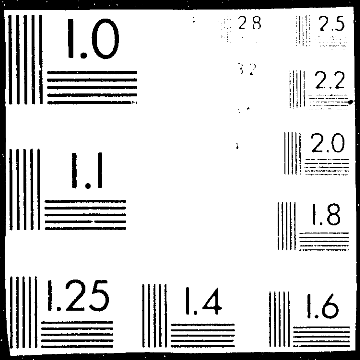
1 OF 1



in the sequential indicator realizations caused by transformation from the class partitions to the real values. Deutsch and Journel (1992) acknowledge the extra nugget artificially reducing the nugget effect of the indicator semi- of the problem

Conf-9305268--1

All papers must include the following statement:
This work performed at Sandia National Laboratories is supported by the U.S. Department of Energy under contract DE-AC04-76DP00789.

STOCHASTIC SIMULATION FOR IMAGING SPATIAL UNCERTAINTY: COMPARISON AND EVALUATION OF AVAILABLE ALGORITHMS

C. A. GOTWAY
Department of Biometry
University of Nebraska
Lincoln, NE 68583-0712

B.M. RUTHERFORD
Sandia National Laboratories
P.O Box 5800
Albuquerque, NM 87185

Abstract. Stochastic simulation has been suggested as a viable method for characterizing the uncertainty associated with the prediction of a nonlinear function of a spatially-varying parameter. Geostatistical simulation algorithms generate realizations of a random field with specified statistical and geostatistical properties. A nonlinear function (called a transfer function) is evaluated over each realization to obtain an uncertainty distribution of a system response that reflects the spatial variability and uncertainty in the parameter. Crucial management decisions, such as potential regulatory compliance of proposed nuclear waste facilities and optimal allocation of resources in environmental remediation, are based on the resulting system response uncertainty distribution.

Many geostatistical simulation algorithms have been developed to generate the random fields, and each algorithm will produce fields with different statistical properties. These different properties will result in different distributions for system response, and potentially, different managerial decisions. The statistical properties of the resulting system response distributions are not completely understood, nor is the ability of the various algorithms to generate response distributions that adequately reflect the associated uncertainty.

This paper reviews several of the algorithms available for generating random fields. Algorithms are compared in a designed experiment using seven exhaustive data sets with different statistical and geostatistical properties. For each exhaustive data set, a number of realizations (both unconditional and data-conditioned) are generated using each simulation algorithm. The realizations are used with each of several deterministic transfer functions to produce a cumulative uncertainty distribution function of a system response. The uncertainty distributions are then compared to the single value obtained from the corresponding exhaustive data set. The results of the study facilitate comparisons between the individual methods, allow an assessment of the consistency of the simulation algorithms, and indicate potential for bias or imprecision.

INTRODUCTION

Stochastic simulation provides a way to incorporate various types of uncertainty into prediction of a complex system response. Usually, some information is available on a parameter of interest (for example, the permeability of a sandstone formation), but the transfer function (a groundwater flow model, for example) may require a detailed spatial map of this parameter. The exhaustive sampling necessary to obtain such a map is usually not feasible. One alternative is to generate realizations of a random field that share the available information on the parameter of interest. These realizations serve as input to the transfer function that computes a system response for each. If the realizations characterize the spatial uncertainty of the parameter of interest, the resulting distribution of predicted system response values will reflect the uncertainty (see Figure 1). This approach, proposed in Journel (1988), is widely used in hydrology, petroleum engineering, and the environmental sciences. Crucial management decisions such, as potential regulatory compliance of proposed nuclear waste sites and optimal allocation of resources in environmental remediation, are based on the resulting system response uncertainty distribution.

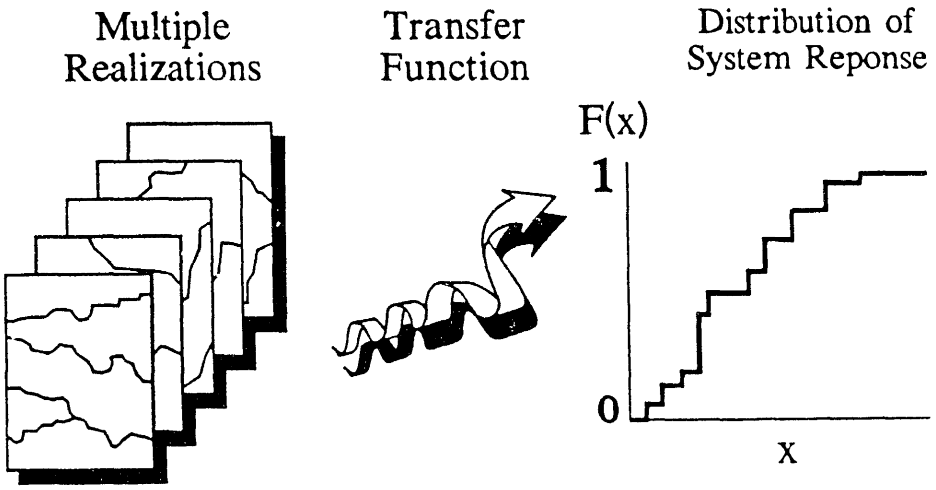


Figure 1. Schematic Illustration Of Stochastic Simulation. Modified from Journel (1988).

Many different stochastic simulation algorithms could be used to construct the realizations and each may assign different statistical and spatial features to the generated fields. These differences may be due to the order in which the simulated values are obtained (the method of generation), the statistical and geostatistical inputs to the algorithms (i.e., the summary functions such as the variogram), and the degree to which the summary functions of the realizations match the specified input summary functions. Consequently, the distribution of predicted response values, and the resulting inferences drawn from this distribution, depend on the particular simulation algorithm implemented. For the scientist using stochastic simulation, the differences among the algorithms and the resulting effects on the system response uncertainty distributions need to be understood so that appropriate algorithms are used in each application.

To date, very little has been done to evaluate and compare these algorithms as they are currently being used. Journel and Alabert (1989) use the Berea Sandstone exhaustive data set to compare sequential indicator simulation to Gaussian simulation. They recommend the use of indicator methods when it is necessary to characterize strong connectivity of extreme (high or low) values. Deutsch and

Journel (1992b) use the same data set to compare sequential indicator simulation, sequential Gaussian si ulation, and simulated annealing using petroleum industry-based transfer functions. Based on results from one set of simulated system response distributions, they find all methods to be feasible, accurate, and precise. Hansen (1992) uses several indicator-based synthetic exhaustive data sets to evaluate the sequential indicator simulation algorithm. The results of this study indicate that the sequential indicator simulation algorithm, when applied to problems where conditioning data are available, may "over-condition" the data, causing the response uncertainty distribution to be very precise but biased for the true response as obtained from the exhaustive data sets. Clearly, much more work is needed to evaluate the many geostatistical simulation algorithms over the range of applications considered in practice.

The purpose of this paper is to present the results of a comprehensive study designed to evaluate and compare geostatistical simulation algorithms using a number of different exhaustive data sets that represent a variety of spatial phenomena. The objective of the study, discussed in subsequent sections, is to obtain more information on how the different simulation algorithms work in basic simulation applications. Using them in a designed experiment with many exhaustive data sets will facilitate comparison of realizations and response uncertainty distributions produced by the various simulation algorithms, allow an assessment of the robustness of the methods to differences in underlying data distributions, and indicate potential biases or imprecision due to a particular algorithm or method. Section II gives the specifics of the design of this experiment, and Section III summarizes the results. A discussion of the findings and recommendations for implementation and future research are then given in Section IV.

DESIGN OF THE COMPARATIVE EXPERIMENT

The basic experiment is as follows. For each of seven exhaustive data sets, 200 realizations of a random field model are generated using the simulation algorithms appropriate for that data set. Input to each simulation method is based on "truth" as derived from the exhaustive data set. Truth will be quantified using appropriate "summary functions," i.e., functions that incorporate the statistical and geostatistical features of the field, such as the semivariogram or indicator semivariogram. One-hundred of the fields are conditioned on "data" randomly selected from the original exhaustive data set. The remaining 100 fields are left unchanged as unconditional realizations. For each realization, several transfer functions are computed, each giving one value of the system response. The results of the experiment provide several uncertainty distributions for every simulation algorithm, each of which corresponds to one of the exhaustive-data-set/transfer-function scenarios. Half of these distributions will be based on unconditional realizations, and the other half will be data-conditioned.

A. *Simulation Algorithms and Software*

A brief description of all algorithms and software used in this st dy is provided below. Because of time and space constraints, we have omitted several promising simulation algorithms such as fractal simulation algorithms, spectral algorithms, Boolean and random set algorithms, and nearest-neighbor approaches. These methods will be explored in subsequent studies.

1. LU Decomposition. The LU decomposition method is based on an LU or Cholesky-type of decomposition of the covariance matrix between data locations

and grid locations. Specifically, this covariance matrix can be decomposed as

$$C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = LU = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{pmatrix},$$

where C_{11} is the covariance between data at data locations, C_{22} is the covariance between data at grid locations, and C_{12} is the covariance between data at data locations and those at grid locations. A conditional Gaussian simulation is obtained by simulating a vector ϵ of independent normal random variables with mean zero and unit variance, and using the data vector z in the transformation

$$\begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} L_{11}^{-1} z \\ \epsilon \end{pmatrix} = \begin{pmatrix} z \\ L_{21} L_{11}^{-1} z + L_{22}\epsilon \end{pmatrix}.$$

Further details of this algorithm can be found in Davis (1987a), Cressie (1991), and Dowd (1992). LU decomposition is relatively easy to implement, can handle any type of covariance function and anisotropy, and can incorporate data conditioning efficiently. However, the amount of storage required can limit the size of the simulation grid that can be efficiently considered. Moreover, when the simulation grid size is large and the covariance matrix is sparse, numerical inaccuracies may result. To circumvent these difficulties, approximations that provide more efficient and stable calculations have been introduced by Quimby (1986) and Davis (1987b).

For the continuous-variable simulations required in this manuscript, the basic LU decomposition algorithm in Deutsch and Journel (1992a) was used for the continuous-variable simulations. The random number generator provided in the software was replaced with one found in Press, et al. (1986). In implementing the simulations required for this study, we found storage space and run times to be more of a nuisance than a limitation. Each conditional simulation took about 16 minutes on an IBM RS6000 workstation, which given the computational nature of many environmental applications, is hardly a limitation.

2. Turning Bands. The turning bands method was developed to ease the computational burden in generating three dimensional fields. The method works by simulating one-dimensional processes on lines regularly spaced in two- or three-dimensions. The one-dimensional simulations are then projected onto the spatial coordinates and averaged to give the required two- or three-dimensional simulated value. The turning bands algorithm is a fast and efficient method of random field generation, but the use of a separate data-conditioning step based on kriging can reduce its efficiency for generating conditional random fields. Perhaps the biggest drawback of the method is the limitation on the choice of covariance function that can be specified. One list of possible choices is provided in Zimmerman and Wilson (1990), and additional descriptions and properties of this algorithm can be found in Journel (1974), and Mantoglou and Wilson (1982).

The turning bands computer code TUBA (Zimmerman and Wilson, 1990) was used to generate the continuous-variable simulations since this code provides a high degree of flexibility in the choice of turning bands parameters. To reduce banding artifacts due to the one-dimensional line processes, 64 turning bands were used in every simulation at the recommendation of the author of the TUBA code. The software allows the user several choices for the covariance function, but it does not generate fields with a specified nugget effect. It does not (nor does any other turning bands code known to the authors) easily incorporate zonal anisotropy or anisotropy that departs from the coordinate directions. To use the turning bands code in such situations requires the addition of two or more realizations with simple covariance structures. These additional computations were done for simulations in which a nugget effect was required, but, in the cases where complex semivariogram models were necessary, approximate models, compatible with the turning bands software, were used.

3. Sequential Gaussian and 4. Sequential Indicator Simulation Algorithms. Both of these methods are based on a sequential approach to simulation based on approximations to Bayes' theorem, and are described in Journel and Alabert (1989), Gomez-Hernandez and Srivastava (1990), and Deutsch and Journel (1992a). The basic conditional sequential simulation algorithm is as follows: 1) Define a random path through all grid nodes; 2) Draw a value from the conditional distribution of the random variable at the first grid node given the (n) conditioning data; 3) Add this new value to the conditioning data set; 4) Draw a value from the conditional distribution of the variable at node two given the (n+1) conditioning data, and 5) Repeat until all nodes are simulated. In sequential Gaussian simulation, the conditioning data are first transformed to standard Gaussian values and the semivariogram of the transformed data is specified. Simple kriging is used to obtain estimates of the necessary conditional distributions. At each node, the kriged value obtained from simulated and conditioning data, and the associated kriging variance are used to specify the conditional Gaussian distribution. Realizations are then drawn randomly from this distribution. Finally, the results of the Gaussian simulation are transformed back to the original data space. In the sequential indicator simulation approach, no assumptions are made about the parametric form of the conditional distributions. The conditioning data are transformed to indicators defined by threshold values based on available data and other relevant information. Estimates of the conditional distributions at each grid node are given by simple indicator kriging using corresponding indicator semivariograms.

As discussed in Dowd (1992) these methods have several advantages including automatic handling of anisotropies and data conditioning, and fast computer implementation since an efficient kriging algorithm with a moving neighborhood search capability is all that is required. However, since sequential methods are relatively new, their properties and limitations, if any, are unknown. Artifacts of these algoithms could be present in the generated fields or in the system response uncertainty distribution. In particular, the conditional distributions obtained from the sequential indicator simulation algorithm do not respect the properties of cumulative distribution functions. It is possible to obtain probability estimates larger than 1, less than zero, and often the resulting conditional distributions are not monotonic. Although an artificial correction is used to force the desired properties, it is not clear what effect this correction may have on the realizations and the system response distribution.

Both sequential algorithms were used for all continuous-variable simulations using the software given in Deutsch and Journel (1992a). As with LU decomposition, the random number generator was replaced by that given in Press, et al., (1986). Both algorithms were very flexible, efficient, and easy to use. However, several user-specified parameters- such as the use of simple kriging versus ordinary kriging, the maximum number of simulated nodes retained for kriging, octant-search parameters, and, in particular, upper and lower tail extrapolation choices can affect the efficiency of the algorithms, the nature of the realizations, and the resulting uncertainty distributions.

Sequential indicator simulation is especially straightforward when applied to generating realizations of a categorical variable. Thus, this algorithm was also used to produce realizations of the GCD and Boolean exhaustive data sets (discussed below).

5. Truncated Gaussian Random Function Approach This method, discussed in Matheron et al., (1987), Galli et al., (1990), and Dowd (1992) describes K lithofacies or lithologic units by using one indicator function per facies. Indicator simulation is accomplished by generating a Gaussian random field and then truncating the Gaussian values to achieve the categorical simulation. Specifically, suppose $Y(s)$ is a standard Gaussian random variable. The associated indicator transform is

$$I(s; y_i) = \begin{cases} 1 & \text{if } Y(s) \in (y_{i-1}, y_i] \\ 0 & \text{otherwise} \end{cases}$$

so that a point s belongs to category i if $Y(s) \in (y_{i-1}, y_i]$. The thresholds, y_i, are determined according to the proportion of values that fall into each category. Let Φ be the standard Gaussian distribution function, and let p_i be the proportion of values in category i. Then

$$p_1 \equiv P[-\infty < Y(s) \leq y_1] \quad \rightarrow y_1 = \Phi^{-1}(p_1).$$

In general, $y_{k-1} = \Phi^{-1}(p_1 + p_2 + \ldots + p_{k-1})$. After the Gaussian thresholds have been determined, it is then necessary to determine the covariance structure of the Gaussian random variables. This structure depends on the structure of the indicator covariances which may be computed and modeled from the data. The relationship between the two covariance functions can be specified using Hermite polynomial expansions, in which case (Dowd, 1992)

$$C_I(h) = \sum_{i=1}^{K}\sum_{j=1}^{K} c_i c_j g(y_i) g(y_j) + \sum_{n=1}^{\infty} \frac{H_{n-1}(y_i)H_{n-1}(y_j)}{n!}(C_Y(h))^n,$$

where $H_{n-1}(x)$ are Hermite polynomials, g(y) is the standard Gaussian density, c_i is a unique integer value assigned to each category, and $C_I(h)$, and $C_Y(h)$ are the covariance functions of the indicator variables and the Gaussian variables, respectively. Conditioning data at locations $\{s_\alpha\}$, are replaced by standard Gaussian random variables with covariance function $C_Y(h)$ such that $y_{j-1} \leq y(s_\alpha) < y_j$, where j indexes the category to which s_α belongs. Finally, a usual conditional Gaussian simulation is performed and then back-transformed to obtain the associated indicator values.

There are many ways to determine $C_Y(h)$ from the above equation. To the authors' knowledge, there is no published theory on an optimal solution for $C_Y(h)$ or even an accepted methodolgy for obtaining any satisfactory solution. The approach used in this study was to obtain values of $C_Y(h)$ that satisfy the above equation using a Golden Section Search algorithm (Press, et al., 1986) for all relevant lags h, rather than to use a specific parametric model. Once $C_Y(h)$ has been established, Gaussian random fields were generated via simulated annealing (Kirkpatrick, et al., 1983; Deutsch and Journel 1992a, 1992b) to force each realization to match $C_Y(h)$. The determination of $C_Y(h)$ is the most computationally intensive aspect of this algorithm, and the lack of a straightforward method for conditioning to indicator data is also a limitation. For problems with more than two categories, consideration of indicator cross covariances is required, and additional computations are required.

Clearly, in this type of study, it is impossible to divorce the simulation method or algorithm from the computer software used to implement the algorithm. In constructing the software, there are often various choices for the computational details of algorithm implementation (e.g., the number of turning bands lines, cdf interpolation and extrapolation methods, search strategies) that are specified by the user. Every attempt was made to provide the most complete input to each simulation code. The goal of this study is not to discredit any of the approaches or to determine the best algorithm, since no algorithm can be best for all applications. Instead, we hope to obtain more information on how the algorithms work in simulation applications like those illustrated in Figure 1 and to indicate new research directions for investigating the properties of the algorithms and developing new algorithms.

B. *Data conditioning*

For the conditional realizations, N=100 sets of r=100 data values were randomly selected from each exhaustive data set. Each realization was then conditioned on one of the N sets of 100 values. Although in practice all realizations are conditioned to the same set of data values, in a study such as this, care must be taken not to introduce bias into the experiment due to the particular data set chosen. Since it is impossible to determine a "good" data set *apriori*, multiple sets of conditioning data were used.

C. *Exhaustive Data Sets*

Seven exhaustive data sets were used in this comparative study. Three are real exhaustive data sets taken from the authors' project work and from the literature. The advantage of these data sets is that they reflect real spatial phenomema and the statistical and geostatistical properties of these data sets reflect what is actually encountered in practice. However, results based on real exhaustive data sets are limited to the particular properties exhibited by the data. In order to broaden the scope of inference of a study such as the one described here, it is necessary to consider exhaustive data sets with a variety of statistical and geostatistical properties. Thus, the remaining four exhaustive data sets were generated synthetically so that their statistical and geostatistical features could be controlled and varied. A brief description of each exhaustive data set is provided below and corresponding gray-scale maps are shown in Figure 2.

1. Multivariate Gaussian Exhaustive Data Set. This data set was obtained by generating 1600 independent standard Gaussian random variables on a 40 x 40 grid of unit spacing. The covariance matrix was derived from a prespecified exponential semivariogram model with a zero nugget, unit sill, and range 7. A Cholesky decomposition algorithm from the IML procedure in SAS (Statistical Analysis System Version 6.01, 1992) was used to induce the desired covariance structure. The ensemble average was then standardized to mean zero and unit variance to produce the exhaustive data set.

2. Uniform Exhaustive. The uniform exhaustive data set consists of 1600 values on a 40 x 40 regular grid with unit spacing and was obtained by generating a set of uniform random variables over the grid and then applying simulated annealing to more closely match the desired spatial covariance. The simulated annealing algorithm given in GSLIB (Deutsch and Journel, 1992a) was used to force the image to match (to the extent permitted by this algorithm) two summary functions: 1) the histogram of a uniform probability distribution on $[-\sqrt{3},\sqrt{3}]$, and 2) an isotropic exponential semivariogram with zero nugget, unit sill, and range 10.

3. Indicator-Covariance Specified Exhaustive. This exhaustive data set was obtained by modifying the simulated annealing algorithm given in Deutsch and Journel (1992a) to anneal an initial image to specified indicator semivariograms. The resulting exhaustive data set was forced to have mean zero, unit variance, and indicator semivariograms specified as
Cut-off -2: isotropic sph ($c_o = 0.00, c_s = 0.16, a = 7.00$),
Cut-off -1: isotropic sph ($c_o = 0.00, c_s = 0.25, a = 5.00$),
Cut-off 0: anisotropic sph ($c_o = 0.00, c_s = 1.00, ax = 7.00, ay = 3.50$),
Cut-off 1: isotropic sph ($c_o = 0.00, c_s = 1.25, a = 10.0$),
Cut-off 2: anisotropic sph ($c_o = 0.00, c_s = 1.50, ax = 30.00, ay = 15.00$).
The notation " sph" refers to a spherical semivariogram model with nugget c_o, sill c_s, and range a. The notation "anisotropic sph" specifies a model of geometric anisotropy, where the parametric form is that of a spherical semivariogram model, but with a range of ax in the N 90° E direction and a range of ay in the N 0° E direction.

4. Reduced Walker Lake Exhaustive. To construct the exhaustive data set called the "Reduced Walker Lake" exhaustive data set, an initial exhaustive variable, U', was selected from a portion of the U variable of the original Walker Lake Data Set (Isaaks and Srivastava, 1989). The reduction was done in order to facilitate comparisions with results obtained for other continuous-variable exhaustive data sets that consist of 1600 data points on a regular 40 x 40 grid with unit spacing. Following the approach of Desbarats and Srivastava (1991), the variable U' was then transformed to create a variable T which is similar to the log-normally distributed transmissivity parameter common in hydrogeologic applications.

5. Berea Sandstone Exhaustive. The Berea sandstone data set consists of 1600 air permeameter measurements (in millidarcies) taken from a 2 x 2 foot vertical slab of Berea sandstone (Giordano, et al., 1985). The locations of these measurements are equally spaced over a 40 x 40 grid. The Berea sandstone has been used in many comparative geostatistical studies such as Journel and Alabert (1989), Deutsch and Journel (1992b), and Rossi and Posa (1992). Figure 2e shows the the banding in the N 123°E direction that is characteristic of this exhaustive data set.

6. GCD Exhaustive. The GCD exhaustive data set presented in this manuscript is a coarser version of the one discussed in Gotway, et al. (1993) obtained by digitizing a photo-mosaic map of a trench wall at the Greater Confinement Disposal Facility in Southern Nevada. Each node on the resulting grid is associated with one of two hydrogeologic units and is assigned a value of 1 (for the black unit) or 0 (for the white unit). For the purposes of this paper, the exact nature of each unit is not important.

7. Boolean Exhaustive. This exhaustive data set was obtained by Boolean simulation of half-ellipses. Ellipses were horizontally oriented with anisotropy ratios of 2 or 3 and minor axes lengths varying from 1-5 units. The target coverage percentage was specified at 20 percent. Each node on the 80 x 23 grid was given a value of 1 if it was contained in at least one of the ellipses and the value 0 if it was not contained in any ellipse.

Summary functions necessary for input into the various simulation algorithms were obtained directly from the exhaustive data sets. For the Gaussian-based simulation algorithms, the continuous, non-Gaussian, exhaustive data sets were transformed to normality prior to simulation. Semivariograms (N 0°E, N 45°E, N 90°E, and N 135°E and omnidirectional), were then determined from the transformed

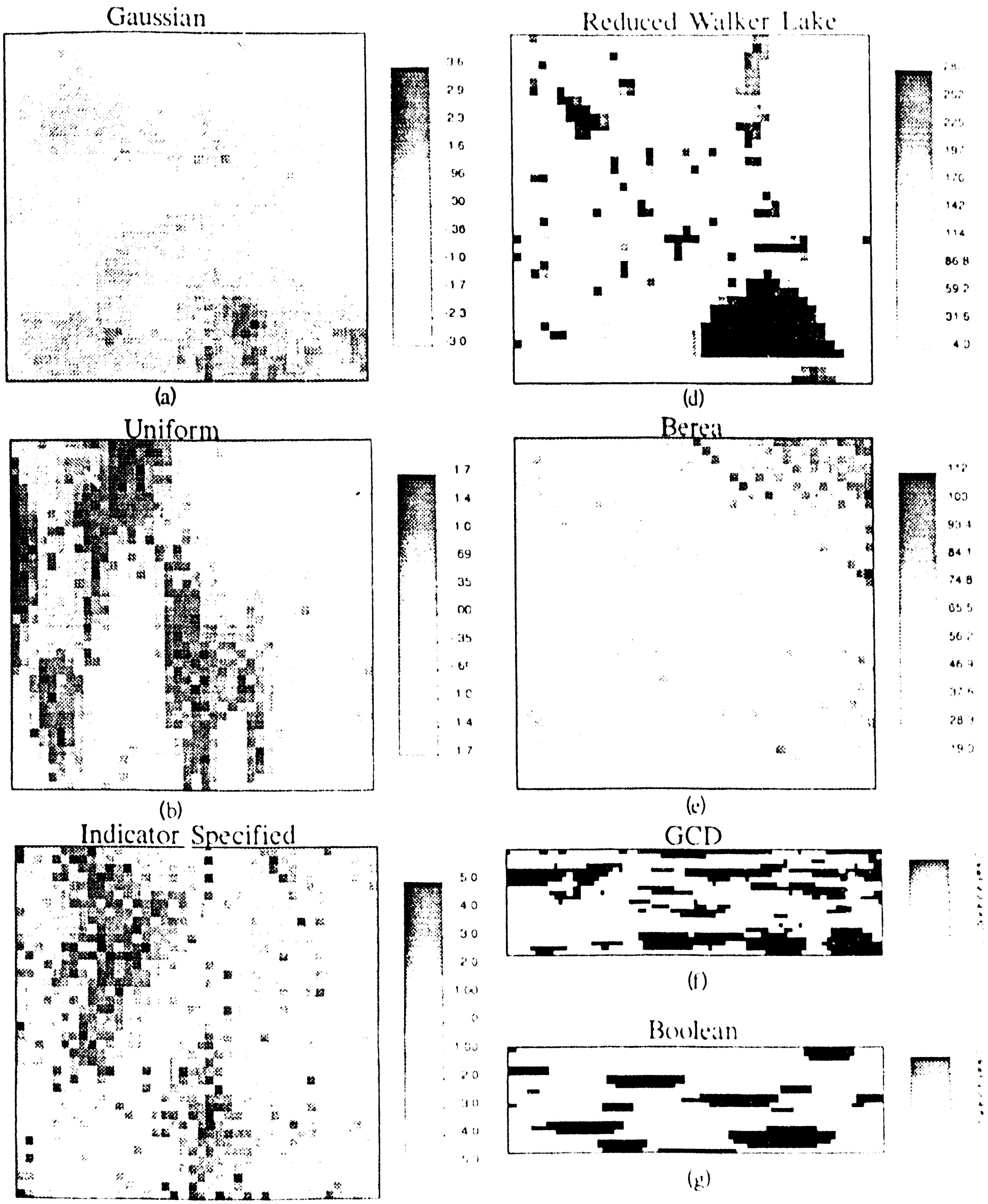


Figure 2. Exhaustive Data Sets.

data and used to generate the realizations. For the real data sets where additional information indicated the principal directions of anisotropy, semivariograms corresponding to these directions were used. After simulation, the generated values were back-transformed, if necessary, based on the original exhaustive distribution. Indicator semivariograms were always obtained from the original exhaustive data set and at least five indicator thresholds were used for each exhaustive data set. In general, a collection of semivariogram models was used in modeling all of the indicator semivariograms derived from the continuous-variable exhaustive data sets. The exception concerns the Berea sandstone exhaustive data set where, following Deutsch and Journel (1992b), the median-indicator approximation (Journel, 1983) was used.

All semivariograms were modeled using a combination of weighted-least-squares-regression fitting (Cressie, 1985) and visual fitting techniques. Anisotropic models and nested structures were used where appropriate. Additional constraints were placed on the choice of indicator semivariogram models in order to repect their theoretical properties.

D. *Transfer functions*

Many important applications of stochastic simulation are flow related. Measures of groundwater travel time, contaminant breakthrough time, and bed thickness all rely on some quantitative notion of connectivity that reflects the degree to which regions of high or low values are related. Many of the transfer functions used here were selected to provide simple indicators of connectivity. They are used in place of actual groundwater flow or transport codes since the complexities involved in boundary condition determination and assumptions, potential errors introduced by necessary grid-discretization, and model calibration issues would only detract from the ability to detect differences in statistical properties among the simulation algorithms.

There are other important transfer functions that are not necessarily flow related. For example, the ability to accurately predict the proportion of values above a specified threshold is important in mining and environmental restoration applications. In these and many other applications, incorporation of major geologic features may also be a primary concern.

The transfer functions described below were selected to encompass a variety of applications like those described above. For the categorical problems, the transfer functions are called discrete path, cluster, and discrete average cost. These functions operate on binary data sets. For the continuous-variable problems, the transfer functions are called minimum cost path, geometric mean average and range, and threshold proportion. The transfer functions are described in the following paragraphs.

Discrete Path. At each x-node along the upper boundary of the data set, a particle is released. The particle can move downward or diagonally downward, but if no path is available it can move only horizontally. Barriers to movement are considered to be the higher coded (black) materials in both problems considered here. The initial direction of horizontal movement is left to right. The direction is changed whenever a barrier is encountered during an attempted horizontal move. The output from the transfer function is the number of particles reaching the lower boundary.

Cluster. This algorithm counts the number of clusters of barrier material. The barrier materials are considered to be part of the same cluster if they are connected diagonally, horizontally, or vertically.

Discrete Average Cost. Particles are released from each node on the left boundary and travel horizontally through the region with unit penalty for each move within a white region. Particles can move right or diagonally right to avoid the black regions where movement costs are five times as great. Output is the average cost.

Minimum-Cost Path. In this algorithm, movement costs are based on the reciprocals of the data values. The minimum cost path from the upper boundary to the lower boundary is computed using a dynamic programming algorithm allowing downward and diagonally-downward movement. The output is the minimum cost.

Geometric Mean. The geometric mean is computed for each interior grid node as the product of the 25 closest nodes. For data sets with negative measurements, a constant was added to assure all values were positive. A second fixed constant was used to control the magnitude of the results. Two transfer functions were considered: the difference between the maximum and minimum of all the geometric means; and the average of all geometric means.

Threshold Proportion. This transfer function is the proportion of values that are greater than the 90th percentile of the exhaustive data set.

The transfer functions are applied first to the exhaustive data sets to obtain the "true" value to be used as a basis for comparison, and then to each realization. Comparisons between these values are then used in Section III to draw inferences concerning the simulation techniqies.

RESULTS

The results of this study reveal a number of differences between the simulation algorithms. The differences can be seen in 1) the accuracy of the uncertainty distributions, 2) the characteristics of the realizations input to the transfer functions, and 3) the shapes of the system response distributions. Results in each of these areas are summarized and discussed below.

To help assess the methods, each system response uncertainty distribution was compared to the true value computed from the exhaustive data set. Table 1 gives a summary of the overall results. Columns one and two provide the proportion of system response distributions that 1) contained the true value within the range of response values, and 2) contained the true value within the 90th and 10th percentiles. The last three columns of Table 1 provide additional measures pertaining to the bias, precision, and accuracy of the response uncertainty distributions produced by each algorithm. The bias measure (column 3) is the absolute difference between the median of the uncertainty distribution and the true value, divided by the true value, then averaged over all exhaustive-data set/transfer-function combinations. Precision (column 4) is measured as the difference between the 90th and 10th percentiles of each uncertainty distribution divided by the corresponding percentile difference for the uncertainty distributions obtained using the unconditional LU decomposition simulation algorithm or the unconditional categorical sequential indicator algorithm (as arbitrary references), averaged over all exhaustive data-set/transfer-function combinations. Simulation algorithms that consistently produce uncertainty distributions that are more precise than those obtained using unconditional LU decomposition (or categorical sequential indicator) simulation will have precision values less than 1.00. Column 5 combines the measures of bias and precision into a measure of accuracy. Accuracy is measured as the weighted average of the absolute differences between system response value and true value, where the weights are the reciprocals of the across-simulation-method average standard deviation of the uncertainty distributions for each exhaustive data set/transfer function scenario. The weights are used to account for differences in the magnitude of the values in the different scenarios. Lower numbers for the accuracy measure are indicative of uncertainty distributions whose values are consistently close to those computed from the exhaustive data sets.

SIMULATION ALGORITHM	PROPORTION OF DISTRIBUTIONS CONTAINING THE TRUE VALUE OF (X_{min}, X_{max})		PROPORTION OF DISTRIBUTIONS CONTAINING THE TRUE VALUE OF (X_{10}, X_{90})		BIAS		PRECISION		ACCURACY	
	Unc	Cond	Unc	Cond	Unc	Cond	Unc	Cond	Unc	Cond
TURNING BANDS	18/20	19/20	17/20	18/20	0.35	0.18	0.71	.45	1.14	1.52
LU DECOMPOSITION	20/20	19/20	19/20	17/20	0.30	0.17	1.00	0.45	1.25	1.47
SEQUENTIAL GAUSSIAN	20/20	18/20	18/20	17/20	0.31	0.17	0.71	0.37	1.04	1.52
SEQUENTIAL INDICATOR (continuous)	17/20	17/20	13/20	14/20	0.51	0.32	0.89	0.37	1.47	1.75
SEQUENTIAL INDICATOR (categorical)	5/6	5/6	5/6	5/6	0.35	0.28	1.03	.91	1.33	1.32
TRUNCATED GAUSSIAN	4/6	4/6	3/6	3/6	0.43	0.42	.90	.93	1.64	1.72

Table 1. Summary of Uncertainty Distributions Combined Across All Exhaustive Data Set/Transfer Function Scenarios. The bias, precision, and accuracy measures are those discussed in the text.

The first column of Table 1 shows that most of the uncertainty distributions produced by the continuous-variable simulation algorithms did contain the true value. The exceptions are the uncertainty distributions based on transfer functions used with the Reduced Walker Lake exhaustive data set. The second column of Table 1 gives the proportion of uncertainty distributions containing the true value within the 10th and 90th percentiles. On average, this number should be around 80%, or 16/20, for the continuous-variable simulation algorithms. Thus, from the table we can see that the probability content assigned to specified intervals appears to be accurate for uncertainty distributions based on conditional Gaussian-based algorithms, but may be too low for uncertainty distributions produced using the sequential indicator algorithm, or too high if unconditional LU decomposition or sequential Gaussian methods are used for problems of the type presented here.

The last three columns of Table 1 show that the uncertainty distributions obtained using the unconditional Gaussian-based simulation algorithms have similar values for the bias measure, but that those obtained using the unconditional turning bands and sequential Gaussian algorithms are more precise than those based on the unconditional LU decomposition algorithm. Conditioning the simulation reduces the differences in the uncertainty distributions produced by the various methods, but, on average, uncertainty distributions produced by the Gaussian-based simulation algorithms are more accurate than those produced using sequential indicator simulation. These results differ from those presented by Journel and Deutsch (1993) in which response uncertainty distributions based on unconditional realizations generated by the sequential Gaussian simulation algorithm were compared to those obtained using sequential indicator simulation. Based on one uncertainty distribution that did not contain the true value, Journel and Deutsch (1993) conclude that Gaussian random field models may produce uncertainty distributions and probability intervals that are too narrow. Clearly, the results of our study do not support their contention. Although the Gaussian realizations produced an inaccurate response uncertainty distribution for the exhaustive-data-set/transfer function scenario considered in Journel and Deutsch (1993), this may be due to the use of the sequential Gaussian simulation algorithm, and conclusions based on this algorithm cannot be extended to Gaussian-based methods in general.

Most of the response uncertainty distributions obtained using the categorical simulation algorithms were also accurate. The primary exceptions are the results for the Boolean exhaustive data set and the discrete path transfer function, where none of the uncertainty distributions contain the true value. Of course, in practical applications, transfer functions are not computed on lithologic simulations alone. Usually, the lithologic simulations are the first stage of a two-stage simulation procedure designed to incorporate the large-scale geologic features of a region as well as capture the small-scale features of an associated parameter of interest. Both the discrete path and cluster transfer functions reflect the ability of the simulation methods to accurately portray lithologic features. This suggests that perhaps other simulation methods that utilize different summary functions which characterize shape and connectedness (such as Boolean and random set algorithms) might be more appropriate for these transfer functions. For transfer functions such as the discrete average cost function, both categorical simulation algorithms will likely produce accurate response uncertainty distributions. Although uncertainty distributions produced by the truncated Gaussian approach show about the same variability as those produced by categorical sequential indicator simulation, the bias and accuracy measures indicate that the truncated Gaussian approach has a higher average bias. Realizations generated using the truncated Gaussian approach tend to have a larger number of isolated points than those obtained using sequential indicator simulation, and this adversely affects the uncertainty distributions produced by the truncated Gaussian algorithm.

For a particular exhaustive data set, the realizations (taken as an ensemble) produced by the LU decomposition and turning bands algorithms are very similar. The exceptions are the realizations generated using the turning bands algorithim with the Berea exhaustive data set. This particular data set exhibits strong zonal anisotropy that is not parallel to the coordinate axes, and the simplifications made to semivariogram models for use with the turning bands software hindered its ability to portray the banding characteristic of the Berea exhaustive data set. It is surprising that even with several such simplifications, the turning bands algorithm gave fairly accurate results. None of the realizations produced by the turning bands method showed artifacts due to the lines necessary for the one-dimensional simulations.

There are notable visual differences between the realizations produced by the LU decomposition and turning bands methods and those generated using the sequential simulation algorithms. The realizations provided by the sequential methods tend to exhibit more clustering of similar values than realizations produced by the other methods. Figure 3 presents two typical realizations of the Reduced Walker Lake data set; one generated using LU Decomposition, and the other generated using sequential indicator simulation. The clustering is probably an artifact of the kriging on which the sequential approaches are based and is discussed in Hansen (1992) but was not anticipated here since simple kriging was used throughout. Clustering of similar values is particularly pronounced in realizations produced by sequential indicator simulation, and may be due in part to the nugget effect induced in the sequential indicator realizations caused by transformation from the class partitions to the real values. Deutsch and Journel (1992) acknowledge the extra nugget component and suggest artificially reducing the nugget effect of the indicator semivariograms prior to simulation. This suggestion may alleviate part of the problem, but would not affect the clustering problem because it would reduce within class variability and not the extra variability resulting from the discrete partitioning. Figure 3 also illustrates another important difference between the realizations produced by sequential indicator simulation and those obtained by Gausian-based methods. The use of indicator semivariograms enables realizations of the Reduced Walker Lake data set generated using sequential indicator simulation to capture the ridge of high values characteristic of this data set. Although the Gaussian-based methods capture the large area of high values, the entire ridge is much less clearly defined in these realizations, with no immediate "drop-off" as seen in the exhaustive data set.

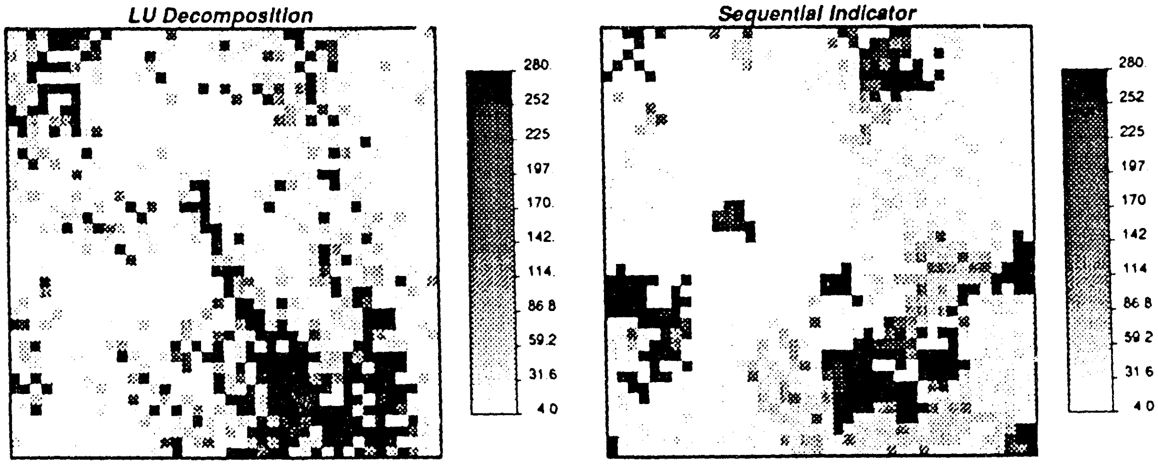


Figure 3. Conditional Realizations of the Reduced Walker Lake Exhaustive Data Set.

The categorical realizations do not usually reflect the shape or position of key features of the categorical exhaustive data sets, even after data-conditioning. Some typical realizations are shown in Figure 4. As mentioned earlier, the conditional realizations obtained using the truncated Gaussian approach have more isolated points than those produced using the categorical sequential indicator approach. This could be due to the details of the implementation of the truncated Gaussian algorithm, such as the choice of an annealing step used in generating the Gaussian realizations or the use of a golden section search algorithm in determining the Gaussian covariance structure, and may not be artifacts of the truncated Gaussian approach in general. At present, there does not appear to be any theory or guidance in the literature as to choices for specific details essential for software implementation of this algorithm or the effect of these choices on the generated realizations.

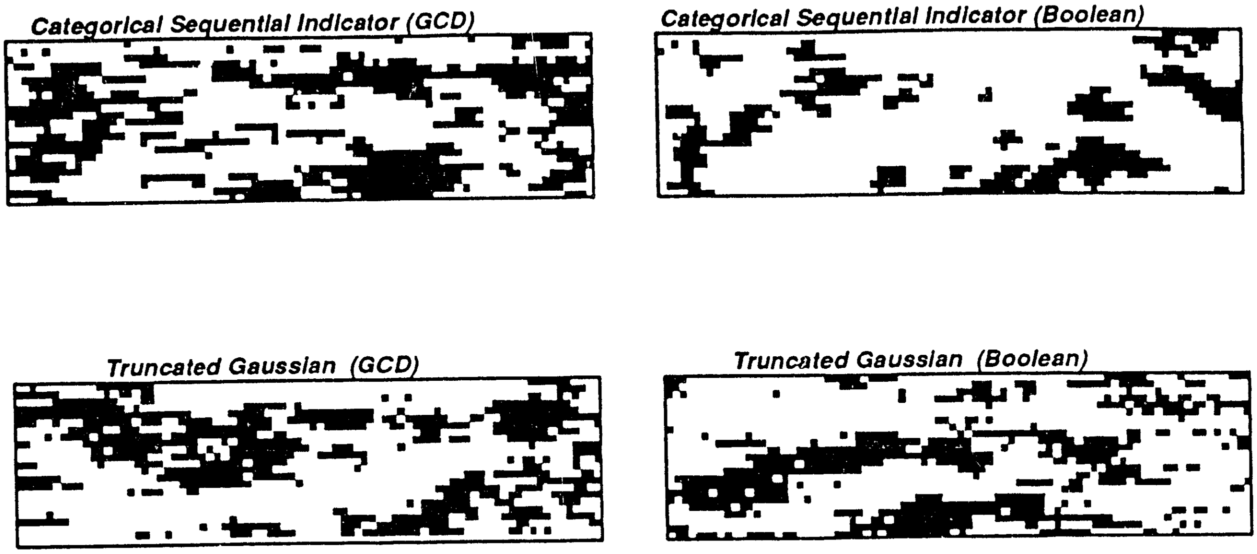

Categorical Sequential Indicator (GCD)
Categorical Sequential Indicator (Boolean)
Truncated Gaussian (GCD)
Truncated Gaussian (Boolean)

Figure 4. Conditional Realizations of the GCD and Boolean Exhaustive Data Sets.

Different components of the simulation methods appear to affect the shape of the uncertainty distrbution of system response. An example is shown in Figure 5, which gives the results of the LU decomposition and sequential Gaussian simulation algorithms for the indicator-covariance specified exhaustive data set using the geometric mean range transfer function. In this situation, the exhaustive data set, transfer function, and summary functions are exactly the same. The sole difference lies with the particular algorithm used. Another example is given in Figure 6, which shows the results of the turning bands and sequential indicator simulation algorithms for the uniform exhaustive data set and the geometric mean average transfer function. It is not clear if the differences between these distributions are due to method of generation (sequential vs. non-sequential), summary function specification (semivariogram vs. indicator semivariograms), or algorithm implementation details. A final example is given in Figure 7, which shows response distributions for the minimum cost transfer function and the Gaussian exhaustive data set obtained using the sequential Gaussian and turning bands algorithms. Here, the spread of the two uncertainty distributions is quite different. These examples, and many others not presented here, show that the uncertainty attributed to a system response prediction can depend on the particular simulation algorithm used and may differ substantially even among methods that are based on the same assumptions and use the same available information.

In general, the effect of data conditioning on the response uncertainty distributions is clear: Conditioning decreases the variability in the system response distribution. In particular, the tails of the distribution are pulled toward the center, the median is generally moved closer to the true value, and most outliers are

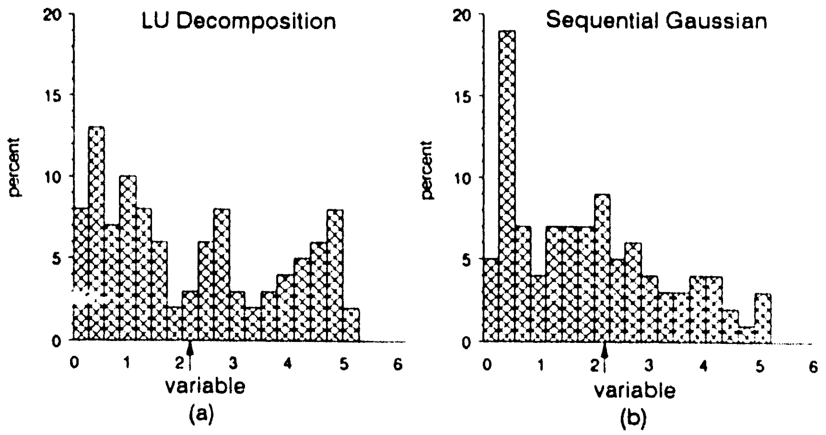


Figure 5. Uncertainty Distributions for the Geometric Mean Range Transfer Function Based on Unconditional Realizations of the Indicator-Covariance Specified Exhaustive Data Set. The arrow denotes the true transfer function value.

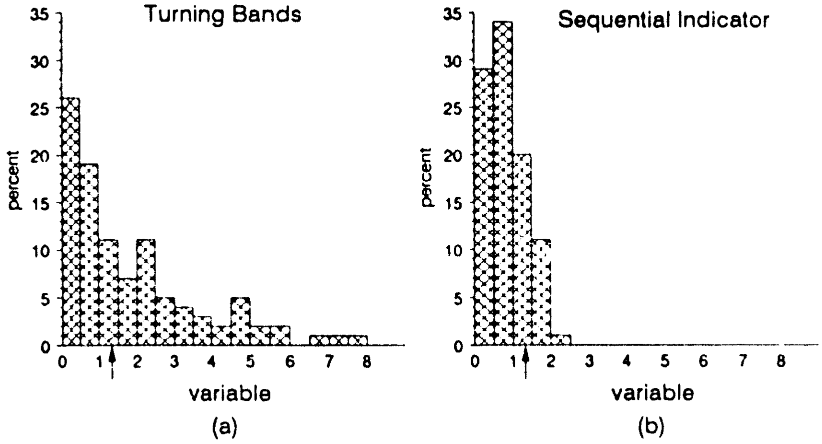


Figure 6. Uncertainty Distributions for the Geometric Mean Average Transfer Function Obtained from Unconditional Realizations of the Uniform Exhaustive Data Set. The arrow denotes the true transfer function value.

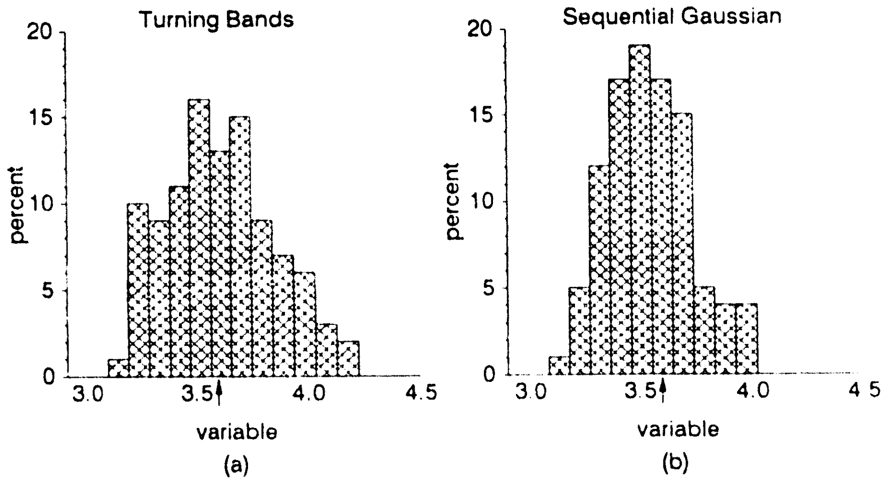


Figure 7. Uncertainty Distributions for the Minimum Cost Transfer Function Obtained from Unconditional Realizations of the Gaussian Exhaustive Data Set. The arrow denotes the true transfer function value

eliminated (see Figure 8 for a typical example). Conditioning the realizations will also reduce algoi.thmic effects but not eliminate them. An example is given in Figure 9, which shows uncertainty distributions produced using conditional LU decomposition and sequential indicator simulation algoiithms for the indicator-covariance specified exhaustive data set and the geometric mean range transfer function. Here, the two uncertainty distributions are very different even after conditioning on 100 data points. In a few other situations, such as that depicted in Figure 10, it would appear that the increase in precision due to data conditioning was obtained at the expense of bias, since the conditioned distribution no longer contains the true response. This example (and the t.aree other cases observed in this study) support the observations in Hansen (1992), where the tendency was termed "over-conditioning."

All of the simulation algorithms used in this study had the benefit of complete exhaustive information wherever possible. The exceptions were the tail extrapolation choices in the sequential indicator algorithm necessary to extrapolate the conditional distribution estimated at the lowest (highest) threshold to a specified minimum (maximum). As mentioned earlier, the nature of the response distributions produced by the sequential simulation methods can be greatly affected by some parameters that are difficult to determine from the data. Tail extrapolation choices coupled with the choice of specified data minimum and maximum are two such important parameters. In this study, linear extrapolation to the exhaustive data minimum and maximum was used. Results obtained using linear interpolation/extrapolation between exhaustive quantiles (information provided to the other techniques when transforming and back-transforming) produced uncertainty ranges with tighter bounds. Consequently, results based on this interpolation/extrapolation choice were less accurate than those presented here. In addition, a third set of realizations was generated using linear extrapolation to specified minimum and maximum data values that were 10% beyond those given by the exhaustive data. This was done because, in practical applications, exhaustive information will not be available, and a researcher using simulation may decide to let the simulated values fall slightly outside of the data range in order to create the "tails" of the distribution. Based on this third set of realizations, the measures in Table 1 obtained for the Gaussian-based methods did not change much, but some of the measures for sequential indicator were very different. Specifically, most uncertainty distributions had much higher ranges (precision measures were 3.20 (unc.) and 1.43 (cond.)), but were no more accurate than the previous distributions (bias measures were 0.56 (unc.) and 0.30 (cond.)). Clearly, the choice of such parameters is very important, and careful evaluation is required before using realizations generated by any simulation algorithm to assess the uncertainty of a system response.

SUMMARY AND CONCLUSIONS

Several very broad issues are illustrated by the results of this study. The first is the effect of particular simulation algorithms on the resulting uncertainty distribution. Even for simulation algorithms that use the same summary functions, the shape and spread of the resulting uncertainty distributions based on a particular transfer function can be quite different. This lack of consistency implies that different algorithms may produce very different predictions of system response and poses additional problems for applications whose major interest lies in quantifying the uncertainty associated with particular events.

Second, there are some transfer functions (like the discrete path transfer function) for which producing the distribution of system response based on just two summary functions (like the cdf and semivariogram) may not be appropriate. We

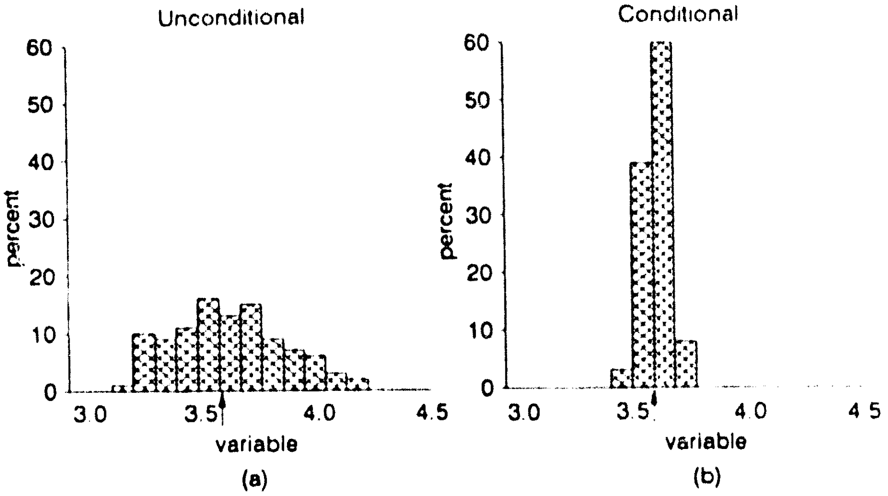


Figure 8. Uncertainty Distributions for the Minimum Cost Transfer Function Obtained from Realizations of the Gaussian Exhaustive Data Set Generated by the Turning Bands Algorithm. The arrow denotes the true transfer function value

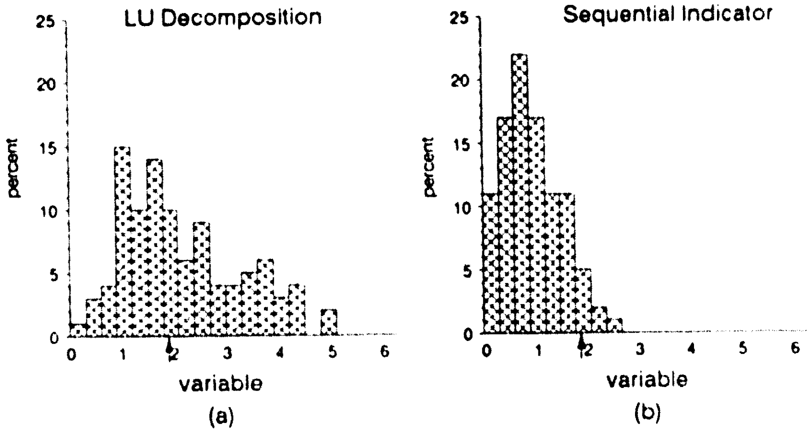


Figure 9. Uncertainty Distributions for the Geometric Mean Range Transfer Function Obtained from Conditional Realizations of the Indicator-Covariance Specified Exhaustive Data Set. The arrow denotes the true transfer function value.

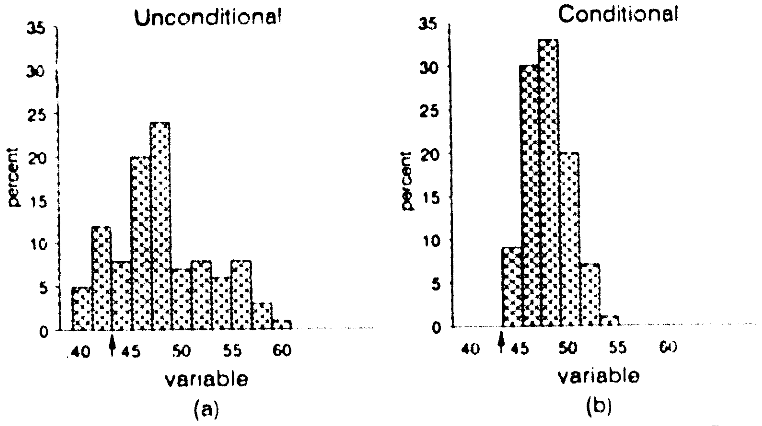


Figure 10. Uncertainty Distributions for the Minimum Cost Transfer Function Obtained from Realizations of the Berea Sandstone Exhaustive Data Set Generated by Sequential Gaussian Simulation The arrow denotes the true transfer function value.

observed that for lithologic simulation in which the shape and connectedness of units are important, simulation algorithms that utilize summary functions that can measure these properties might be more appropriate than those considered in this study.

Third, conditioning on sample data will improve the precision associated with the system response distribution and reduce, but not eliminate, algorithmic effects. A few cases of "over-conditioning" were observed but did not appear to be a chronic problem.

Finally, our results indicate that, overall, Gaussian based simulation models can incorporate the essential spatial features of a spatially-varying parameter. Although Gaussian models maximize spatial disorder, and Journel and Deutsch (1993) maintain that they could yield a response uncertainty space that is too narrow, our results do not support this contention. We have found that the Gaussian-based approaches tend to produce uncertainty distributions that are more accurate than those obtained using the sequential indicator algorithm. For imaging spatial uncertainty in a continuous variable, this study suggests that, for the variety of exhaustive-data-set/transfer-function scenarios considered, Gaussian-based approaches, and the sequential Gaussian simulation algorithm in particular, are flexible and accurate methods for stochastic simulation of random fields.

There are many other interesting studies that could have been done within this project, such as considering the effects of summary function estimation from limited data, conditioning sample size and location, and transfer function approximations. For the practicioner, these are real unknowns. To look at these factors over all of the various simulation algorithms would be unnecessarily tedious and redundant. This study provides additional information on the simulation algorithms that appear to work well in a variety of exhaustive data set/transfer function combinations. Studies that look at the effects of other factors not yet investigated can then be designed around these algorithms. In particular, our results suggest that there is still much work to be done in recovering information lost due to discretization of values necessary for the sequential simulation algorithm and in developing simulation methodolgy for categorical variables. In particular, refinements and improvements to the methods illustrated in this paper so that they might more adequately capture continuity and shape, and the study and evaluation of the properties of other categorical simulation approaches such as Boolean and random set algorithms, nearest neighbor approaches, and further development of fractal simulation methods could provide intersting and useful research directions.

ACKNOWLEDGEMENTS

We are grateful to Tony Zimmerman for his assistance with both the implementation of the turning bands algorithm, and the computation of a groundwater travel *time transfer* function that was not presented in this paper. Comments and suggestions made by Noel Cressie, Bob Easterling, Kathy Hansen, and two reviewers were also very helpful.

REFERENCES

Cressie, N. (1985). "Fitting variogram models by weighted least squares," International Association for Mathematical Geology, 17, 563-586.

Cressie, N. (1991). Statistics for Spatial Data, Wiley, New York.

Davis, M.W. (1987a). "Generating large stochastic simulations via the LU triangular decomposition of the covariance matrix," Mathematical Geology, 19, 91-98.

Davis, M.W. (1987b). "Generating large stochastic simulations the matrix polynomial approximation method," Mathematical Geology, 19, 99-107.

Desbarats, A.J., and Srivastava, R.M. (1991). "Geostatistical characterization of groundwater flow parameters in a simulated aquifer," Water Resources Research, 27, 687-698.

Deutsch, C.V., and Journel, A. G. (1992a). GSLIB: Geostatistical Software Library and User's Guide, Oxford University Press, New York.

Deutsch, C.V., and Journel, A.G. (1992b). "Annealing techniques applied to the integration of geological and engineering data," Stanford Center for Reservoir Forecasting, Stanford, University.

Dowd, P. A. (1992). "A review of recent developments in geostatistics," Computers and Geosciences, 17, 1481-1500.

Galli, A., Guerillot, D., Ravenne, C., and HERESIM Group, (1990). "Combining geology, geostatistics, and multiphase fluid flow for 3D reservoir studies," in D. Guerillot and O. Guillon (eds.), Proceedings of the 2nd European Conference on the Mathematics of Oil Recovery, Editions Technip, Paris, pp. 11-19.

Giordono, R., Salter, S., and Mohanty, K. (1985). "The effects of permeability variations on flow in porous media," SPE paper 14365, 60th SPE Annual Conference, Las Vegas, NV.

Gomez-Hernandez, J.J. and Srivastava, R.M. (1990). "ISIM3D: An ANSI-C three dimensional multiple indicator conditional simulation program," Computers and Geosciences, 16, 395-440.

Gotway, C.A., Conrad, S.H., Zimmerman, D.A., and McCord, J.T. (1993). "Sequential indicator simulation of lithology with application to vadose zone water flow and transport, Water Resources Research, submitted.

Hansen, K.M. (1992). "The use of sequential indicator simulation to characterize geostatistical uncertainty," Technical Report SAND91-0758, Sandia National Laboratories, Albuquerque, NM.

Isaaks, E.H., and Srivastava, R.M. (1989). Applied Geostatistics, Oxford, New York.

Journel, A.G. (1974). "Geostatistics for conditional simulation of ore bodies," Economic Geology, 69, 673-687.

Journel, A.G. (1983). "Nonparametric estimation of spatial distributions," Journal of the International Association for Mathematical Geology, 15, 445-468.

Journel, A.G. (1988). Fundamentals of Geostatistics in Five Lessons, American Geophysical Union, Washington, D.C.

Journel, A.G., and Alabert, F. (1989). "Non-Gaussian data expansion in the earth sciences," Terra Nova, 1, 123-134.

Journel, A.G., and Deutsch, C.V. (1993). "Entropy and spatial disorder," Mathematical Geology, 25, 329-355.

Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P. (1983). "Optimization by simulated annealing," Science, 220, 671-680.

Mantoglou, A. and Wilson, J.L. (1982). "The turning bands method for simulation of random fields using line generation by a spectral method," Water Resources Research, 18, 1379-1384.

Matheron, G., Beucher, H., de Fouquet, C. and Galli, A. (1987). "Conditional simulation of the geometry of fluvio-deltaic reservoirs," SPE paper #16753, 62nd Conference of the Society of Petroleum Engineers, Dallas, TX.

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. (1986). Numerical Recipes, Cambridge University Press, Cambridge.

Quimby, W.F. (1986). "Selected topics in spatial analysis: Nonstationary vector kriging, large scale conditional simulation of three dimensional random fields, and hypothesis testing in a correlated random field," PhD dissertation, Department of Statistics, University of Wyoming, Laramie, WY.

Rossi, M., and Posa, D. (1992). "A Non-parametric bivariate entropy estimator for spatial processes," Mathematical Geology, 24, 539-552.

Statistical Analysis System (1990). SAS User's Guide, SAS Institute, Cary, NC.

Zimmerman, D.A., and Wilson, J.L. (1990). Description of and user's manual for TUBA: a computer code for generating two dimensional random fields via the Turning Bands Method, SEASOFT, Albuquerque, NM.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

END

DATE
FILMED

11/16/93