

An Improved Perturbation Mechanism for Simulated Annealing Simulation¹

C. V. Deutsch² and X. H. Wen³

Simulated annealing (SA) is being increasingly used for the generation of stochastic models of spatial phenomena because of its flexibility to integrate data of diverse types and scales. The major shortcoming of SA is the extensive CPU requirements. We present a perturbation mechanism that significantly improves the CPU speed. Two conventional perturbation mechanisms are to (1) randomly select two locations and swap their attribute values, or (2) visit a randomly selected location and draw a new value from the global histogram. The proposed perturbation mechanism is a modification of option 2: each candidate value is drawn from a local conditional distribution built with a template of kriging weights rather than from the global distribution. This results in accepting more perturbations and in perturbations that improve the variogram reproduction for short scale lags. We document the new method, the increased convergence speed, and the improved variogram reproduction. Implementation details of the method such as the size of the local neighborhood are considered.

KEY WORDS: geostatistics, stochastic simulation, Monte Carlo simulation.

INTRODUCTION

In the SA approach to stochastic simulation, creation of a realization is formulated as an optimization problem to be solved by the numerical optimization technique of simulated annealing. Seminal references for the application of these techniques to spatial problems include Farmer (1992), Geman and Geman (1984), Kirkpatrick, Gelatt, and Vecchi, (1983), and Metropolis (1953). Other papers focused on the geostatistical aspects of this problem include Deutsch and Cockerham (1994) and Datta-Gupta, Lake, and Pope (1995).

The key elements of the application of SA to the creation of stochastic realizations are (1) a method to generate an initial realization, (2) a quantitative objective function, (3) a perturbation mechanism, (4) a fast approach to update the objective function after each perturbation, and (5) a set of rules that deter-

¹Received 20 August 1997; accepted 29 October 1997.

²Department of Civil & Environmental Engineering, University of Alberta, Alberta, Canada. e-mail: cdeutsch@civil.ualberta.ca

³Mobil Oil Company. e-mail: Xian-Huan_Wen@email.mobil.com

mine which perturbations are kept. The focus of this paper is on the perturbation mechanism.

In practice, the initial realization is almost always generated by assigning the value at each cell by drawing randomly from the global distribution.

Local fast updating of the objective function is critical for the practical implementation of simulated annealing-based simulation algorithms. This is possible for most spatial statistics including the variogram, indicator variograms, and multiple-point spatial statistics.

An equally important implementation consideration is the perturbation mechanism. The two most common perturbation mechanisms are (1) swap the values at two randomly chosen locations, and (2) replace the value at a randomly chosen location by a value drawn from the global histogram. Although these approaches permit simulated annealing-based simulation to converge, we show dramatic convergence and CPU-speed improvements by adopting an idea used in techniques such as the Gibbs sampler (see Geman and Geman, 1984), or iterative simulation techniques (see Srivastava, 1994).

The idea is to randomly choose a location to perturb and then draw a new value from a *local* conditional distribution instead of the global distribution. We describe the use of median indicator kriging with nearby cell values to build this local distribution. The CPU and convergence advantages will be documented. We then apply the new method to the Walker Lake dataset (topographic-based data introduced in Isaaks and Srivastava, 1989) to illustrate the practical application of the proposed perturbation mechanism.

THE PROPOSED METHOD

After selecting a grid node to perturb, a local distribution (ccdf) is constructed by median indicator kriging with the values at nearby grid node locations. Median indicator kriging, or the mosaic model, is convenient for two reasons: (1) we do not assume a parametric shape for the conditional distribution, and (2) the cost of building the local distribution is minimized by considering a single indicator variogram. Further, because the configuration of local data can be kept the same everywhere (except near the edges of the grid) kriging weights are calculated only once at the beginning. The perturbation mechanism consists of drawing a new value from the local conditional distribution.

Kriging weights are obtained by using the median indicator variogram and a small local template that excludes the collocated grid node that is being perturbed (Fig. 1, upper left). The kriging weights provide a direct estimate of local proportions of categorical variables. For continuous variables, there is a need to provide a continuous ccdf model between the available quantile values. A straightforward linear interpolation between the global minimum, the available quantiles, and the global maximum is considered.

In summary, the implementation steps for this perturbation mechanism are illustrated (Fig. 1) and described below:

1. Establish kriging weights for a template of points (excluding the location being perturbed) using the median indicator variogram.
2. Sort the data within the template in ascending order, $z_{(1)}, z_{(2)}, z_{(3)}, \dots, z_{(n)}$, with kriging weights: $w_{(1)}, w_{(2)}, w_{(3)}, \dots, w_{(n)}$.

Template of Kriging Weights

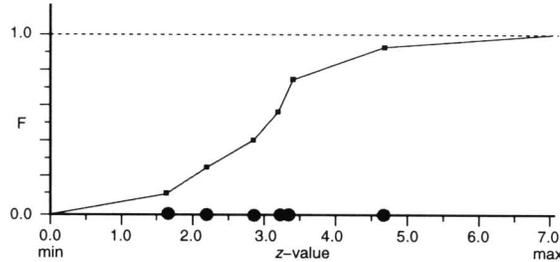
	0.15	0.20
0.15	j'	0.15
0.20	0.15	

$$\gamma(h) = 0.1 + 0.9 \text{ Sph} \left(\sqrt{\left(\frac{hx}{10}\right)^2 + \left(\frac{hy}{5}\right)^2} \right)$$

x is in +45° direction and
y is in -45° direction

A Example z values

	3.28	2.80
4.65	j'	3.33
1.63	2.17	



B Example z values

	0.32	0.17
0.57	j'	1.45
2.81	0.83	

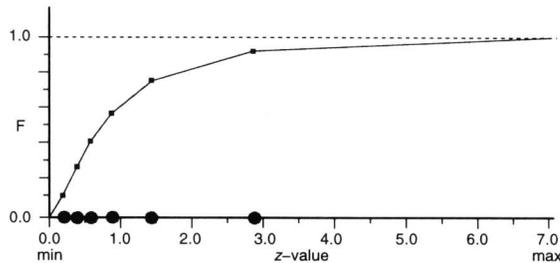


Figure 1. An illustration of how local distributions at location j' are constructed by applying kriging weights to local data. Two examples are shown: A, with values that are in a relatively high area, and B, with values that are in a relatively low area.

3. Calculate the cdf values for each datum:

$$cp_{(i)} = \sum_1^i w_{(i)}, \quad i = 1, \dots, n$$

where the weights $w_{(1)}, w_{(2)}, w_{(3)}, \dots, w_{(n)}$ sum to one, $cp_{(0)}$ at z_{\min} is 0.0, and $cp_{(n+1)}$ at z_{\max} is 1.0.

4. Establish intermediate ccdf values:

$$F(z_{(i)}) = \frac{cp_{(i-1)} + cp_{(i)}}{2}, \quad i = 1, \dots, n$$

5. Linear interpolation allows a complete specification of the relation between $F(z)$ and z . More elaborate ‘tail’ extrapolation methods could be considered for highly skewed data distributions with limited data (c.f. Deutsch and Journel, 1992).
6. Lastly, a new candidate value $z^{\text{new}}(\mathbf{u}_j)$ is drawn for location \mathbf{u}_j from this local ccdf. This candidate value is more likely to be accepted than a value drawn from the global distribution because it is consistent with other cell values in the local neighborhood. Moreover, because this value is drawn from a local conditional distribution built by kriging, the variogram of the perturbed model is more likely to be improved. Note that a candidate value $z^{\text{new}}(\mathbf{u}_j)$ is drawn from the global distribution if the variance of the local distribution is greater than that of the target histogram. An explanation for this exception is given later.

The weights at the top of Figure 1 consider an anisotropic spherical variogram model. Two different sets of local data, hence conditional distributions (A and B) are shown in Figure 1.

IMPLEMENTATION

The revised perturbation mechanism was implemented in a simulated annealing-based simulation program. We start with a small 100 by 100 2-D example (Fig. 2). The target histogram for this initial example is standard normal (mean 0, variance 1). The target variogram is a ‘typical’ isotropic variogram consisting of a 20% nugget effect and a single spherical variogram structure with a range of 20 grid nodes.

Simulation with the new perturbation mechanism reaches the target threshold of 0.005 in 28 CPU seconds on an Silicon Graphics Indigo 2 workstation (approximately twice as fast as a 200 MHz PC), that is, in just over 200,000 perturbations. Convergence with the old perturbation mechanism takes more than twice as long, and we see that the objective function decreases very slowly

after 500,000 perturbations (Fig. 2, bottom). For comparison, sequential indicator simulation (median IK approach in sisim program) takes 18 sec for this 10000 cell problem.

Note the excellent reproduction of the small-distance variogram lags with the new perturbation mechanism (Fig. 2, right center). The old perturbation mechanism, drawing from the global distribution, leads to many unaccepted perturbations once the large-scale features of the realization are established, hence, slow convergence.

Initial Variance Inflation

Another characteristic feature of the new perturbation scheme is that the objective function initially climbs above the starting objective function before decreasing as the temperature is lowered (Fig. 2, bottom). The explanation for this is due to the initial random realization and the approach to get the local distribution: at the start, because of randomly located high and low values, (1) the local distributions can have greater variance than the global distribution, (2) more extreme values are drawn, and (3) these extreme values are kept because of the initial high temperature.

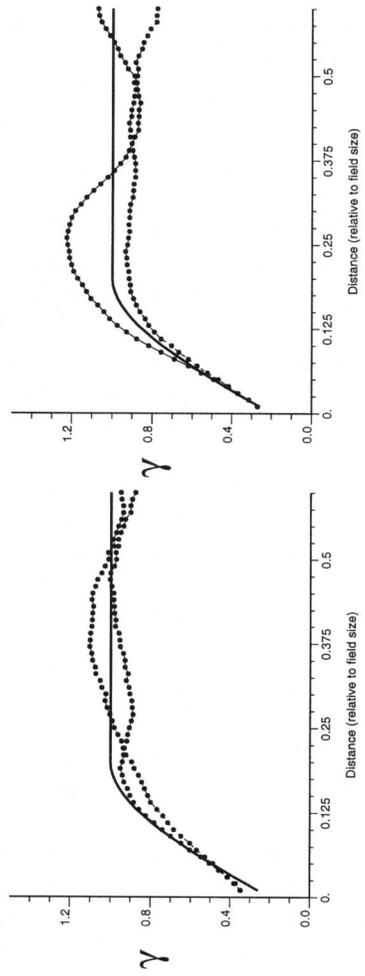
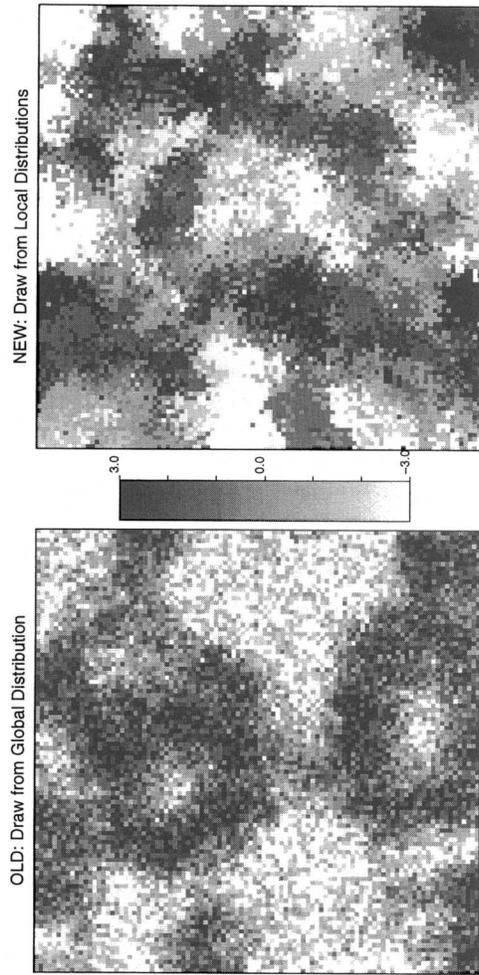
One approach to overcome this initial "variance inflation" is to draw from the target global distribution if the variance of the local distribution is too high, i.e., greater than the variance of the target histogram. The objective function vs. number of perturbations is shown (Fig. 3) for three options: (1) draw from global distribution—solid line, (2) draw from local distribution—short dashed line, and (3) draw from local unless variance too high—long dashed line. The hybrid approach keeps the objective function from climbing and leads to equally low final objective function values.

Size of the Template

The only additional parameter required with this revised perturbation mechanism is the size of the template, n_{tem} ; the median indicator variogram can be identified to the z -variogram, the kriging weights can be precalculated, and the local distributions constructed automatically.

Six realizations are shown (Fig. 4) that were constructed using different template sizes. The top left realization, constructed with a template with two grid nodes aligned in the Y (vertical) direction, shows an artifact vertical banding. The template should have at least one grid node in each principal direction. The remaining realizations appear quite similar indicating that a large template may not be required.

There is a small CPU penalty for using a larger template, that is, it takes some CPU time to assemble the local data into a distribution. A perturbation with $n_{\text{tem}} = 4$ takes 5% more CPU time to consider than simply drawing from



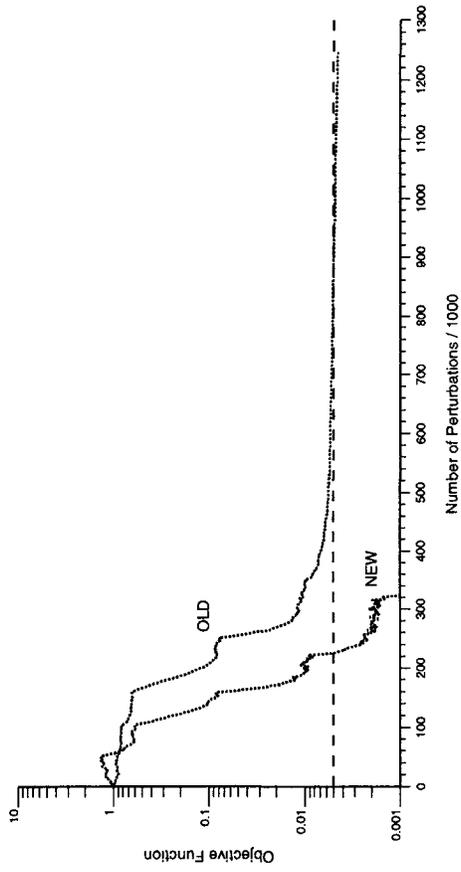


Figure 2. A realization using the old perturbation mechanism (draw from global distribution) and one using the new perturbation mechanism (draw from a local distribution), the corresponding variograms (at objective function of 0.005), and the reduction of the objective function versus the number of perturbations.

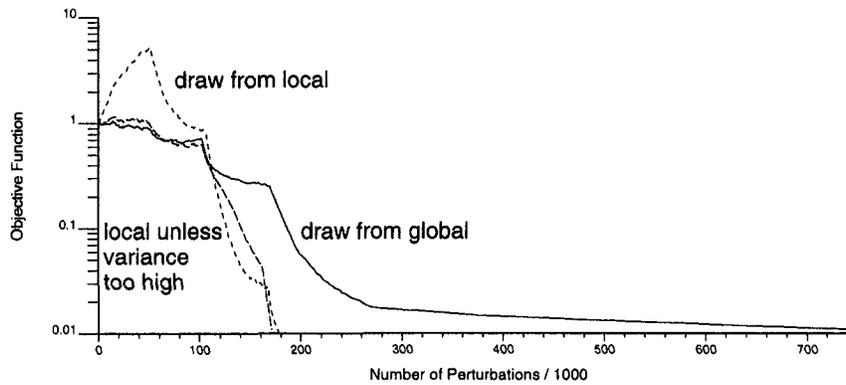


Figure 3. Objective function versus number of perturbations for three options: (1) draw from global distributions—solid line, (2) draw from local distribution—short dashed line, and (3) draw from local unless variance too high—long dashed line.

the global distribution. A perturbation with $n_{\text{tem}} = 40$ takes 45% more CPU time. From this perspective, a smaller template is to be preferred. A larger template, however, considers more spatial information and requires fewer perturbations to achieve convergence. The CPU time vs. template size can be plotted (Fig. 5) to determine a minimum CPU value. The results on Fig. 5 are typical, that is, a significant decrease in CPU time for using the local perturbation mechanism and a gradual increase in CPU time for templates larger than 8–12.

CPU Speed Improvements

The speed improvement depends on the desired objective function value. The comparisons (Fig. 2, bottom; Fig. 3) are representative of many experimental runs. There is a small improvement by using the local updating for large stopping values of the objective function. For small stopping values, there may be dramatic improvements in CPU speed because drawing from the global distribution (OLD method, Fig. 2) is inefficient at reaching low objective function values.

CPU time is shown (Fig. 6) to a minimum objective function of 0.025, which is quite high, on a Silicon Graphics Indigo 2 workstation for models of different size. In all cases, the objective function includes a histogram and a variogram with 99 lags. The solid line is time with the old perturbation mechanism and the dashed line is with the new perturbation mechanism. Note the linear scaling of CPU time with the size of the model. This figure implies that

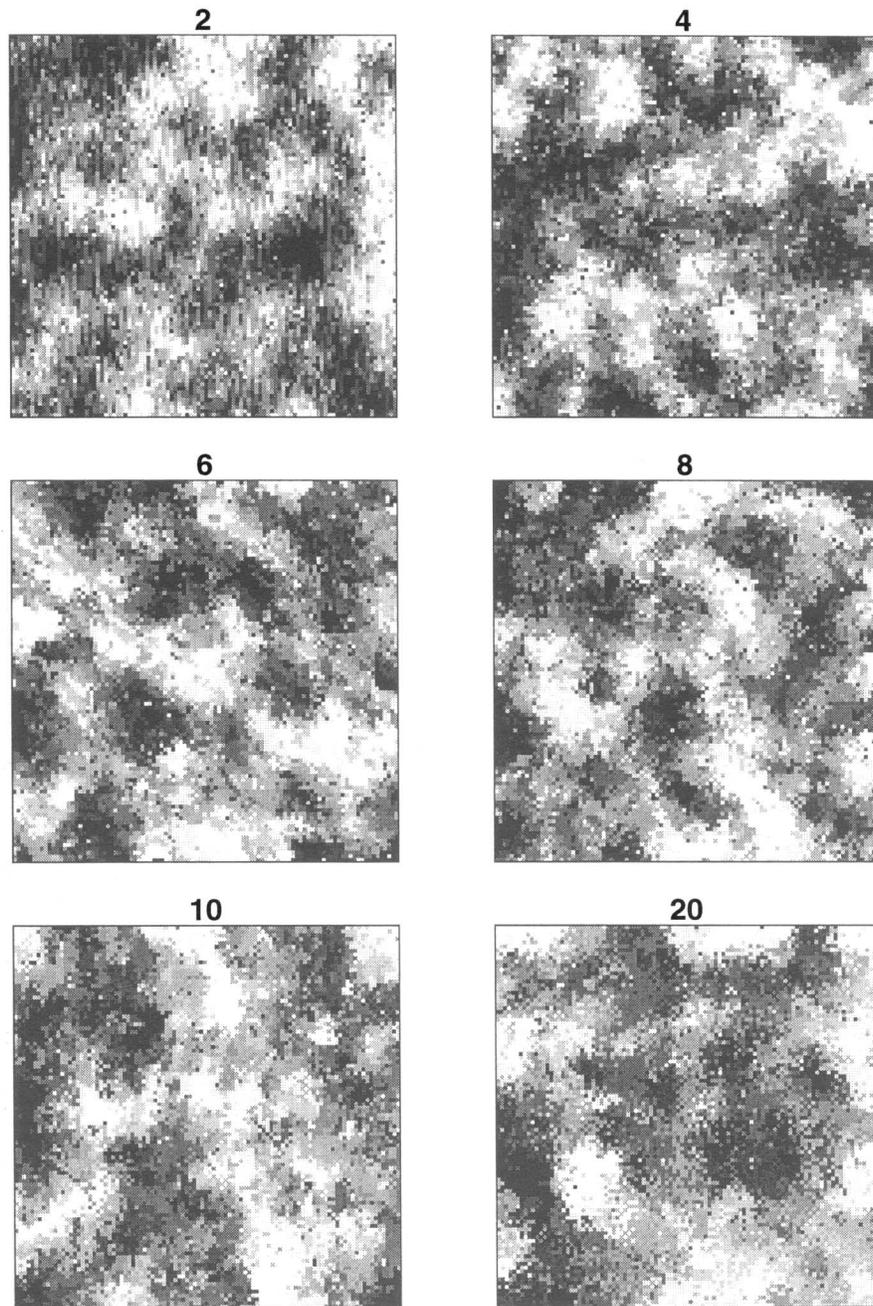


Figure 4. Six realizations using different template sizes: note the artifacts when two data are used (aligned vertically) and the similarity of the remaining realizations.

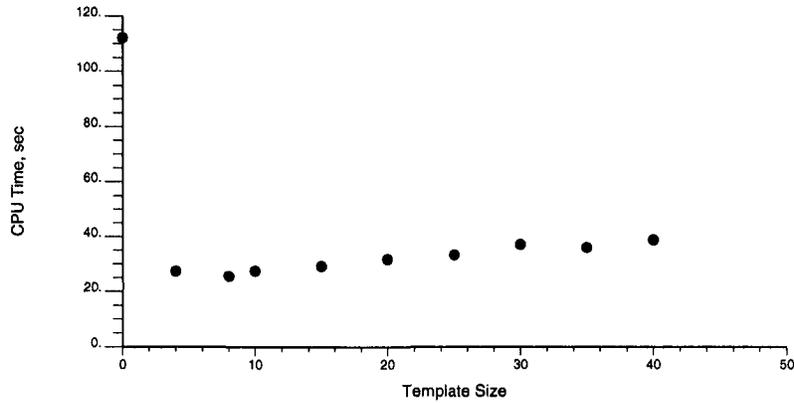


Figure 5. CPU time (Silicon Graphics Indigo 2) vs. template size for 10000 cell example to reach objective function of 0.01. Minimum CPU value of 25.44 is at template size of 8.

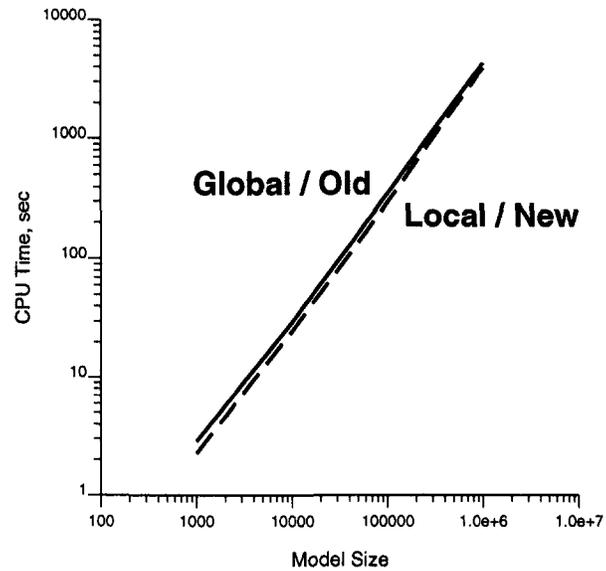


Figure 6. The CPU time to a minimum objective function of 0.025 on a SGI workstation. The solid line is the old perturbation mechanism and the dashed line is the new perturbation mechanism. Note the linear scaling of CPU time with the size of the model. Note also that the final objective function value of 0.025 is quite high. It would take the old perturbation mechanism significantly longer to achieve a lower objective function value of, say, 0.01.

there is marginal improvement with new perturbation mechanism; however, the new procedure allows a lower final objective function that would take significantly longer with the old mechanism.

CASE STUDY

A case study using the southwest quadrant of the Walker Lake dataset (Isaaks and Srivastava, 1989) is shown to further illustrate the practical applicability of the proposed approach. Figure 7 shows the reference grid of 130×150 elevation-related data values, a sample location map of 100 data used for the conditional simulation, a histogram of the 19,500 reference values, and the median indicator variogram and fitted isotropic model. Although the full set of Walker lake data show significant anisotropy in the $-15/75$ directions, this quadrant shows no significant anisotropy. The median indicator variogram is used because of its robustness with respect to the highly skewed data distribution and the appropriateness for the proposed perturbation mechanism.

As a reference, Figure 8 shows a realization of sequential indicator simulation. Median indicator kriging with nine thresholds (based on the deciles) was used. The realization matches the spatial character and long range trends of the reference distribution (Fig. 7, upper left). The minor mismatch between the input variogram and that of the final realization is common in sequential simulation procedures.

Figure 9 shows results simulated annealing simulation using the proposed perturbation mechanism (top) and the conventional perturbation mechanism (bottom—drawing from global distribution). Note how the median indicator variogram of the proposed method is much closer to the input model, particularly at short distances. Also note the more “patchy” appearance of the realization generated using the local mechanism (similar to the sequential simulation result in Fig. 8). The improved result of the new method is also shown by the objective function in Figure 10; the result is much lower than the conventional method.

A quantitative comparison of realizations from different methods could be attempted using measures of spatial entropy (Journel and Deutsch, 1993) or, perhaps, multiple point statistics or the results of flow simulation (Deutsch, 1994).

CONCLUSIONS

The practical application of simulated annealing methods to geostatistical simulation problems depends on careful implementation. The improved perturbation mechanism, borrowed from other iterative simulation methods, leads to better reproduction of small scale variogram lags, faster convergence, and convergence to lower final objective functions than historically attainable.

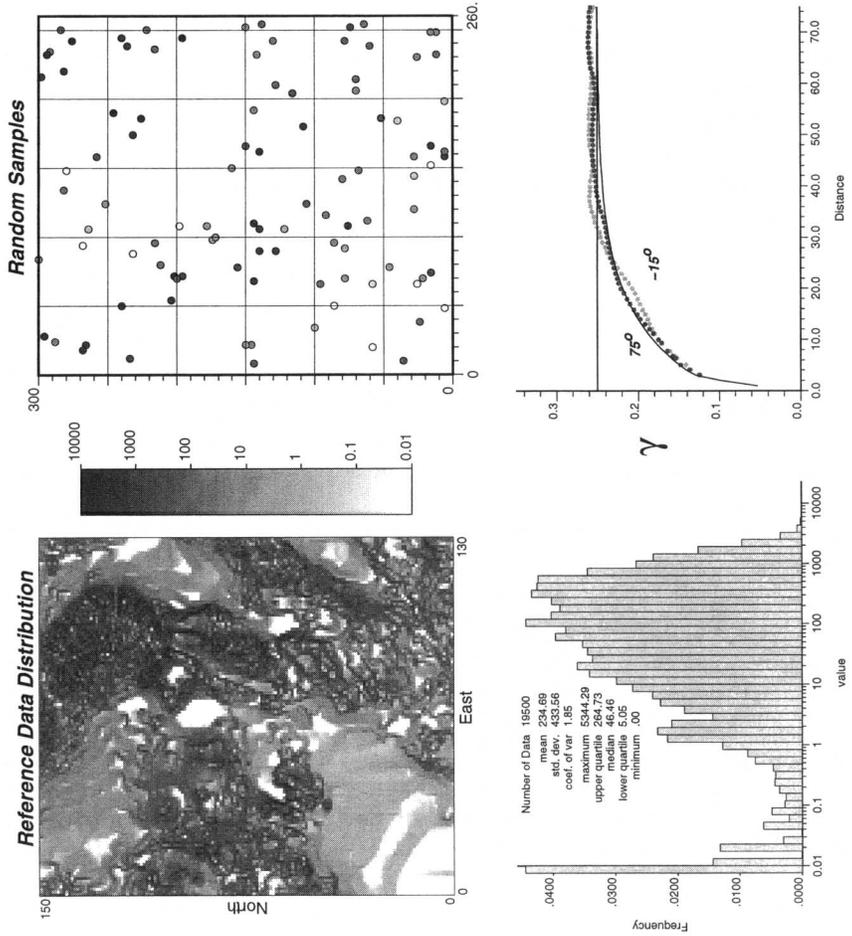


Figure 7. Walker Lake data: (upper left) gray scale image of reference grid of 130 by 150 values, (upper right) locations of 100 randomly chosen locations, (lower left) histogram of 19,500 reference values, and (lower right) directional median indicator semivariograms with isotropic fitted model.

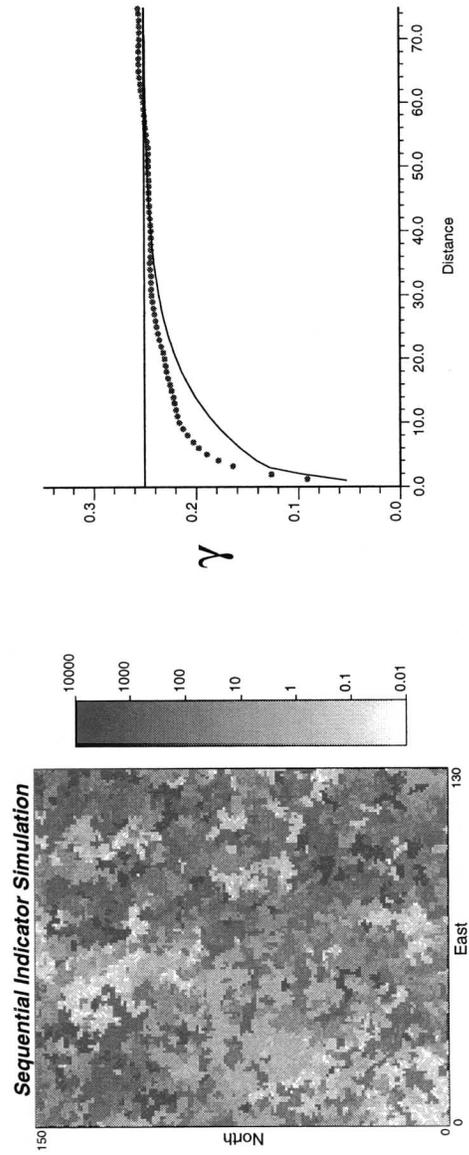


Figure 8. A realization of sequential indicator simulation (median IK approach) and resulting isotropic variogram.

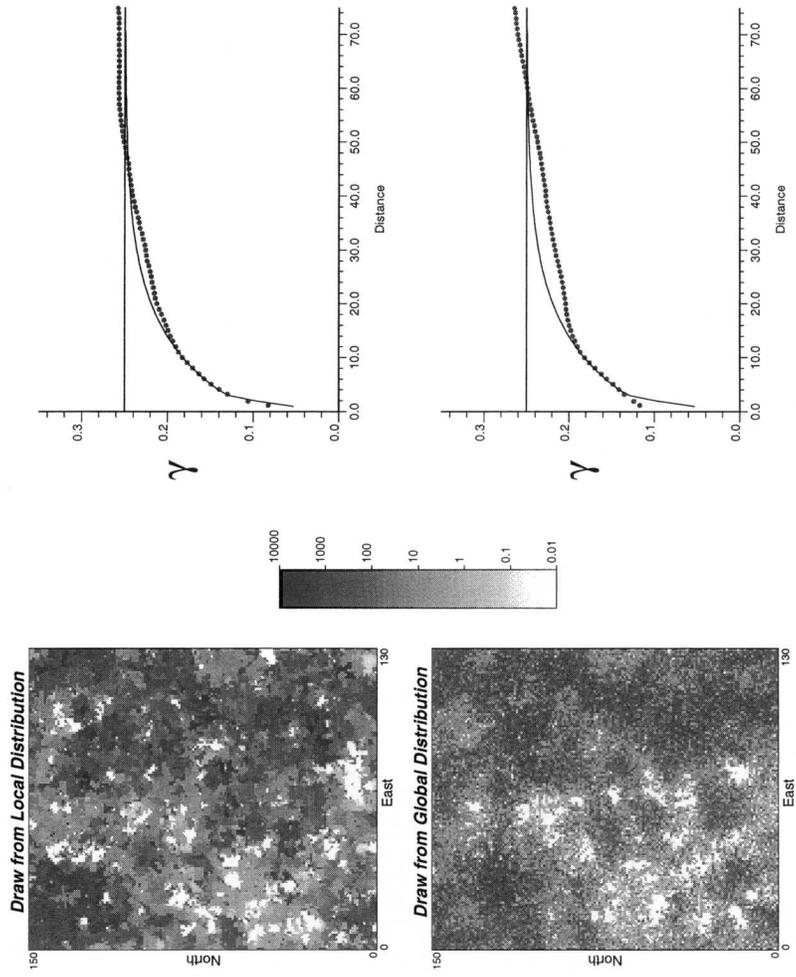


Figure 9. A realization of simulated annealing simulation with drawing from the local distributions (top) and the conventional approach of drawing from the global distribution (bottom). The corresponding isotropic variogram and input model are shown on the right.

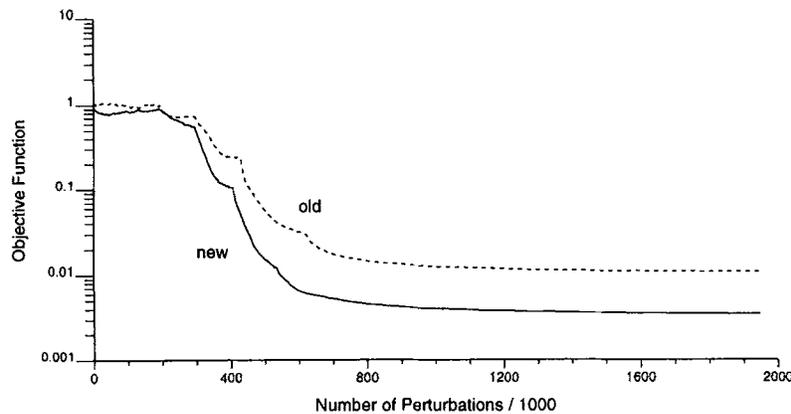


Figure 10. The objective function (total of all components) vs. the number of perturbations. The result of drawing from the local distributions is labeled "new" and the results from the conventional approach is labeled "old."

The sole additional parameter is the number of nearby cell values to use in building the local distribution. We showed that a template size of 8–12 values balanced the improvement brought by the local perturbation mechanism with the additional CPU cost of building local distributions with a large number of nearby values.

This improvement is necessary to achieve the full potential of simulated annealing methods to integrate data of diverse types and scales together with conventional geostatistical constraints such as the histogram and variogram.

REFERENCES

- Datta-Gupta, A., Lake, L. W., and Pope, G. A., 1995, Characterizing heterogeneous permeability media with spatial statistics and tracer data using sequential simulation annealing: *Math. Geology*, v. 27, no. 6, p. 763–787.
- Deutsch, C. V., 1994, Algorithmically-defined random function models, in Dimitrakopoulos, R., ed., *Geostatistics for the next century*: Kluwer, Dordrecht, Holland, p. 422–435.
- Deutsch, C. V., and Cockerham, P. W., 1994, Practical considerations in the application of simulated annealing of stochastic simulation: *Math. Geology*, v. 26, no. 1, p. 67–82.
- Deutsch, C. V., and Journel, A. G., 1992, *GSLIB: Geostatistical software library and user's guide*: Oxford University Press, New York, 340 p.
- Farmer, C. L., 1992, Numerical rocks, in King, P. R., ed., *The mathematical generation of reservoir geology*: Clarendon Press, Oxford.
- Geman, S., and Geman, D., 1984, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images: *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-6, no. 6, p. 721–741.
- Journel, A. G., and Deutsch, C. V., 1993, Entropy and spatial disorder: *Math. Geology*, v. 25, no. 3, p. 329–355.

- Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P., 1983, Optimization by simulated annealing: *Science*, v. 220, no. 4598, p. 671-680.
- Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E., 1953, Equation of state calculations by fast computing machines: *J. Chem. Phys.*, v. 21, no. 6, p. 1087-1092.
- Srivastava, R. M., 1994, An overview of stochastic methods for reservoir characterization, *in* *Stochastic modeling and geostatistics: Principles, methods, and case studies*: Am. Assoc. Pet. Geologists, Tulsa, OK, 379 p.