333333
5786655577655455566654443223333321122112332323
5666676556676554445676656443233433322221233
555666665555433456665543322332233232211133
456666655444443334455543323233333333333321121
5655666553344333444455443234333333333334343112
566665544443443334445555444432334332334543212
66655543334454333344554444323334445444234
66655444343335523334445445554433232345454444
545544433323433333333333333333333333333333333
554444454332233333333333333333333333333333333
444443454442211233222245555543334334333333333
33444445442112233312344456665434333433444555
244444443211222332234455666544334232455555457
24444433211212233233333345665443321214544554567
23454443321113323333334566544332123444555666778
334444332201243333233456543323332433445576778
33444433222234332234566543343333334446778888
443444322312323222345553334432233333356677889
2434443222221122223344343443122224345567789
3446543333212112222233443323232312344455677899
3344554322210011211223443321210322233355567789
33445654452321111131664433213321223224465667788
235656665321112222244433212233234334567789
346667665432221101433433332333333334456677999
456676544422000133333321233322232233345677998
5566555544443201223333222232222222323234587877
65665445654433210123332222223222222323457776777
675555554433343211133333222232222223466677777
665556654444452322443322223222232223345676678
6666565544445432232443322223222222322244656678
56655544543244321235543222112223212221123455556
5665454666543333222133223232121233556556
675445666554323344455532122333211111234555555
666566655544443433554455442123332211111134554555
76676656655555554445556654343332210123234545555
78888665555555445566654454332211213444445555
777775554455444433321133211012232333333333333
776776555544555665445666433331111131233344334
7767666664555546554455444333332112233333333333
666667666555345455444544433222210122232222223
77666765554454554444444432232223222222222112
77867775565555555444443542232222233423211222
78886665445555544433344311112233234221111112
6788776545446666343422334444311023333343210011
6778776544567654433333422221234434311010
6677778765446665644443354444322123334434342000

LEGEND

Table with 3 columns: MAP VALUE, LOW RANGE, HIGH RANGE. Values range from 0 to 9 and 0.12E+01 to 0.47E+01.

Figure 5B Transformed, nonconditional simulation, variable 2

is to require the simulation model to assume observed data values at spatial locations where the random function, Z, has been observed

\*\*\*\*\* CO-SIMULATION PROGRAM \*\*\*\*\*
RESULTS FOR VARIABLE 2

CO-SIMULATION GRID DIMENSIONS

NUMBER OF ROWS = 50
NUMBER OF COLUMNS = 50
INCREMENT IN X = 5.000
INCREMENT IN Y = 5.000
MAXIMUM Y COORDINATE = 250.000
MINIMUM X COORDINATE = 0.000

MODEL PARAMETERS

BEGINNING RANDOM SEED = 0.666666663000+09
FINAL RANDOM NUMBER SEED = 0.191055446200+10
MEAN OF RANDOM NUMBERS = 0.001
VARIANCE OF SPATIAL STRUCTURE = 80.000
VARIANCE OF SPATIAL NOISE = 0.000

INITIAL CONSTANTS

NGMAX = 75
NR = 8
KD = 17
C = 0.171

Figure 5C Initial information for variable 2

CLASS	GAMMA
1	0.195
2	0.404
3	0.589
4	0.743
5	0.823
6	0.880
7	0.931
8	0.976
9	0.990
10	1.018
11	1.086
12	1.285
13	1.556
14	1.817
15	2.010
16	2.095
17	2.110
18	2.003
19	1.861
20	1.744

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = -0.339E-01  
 VARIANCE = 0.106E+01

Figure 6A Variogram for Figure 5A

Each conditioning data value is associated with a discrete spatial location within the model, collectively, these data represent observations of  $Z$  and are denoted  $Z_{ob}(x)$ . At each conditioning location within the model, a value has previously been simulated. These represent simulated observations of  $Z$  and are denoted  $Z_{nc}(x)$ .

At each grid intersection,  $g$ , of the model, kriging is used to yield

$$Z_{ob}^*(g) = \sum_{i=1}^m \lambda_i Z_{ob}(x_i),$$

$$Z_{nc}^*(g) = \sum_{i=1}^m \lambda_i Z_{nc}(x_i), \quad (20)$$

where  $m$  is the total number of conditioning data. Conditioning is achieved as

$$Z_{cs}(g) = Z_{nc}(g) - Z_{nc}^*(g) + Z_{ob}^*(g), \quad (21)$$

and this yields the final, conditionally simulated RF,  $Z_{cs}$ .

**COSIMULATION VS SIMULATION**

By analogy with simulation, for example conditioned simulation of a single random function we may write

$$\bar{Z}(x) = \bar{Z}^*(x) + [\bar{Z}(x) - \bar{Z}^*(x)], \quad (22)$$

where  $\bar{Z}^*(x)$  is the cokriging estimation (block or punctual as appropriate) given by

$$\bar{Z}^*(x) = \sum_{j=1}^n \bar{Z}(x_j) \Gamma_j, \quad (23)$$

and the  $\Gamma_1, \dots, \Gamma_n$  are given by

$$\sum_{j=1}^n \bar{\gamma}(x_i - x_j) \Gamma_j + \bar{\mu} = \bar{\gamma}(x_i - x), \quad (24)$$

$$\sum_{j=1}^n \Gamma_j = I$$

The derivation of these equations is given by Myers (1982) In the same manner as in the one variable version

$$\bar{Z}^*(x), \bar{Z}(x) - \bar{Z}^*(x),$$

are orthogonal, that is,

$$E[\bar{Z}^*(x)]^T [\bar{Z}(x) - \bar{Z}^*(x)],$$

is a matrix with all zeros. To generate simulated values for  $\bar{Z}(x)$ , it would be sufficient to generate  $\bar{Z}_s(x) - \bar{Z}^*(x)$  isomorphic to  $\bar{Z}(x) - \bar{Z}^*(x)$ , that is, with the same first two moments. The isomorphic simulation then is added to  $\bar{Z}^*(x)$  to obtain the coconditioned cosimulation. There are at least three possible approaches

- (1) Simulate each component of  $\bar{Z}(x) - \bar{Z}^*(x)$ , independently, this corresponds to utilizing only the diagonal of the variogram matrix for  $\bar{Z}(x)$ .
- (2) If  $\bar{Z}(x) = \bar{Y}(x)A$  where the components of  $\bar{Y}$  are uncorrelated and hence  $\bar{Y}$  has a diagonal variogram matrix  $D$  then  $\bar{\gamma}_z = A^T D A$ , by simulating separately the components of  $\bar{Y}$ , the cross-correlation of the components of  $\bar{Z}$  are captured by the matrix  $A$ . This also simplifies cokriging but it severely restricts  $\bar{\gamma}$  to make such an assumption. This method is described in greater detail in Journel and Huujbregts (1978).
- (3) In the one variable version of the Turning Bands method, the covariance function, for example variogram, for the random function in 2- or 3-space is given as an integral of the corresponding covariance in 1-space. This in turn is represented as the self convolution of a function defined in 1 space which then is used to apply the moving average technique along a line. For true cosimulation using Turning Bands we would need the following representations

$$\bar{\sigma}_n(h) = \int \bar{\sigma}_1(\langle h, s \rangle) \omega_n(ds), \quad (25)$$

CLASS	GAMMA
1	1.291
2	2.675
3	3.900
4	4.917
5	5.448
6	5.826
7	6.164
8	6.460
9	6.550
10	6.734
11	7.185
12	8.502
13	10.290
14	12.024
15	13.300
16	13.863
17	13.965
18	13.253
19	12.315
20	11.540

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.800E+01  
 VARIANCE = 0.700E+01

Figure 6B Variogram for Figure 5B

OBS	REL	CUML	LOW
140	0.101	0.101	-0.5
134	0.097	0.199	-0.4
149	0.108	0.307	-0.3
120	0.087	0.393	-0.2
128	0.093	0.486	-0.1
152	0.110	0.596	-0.0
137	0.099	0.696	0.1
149	0.108	0.804	0.2
136	0.099	0.902	0.3
135	0.098	1.000	0.4

1380

LOCATION INDEX RANGE = 4 TO 70

Figure 7A Histogram of random numbers used for simulation which yielded Figure 5

OBS	REL	CUML	LOW
42	0.017	0.017	-2.7
160	0.064	0.081	-2.1
380	0.152	0.233	-1.4
573	0.229	0.462	-0.8
470	0.188	0.650	-0.2
457	0.183	0.833	0.4
269	0.108	0.941	1.0
105	0.042	0.983	1.6
33	0.013	0.996	2.2
10	0.004	1.000	2.8

2499

Figure 7B Histogram for Figure 5A

OBS	REL	CUML	LOW
42	0.017	0.017	1.2
160	0.064	0.081	2.8
380	0.152	0.233	4.4
573	0.229	0.462	5.9
470	0.188	0.650	7.5
457	0.183	0.833	9.1
269	0.108	0.941	10.6
105	0.042	0.983	12.2
33	0.013	0.996	13.8
10	0.004	1.000	15.3

2499

Figure 7C Histogram for Figure 5B

\*\*\*\* CONDITIONING RESULTS \*\*\*\*

VARIogram PARAMETERS						
VARIABLE	NUGGET	SILL	RANGE	ANIS	RATIO	
1	0.200	1.300	80.000	0.000	1.000	
2	1.600	10.000	80.000	0.000	1.000	

  

CROSS-CORRELATION PARAMETERS						
PAIR	NUGGET	SILL	RANGE	ANIS	RATIO	
1	1.800	11.300	80.000	0.000	1.000	

  

CONDITIONING DATA						
EAST	NORTH	VAR 1	VAR 2	VAR 3	VAR 4	VAR 5
112.180	106.510	11.000	18.100			
132.360	91.170	7.000	10.200			
133.210	102.280	7.000	15.600			
132.550	63.370	6.000	6.100			
1.000	135.640	5.000	1.700			
248.490	57.810	5.000	2.300			
71.850	182.890	5.000	1.000			
155.390	104.170	6.000	4.500			
173.980	94.830	5.000	2.100			
97.860	141.200	6.000	6.200			
164.320	141.200	5.000	2.300			
163.430	108.810	6.000	11.700			
143.470	152.310	6.000	7.600			
220.000	20.000	5.000	1.500			
167.240	71.710	6.000	2.300			
220.000	100.000	5.000	2.000			
44.410	98.950	6.000	3.200			
131.440	90.050	7.000	9.900			
220.000	145.000	5.000	2.000			
220.000	225.000	5.000	2.000			
180.000	145.000	6.000	3.000			
180.000	225.000	5.000	2.000			
140.000	185.000	5.000	2.000			
100.000	225.000	5.000	2.000			
60.000	225.000	5.000	2.000			
20.000	185.000	5.000	2.000			
220.000	185.000	5.000	2.000			
140.000	225.000	5.000	2.000			

Figure 8 Output from PRT3 documenting conditioning information



CLASS	GAMMA
1	0.199
2	0.372
3	0.566
4	0.685
5	0.838
6	0.987
7	1.081
8	1.163
9	1.199
10	1.248
11	1.226
12	1.218
13	1.171
14	1.102
15	0.989
16	0.821
17	0.645
18	0.490
19	0.461
20	0.507

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.531E+01  
 VARIANCE = 0.115E+01

Figure 10A Variogram for Figure 9A

$$\bar{Z}_0(x) = \frac{1}{\sqrt{15}} \sum_{i=1}^{15} \bar{y}_i(k_i) \quad (29)$$

Here,

$$\begin{aligned} \bar{Z}_0 &= (\bar{Z}_1, \bar{Z}_2, \dots, \bar{Z}_m) \\ \bar{Y}_i &= (\bar{Y}_{i1}, \bar{Y}_{i2}, \dots, \bar{Y}_{im}) \end{aligned} \quad (30)$$

where  $m$  is the number of random functions to be simulated. Subsequent to solving Equation 29, each random function is transformed using either a linear or Hermitian Transformation.

Although cokriging is more expensive than ordinary kriging (Carr, Myers, and Glass, 1985), it relies on auto- and cross-correlation for linear estimation of regionalized variables. It is emphasized that the only difference between conditional simulation and coconditional simulation is the method used for conditioning.

Using cokriging, Equation 21 becomes

$$\bar{Z}_{cs}(g) = \bar{Z}_{nc}(g) - \bar{Z}_{nc}^*(g) + \bar{Z}_{ob}^*(g) \quad (31)$$

where

$$\begin{aligned} \bar{Z}_{cs}(g) &= (Z_{cs,1}(g), Z_{cs,2}(g), \dots, Z_{cs,m}(g)) \\ \bar{Z}_{nc}(g) &= (Z_{nc,1}(g), Z_{nc,2}(g), \dots, Z_{nc,m}(g)) \\ \bar{Z}_{nc}^*(g) &= (Z_{nc,1}^*(g), Z_{nc,2}^*(g), \dots, Z_{nc,m}^*(g)) \\ \bar{Z}_{ob}^*(g) &= (Z_{ob,1}^*(g), Z_{ob,2}^*(g), \dots, Z_{ob,m}^*(g)) \end{aligned} \quad (32)$$

The vectors,  $\bar{Z}_{nc}^*$  and  $\bar{Z}_{ob}^*$ , are developed using cokriging.

**PROGRAM DESCRIPTION**

COSIM is a FORTRAN IV coconditional simulation program which can simulate and condition from one to five random functions. This program was developed on an IBM 4331 computational system. It provides a general technique for single or multiple variable conditional simulation. If only one random

function is to be simulated, COSIM is a standard conditional simulation program. Some elements of COSIM are modeled after the nonconditional simulation program, SIMUL (Journal and Huijbregts, 1978, p 537-545). COSIM is a general, powerful, single program for general geostatistical applications.

COSIM is essentially the merging of a nonconditional simulation program, looped over the desired number of random functions, with the program, COKRIG (Carr, Myers, and Glass, 1985). Cokriging is employed for the conditioning of the coconditionally simulated random functions.

As revealed in the program listing of Appendix 1, COSIM comprises a main program supported by 23 subroutines and function subprograms. The 23 subroutines and functions are listed alphabetically following the main program. In brief, each of these program sections is described here:

- A Main Program (COSIM) controls
  - 1 ICOSOHEDRON Vector definition
  - 2 Data acquisition
  - 3 Nonconditional simulation of all random functions
  - 4 Statistical aspects of nonconditional simulations
  - 5 Transformation
  - 6 Conditioning
  - 7 Statistical aspects of the conditioned model
- B Subroutine AFORM
  - Forms the intersample covariance/cross-covariance matrix, a function of conditioning data. The matrix has the dimension  $A(N,N)$ , where  $N = (\text{No. of conditioning data} + 1) \times \text{no. of RF}$ ,  $N$  has a maximum value of 100 under current program dimensioning.
- C Subroutine COKRIG
  - Performs actual conditioning through cokriging and solves Equation 24.
- D Functions COVAR and CROSS

CLASS	GAMMA
1	1.452
2	3.264
3	5.068
4	6.677
5	7.807
6	8.797
7	9.675
8	10.237
9	10.179
10	9.793
11	9.886
12	7.890
13	6.971
14	6.507
15	6.785
16	7.622
17	8.581
18	8.989
19	9.309
20	9.268

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.341E+01  
 VARIANCE = 0.875E+01

Figure 10B Variogram for Figure 9B

OBS	REL	CUML	LOW	
15	0.006	0.006	1.9	+
132	0.053	0.059	2.8	+ *****
282	0.113	0.172	3.6	+ *****
758	0.303	0.475	4.4	+ *****
801	0.321	0.796	5.2	+ *****
358	0.143	0.939	6.1	+ *****
90	0.036	0.975	6.9	+ *****
36	0.014	0.989	7.7	+ **
17	0.007	0.996	8.5	+ *
10	0.004	1.000	9.4	+ *

2499

Figure 11A Histogram for Figure 9A

OBS	REL	CUML	LOW	
60	0.024	0.024	-3.5	+ ****
265	0.106	0.130	-1.5	+ *****
696	0.279	0.409	0.5	+ *****
695	0.278	0.687	2.5	+ *****
441	0.176	0.863	4.4	+ *****
194	0.079	0.941	6.4	+ *****
84	0.034	0.974	8.4	+ *****
30	0.012	0.986	10.4	+ **
19	0.008	0.994	12.4	+ *
15	0.006	1.000	14.3	+ *

2499

Figure 11B Histogram for Figure 9B

Compute the covariance and cross-covariance values respectively as a function of separation distance

CLASS	GAMMA
1	4.838
2	6.012
3	6.876
4	7.427
5	8.270
6	8.771
7	9.448
8	9.930
9	10.177
10	10.591
11	10.843
12	10.779
13	10.646
14	10.478
15	10.201
16	9.791
17	8.801
18	7.541
19	6.872
20	6.814

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.138E+02  
 VARIANCE = 0.936E+01

Figure 12A Cross-variogram developed using nonconditional cosimulation

CLASS	GAMMA
1	1.698
2	3.814
3	6.033
4	7.863
5	9.349
6	10.870
7	12.283
8	13.376
9	13.679
10	13.448
11	12.332
12	10.780
13	8.846
14	7.325
15	6.678
16	6.820
17	7.338
18	7.508
19	8.003
20	8.131

MEAN AND VARIANCE OF THE SIMULATION  
 MEAN = 0.872E+01  
 VARIANCE = 0.118E+02

Figure 12B Cross-variogram developed using cokriging for conditioning

- E Function DRAND (Schrage, 1979)  
This is a double precision random-number generator, useable on a 16 bit or better CPU
- F Subroutine EQSOLV  
Inverts the matrix, A, prepared by AFORM  
Because the matrix, A, comprises all conditioning data, there is need to invert this matrix only once EQSOLV completes this inversion relative to Kuo (1965, p 168-169), modified to accommodate submatrix structure
- G Subroutine HGRAM  
Plots histograms of random numbers and simulation results
- H Subroutine INIT  
Accesses all program data
- I Subroutine MATML1  
Matrix multiplication for the definition of the icosohedron vectors
- J Subroutine MATMUL  
Matrix multiplication for various operations in the conditioning using cokriging
- K Subroutine NUGGET  
Adds a nugget effect to the model as part of a linear transformation  
Method used  
 $Z_T = Z_T + 5 (DRAND(x) - 0.5)$  (33)  
where  $Z_T$  is that computed by Equation 16
- L Subroutine PROPT  
Prints program options
- M Subroutine PRT1  
Prints program constants for each simulated random function
- N Subroutine PRT2  
Prints a printer map of each simulated RF in both the nonconditional and coconditional stages This subroutine also initializes the histogram plot for each simulation



SIMULATION RESULTS

IF NCOL.GT.70, THE FOLLOWING  
IS EVERY OTHER COLUMN BY EVERY  
OTHER ROW.

```

45434655455576756555767777445566546666545757656545
57476445554576565457475755655774475556665788456457
567666767568676765686666756656556547668765665465
8755766866754646576756554575665767777767777546436
6898768996576488685765766555565766688876666764546
78899867677555756577345555636555655685675566857574
67887987887777666575564545567688877875585756765646
658998876986665476466765566767676767676767676767676
8689777887868876554665646757458674567655566877664
86777786877678754663445637566547576666554586769636
778997977588876756455443564474764445767788886465
7765976774587654766545555555555455646646565476776444
465686653665485346644644345323543363667645765647665
77578685435467646544533544443443325444444447543666
664656563445465357454452422443443556646443332544444
664645565436543675444422642424244663524336464535
6534545632544563635433114343334323332344335533443
44465746533467654634433331222344324453434225442554
4333554545756554441344235443225354322353333333535
56523454333665453131342454454235433442444453234
445643675544544343423334447643554433422353254433
5443446654454446445444444444444444444444444444444
5653546666556553344256333653454666636433414544332
56445565534456334334445342644766645555444244553
574454664533465565553454433676665664645423312425
8456544543345466745664544555566877977755255343244
465634433337546476677446667557678688676744444343424
4643443354457787877577758575698677977755255343244
6344324334676755756757678888695866653545332525
45232424756556675876787678777654464334534534
5334433235467776676757665789789888864653343433554
3232333433555756466785588888788698867552343543634
3242324222354564756765776686777877665333323336543
3334331002334665575666577688885776645354334453445
33243413002345443447487767778877866234432236435
333453231323443456786866858786767574543433334433
454336413233223466567687787888677756453445445353
52344332212233443454577566666665666575653434655342
4233424342124143353658485687767766653365555644534
43234542233234352434544546868686646435534536643354
4345433232344345433134766577778787645543643544443
6355535644413434543335576675676767766553556344535
545546653123334353442567644776555575423535444444
544535752545454545454545454545454545454545454545
43348665353535353535353535353535353535353535353535
4446466754444546243756555776965555356465565533
47475665754336554465338575764PR65655644364755534
4457775444545643456665767788558656445456774577644
64775863444653666566457766765776643545655657656
5644666563554665767658889867776653533565576765564
    
```

LEGEND

MAP	VALUE	LOW RANGE	HIGH RANGE
0		-0.29E+01	-0.94E+00
1		-0.94E+00	0.10E+01
2		0.10E+01	0.30E+01
3		0.30E+01	0.50E+01
4		0.50E+01	0.70E+01
5		0.70E+01	0.90E+01
6		0.90E+01	0.11E+02
7		0.11E+02	0.13E+02
8		0.13E+02	0.15E+02
9		0.15E+02	0.17E+02

Figure 14A Linear transformation with nugget, variable 2

*Portability of COSIM*

Three major considerations must be accounted for when attempting the execution of COSIM on computational systems other than the IBM 4331 System First, SUBROUTINE INIT utilizes free formats for data acquisition, for example,

```
READ(5,*) MVAR
```

The FORTRAN compiler, if not amenable to this type of read statement, might dictate the use of a formal format structure Second, the random-number generation function, DRAND, is compatible with 16 bit or better CPUs COSIM prints a histogram of random numbers at the beginning of each nonconditional simulation If this histogram is unacceptable, DRAND may need to be replaced by a system function Finally, a temporary storage file is needed for the nonconditional simulation phase This file is designated as unit 1 and is used at statements COS02890 and COS03110 within the program Some systems may not accept the unit 1 designation

*Input description/dimensioning limitations*

Data input to COSIM is controlled entirely by Subroutine INIT Data entry, at present, is afforded through free format style The following is essentially a reproduction of the data input guide which occurs at the beginning of COSIM (see Appendix 1)

*Input guide* Figure 1 presents an example data file for COSIM Its entries are

- Record 1 READ (5,\*) MVAR  
MVAR = No of RANDOM FUNCTIONS  
To be simulated (From 1 to a maximum of 5)  
If MVAR = 1, COSIM is a standard conditional simulation program
- Record 2 READ (5,\*) NROW, NCOL, XDIM, YDIM  
NROW = No of rows in model grid  
NCOL = No of columns in model grid  
XDIM = X - distance between columns  
YDIM = Y - distance between rows

SIMULATION RESULTS

IF NCOL.GT.70, THE FOLLOWING  
IS EVERY OTHER COLUMN BY EVERY  
OTHER ROW.

```

3444643344433444344434443321110110000000000121222
4454544433444454443334332211110000000001112223
45444433444454443333332211112010000001112111
44444334443444333344333531010121100000001112111
344445434445433222444443211122211111010000111
3334444343343211213443333110111011120110000011
33444443333221212123333233110121111111100010
34444332212221212223333221212121111222321100
44443332223321111223333333212011112233232023
4433332221111221211122323334321111122323333
3333332211112212111223233334321111122323333
322233321100221111123343343322122222113343333
22222332000001111112334333221222121133433333
12333223200000110013323443322212112334333334
1233322111000011012233344333122101102333433335
02233221000000010012233332221000023333333345
112322210000111111123443311011223334443446
2123322110000112221111334432112212122345566768
22122322101101101100122333212321001121134445779
1222322101100001000000122321212210000121333445579
11223311111100000000011232220110100122333445799
11223332111000000000000122110010011011233345579
22223432110000000000022310000011111233444567
11443444331100000001112233210011112333344567
134444433220110000001222111101101112334456999
33444433323210000001222100001111010011234456999
33433333223110000001111001111101011223657566
4443333433221100000111010100100100000133566455
444443433321231000012111100001021000113446554
44333433322332101012212100102102000112344456
44443443222331011122332100111100000012233345
4443332443211221100233200000010100000001233434
34433324443212211233331000010110110002334444
44433444432221223333321000022100000012333333
444334444332232233332000122000000013333333
544544444433333323334443121210100001012323333
466666433333333322334443323311000000122323333
555653433333333322335221111000000122212223
66454443334333334433233421111000001011222113
54444444333333443333323211110000012111221212
444444443332233333322311000010000110110111
5444454343333323332232311010010100011110011
466455333334433433222232100110101212011000000
466744433233343332211133221000000102200000000
456554323234444222111232321000001211211100000
45655432234443221111523221000001110000000000
4446566543234443432522123322100111232121211000
    
```

LEGEND

MAP	VALUE	LOW RANGE	HIGH RANGE
0		0.71E+00	0.27E+01
1		0.27E+01	0.47E+01
2		0.47E+01	0.67E+01
3		0.67E+01	0.87E+01
4		0.87E+01	0.11E+02
5		0.11E+02	0.13E+02
6		0.13E+02	0.15E+02
7		0.15E+02	0.17E+02
8		0.17E+02	0.19E+02
9		0.19E+02	0.21E+02

Figure 14B Hermitian transformation, variable 2

OBS	REL	CUML	LOW	
20	0.008	0.008	1.5	+ *
57	0.023	0.031	2.3	+ ****
132	0.053	0.084	3.1	+ *****
318	0.127	0.211	4.0	+ *****
496	0.198	0.409	4.8	+ *****
664	0.266	0.675	5.6	+ *****
437	0.175	0.850	6.4	+ *****
267	0.107	0.957	7.2	+ *****
93	0.037	0.994	8.0	+ *****
15	0.006	1.000	8.8	+ *

2499

Figure 15A Histogram for Figure 13A

OBS	REL	CUML	LOW	
5	0.002	0.002	-0.0	+
18	0.007	0.009	1.0	+
25	0.010	0.019	2.1	+ *
22	0.009	0.028	3.2	+ *
1052	0.421	0.449	4.3	+ *****
670	0.268	0.717	5.3	+ *****
548	0.219	0.936	6.4	+ *****
144	0.058	0.994	7.5	+ *****
13	0.005	0.999	8.5	+
2	0.001	1.000	9.6	+

2499

Figure 15B Histogram for Figure 13B

OBS	REL	CUML	LOW	
4	0.002	0.002	-2.9	+
18	0.007	0.009	-0.9	+ *
107	0.043	0.052	1.0	+ *****
339	0.136	0.187	3.0	+ *****
517	0.207	0.394	5.0	+ *****
557	0.223	0.617	7.0	+ *****
477	0.191	0.808	9.0	+ *****
314	0.126	0.934	11.0	+ *****
143	0.057	0.991	13.0	+ *****
23	0.009	1.000	14.9	+ **

2499

Figure 15C Histogram for Figure 14A

OBS	REL	CUML	LOW	
438	0.175	0.175	0.7	+ *****
566	0.226	0.402	2.7	+ *****
437	0.175	0.577	4.7	+ *****
604	0.242	0.818	6.7	+ *****
330	0.132	0.950	8.7	+ *****
68	0.027	0.978	10.7	+ *****
35	0.014	0.992	12.6	+ **
10	0.004	0.996	14.6	+
1	0.000	0.996	16.6	+
10	0.004	1.000	18.6	+

2499

Figure 15D Histogram for Figure 14B

- |   |   |
|---|---|
| <p>Record 3 READ (5,*) YMAX, XMIN<br/>These are the y and x coordinates respectively of the upper, left corner of the model grid</p> <p>Record 4 READ (5,*) RSEED<br/>RSEED = Any odd, double precision number from 1 to 2147483647 e.g., 666666663 D0</p> <p>Record(s) 5 MODEL TRANSFORMATION<br/>A READ (5,*) ITRANS<br/>ITRANS = 0 (Linear transformation)<br/>ITRANS = 1 (Hermitian transformation)<br/>B-1 If ITRANS = 0<br/>READ (5,*) LVAR(I),</p> | <p>LMEAN(I), RANG(I),<br/>CNUG(I)<br/>where I = 1 to MVAR (i.e., MVAR records are required here)<br/>LVAR(I) = desired variance of variable, I<br/>LMEAN(I) = desired mean of variable, I<br/>RANG(I) = range of spatial structure for I<br/>CNUG(I) = Nugget value for I<br/>B-2 If ITRANS = 1<br/>READ (5,*) (COEF(J),<br/>J = 1,10), RANG (I)<br/>COEF = Ten coefficients of the</p> |
|---|---|

Table 2 Coefficients of Hermite polynomial expansion Coef =  $\psi_N$

N	RF #1	RF #2
0	5 8100	5 9540
1	1 0010	2 9780
2	0 2043	0 2430
3	0 0373	-0 0309
4	-0 0388	0 0425
5	0 0070	0 0050
6	0 0017	-0 0104
7	-0 0022	-0 0029
8	-0 0002	0 0005
9	0 0002	0 0003
Total = 10		

Hermitian Transformation  
RANG = Desired range for variable I

- Record 6 READ (5,\*) NSTOP  
NSTOP = 0 (nonconditional simulation, used to check nonconditional simulation)  
NSTOP = 1 proceed with conditioning
- Record(s) 7 Required only if NSTOP = 1  
VARIOGRAM PARAMETERS  
MVAR records  
READ (5,\*) CO(I), C(I),  
RANGE(I), ANIS(I), RATIO(I)  
CO(I) = Nugget of Variable I  
C(I) = Sill of variable I  
RANGE(I) = Range of variogram, Variable I  
ANIS(I) = Spatial Anisotropy Angle  
RATIO(I) = Range—long/Range—Short  
ISOTROPY ANIS = 0, RATIO = 1
- Record(s) 8 Required only if MVAR > 1 and NSTOP = 1  
Cross—Variogram parameters, requiring MVAR (MVAR - 1)/2 Records  
READ(5,\*) CCO(I), CC(I),  
CRANGE(I), CANIS(I),  
CRATIO(I)  
where these are the nugget, sill, range, anisotropy and ratio of the cross variogram for variable pair, I
- Record 9 Required only if NSTOP = 1  
READ(5,\*) IKRIG  
IKRIG = 0 (all variables fully sampled)  
IKRIG = 1 (some variables undersampled)  
If IKRIG = 1, a zero data value indicates undersampling

- Record(s) 10 Required only if NSTOP = 1  
READ (5,\*) Y, X, (DAT(J), J = 1, MVAR)  
Y = Y coordinate of conditioning point  
X = X coordinate of conditioning point  
DAT = 1 to MVAR Variables  
Program continues reading conditioning DATA  
(1 record per conditioning point) until X = Y = 0
- Record 11 Required only if NSTOP = 1  
READ (5,\*) KTEST  
KTEST = 0 (no testing)  
KTEST = 1 (Testing)
- Record(s) 12 Required only if NSTOP = KTEST = 1  
READ (5,\*) X, Y  
These are coordinates at which the simulation is discretely sampled A maximum of 20 locations, 1 record per location  
X = Y = 0 is the last record

EXAMPLES

These examples are intended to demonstrate the utility of COSIM for the simulation of spatial coregonalization Also documented, in a rough manner, is the CPU efficiency of COSIM Lastly, the Hermitian transformation is compared to that using a linear algorithm

Simulation of spatial coregonalization

For this demonstration, two random functions were nonconditionally simulated, then conditioned using cokriging A linear transformation was used for each RF, moreover, a zero nugget effect was assumed Of further note, each RF was sampled fully

Figure 2 presents the results for the nonconditional simulation of the first RF Figures 3 and 4 supplement this first figure by showing the variogram and histograms for each stage of the simulation Figures 5-7 present the same information for the second RF

Of utmost importance to this discussion, Figures 8-11 present the results for coconditioning of these two RFs Figure 8 is the output yielded by subroutine PRT3 and tabulates the variograms, cross-variograms, and conditioning RFs, whereas Figures 10 and 11 supplement Figure 9 by showing variograms and histograms of the final conditioned simulations

Figure 12 documents the cross-correlation achieved through conditioning For the two simulated RFs presented, a zero nugget was assumed This figure shows the cross-variogram results obtained by adding the two nonconditioned RFs presented in Figures 2 and 5, then compares this to the cross-variogram obtained through coconditioning The actual cross-variogram model had

a sill of 11 3 and that obtained through coconditioning is similar The cross-variogram obtained without conditioning has a lower variance in comparison.

*CPU execution time for COSIM*

In brief, Table 1 summarizes the execution time for COSIM in various applications This table is intended to give only a rough idea of the efficiency of COSIM These CPU times are relative to an IBM 4331 computational system operating in batch mode (not CMS foreground mode)

*An alternate transformation*

Program COSIM contains two alternatives for the transformation of the initial nonconditional simulation The examples given were developed using a linear transformation As an alternative, an Hermitian transformation can be applied This type of transformation is useful if a distribution other than Gaussian is desired

To demonstrate the Hermitian transformation, Figures 13 and 14 compare the linear transformation results to those obtained through Hermitian transformation These are not conditioned Furthermore, Figure 15 compares the statistical attributes of each type of transformation Table 2 lists the coefficients of the Hermite polynomial expansion for each RF

REFERENCES

Carr, J R , Myers, D E , and Glass, C E , 1985, Co-kriging—a computer program Computers & Geosciences, v 11, no 2, p 111-127  
 Carr, J R , and McCallister, P , 1985, An application of co-kriging for the estimation of Tripartite Earthquake Response Spectra Jour Geology, v 17, no 5, p 527-545  
 Journel, A , 1974a, Geostatistics for conditional simulation of ore bodies Economic Geology, v 69, no 5, p 673-687  
 Journel, A , 1974b, Simulations conditionnelles—theorie et pratique unpubl doctoral dissertation, Ecole des Mines, Fontainebleau  
 Journel, A , and Huybregts, Ch J , 1978, Mining geostatistics Academic Press, London, 600 p  
 Kim, Y C , Myers, D E , and Knudsen, H P , 1977, Advanced geostatistics in ore reserve estimation and mine planning—practitioner's guide Prepared at Univ Arizona, Tucson, for U S ERDA, Grand Junction, Colorado, GJBX-65 (77), 234 p  
 Kuo, S S , 1965, Numerical methods and computers Addison-Wesley, Reading, Massachusetts, 341 p  
 Myers, D E , 1982, Matrix formulation of co-kriging Jour Math Geology, v 14, no 3, p 249-257  
 Myers, D E , and Carr, J R , 1984, Co-kriging and principal component analysis, bentonite data revisited Cycle de Formation Specialisee en Geostatistique, Reunion des 14-15 Juin, Sur La Geostatistique, p 65-77  
 Schrage, L , 1979, A more portable FORTRAN random number generator ACM Trans Mathematical Software, v 5, no 3, p 132-138

APPENDIX 1

*Program listing*

```

C.....PROGRAM COSIM
C          PROGRAM COSIM :      CC-CONDITIONAL SIMULATION OF SPATIAL
C                                COREGIONALIZATION.
C          PROGRAM AUTHOR:      DR. JAMES RUSSELL CARR
C                                DEPARTMENT OF GEOLOGGICAL ENGINEERING
C                                UNIVERSITY OF MISSOURI
C                                ROLLA, MISSOURI, U.S.A. 65401
C                                TELEPHONE: (314) 341-4867
C          PROGRAM LANGUAGE:     FORTRAN IV (IBM 4331)
C                                STORAGE : 130 K
C                                VERSION : JUNE, 1984
C          GENERAL INFORMATION:
C                                PROGRAM COSIM PROVIDES THE FOLLOWING
C                                CAPABILITIES:
C          1. NON-CONDITIONAL SIMULATION OF N
C              VARIABLES (N = 1 TO 5).
C          2. A PRINT OF NON-CONDITIONAL RESULTS
C          3. VARIOGRAM OF NON-CONDITIONAL RESULTS
C          4. HISTOGRAMS OF RANDOM NUMBERS AND OF
C              NON-CONDITIONAL SIMULATIONS
C          5. TRANSFORMATION OF NON-
C              CONDITIONAL SIMULATIONS
C          6. A PRINT OF TRANSFORMATION RESULTS
C              ALONG WITH HISTOGRAMS AND VARIOGRAMS
C          7. CONDITIONING THROUGH CO-KRIGING
C          8. HISTOGRAMS AND VARIOGRAMS OF
C              CONDITIONING RESULTS
C          9. A PRINT OF CONDITIONING RESULTS
C          10. COMPUTATION OF CROSS-VARIOGRAMS OF
C              COREGIONALIZATIONS.
C          11. A PRINT OF SIMULATED VALUES AT TEST
C              LOCATIONS ( 20 LOCATIONS MAX. )
C          PROGRAM LIMITATIONS:
C          1. SPHERICAL SPATIAL LAW ONLY
C          2. LINEAR OR HERMITIAN TRANSFORMATION
C          3. EAST-WEST VARIOGRAM ONLY
C          4. 10 BIN HISTOGRAM COMPUTATION
C          5. 65 X 65 SIMULATION, MAXIMUM
C          6. 16 BIT MINIMUM MACHINE CAPABILITY
C              SUPPORTING DOUBLE PRECISION USAGE
C          7. 10-STEP DIVISION PRINT CODE
    COS00010
    COS00020
    COS00030
    COS00040
    COS00050
    COS00060
    COS00070
    COS00080
    COS00090
    COS00100
    COS00110
    COS00120
    COS00130
    COS00140
    COS00150
    COS00160
    COS00170
    COS00180
    COS00190
    COS00200
    COS00210
    COS00220
    COS00230
    COS00240
    COS00250
    COS00260
    COS00270
    COS00280
    COS00290
    COS00300
    COS00310
    COS00320
    COS00330
    COS00340
    COS00350
    COS00360
    COS00370
    COS00380
    COS00390
    COS00400
    COS00410
    COS00420
    COS00430
    COS00440
    COS00450
    COS00460
    COS00470
    COS00480
    COS00490
    COS00500
    
```





```

C.....COMPUTE NMORE RANDOM VARIABLES PER VECTOR.  RANDOM NUMBER      COS02550
C.....GENERATION IS PROVIDED BY FUNCTION DRAND.  THIS FUNCTION        COS02560
C.....RETURNS RANDOM NUMBERS IN THE RANGE 0 TO 1.  IN CONDITIONAL    COS02570
C.....SIMULATION, AN INITIAL DISTRIBUTION IS NEEDED HAVING A MEAN    COS02580
C.....OF ZERO.  DRAND RETURNS VALUES HAVING A MEAN OF 0.5;  HENCE,  COS02590
C.....TO OBTAIN A MEAN OF ZERO, 0.5 IS SIMPLY SUBTRACTED FROM THE    COS02600
C.....VALUE,  DRAND.  COS02610
C..... DO 10 I = 1,NMORE COS02620
RDOM(I) = DRAND(RSEED) - 0.50 COS02630
IF(RDOM(I).GT.RMAX) RMAX = RDOM(I) COS02640
IF(RDOM(I).LT.RMIN) RMIN = RDOM(I) COS02650
RMEAN = RMEAN + RDOM(I) COS02660
10 CONTINUE COS02670
XMN = -0.50 COS02680
XMAX = 0.50 COS02690
DO 30 I = 1,NMORE COS02700
DO 30 J = 1,10 COS02710
XOLD = FLOAT(J - 1)*0.10 + XMN COS02720
XNEW = FLOAT(J) *0.10 + XMN COS02730
IF(RDOM(I).GE.XOLD.AND.RDOM(I).LT.XNEW) BIN(J) = BIN(J) + 1 COS02740
30 CONTINUE COS02750
C..... COMPUTE NGMAX CORRELATED RANDOM VARIABLES FOR EACH OF THE    COS02760
C..... 15 VECTORS.  COS02770
C..... DO 40 J = 1,NGMAX COS02780
KN = J + KD COS02790
M = 0 COS02800
DO 40 I = J,KN COS02810
M = M + 1 COS02820
DIFF = M - NR - 1 COS02830
BUF(J) = BUF(J) + (DIFF * RDOM(I) * C9) COS02840
40 CONTINUE COS02850
WRITE(1) (BUF(JK),JK = 1,NGMAX) COS02860
CONTINUE COS02870
100 WRITE(6,120) COS02880
120 FORMAT(1H1,T30,'HISTOGRAM OF RANDOM NUMBERS',/) COS02890
CALL HGRAM(RMIN,RMAX) COS02900
C..... USING THE CORRELATED ONE DIMENSIONAL RANDOM VARIABLES      COS02910
C..... STORED ON UNIT 1, FORM THE 3-D SIMULATION.  (JOURNAL        COS02920
C..... AND HUIJBREGTS, 1978, P. 499).  COS02930
C..... XBEGIN = - FLOAT(( NCOL + 1) / 2) COS02940
YBEGIN = FLOAT(( NROW + 1) / 2) COS02950
NVERT = 1 COS02960
DO 150 I = 1,NROW COS02970
DO 150 J = 1,NCOL COS02980
RF(I,J,IVAR) = 0.0 COS02990
150 CONTINUE COS03000
REWIND 1 COS03010
SQ15 = (1.0 / SQRT(15.0)) COS03020
LOCMIN = 73057 COS03030
LOCMAX = -73057 COS03040
DO 200 IRF = 1,15 COS03050
READ(1) (BUF(JK),JK = 1,NGMAX) COS03060
DO 190 M = 1,NVERT COS03070
DO 190 I = 1,NROW COS03080
DO 190 J = 1,NCOL COS03090
XNOW = XBEGIN + FLOAT(J) COS03100
YNOW = YBEGIN - FLOAT(I) COS03110
ZNOW = 1.0 COS03120
DUM1 = ICOS(1,IRF)*XNOW + ICOS(2,IRF)*YNOW + COS03130
1 ICOS(3,IRF)*ZNOW + 0.5 COS03140
LOC = DUM1/2.0 + NGMAX/2 COS03150
IF(LOC.GT.LOCMAX) LOCMAX = LOC COS03160
IF(LOC.LT.LOCMIN) LOCMIN = LOC COS03170
IF(LCC.LE.0) GO TO 190 COS03180
RF(I,J,IVAR) = RF(I,J,IVAR) + BUF(LCC) * SQ15 COS03190
190 CONTINUE COS03200
200 CONTINUE COS03210
WRITE(6,210) LOCMIN,LOCMAX COS03220
210 FORMAT(/,T5,'LOCATION INDEX RANGE = ',I10,' TO ',I10,/) COS03230
RMEAN = RMEAN / FLOAT(15*NMORE) COS03240
CALL PRT1(NGMAX,NR,KD,C9,BSEED,RMEAN,IVAR) COS03250
CALL PRT2(IVAR) COS03260
CALL VARG(XMEAN,VAR,IVAR) COS03270
CALL TRANS(XMEAN,VAR,IVAR) COS03280
CALL PRT2(IVAR) COS03290
CALL VARG(XMEAN,VAR,IVAR) COS03300
C..... RETURN FOR OTHER VARIABLES COS03310
C..... 3000 CONTINUE COS03320
C..... CONDITION THE MODEL IF DESIRED.  COS03330
C..... IF(NSTOP.EQ.0) GO TO 3350 COS03340
CALL COKRIG(KOUNT,DET) COS03350
IF(DET.EQ.0.0) GO TO 5000 COS03360
C..... PRINT MODEL AND STATISTICAL ASPECTS.  COS03370
C..... CALL PRT3(KOUNT) COS03380
DO 3100 I = 1,MVAR COS03390
CALL PRT2(I) COS03400
CALL VARG(XMEAN,VAR,I) COS03410
3100 CONTINUE COS03420
IF(KTEST.GT.0) CALL PTEST(KASTLE) COS03430
C..... CHECK CROSS-CORRELATION RESULTS.  COS03440
C..... COS03450
COS03460
COS03470
COS03480
COS03490
COS03500
COS03510
COS03520
COS03530
COS03540
COS03550
COS03560

```

```

C...
DO 3300 I = 1,MVAR
DO 3300 J = 1,MVAR
IF(J.LE.I) GO TO 3300
DO 3150 K = 1,NROW
DO 3150 L = 1,NCOL
RF(K,L,6) = RF(K,L,I) + RF(K,L,J)
3150 CONTINUE
CALL VARG(XMEAN,VAR,6)
3300 CONTINUE
3350 CONTINUE
C...
C.....AS A FINAL CHECK, PRINT VECTORS OF ICOSOHEDRON.
C...
WRITE(6,3500)
DO 3400 I = 1,3
WRITE(6,4500)(ICOS(I,JK),JK = 1,15)
3400 CONTINUE
3500 FORMAT(1H1,T30,'CHECK ON 15 VECTORS',/)
4500 FORMAT(T3,15F5.2)
CALL PROP
5000 CONTINUE
STOP
END
C
SUBROUTINE AFORM(KOUNT)
COMMON /DAT/ X(100),Y(100),DAT(100,5)
COMMON /PARM/ MVAR
COMMON /VAR/ CO(5),C(5),RANGE(5),ANIS(5),RATIO(5)
COMMON /CVAR/ CCU(10),CC(10),CRANGE(10),CANIS(10),CRATIO(10)
COMMON /FORM/ A(100,100)
COMMON /OPT/ NSTOP,KTEST,IKRIG,ITRANS
C...
C.....THIS SUBROUTINE FORMS THE INTERSAMPLE COVARIANCE/CROSS-
C.....COVARIANCE MATRIX, A, FOR ALL DATA TO WHICH THE NON-
C.....CONDITIONAL SIMULATION IS TO BE CONDITIONED.
C...
DO 1000 II = 1,KOUNT
DO 750 JJ = 1,KOUNT
KPOS = 0
DO 500 KK = 1,MVAR
KTOT = (II - 1) * MVAR + KK
LTOT = (JJ - 1) * MVAR + KK
DO 500 LL = 1,MVAR
NTOT = (JJ - 1) * MVAR + LL
MTOT = (II - 1) * MVAR + LL
DIFX = X(II) - X(JJ)
DIFY = Y(II) - Y(JJ)
IF(LL.NE.KK) GO TO 100
DISTAN = SQRT((DIFX * COS(ANIS(LL)) + DIFY * SIN(ANIS
1 2 (LL)))**2 + (RATIO(LL) * (DIFY * COS(ANIS(LL))
- DIFX * SIN(ANIS(LL))))**2)
I9 = LL
A(KTOT,NTOT) = COVAR(DISTAN,I9)
GO TO 500
CONTINUE
100 IF(LL.LT.KK) GO TO 500
KPOS = KPOS + 1
DISTAN = SQRT((DIFX * COS(CANIS(KPOS)) + DIFY * SIN
1 2 (CANIS(KPOS)))**2 + (CRATIO(KPOS) * (DIFY *
3 COS(CANIS(KPOS)) - DIFX * SIN(CANIS(KPOS))))
**2)
I8 = KK
I9 = LL
A(KTOT,NTOT) = CROSS(DISTAN,KPOS,I8,I9)
A(MTOT,LTOT) = A(KTOT,NTOT)
CONTINUE
500 CONTINUE
750 CONTINUE
C...
C.....LAST MATRIX IN EACH ROW OF MATRIX, A, IS A UNIT MATRIX.
C...
DO 900 MM = 1,MVAR
IA = (II - 1) * MVAR + MM
DO 900 NN = 1,MVAR
KM = KOUNT * MVAR + NN
IF(MM.EQ.NN) A(IA,KM) = 1.0
IF(MM.NE.NN) A(IA,KM) = 0.0
CONTINUE
900 CONTINUE
1000 CONTINUE
C...
C.....FORM LAST ROW OF A: A ROW OF UNIT AND NULL MATRICES.
C...
DO 1200 LM = 1,KOUNT
DO 1200 MM = 1,MVAR
IA = KOUNT * MVAR + MM
DO 1200 NN = 1,MVAR
NTOT = (LM - 1) * MVAR + NN
IF(MM.EQ.NN) A(IA,NTOT) = 1.0
IF(MM.NE.NN) A(IA,NTOT) = 0.0
CONTINUE
1200 CONTINUE
C...
C.....LAST MATRIX IN THIS ROW IS NULL
C...
DO 1400 MM = 1,MVAR
IA = KOUNT * MVAR + MM
DO 1400 NN = 1,MVAR
IB = KOUNT * MVAR + NN
A(IA,IB) = 0.0
CONTINUE
1400 CONTINUE
C...
C.....ACCOUNT FOR UNDERSAMPLED RANDOM FUNCTIONS IF DESIRED.

```

COS03570  
 COS03580  
 COS03590  
 COS03600  
 COS03610  
 COS03620  
 COS03630  
 COS03640  
 COS03650  
 COS03660  
 COS03670  
 COS03680  
 COS03690  
 COS03700  
 COS03710  
 COS03720  
 COS03730  
 COS03740  
 COS03750  
 COS03760  
 COS03770  
 COS03780  
 COS03790  
 COS03800  
 COS03810  
 COS03820  
 COS03830  
 COS03840  
 COS03850  
 COS03860  
 COS03870  
 COS03880  
 COS03890  
 COS03900  
 COS03910  
 COS03920  
 COS03930  
 COS03940  
 COS03950  
 COS03960  
 COS03970  
 COS03980  
 COS03990  
 COS04000  
 COS04010  
 COS04020  
 COS04030  
 COS04040  
 COS04050  
 COS04060  
 COS04070  
 COS04080  
 COS04090  
 COS04100  
 COS04110  
 COS04120  
 COS04130  
 COS04140  
 COS04150  
 COS04160  
 COS04170  
 COS04180  
 COS04190  
 COS04200  
 COS04210  
 COS04220  
 COS04230  
 COS04240  
 COS04250  
 COS04260  
 COS04270  
 COS04280  
 COS04290  
 COS04300  
 COS04310  
 COS04320  
 COS04330  
 COS04340  
 COS04350  
 COS04360  
 COS04370  
 COS04380  
 COS04390  
 COS04400  
 COS04410  
 COS04420  
 COS04430  
 COS04440  
 COS04450  
 COS04460  
 COS04470  
 COS04480  
 COS04490  
 COS04500  
 COS04510  
 COS04520  
 COS04530  
 COS04540  
 COS04550  
 COS04560  
 COS04570  
 COS04580

```

C... IF(IKRIG.EQ.1) CALL UNSAM(KOUNT)
      RETURN
      END
C
      SUBROUTINE COKRIG(KOUNT,DET)
      COMMON /SIM/ RF(65,65,6)
      COMMON /DAT2/ X(100),Y(100),DAT(100,5)
      COMMON /PARM/ MVAR
      COMMON /OPT/ NSTOP,KTEST,IKRIG,ITRANS
      COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
      COMMON /FORM/ A(100,100)
      COMMON /VAR/ CO(5),C(5),RANGE(5),ANIS(5),RATIO(5)
      COMMON /CVAR/ CCO(10),CC(10),CRANGE(10),CANIS(10),CRATIO(10)
      DIMENSION XMEAS(100,5),WEIGHT(100,5),SIMU(100,5)
      DIMENSION TEMP(5,5),TEMP1(5,5),TEMP2(5,5),EST(5,2)
C... THIS SUBROUTINE CONTROLS THE CONDITIONING, USING COKRIGING,
C... OF THE NON-CONDITIONAL SIMULATION TO THE SAMPLE DATA.
C... STEP 1. FIND SIMULATED VALUES (SIMU) CLOSEST
C... TO THE ACTUAL DATA LOCATIONS.
C...
      YBEGIN = YMAX + 0.5 * YDIM
      XBEGIN = XMIN - 0.5 * XDIM
      DO 10 I = 1,KOUNT
      IROW = (YBEGIN - Y(I)) / YDIM
      JCOL = (X(I) - XBEGIN) / XDIM
      IF(IROW.LE.0) IROW = 1
      IF(IROW.GT.NROW) IROW = NROW
      IF(JCOL.LE.0) JCOL = 1
      IF(JCOL.GT.NCOL) JCOL = NCOL
      DO 10 J = 1,MVAR
      SIMU(I,J) = RF(IROW,JCOL,J)
      CONTINUE
10
C... STEP 2. FORM THE INTERSAMPLE COVARIANCE/CROSS-COVARIANCE
C... MATRIX, A.
C...
      CALL AFORM(KOUNT)
C... STEP 3. INVERT THE MATRIX, A.
C...
      NN = KOUNT + 1
      CALL EQSOLV(NN,DET)
      IF(DET.EQ.0.0) WRITE(6,30)
30      FORMAT(/,'A-MATRIX IS SINGULAR',/)
      IF(DET.EQ.0.0) RETURN
C... STEP 4. USING A-INV, CONDITION THE MODEL.
C...
      DO 1000 II = 1,NROW
      YCORD = YBEGIN - FLOAT(II) * YDIM
      DO 1000 JJ = 1,NCOL
      XCORD = XBEGIN + FLOAT(JJ) * XDIM
C... STEP 4A. FORM THE MEASUREMENT VECTOR FOR LOCATION (II,JJ)
C...
      DO 500 KK = 1,KOUNT
      DIFX = X(KK) - XCORD
      DIFY = Y(KK) - YCORD
      IF(IKRIG.EQ.0) GO TO 50
      IF(DIFX.EQ.0.0.AND.DIFY.EQ.0.0) DIFX = 0.20
50      CONTINUE
      KPOS = 0
      DO 200 I = 1,MVAR
      KZ = (KK - 1) * MVAR + I
      DO 200 J = 1,MVAR
      KW = (KK - 1) * MVAR + J
      IF(J.NE.I) GO TO 100
      DISTAN = SQRT((DIFX * COS(ANIS(J)) + DIFY * SIN(ANIS
      (J)))**2 + (RATIO(J)*DIFY*COS(ANIS(J)) -
      DIFX * SIN(ANIS(J)))**2)
1/2      I9 = J
      XMEAS(KZ,J) = COVAR(DISTAN,I9)
      GO TO 200
      CONTINUE
      IF(J.LT.I) GO TO 200
100      KPOS = KPOS + 1
      DISTAN = SQRT((DIFX * COS(CANIS(KPOS)) + DIFY * SIN
      (CANIS(KPOS)))**2 + (RATIO(KPOS) * (DIFY *
      COS(CANIS(KPOS)) - DIFX * SIN(CANIS(KPOS)))
      **2)
1/2      I8 = I
1/3      I9 = J
      XMEAS(KZ,J) = CROSS(DISTAN,KPOS,I8,I9)
      XMEAS(KW,I) = XMEAS(KZ,J)
200      CONTINUE
500      CONTINUE
C... LAST MATRIX IS A UNIT MATRIX
C...
      DO 600 I = 1,MVAR
      KZ = KOUNT * MVAR + I
      DO 600 J = 1,MVAR
      IF(I.EQ.J) XMEAS(KZ,J) = 1.0
      IF(I.NE.J) XMEAS(KZ,J) = 0.0
600      CONTINUE
C... MODIFY THIS VECTOR FOR UNDERSAMPLING IF DESIRED.
C...

```

```

IF(IKRIG.EQ.1) CALL UNVEC(XMEAS,KOUNT)
C...
C.....STEP 4B. FORM THE WEIGHTING VECTOR FOR LOCATION (I,JJ).
C...
DO 650 I = 1,NN
DO 650 J = 1,MVAR
KZ = (I - 1) * MVAR + J
DO 650 K = 1,MVAR
WEIGHT(KZ,K) = 0.0
650 CONTINUE
DO 800 I = 1,NN
DO 800 J = 1,NN
DO 700 K = 1,MVAR
KZ = (I - 1) * MVAR + K
KT = (J - 1) * MVAR + K
DO 700 L = 1,MVAR
KW = (J - 1) * MVAR + L
TEMP(K,L) = A(KZ,KW)
TEMP1(K,L) = XMEAS(KT,L)
700 CONTINUE
CALL MATMUL(TEMP,TEMP1,TEMP2,MVAR,MVAR,MVAR)
DO 800 K = 1,MVAR
KZ = (I - 1) * MVAR + K
DO 800 L = 1,MVAR
WEIGHT(KZ,L) = WEIGHT(KZ,L) + TEMP2(K,L)
800 CONTINUE
C...
C.....STEP 4C. USING THE WEIGHTING VECTOR, COMPUTE KRIGED EST.
C...
DO 810 I = 1,MVAR
EST(I,1) = 0.0
EST(I,2) = 0.0
810 CONTINUE
DO 900 I = 1,KOUNT
DO 900 J = 1,MVAR
DO 900 K = 1,MVAR
KZ = (I - 1) * MVAR + K
IF(IKRIG.EQ.0) GO TO 890
IF(DAT(I,K).EQ.0.0) SIMU(I,K) = 0.0
890 CONTINUE
EST(J,1) = EST(J,1) + DAT(I,K) * WEIGHT(KZ,J)
EST(J,2) = EST(J,2) + SIMU(I,K) * WEIGHT(KZ,J)
900 CONTINUE
C...
C.....STEP 4D. CONDITION THE MODEL: RF = RF - SIM.EST + DAT.EST.
C...
DO 950 I = 1,MVAR
RF(I,JJ,I) = RF(I,JJ,I) - EST(I,2) + EST(I,1)
950 CONTINUE
1000 CONTINUE
RETURN
END
C
C
FUNCTION COVAR(DIST,K)
COMMON /VAR/ CO(5),C(5),RANGE(5),ANIS(5),RATIO(5)
C...
C.....THIS FUNCTION EVALUATES THE MODEL COVARIANCE ASSOCIATED
C.....WITH THE SEPARATION DISTANCE, DIST.
C...
IF(DIST.GE.RANGE(K)) GO TO 120
B = C(K) - CO(K)
DUM1 = CO(K) + B*(1.5*DIST/RANGE(K) - 0.5*(DIST/
1 RANGE(K))**3)
IF(DIST.EQ.0.0) DUM1 = 0.0
COVAR = C(K) - DUM1
RETURN
120 CONTINUE
COVAR = 0.0
RETURN
END
C
C
FUNCTION CROSS(DIST,K,IA,IB)
COMMON /CVAR/ CCO(10),CC(10),CRANGE(10),CANIS(10),CRATIO(10)
C...
C.....THIS FUNCTION EVALUATES THE CROSS-COVARIANCE VALUE
C.....CORRESPONDING TO THE DISTANCE, DIST.
C...
IF(DIST.GE.CRANGE(K)) GO TO 120
B = CC(K) - CCO(K)
DUM1 = CCO(K) + B*(1.5*DIST/CRANGE(K)
1 - 0.5*(DIST/CRANGE(K))**3)
IF(DIST.EQ.0.0) DUM1 = 0.0
DUM2 = CC(K) - DUM1
GO TO 400
CONTINUE
DUM2 = 0.0
400 CONTINUE
DUM3 = COVAR(DIST,IA)
DUM4 = COVAR(DIST,IB)
CROSS = 0.5 * (DUM2 - DUM3 - DUM4)
RETURN
END
C
C
FUNCTION DRAND(IX)
C...
C.....THIS IS A PORTABLE RANDOM NUMBER GENERATION
C.....FUNCTION. REFERENCE:
C.....
C.....SCHRAGE,L., "A MORE PORTABLE FORTRAN RANDOM
C.....

```

COS05610  
COS05620  
COS05630  
COS05640  
COS05650  
COS05660  
COS05670  
COS05680  
COS05690  
COS05700  
COS05710  
COS05720  
COS05730  
COS05740  
COS05750  
COS05760  
COS05770  
COS05780  
COS05790  
COS05800  
COS05810  
COS05820  
COS05830  
COS05840  
COS05850  
COS05860  
COS05870  
COS05880  
COS05890  
COS05900  
COS05910  
COS05920  
COS05930  
COS05940  
COS05950  
COS05960  
COS05970  
COS05980  
COS05990  
COS06000  
COS06010  
COS06020  
COS06030  
COS06040  
COS06050  
COS06060  
COS06070  
COS06080  
COS06090  
COS06100  
COS06110  
COS06120  
COS06130  
COS06140  
COS06150  
COS06160  
COS06170  
COS06180  
COS06190  
COS06200  
COS06210  
COS06220  
COS06230  
COS06240  
COS06250  
COS06260  
COS06270  
COS06280  
COS06290  
COS06300  
COS06310  
COS06320  
COS06330  
COS06340  
COS06350  
COS06360  
COS06370  
COS06380  
COS06390  
COS06400  
COS06410  
COS06420  
COS06430  
COS06440  
COS06450  
COS06460  
COS06470  
COS06480  
COS06490  
COS06500  
COS06510  
COS06520  
COS06530  
COS06540  
COS06550  
COS06560  
COS06570  
COS06580  
COS06590  
COS06600  
COS06610  
COS06620



```

80 CONTINUE
90 CONTINUE
   IPIVOT(ICOL) = IPIVOT(ICOL) + 1
C....THE FOLLOWING STATEMENTS PUT THE PIVOT ELEMENT ON DIAGONAL.
C....
100 IF(IROW - ICOL) 100,150,100
    DET = -DET
    DO 130 L = 1,N
    DO 120 LI = 1,MVAR
    LL = (IROW - 1) * MVAR + LI
    DO 120 LJ = 1,MVAR
    LM = (L - 1) * MVAR + LJ
    TEMP(LI,LJ) = A(LL,LM)
120 CONTINUE
    DO 130 LI = 1,MVAR
    LK = (IROW - 1) * MVAR + LI
    LL = (ICOL - 1) * MVAR + LI
    DO 130 LJ = 1,MVAR
    LM = (L - 1) * MVAR + LJ
    A(LK,LM) = A(LL,LM)
    A(LL,LM) = TEMP(LI,LJ)
130 CONTINUE
150 CONTINUE
    INDEX(I,1) = IROW
    INDEX(I,2) = ICOL
    DO 160 LI = 1,MVAR
    IP = (I - 1) * MVAR + LI
    IA = (ICOL - 1) * MVAR + LI
    DO 160 LJ = 1,MVAR
    IB = (ICOL - 1) * MVAR + LJ
    PIVOT(IP,LJ) = A(IA,IB)
    TEMP(LI,LJ) = PIVOT(IP,LJ)
160 CONTINUE
    W = TRACE(TEMP)
    DET = DET * W
C....THE FOLLOWING STATEMENTS DIVIDE PIVOT ROW BY PIVOT ELEMENT.
C....
    DO 170 LI = 1,MVAR
    IA = (ICOL - 1) * MVAR + LI
    DO 170 LJ = 1,MVAR
    IB = (ICOL - 1) * MVAR + LJ
    IF(LI.EQ.LJ) A(IA,IB) = 1.0
    IF(LI.NE.LJ) A(IA,IB) = 0.0
170 CONTINUE
    DO 200 L = 1,N
    DO 180 LI = 1,MVAR
    IP = (I - 1) * MVAR + LI
    IA = (ICOL - 1) * MVAR + LI
    DO 180 LJ = 1,MVAR
    IB = (L - 1) * MVAR + LJ
    TEMP(LI,LJ) = A(IA,IB)
    TEMP1(LI,LJ) = PIVOT(IP,LJ)
180 CONTINUE
    CALL SCALG(TEMP1,TEMP,TEMP2)
    DO 200 LI = 1,MVAR
    IA = (ICOL - 1) * MVAR + LI
    DO 200 LJ = 1,MVAR
    IB = (L - 1) * MVAR + LJ
    A(IA,IB) = TEMP2(LI,LJ)
200 CONTINUE
C....REDUCE THE NON PIVOT ROWS.
C....
    DO 1350 LI = 1,N
    IF(LI - ICOL) 210,1350,210
210 CONTINUE
    DO 220 LJ = 1,MVAR
    IA = (LI - 1) * MVAR + LJ
    DO 220 LM = 1,MVAR
    IB = (ICOL - 1) * MVAR + LM
    TEMP(LJ,LM) = A(IA,IB)
    A(IA,IB) = 0.0
220 CONTINUE
    DO 240 L = 1,N
    DO 230 LJ = 1,MVAR
    IA = (ICOL - 1) * MVAR + LJ
    DO 230 LM = 1,MVAR
    IB = (L - 1) * MVAR + LM
    TEMP1(LJ,LM) = A(IA,IB)
230 CONTINUE
    CALL MATMUL(TEMP,TEMP1,TEMP2,MVAR,MVAR,MVAR)
    DO 240 LJ = 1,MVAR
    IA = (LI - 1) * MVAR + LJ
    DO 240 LM = 1,MVAR
    IB = (L - 1) * MVAR + LM
    A(IA,IB) = A(IA,IB) - TEMP2(LJ,LM)
240 CONTINUE
1350 CONTINUE
C....THE FOLLOWING STATEMENTS INTERCHANGE COLUMNS.
C....
    DO 3000 I = 1,N
    L = N - I + 1
    IF(INDEX(L,1) - INDEX(L,2)) 2000,3000,2000
2000 JROW = INDEX(L,1)
    JCOL = INDEX(L,2)
    DO 2500 K = 1,N
    DO 2400 KI = 1,MVAR
    IA = (K - 1) * MVAR + KI
    DO 2400 KJ = 1,MVAR
    IB = (JROW - 1) * MVAR + KJ

```

COS07660  
COS07670  
COS07680  
COS07690  
COS07700  
COS07710  
COS07720  
COS07730  
COS07740  
COS07750  
COS07760  
COS07770  
COS07780  
COS07790  
COS07800  
COS07810  
COS07820  
COS07830  
COS07840  
COS07850  
COS07860  
COS07870  
COS07880  
COS07890  
COS07900  
COS07910  
COS07920  
COS07930  
COS07940  
COS07950  
COS07960  
COS07970  
COS07980  
COS07990  
COS08000  
COS08010  
COS08020  
COS08030  
COS08040  
COS08050  
COS08060  
COS08070  
COS08080  
COS08090  
COS08100  
COS08110  
COS08120  
COS08130  
COS08140  
COS08150  
COS08160  
COS08170  
COS08180  
COS08190  
COS08200  
COS08210  
COS08220  
COS08230  
COS08240  
COS08250  
COS08260  
COS08270  
COS08280  
COS08290  
COS08300  
COS08310  
COS08320  
COS08330  
COS08340  
COS08350  
COS08360  
COS08370  
COS08380  
COS08390  
COS08400  
COS08410  
COS08420  
COS08430  
COS08440  
COS08450  
COS08460  
COS08470  
COS08480  
COS08490  
COS08500  
COS08510  
COS08520  
COS08530  
COS08540  
COS08550  
COS08560  
COS08570  
COS08580  
COS08590  
COS08600  
COS08610  
COS08620  
COS08630  
COS08640  
COS08650  
COS08660  
COS08670  
COS08680

```

        IC          = (JLUL - 1) * MVAR + KJ
        TEMP(KI,KJ) = A(IA,IB)
        A(IA,IB)    = A(IA,IC)
        A(IA,IC)    = TEMP(KI,KJ)
2400    CONTINUE
2500    CONTINUE
3000    CONTINUE
8100    RETURN
        END
C
C
        SUBROUTINE HGRAM(XMIN,XMAX)
        COMMON /HIST/ BIN(10)
        CHARACTER*1 FREQ(50)
        DIMENSION IOBS(10),CEL(10),CUML(10)
        INTEGER BIN,BMAX
        REAL INC,INT,CEL,CUML
C....
C.....THIS SUBROUTINE PLOTS A HISTOGRAM.
C....
        ISUM        = 0
        DO 4 I       = 1,10
        ISUM        = ISUM + BIN(I)
4       CONTINUE
        DO 5 I       = 1,10
        IOBS(I)     = BIN(I)
        DUM1        = FLOAT(BIN(I))
        DUM2        = FLOAT(ISUM)
        CEL(I)      = DUM1 / DUM2
        CUML(I)     = 0.0
        DO 5 J       = 1,I
        CUML(J)     = CUML(J) + CEL(J)
5       CONTINUE
        DO 10 I      = 1,50
        FREQ(I)     = '*'
10      CONTINUE
        BMAX        = -73057
        DO 15 I      = 1,10
        IF(BIN(I) .GT. BMAX) BMAX = BIN(I)
15      CONTINUE
        DO 16 I      = 1,10
        BIN(I)      = BIN(I) * 50/BMAX
16      CONTINUE
        INC         = (XMAX - XMIN)/10.0
        BEGIN       = XMIN - INC
        WRITE(6,17)
        FORMAT(/,T5,'OBS',3X,'REL',2X,'CUML',3X,'LOW',/)
        DO 20 I      = 1,10
        INT         = BEGIN + FLOAT(I) * INC
        J           = BIN(I)
        WRITE(6,30) IOBS(I),CEL(I),CUML(I),INT,(FREQ(JK),JK = 1,J)
20      CONTINUE
        WRITE(6,25) ISUM
        FORMAT(/,T2,I6)
25      FORMAT(T2,I6,2F6.3,F6.1,2X,'+',1X,50A1)
30      RETURN
        END
C
C
        SUBROUTINE INIT(KOUNT,KASTLE)
        COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
        COMMON /MODEL/ RSEED,RANG(5),CNUG(5)
        COMMON /HERM/ COEF(10,5)
        COMMON /OPT/  NSTOP,KTEST,IKRIG,ITRANS
        COMMON /TEST/ TX(20),TY(20)
        COMMON /VAR/  CO(5),C(5),RANGE(5),ANIS(5),RATIO(5)
        COMMON /CVAR/ CCO(10),CC(10),CRANGE(10),CANIS(10),CRATIO(10)
        COMMON /DAT2/ X(100),Y(100),DAT(100,5)
        COMMON /LTRAN/ LVAR(5),LMEAN(5)
        COMMON /PARM/ MVAR
        DIMENSION DUM(5)
        REAL LVAR,LMEAN
        DOUBLE PRECISION RSEED
C....
C.....THIS SUBROUTINE ACCESSES USER SUPPLIED DATA.
C....
        KOUNT       = 0
        NSTOP       = 0
        KTEST       = 0
        KASTLE      = 0
C....
C.....1. INPUT NUMBER OF RANDOM FUNCTIONS TO BE SIMULATED.
C....
        READ(5,*) MVAR
C....
C.....INPUT INITIAL CONSTANTS
C....
C.....2. ESTABLISH GRID DIMENSIONS
C....
        READ(5,*) NROW,NCOL,XDIM,YDIM
C....
C.....3. ESTABLISH GRID GEOGRAPHIC REGISTRATION
C....
        READ(5,*) YMAX,XMIN
C....
C.....4. ESTABLISH SIMULATION SPATIAL PARAMETERS
C....
        READ(5,*) RSEED
C....
C.....5. ESTABLISH MODEL TRANSFORMATION
C....
        READ(5,*) ITRANS
        IF(ITRANS.EQ.1) GO TO 15
    
```

COS08690  
 COS08700  
 COS08710  
 COS08720  
 COS08730  
 COS08740  
 COS08750  
 COS08760  
 COS08770  
 COS08780  
 COS08790  
 COS08800  
 COS08810  
 COS08820  
 COS08830  
 COS08840  
 COS08850  
 COS08860  
 COS08870  
 COS08880  
 COS08890  
 COS08900  
 COS08910  
 COS08920  
 COS08930  
 COS08940  
 COS08950  
 COS08960  
 COS08970  
 COS08980  
 COS08990  
 COS09000  
 COS09010  
 COS09020  
 COS09030  
 COS09040  
 COS09050  
 COS09060  
 COS09070  
 COS09080  
 COS09090  
 COS09100  
 COS09110  
 COS09120  
 COS09130  
 COS09140  
 COS09150  
 COS09160  
 COS09170  
 COS09180  
 COS09190  
 COS09200  
 COS09210  
 COS09220  
 COS09230  
 COS09240  
 COS09250  
 COS09260  
 COS09270  
 COS09280  
 COS09290  
 COS09300  
 COS09310  
 COS09320  
 COS09330  
 COS09340  
 COS09350  
 COS09360  
 COS09370  
 COS09380  
 COS09390  
 COS09400  
 COS09410  
 COS09420  
 COS09430  
 COS09440  
 COS09450  
 COS09460  
 COS09470  
 COS09480  
 COS09490  
 COS09500  
 COS09510  
 COS09520  
 COS09530  
 COS09540  
 COS09550  
 COS09560  
 COS09570  
 COS09580  
 COS09590  
 COS09600  
 COS09610  
 COS09620  
 COS09630  
 COS09640  
 COS09650  
 COS09660  
 COS09670  
 COS09680  
 COS09690  
 COS09700  
 COS09710

```

DO 10 I = 1,MVAR
10  READ(5,*) LVAR(I),LMEAN(I),RANG(I),CNUG(I)
    CONTINUE
    GO TO 30
15  CONTINUE
    DO 20 I = 1,MVAR
    READ(5,*) (COEF(JK,I), JK = 1,10),RANG(I)
    CNUG(I) = 0.0
20  CONTINUE
30  CONTINUE
C...
C.....6. INPUT CONDITIONING OPTION
C...
    READ(5,*) NSTOP
C...
C.....7. INPUT VARIOGRAM AND CROSS VARIOGRAM PARAMETERS.
C...
    IF(NSTOP.EQ.0) GO TO 1000
    DO 100 I = 1,MVAR
    READ(5,*) CO(I),C(I),RANGE(I),ANIS(I),RATIO(I)
    ANIS(I) = ANIS(I) * 0.01745329
100  CONTINUE
    IF(MVAR.EQ.1) GO TO 160
    N = (MVAR * (MVAR - 1)) / 2
    DO 150 I = 1,N
    READ(5,*) CCO(I),CC(I),CRANGE(I),CANIS(I),CRATIO(I)
    CANIS(I) = CANIS(I) * 0.01745329
150  CONTINUE
160  CONTINUE
C...
C.....8. INPUT CONDITIONING DATA.
C...
    READ(5,*) IKRIG
200  CONTINUE
    READ(5,*) D1,D2,(DUM(JK),JK = 1,MVAR)
    IF(D1.EQ.0.0.AND.D2.EQ.0.0) GO TO 500
    KOUNT = KOUNT + 1
    X(KOUNT) = D2
    Y(KOUNT) = D1
    DO 250 I = 1,MVAR
    DAT(KOUNT,I) = DUM(I)
250  CONTINUE
    GO TO 200
C...
C.....9. INPUT TEST LOCATIONS IF DESIRED.
C...
500  CONTINUE
    READ(5,*) KTEST
    IF(KTEST.EQ.0) GO TO 1000
600  CONTINUE
    READ(5,*) D1,D2
    IF(D1.EQ.0.0.AND.D2.EQ.0.0) GO TO 1000
    KASTLE = KASTLE + 1
    TX(KASTLE) = D1
    TY(KASTLE) = D2
    GO TO 600
1000 CONTINUE
    RETURN
    END
C
C
    SUBROUTINE MATML(A,B,C,I,J,K)
C...
C.....THIS SUBROUTINE IS DESIGNED FOR THE MULTIPLICATION OF
C.....TWO MATRICES, A AND B. FORMULA: A X B = C. NOTE:
C.....THE ORDER OF MULTIPLICATION SHOWN IS ESSENTIAL.
C...
    DIMENSION A(3,3),B(3,3),C(3,3)
    DO 10 LI = 1,3
    DO 10 LJ = 1,3
    C(LI,LJ) = 0.0
    DO 10 LK = 1,3
    C(LI,LJ) = C(LI,LJ) + A(LI,LK) * B(LK,LJ)
10  CONTINUE
    RETURN
    END
C
C
    SUBROUTINE MATMUL(A,B,C,J,K,L)
    DIMENSION A(5,5),B(5,5),C(5,5)
C...
C.....THIS SUBROUTINE PERFORMS A GENERAL MATRIX
C.....MULTIPLICATION OF THE FORM: C = A X B.
C...
    DO 100 I = 1,J
    DO 100 M = 1,L
    C(I,M) = 0.0
    DO 100 N = 1,K
    C(I,M) = C(I,M) + A(I,N) * B(N,M)
100  CONTINUE
    RETURN
    END
C
C
    SUBROUTINE NUGGET(K)
C...
C.....THIS SUBROUTINE ADDS A NUGGET EFFECT TO THE MODEL.
C...
    COMMON /MODEL/ RSEED,RANG(5),CNUG(5)
    COMMON /SIM/ RF(65,65,6)
    COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
    DOUBLE PRECISION RSEED
    DUM = CNUG(K)

```

COS09720  
 COS09730  
 COS09740  
 COS09750  
 COS09760  
 COS09770  
 COS09780  
 COS09790  
 COS09800  
 COS09810  
 COS09820  
 COS09830  
 COS09840  
 COS09850  
 COS09860  
 COS09870  
 COS09880  
 COS09890  
 COS09900  
 COS09910  
 COS09920  
 COS09930  
 COS09940  
 COS09950  
 COS09960  
 COS09970  
 COS09980  
 COS09990  
 COS10000  
 COS10010  
 COS10020  
 COS10030  
 COS10040  
 COS10050  
 COS10060  
 COS10070  
 COS10080  
 COS10090  
 COS10100  
 COS10110  
 COS10120  
 COS10130  
 COS10140  
 COS10150  
 COS10160  
 COS10170  
 COS10180  
 COS10190  
 COS10200  
 COS10210  
 COS10220  
 COS10230  
 COS10240  
 COS10250  
 COS10260  
 COS10270  
 COS10280  
 COS10290  
 COS10300  
 COS10310  
 COS10320  
 COS10330  
 COS10340  
 COS10350  
 COS10360  
 COS10370  
 COS10380  
 COS10390  
 COS10400  
 COS10410  
 COS10420  
 COS10430  
 COS10440  
 COS10450  
 COS10460  
 COS10470  
 COS10480  
 COS10490  
 COS10500  
 COS10510  
 COS10520  
 COS10530  
 COS10540  
 COS10550  
 COS10560  
 COS10570  
 COS10580  
 COS10590  
 COS10600  
 COS10610  
 COS10620  
 COS10630  
 COS10640  
 COS10650  
 COS10660  
 COS10670  
 COS10680  
 COS10690  
 COS10700  
 COS10710  
 COS10720  
 COS10730  
 COS10740

```

STD      = SQRT(DUM)
IF(STD.EQ.0.0) GO TO 100
DO 90 I = 1,NROW
DO 90 J = 1,NCOL
W(I,J,K) = (DRAND(RSEED) - 0.50) * STD * 5.0
RF(I,J,K) = RF(I,J,K) + W
90 CONTINUE
100 CONTINUE
RETURN
END
C
C
SUBROUTINE PROPT
COMMON /OPT/ NSTOP,KTEST,IKRIG,ITRANS
C...
C.....SUBROUTINE TO PRINT PROGRAM OPTIONS.
C...
WRITE(6,50)
IF(NSTOP.EQ.0) WRITE(6,100)
IF(NSTOP.NE.0) WRITE(6,150)
IF(KTEST.EQ.0) WRITE(6,200)
IF(KTEST.NE.0) WRITE(6,250)
IF(IKRIG.EQ.0) WRITE(6,300)
IF(IKRIG.NE.0) WRITE(6,350)
IF(ITRANS.EQ.0) WRITE(6,400)
IF(ITRANS.NE.0) WRITE(6,450)
50 FORMAT(1H1,T30,'PROGRAM OPTIONS IN EFFECT:',//)
100 FORMAT(T30,'NON-CONDITIONAL SIMULATION ONLY',//)
150 FORMAT(T30,'SIMULATION WAS CONDITIONED',//)
200 FORMAT(T30,'NO TEST LOCATIONS DESIRED',//)
250 FORMAT(T30,'TEST LOCATIONS WERE SPECIFIED',//)
300 FORMAT(T30,'ALL VARIABLES WERE FULLY SAMPLED',//)
350 FORMAT(T30,'SOME VARIABLES WERE UNDER-SAMPLED',//)
400 FORMAT(T30,'LINEAR TRANSFORMATION EFFECTED',//)
450 FORMAT(T30,'HERMITIAN TRANSFORMATION EFFECTED',//)
RETURN
END
C
C
SUBROUTINE PRT1(I,J,K,C,BSEED,RMEAN,IVAR)
COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
COMMON /MODEL/ RSEED,RANG(5),CNUG(5)
DOUBLE PRECISION RSEED,BSEED
C...
C.....SUBROUTINE TO PRINT INITIAL PROGRAM RESULTS AND CONSTANTS.
C...
C.....1. GRID DEFINITION.
C...
WRITE(6,50) IVAR
50 FORMAT(1H1,T30,'***** CO-SIMULATION PROGRAM *****',//,
1 T30,'RESULTS FOR VARIABLE ',I10,//)
100 WRITE(6,100) NROW,NCOL,XDIM,YDIM,YMAX,XMIN
FORMAT(1H0,T30,'CO-SIMULATION GRID DIMENSIONS',//,
1 T20,'NUMBER OF ROWS = ',T50,I10,/,
2 T20,'NUMBER OF COLUMNS = ',T50,I10,/,
3 T20,'INCREMENT IN X = ',T50,F10.3,/,
4 T20,'INCREMENT IN Y = ',T50,F10.3,/,
5 T20,'MAXIMUM Y COORDINATE = ',T50,F10.3,/,
6 T20,'MINIMUM X COORDINATE = ',T50,F10.3,/)
C...
C.....2. MODEL PARAMETERS
C...
WRITE(6,200) BSEED,RSEED,RMEAN,RANG(IVAR),CNUG(IVAR)
200 FORMAT(//,T30,'MODEL PARAMETERS',//,
1 T20,'BEGINNING RANDOM SEED = ',T50,D20.11,/,
2 T20,'FINAL RANDOM NUMBER SEED = ',T50,D20.11,/,
3 T20,'MEAN OF RANDOM NUMBERS = ',T50,F20.3,/,
4 T20,'RANGE OF SPATIAL STRUCTURE = ',T50,F20.3,/,
5 T20,'VARIANCE OF SPATIAL NOISE = ',T50,F20.3,/)
C...
C.....3. CONSTANTS COMPUTED BY PROGRAM
C...
WRITE(6,300) I,J,K,C
300 FORMAT(//,T30,'INITIAL CONSTANTS',//,
1 T20,'NGMAX = ',T50,I10,/,
2 T20,'NR = ',T50,I10,/,
3 T20,'KD = ',T50,I10,/,
4 T20,'C = ',T50,F10.3,/)
RETURN
END
C
C
SUBROUTINE PRT2(IJK)
DIMENSION BUF(100)
COMMON /SIM/ RF(65,65,6)
COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
COMMON /HIST/ BIN(10)
INTEGER BUF,BIN
C...
C.....THIS SUBROUTINE PRINTS A PICTURE OF THE SIMULATION RESULTS.
C...
DO 5 I = 1,10
BIN(I) = 0
5 CONTINUE
N = NCOL
IF(NCOL.GT.70) N = NCOL / 2
C...
C.....FIND THE MAXIMUM AND MINIMUM VALUES OF THE SIMULATED RF.
C...
RFMIN = 73057.00
RFMAX = -73057.00
DO 10 I = 1,NROW
DO 10 J = 1,NCOL

```

```

B      = RF(I,J,IJK)
IF(B.LT.RFMIN) RFMIN = B
IF(B.GT.RFMAX) RFMAX = B
10    CONTINUE
      DIFF      = RFMAX - RFMIN
      RFINC     = DIFF / 10.0
      DO 20 I   = 1,NROW
      DO 20 J   = 1,NCOL
      DO 20 K   = 1,10
      XOLD     = FLOAT(K - 1)*RFINC + RFMIN
      XNEW     = FLOAT(K) *RFINC + RFMIN
      B        = RF(I,J,IJK)
20    IF(B.GE.XOLD.AND.B.LT.XNEW) BIN(K) = BIN(K) + 1
      CONTINUE
      WRITE(6,30)
30    FORMAT(IH1,T30,'HISTOGRAM OF SIMULATION',/)
      CALL HGRAM(RFMIN,RFMAX)
      WRITE(6,400)
      IF(N.NE.NCOL) GO TO 200
      DO 100 I  = 1,NROW
      DO 90 J  = 1,NCOL
      L       = 0
      IF(RF(I,J,IJK).EQ.RFMIN) GO TO 60
      DO 50 K  = 1,10
      L       = K - 1
      VAL2    = RFMIN + RFINC * FLOAT(K)
      VAL1    = RFMIN + RFINC * FLOAT(L)
      B       = RF(I,J,IJK)
50    IF(B.GT.VAL1.AND.B.LE.VAL2) GO TO 60
      CONTINUE
60    CONTINUE
      BUF(J)  = L
90    CONTINUE
      WRITE(6,500) (BUF(JK),JK = 1,N)
100   CONTINUE
      GO TO 300
200   CONTINUE
      DO 290 I  = 1,NROW,2
      J1      = 0
      DO 280 J  = 1,NCOL,2
      J1      = J1 + 1
      L       = 0
      IF(RF(I,J,IJK).EQ.RFMIN) GO TO 260
      DO 210 K  = 1,10
      L       = K - 1
      VAL2    = RFMIN + RFINC * FLCAT(K)
      VAL1    = RFMIN + RFINC * FLOAT(L)
      B       = RF(I,J,IJK)
210   IF(B.GT.VAL1.AND.B.LE.VAL2) GO TO 260
      CONTINUE
260   CONTINUE
      BUF(J1) = L
280   CONTINUE
      WRITE(6,500) (BUF(JK),JK = 1,N)
290   CONTINUE
300   CONTINUE
400   FORMAT(IH1,T30,'SIMULATION RESULTS',/,
1     T20,'IF NCOL.GT.70, THE FOLLOWING',/,
2     T20,'IS EVERY OTHER COLUMN BY EVERY',/,
3     T20,'OTHER ROW.',/)
500   FORMAT(T3,75I1)
C....
C..... COMPUTE LEGEND
C....
      WRITE(6,700)
      DO 600 I  = 1,10
      J       = I - 1
      DUM1    = RFMIN + RFINC * FLCAT(J)
      DUM2    = RFMIN + RFINC * FLOAT(I)
      WRITE(6,800) J,DUM1,DUM2
600   CONTINUE
700   FORMAT(IH1,T30,'LEGEND',/,T22,'MAP VALUE',/,
1     T37,'LOW RANGE',T51,'HIGH RANGE',/)
800   FORMAT(T20,I10,T35,E10.2,T50,E10.2)
      RETURN
      END
C
C
      SUBROUTINE PRT3(KOUNT)
      COMMON /VAR/ CO(5),C(5),RANGE(5),ANIS(5),RATIO(5)
      COMMON /DAT2/ X(100),Y(100),DAT(100,5)
      COMMON /CVAR/ CCO(10),CC(10),CRANGE(10),CANIS(10),CRATIO(10)
      COMMON /PARM/ MVAR
C....
C..... THIS SUBROUTINE PRINTS INITIAL CONDITIONING INFORMATION.
C....
      WRITE(6,100)
      DO 10 I  = 1,MVAR
      WRITE(6,200) I,CO(I),C(I),RANGE(I),ANIS(I),RATIO(I)
10    CONTINUE
      IF(MVAR.EQ.1) GO TO 25
      N      = (MVAR * (MVAR - 1)) / 2
      WRITE(6,300)
      DO 20 I  = 1,N
      WRITE(6,200) I,CCO(I),CC(I),CRANGE(I),CANIS(I),CRATIO(I)
20    CONTINUE
25    CONTINUE
      WRITE(6,400)
      DO 30 I  = 1,KOUNT
      WRITE(6,500) X(I),Y(I),(DAT(I,JK),JK = 1,MVAR)
30    CONTINUE
100   FORMAT(IH1,T30,'**** CONDITIONING RESULTS ****',/,
1     T30,'VARIOGRAM PARAMETERS',/)

```

COS11780  
 COS11790  
 COS11800  
 COS11810  
 COS11820  
 COS11830  
 COS11840  
 COS11850  
 COS11860  
 COS11870  
 COS11880  
 COS11890  
 COS11900  
 COS11910  
 COS11920  
 COS11930  
 COS11940  
 COS11950  
 COS11960  
 COS11970  
 COS11980  
 COS11990  
 COS12000  
 COS12010  
 COS12020  
 COS12030  
 COS12040  
 COS12050  
 COS12060  
 COS12070  
 COS12080  
 COS12090  
 COS12100  
 COS12110  
 COS12120  
 COS12130  
 COS12140  
 COS12150  
 COS12160  
 COS12170  
 COS12180  
 COS12190  
 COS12200  
 COS12210  
 COS12220  
 COS12230  
 COS12240  
 COS12250  
 COS12260  
 COS12270  
 COS12280  
 COS12290  
 COS12300  
 COS12310  
 COS12320  
 COS12330  
 COS12340  
 COS12350  
 COS12360  
 COS12370  
 COS12380  
 COS12390  
 COS12400  
 COS12410  
 COS12420  
 COS12430  
 COS12440  
 COS12450  
 COS12460  
 COS12470  
 COS12480  
 COS12490  
 COS12500  
 COS12510  
 COS12520  
 COS12530  
 COS12540  
 COS12550  
 COS12560  
 COS12570  
 COS12580  
 COS12590  
 COS12600  
 COS12610  
 COS12620  
 COS12630  
 COS12640  
 COS12650  
 COS12660  
 COS12670  
 COS12680  
 COS12690  
 COS12700  
 COS12710  
 COS12720  
 COS12730  
 COS12740  
 COS12750  
 COS12760  
 COS12770  
 COS12780  
 COS12790  
 COS12800

```

2   T3,'VARIABLE',T15,'NUGGET',T27,'SILL',T36,'RANGE',T47,
3   'ANIS',T56,'RATIO',//)
200  FORMAT(I10,5F10.3)
300  FORMAT(//,T30,'CROSS-CORRELATION PARAMETERS',//,
1   T7,'PAIR',T15,'NUGGET',T27,'SILL',T36,'RANGE',T47,'ANIS',
2   T56,'RATIO',//)
400  FORMAT(//,T30,'CONDITIONING DATA',//,
1   T7,'EAST',T16,'NORTH',T27,'VAR1',T37,'VAR2',T47,'VAR3',
2   T57,'VAR4',T67,'VAR5',//)
500  FORMAT(7F10.3)
      RETURN
      END
C
C
      SUBROUTINE PTEST(K)
      COMMON /TEST/ TX(20),TY(20)
      COMMON /PARM/ MVAR
      COMMON /SIM/  RF(65,65,6)
      COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
C...
C..... SUBROUTINE PTEST PRINTS THE VALUES WITHIN THE SIMULATED
C..... MODEL CLOSEST TO LOCATIONS: (TX,TY).
C...
      WRITE(6,100)
      YBEGIN = YMAX + 0.5 * YDIM
      XBEGIN = XMIN - 0.5 * XDIM
      DO 10 I = 1,K
      IROW = (YBEGIN - TY(I)) / YDIM
      JCOL = (TX(I) - XBEGIN) / XDIM
      IF(IROW.LE.0) IROW = 1
      IF(IROW.GT.NROW) IROW = NROW
      IF(JCOL.LE.0) JCOL = 1
      IF(JCOL.GT.NCOL) JCOL = NCOL
      WRITE(6,200) IROW,JCOL,TX(I),TY(I),(RF(IROW,JCOL,JK),
1      JK = 1,MVAR)
10   CONTINUE
100  FORMAT(1H1,T30,'SIMULATED VALUES AT TEST LOCATIONS',//,
1   T4,'ROW',T9,'COL',T20,'X',T30,'Y',T37,'VAR1',T47,'VAR2',
2   T57,'VAR3',T67,'VAR4',T77,'VAR5',//)
200  FORMAT(T3,2I4,7F10.3)
      RETURN
      END
C
C
      SUBROUTINE SCALG(A,B,X)
      COMMON /PARM/ MVAR
      DIMENSION A(5,5),B(5,5),X(5,5),TEMP(5,10)
C...
C..... THIS SUBROUTINE PERFORMS THE NORMALIZATION BY THE
C..... PIVOT TERM FOR GAUSS ELIMINATION MATRIX INVERSION.
C...
      MVAR2 = MVAR * 2
      DO 10 I = 1,MVAR
      DO 10 J = 1,MVAR
      TEMP(I,J) = A(I,J)
10   CONTINUE
      DO 20 I = 1,MVAR
      DO 20 J = 1,MVAR
      K = MVAR + J
      TEMP(I,K) = B(I,J)
20   CONTINUE
      DO 30 I = 1,MVAR
      IP = I + 1
      DO 30 K = 1,MVAR
      IF(I-K) 26,30,26
26   IF(TEMP(I,I).EQ.0.0) GO TO 30
      F = (-TEMP(K,I)) / TEMP(I,I)
      DO 27 L = IP,MVAR2
      TEMP(K,L) = TEMP(K,L) + F*TEMP(I,L)
27   CONTINUE
      DO 40 I = 1,MVAR
      DO 40 J = 1,MVAR
      K = MVAR + J
      IF(TEMP(I,I).EQ.0.0) GO TO 40
      X(I,J) = TEMP(I,K) / TEMP(I,I)
40   CONTINUE
      RETURN
      END
C
C
      FUNCTION TRACE(A)
      COMMON /PARM/ MVAR
      REAL A(5,5)
C...
C..... THIS FUNCTION COMPUTES THE TRACE OF SQUARE MATRIX, A.
C...
      TRACE = 0.0
      DO 10 I = 1,MVAR
      TRACE = TRACE + A(I,I)
10   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE TRANS(XMEAN,VAR,K)
      COMMON /SIM/ RF(65,65,6)
      COMMON /LTRAN/ LVAR(5),LMEAN(5)
      COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
      COMMON /MODEL/ RSEED,RANG(5),CNUG(5)
      COMMON /HERM/ COEF(10,5)
      COMMON /CPT/ NSTOP,KTEST,IKRIG,ITRANS
      REAL LVAR,LMEAN,C(12)
      DOUBLE PRECISION RSEED

```

```

C...
C..... THIS SUBROUTINE PROVIDES A LINEAR OR HERMITIAN
C..... TRANSFORMATION FROM A DISTRIBUTION (0,1). FOR THE LINEAR
C..... TRANSFORMATION, THE PROCEDURE USED IS:
C.....
C..... WHERE,
C.....
C..... T = (1 - M1) * S1/S2 + M2
C.....
C..... T = TRANSFORMED VALUE
C..... I = SIM. VALUE IN ORIGINAL DISTRIBUTION
C..... M1 = MEAN OF ORIGINAL DISTRIBUTION
C..... M2 = MEAN OF TRANSFORMED DISTRIBUTION
C..... S2 = STANDARD DEVIATION OF ORIGINAL DIST.
C..... S1 = STANDARD DEVIATION OF TRANSFORMED DIST.
C.....
C..... DO 5 I = 1,12
C(I) = 0.0
5 CONTINUE
IF(I*TRANS.EQ.1) GO TO 200
S1 = SQRT(LVAR(K))
S2 = SQRT(VAR)
DO 100 I = 1,NROW
DO 100 J = 1,NCOL
RF(I,J,K) = (RF(I,J,K) - XMEAN) * S1/S2 + LMEAN(K)
100 CONTINUE
C.....
C..... ADD A NUGGET EFFECT TO THE MODEL IF DESIRED.
C.....
C..... CALL NUGGET(K)
RETURN
200 CONTINUE
C.....
C..... HERMITIAN TRANSFORMATION.
C..... CORRECT THE MODEL FOR DEFICIENCIES THEN APPLY TRANSFORMATION
C.....
C..... S2 = SQRT(VAK)
DO 210 I = 1,NROW
DO 210 J = 1,NCOL
X = (RF(I,J,K) - XMEAN) * 1.0 / S2
DO 205 M = 1,1
C(M) = COEF(M,K) + X * C(M+1) - M*C(M+2)
205 CONTINUE
RF(I,J,K) = C(1)
DO 210 N = 1,12
C(N) = 0.0
210 CONTINUE
RETURN
END
C
C
SUBROUTINE UNSAM(K)
COMMON /FURM/ A(100,100)
COMMON /DAT2/ X(100),Y(100),DAT(100,5)
COMMON /PARM/ MVAR
C.....
C..... THIS SUBROUTINE MODIFIES THE MATRIX, A, TO ACCOUNT
C..... FOR UNDERSAMPLED RANDOM FUNCTIONS. A ZERO DATA
C..... VALUE IS INDICATIVE OF UNDERSAMPLING.
C.....
C..... L = K + 1
C..... LVAR = L * MVAR
DO 100 I = 1,K
DO 90 J = 1,MVAR
IF(DAT(I,J).NE.0.0) GO TO 90
NI = (I - 1) * MVAR + J
DO 80 M = 1,LVAR
A(NI,M) = 0.0
A(M,NI) = 0.0
80 CONTINUE
90 CONTINUE
100 CONTINUE
RETURN
END
C
C
SUBROUTINE UNVEC(XX,K)
COMMON /DAT2/ X(100),Y(100),DAT(100,5)
COMMON /PARM/ MVAR
DIMENSION XX(100,5)
C.....
C..... THIS SUBROUTINE MODIFIES THE MEASUREMENT VECTOR TO
C..... ACCOUNT FOR UNDERSAMPLED RANDOM FUNCTIONS. A ZERO
C..... DATA VALUE IS INDICATIVE OF UNDERSAMPLING.
C.....
C..... DO 100 I = 1,K
C..... DO 90 J = 1,MVAR
C..... IF(DAT(I,J).NE.0.0) GO TO 90
C..... NI = (I - 1) * MVAR + J
C..... DO 80 M = 1,MVAR
C..... XX(NI,M) = 0.0
80 CONTINUE
90 CONTINUE
100 CONTINUE
RETURN
END
C
C
SUBROUTINE VARG(XMEAN,VAR,IJK)
COMMON /SIM/ RF(65,65,6)
COMMON /GRID/ NROW,NCOL,XDIM,YDIM,YMAX,XMIN
DIMENSION GAMMA(20)
C.....
C..... THIS SUBROUTINE COMPUTES THE SEMI-VARIOGRAM OF THE
C..... SIMULATED RANDOM FUNCTION, RF. THE REGULAR GRID

```

COS13840  
 COS13850  
 COS13860  
 COS13870  
 COS13880  
 COS13890  
 COS13900  
 COS13910  
 COS13920  
 COS13930  
 COS13940  
 COS13950  
 COS13960  
 COS13970  
 COS13980  
 COS13990  
 COS14000  
 COS14010  
 COS14020  
 COS14030  
 COS14040  
 COS14050  
 COS14060  
 COS14070  
 COS14080  
 COS14090  
 COS14100  
 COS14110  
 COS14120  
 COS14130  
 COS14140  
 COS14150  
 COS14160  
 COS14170  
 COS14180  
 COS14190  
 COS14200  
 COS14210  
 COS14220  
 COS14230  
 COS14240  
 COS14250  
 COS14260  
 COS14270  
 COS14280  
 COS14290  
 COS14300  
 COS14310  
 COS14320  
 COS14330  
 COS14340  
 COS14350  
 COS14360  
 COS14370  
 COS14380  
 COS14390  
 COS14400  
 COS14410  
 COS14420  
 COS14430  
 COS14440  
 COS14450  
 COS14460  
 COS14470  
 COS14480  
 COS14490  
 COS14500  
 COS14510  
 COS14520  
 COS14530  
 COS14540  
 COS14550  
 COS14560  
 COS14570  
 COS14580  
 COS14590  
 COS14600  
 COS14610  
 COS14620  
 COS14630  
 COS14640  
 COS14650  
 COS14660  
 COS14670  
 COS14680  
 COS14690  
 COS14700  
 COS14710  
 COS14720  
 COS14730  
 COS14740  
 COS14750  
 COS14760  
 COS14770  
 COS14780  
 COS14790  
 COS14800  
 COS14810  
 COS14820  
 COS14830  
 COS14840  
 COS14850  
 COS14860

```

C.....SPACING IS TAKEN ADVANTAGE OF IN THE EAST-WEST DIRECTION.
C.....1. COMPUTE THE MEAN AND VARIANCE OF THE SIMULATION.
C.....
      DO 5 I = 1,20
      GAMMA(I) = 0.0
5      CONTINUE
      VAR = 0.0
      XMEAN = 0.0
      DO 10 I = 1,NROW
      DO 10 J = 1,NCOL
      XMEAN = XMEAN + RF(I,J,IJK)
10     CONTINUE
      NUM = NROW * NCOL
      XMEAN = XMEAN / FLOAT(NUM)
      DO 20 I = 1,NROW
      DO 20 J = 1,NCOL
      VAR = VAR + (RF(I,J,IJK) - XMEAN)**2
20     CONTINUE
      VAR = VAR / FLOAT(NUM - 1)
      NCOL1 = NCOL / 20
      DO 1000 K = 1,20
      KOUNT = 0
      DUM1 = 0.0
      L = K * NCOL1
      I1 = NCOL - L + 2
      DO 100 I = 1,NROW
      DO 100 J = 1,I1
      I4 = J + L
      IF(I4.GT.NCOL) GO TO 100
      KOUNT = KOUNT + 1
      C = RF(I,I4,IJK)
      D = RF(I,J,IJK)
      DUM1 = DUM1 + (C - D)**2
100    IF(KOUNT.GT.5000) GO TO 200
200    CONTINUE
      N2 = 2 * KOUNT
      IF(N2.EQ.0) GO TO 1000
      GAMMA(K) = DUM1 / FLOAT(N2)
1000   CONTINUE
      WRITE(6,2000)
      DO 1500 I = 1,20
      WRITE(6,3000) I,GAMMA(I)
1500   CONTINUE
2000   FORMAT(1H1,T30,'VARIATION RESULTS',/,
1      T25,'CLASS',T35,'GAMMA',/)
3000   FORMAT(T20,I10,T30,F10.3)
4000   WRITE(6,4000) XMEAN, VAR
1      FORMAT(/,T30,'MEAN AND VARIANCE OF THE SIMULATION',/,
1      T20,'MEAN = ',E15.3,T50,'VARIANCE = ',E15.3,/)
      RETURN
      END
C
C
      SUBROUTINE VECT
      COMMON /DVECT/ ICOS(3,15)
      DIMENSION R(3,3),TEMP(3,3),TEMP1(3,3)
C.....THIS SUBROUTINE DEFINES THE 15 VECTORS THROUGH ROTATION BY R.
C.....
      REAL K,ICOS
      K = 0.618033989
      ICOS(1,1) = K
      ICOS(1,2) = 1.0
      ICOS(1,3) = 1.0 + K
      ICOS(2,1) = 1.0
      ICOS(2,2) = -1.0 - K
      ICOS(2,3) = K
      ICOS(3,1) = K + 1.0
      ICOS(3,2) = K
      ICOS(3,3) = -1.0
      R(1,1) = 0.50
      R(1,2) = -(K + 1.0) / 2.0
      R(1,3) = K / 2.0
      R(2,1) = (K + 1.0) / 2.0
      R(2,2) = K / 2.0
      R(2,3) = -0.50
      R(3,1) = K / 2.0
      R(3,2) = 0.50
      R(3,3) = (K + 1.0) / 2.0
C.....COMPUTE REMAINING 12 VECTORS
C.....
      DO 100 M = 1,4
      DO 10 I = 1,3
      DO 10 J = 1,3
      N = (M - 1) * 3 + J
      TEMP(I,J) = ICOS(I,N)
10     CONTINUE
      CALL MATML1(TEMP,R,TEMP1,3,3,3)
C.....TEMP1 CONTAINS NEW ROTATED VECTORS; THIS IS PLACED INTO ICOS.
C.....
      DO 20 I = 1,3
      DO 20 J = 1,3
      N = M * 3 + J
      ICOS(I,N) = TEMP1(I,J)
20     CONTINUE
100    CONTINUE
      RETURN
      END

```

COS14870  
 COS14880  
 COS14890  
 COS14900  
 COS14910  
 COS14920  
 COS14930  
 COS14940  
 COS14950  
 COS14960  
 COS14970  
 COS14980  
 COS14990  
 COS15000  
 COS15010  
 COS15020  
 COS15030  
 COS15040  
 COS15050  
 COS15060  
 COS15070  
 COS15080  
 COS15090  
 COS15100  
 COS15110  
 COS15120  
 COS15130  
 COS15140  
 COS15150  
 COS15160  
 COS15170  
 COS15180  
 COS15190  
 COS15200  
 COS15210  
 COS15220  
 COS15230  
 COS15240  
 COS15250  
 COS15260  
 COS15270  
 COS15280  
 COS15290  
 COS15300  
 COS15310  
 COS15320  
 COS15330  
 COS15340  
 COS15350  
 COS15360  
 COS15370  
 COS15380  
 COS15390  
 COS15400  
 COS15410  
 COS15420  
 COS15430  
 COS15440  
 COS15450  
 COS15460  
 COS15470  
 COS15480  
 COS15490  
 COS15500  
 COS15510  
 COS15520  
 COS15530  
 COS15540  
 COS15550  
 COS15560  
 COS15570  
 COS15580  
 COS15590  
 COS15600  
 COS15610  
 COS15620  
 COS15630  
 COS15640  
 COS15650  
 COS15660  
 COS15670  
 COS15680  
 COS15690  
 COS15700  
 COS15710  
 COS15720  
 COS15730  
 COS15740  
 COS15750  
 COS15760  
 COS15770  
 COS15780  
 COS15790  
 COS15800  
 COS15810  
 COS15820  
 COS15830  
 COS15840  
 COS15850  
 COS15860  
 COS15870  
 COS15880