Grid-less Simulation of a Fluvio-Deltaic Environment

R. Mohan Srivastava, FSS Canada Consultants, Toronto, Canada
MoSrivastava@fssconsultants.ca
and
Marko Maucec, Halliburton Consulting and Project Management, Kuala Lumpur, Malaysia
Jeffrey Yarus, Landmark Software and Services, Houston, USA

Summary

In geostatistics, sequential simulations conventionally work on a grid (either regular or, sometimes, irregular), building realizations of rock or fluid properties by sampling from local probability distributions; they move to a specific location, x, and answer the question "What is a plausible value of Z (which could be an integer value, like a facies code, or a continuous variable, like a rock or fluid property) at this location?" This is accomplished by building a probability distribution of possible Z values at that location, and randomly sampling from this distribution.

An alternative is to fix the value of Z and to answer instead the question "What is a plausible location for this value?" In this approach, the probability distribution that is constructed is a set of possible locations where a specified value of Z might occur. This distribution can be randomly sampled, and the value of Z can be propagated to the chosen location. This approach was first presented at the Stanford Center for Reservoir Forecasting 20 years ago, but has seen little further development until the CSPG Gussow Geoscience Conference in 2011. The idea resurfaced in the presentation by Maučec et al. (2011) and was used in the fracture simulation method proposed by Srivastava (2011). During the open discussion sessions at the 2011 Gussow meeting, the possibility was advanced that innovative reservoir simulation methods exist further down the path opened by these two presentations. Specifically, models of a reservoir's rock and fluid properties could be constructed without a pre-determined grid by propagating rock, fluid or geometric properties to locations that are sequentially chosen.

This paper begins with an explanation of how sequential indicator simulation can be viewed as a simulator of X(z), i.e. of the location where a specific value belongs, rather than in the conventional way, as a simulator of Z(x). It goes on to show how one approach to the simulation of discrete fracture networks can be regarded as a simulator of X(z) and discusses other recent simulation algorithms that have begun to exploit the same idea.

The paper concludes with an example of simulation of fluvio-deltaic channels, the example that has been the hallmark success of the multi-point statistics algorithm, SNESIM. Instead of tapping into multipoint statistics, it is possible to simulate fluvio-deltaic channel geometries that honour well data by sequentially simulating the location of the centerlines of the fluvial channels, and then adding a simulated channel width. This alternative to SNESIM is considerably faster than the multi-point statistics approach and produces realizations that are more geometrically coherent. The explicit simulation of channel centerlines and widths also enables more sophisticated post-processing of the results since morphological properties of the channels (e.g. their curvature, and their inside and outside edges) are more readily accessible from this type of realization than from a pixel-based realization.

Sequential Indicator Simulation

Figure 1 shows an example of sequential simulation for a simple two-category indicator variable that has 0 and 1 as its two possible values. This type of variable is often used to simulate rock type, with the 0 values being used to code one rock type – sandstone, for example – and the 1 values another –

shale, for example. In Figure 1a), all of the indicators in the local neighbourhood (a 5x5 window) have been simulated and all that remains is to simulate the indicator at the central location.
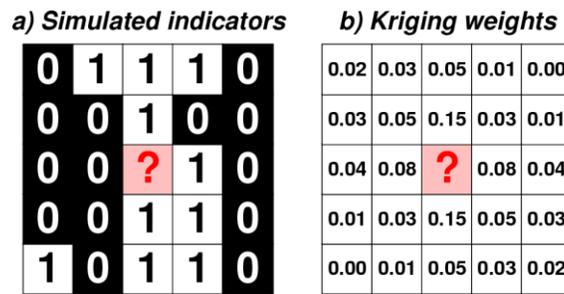

Figure 1: An example of sequential indicator simulation.

In sequential indicator simulation (SIS), kriging weights are calculated (Figure 1b) and the weighted average of the indicators gives us P_1, the probability of encountering a 1 at the central location. The decision on whether to assign a 0 or a 1 at that location is done by drawing a random number, U, and comparing it to P_1. If U is less than P_1, then the central cell is assigned a simulated value of 1; if it is greater than P_1, then the simulated value is 0.

The SIS algorithm is conventionally regarded as a procedure in which we sequentially visit grid nodes and, at each location, build a probability distribution of the unknown variable at that location. In the case of a simple 0/1 indicator variable, the distribution is a binary distribution, with a probability P_0 of encountering a 0 and a probability P_1 of encountering a 1. P_0 is simply the sum of the kriging weights at nearby locations with an indicator of 0, and P_1 is the sum of the other kriging weights, those at locations with an indicator of 1. The step in which a uniformly distributed value, U, is compared to P_1 is a random sampling of the 0/1 variable with P_0 and P_1 being the probabilities of each of the two choices.

Rather than viewing SIS as an algorithm that randomly samples a 0/1 distribution, one could view it as an algorithm that takes a nearby indicator value and assigns it to the central node. The probability of any particular value being propagated to the central node is exactly its kriging weight. The algorithms are identical in their mechanics for the example in Figure 1; the difference is purely conceptual. Rather than viewing the task as one of choosing a plausible value at a given location, the task is viewed as one of taking a value known at one location and propagating it to a plausible location nearby.

Srivastava (1992) presented an iterative indicator simulation procedure that explicitly embraces the notion of propagating known values to plausible nearby locations. Kriging weights determine where each known value can propagate. When a known value is propagated to a new location, it is immediately propagated again to another new location, creating a "thread" of equal values that snakes through the study area in a manner control by the anisotropy expressed in the variogram model used for kriging.. The procedure is fast, but has an undesirable artifact: since each thread is created independently from the others, the resulting simulation ends up with too much short-scale variability. Post-processing is needed to remove this artifact.

Discrete Fracture Network Simulation

Renshaw and Pollard (1994) developed a mechanistic fracture propagation simulator in which randomly-seeded micro-cracks are made to propagate by applying a stress field, calculating the stress field and propagating tips of cracks if the fracture intensity exceeds a critical threshold (Figure 2a).

Srivastava et al. (2004) developed a geostatistical adaptation of this idea using kriging to calculate the probability distribution of the direction of crack growth (Figure 2b), and running the procedure iteratively until a target length distribution had been reached. As shown in Figure 3, this geostatistical procedure shares with Renshaw and Pollard's approach the advantage of producing a visually realistic discrete fracture network (DFN), and has the additional advantage of being easy to condition to known fracture locations.

a) Simulation of crack propagation by solving finite-element stress equations

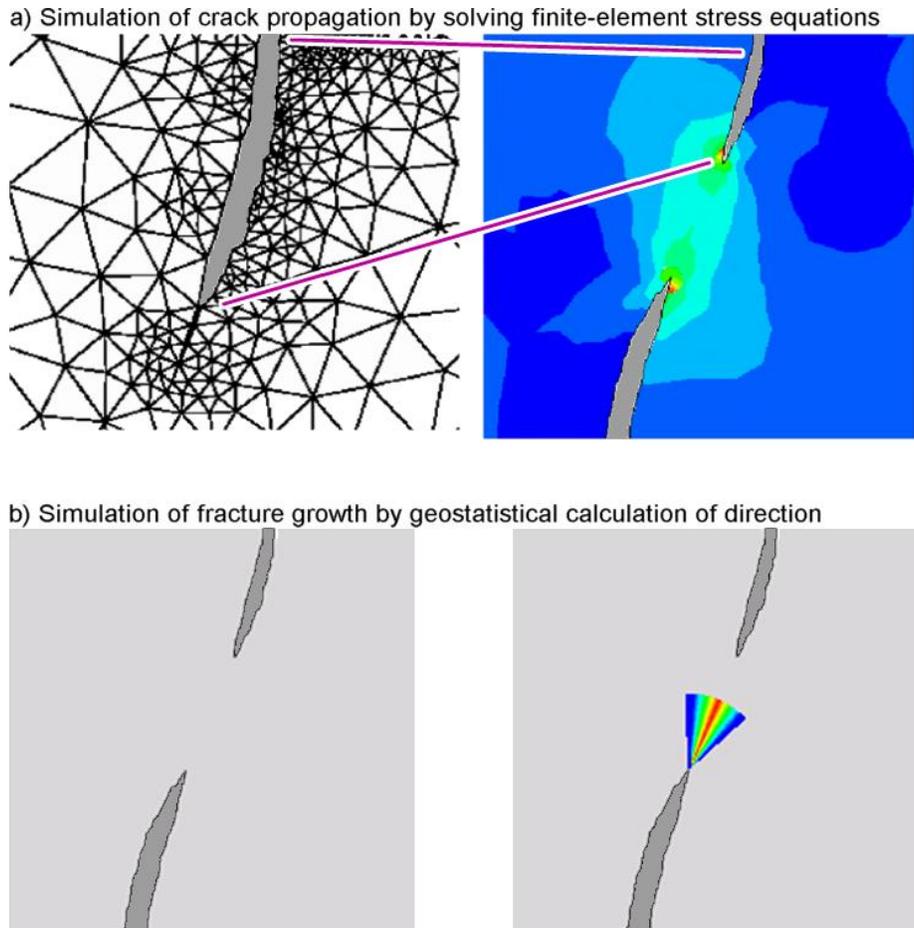b) Simulation of fracture growth by geostatistical calculation of direction

Figure 2: Simulating fracture propagation by: a) solving stress equations, and b) using kriging to calculate a distribution of possible propagation directions (shown as a coloured pie-slice), based on other nearby cracks.
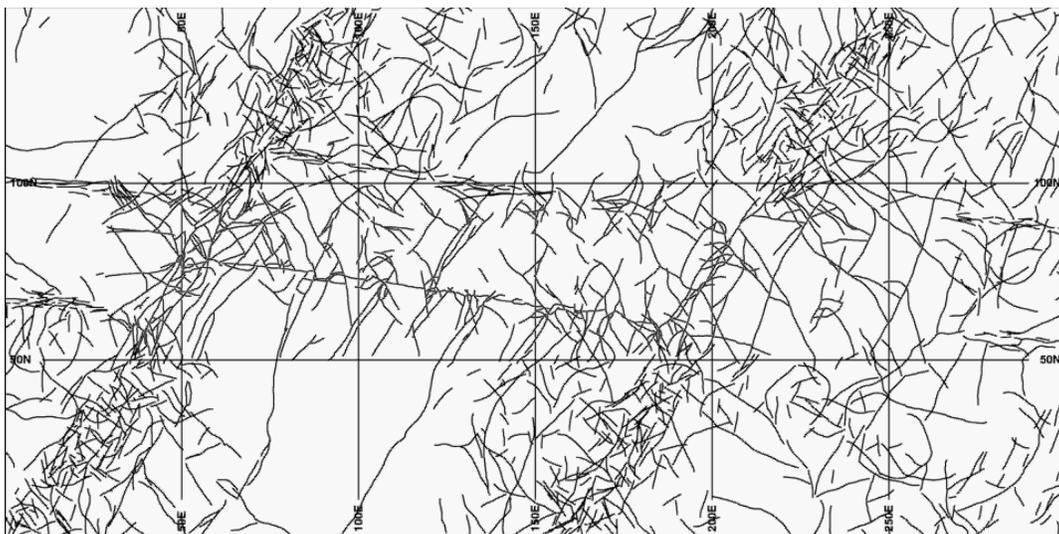
Figure 3: An example of a discrete fracture network simulated using the procedure of Srivastava et al. (2004).

The DFN simulation procedure of Srivastava et al. (2004) is an example of a procedure that exploits the possibility of simulating X(z) instead of Z(x). In this case, the property being simulated is a geometric characteristic: the centerline of a fracture surface. From a location where a centerline is known to occur, the fracture is propagated to a randomly selected nearby location, using surrounding data and a variogram model to ensure that the propagation produces a plausible result.

Other examples of X(z) simulation

The pattern simulation of Arpat (2005) is a procedure that searches for plausible candidate locations of a given multi-point pattern, like having a piece of a jigsaw puzzle in hand, and looking for places where it might fit.

Arpat points out that building a jigsaw puzzle is a useful analogy for building a simulation of a reservoir. Simulation tools that go to a location, x, and try to find a Z value that is appropriate for that location, are analogous to building a jigsaw puzzle by going to a particular location and trying to find the piece that fits there. To take a particular piece and hunt for the location where it belongs is analogous to fixing the Z value and selecting an appropriate location. This alternate strategy, common for building jigsaw puzzles, is not yet frequently employed in geostatistical simulation.

In Arpat's pattern simulation, the Z is not a single value, but rather a multiple-point arrangement, a pattern expressed on a template of a given size and shape. Although the simulation is still implemented within the framework of a predetermined regular grid, it still embodies the X(z) concept by focusing on the question "where does this belong" rather than "what belongs here".

The grid-less "maximum continuity field" (MCF) approach described by Maučec et al. (2011) also expresses some of the X(z) concept. As shown by the DFN simulations discussed earlier, and the fluvio-deltaic simulator discussed in the following section, a procedure that focuses on "where does this belong", rather than "what belongs here", is best implemented in a framework where the simulation is not confined to a predetermined grid. At the same time that the MCF can be understood as a field that locally customizes the anisotropy axes of a variogram model, it can also be understood as a specification of the direction in which a known value is most likely to be repeated. As described by Maučec et al. (2011), this specification does not have to be provided on a predetermined grid. It can, instead, exist as an irregularly spaced set of vectors that condition decisions about directions of propagation wherever such decisions are needed.

A new example: fluvio-deltaic channel geometry

The SNESIM algorithm developed by Strebelle (2000) honours multiple-point statistics that capture complicated spatial patterns expressed by a training image. Figure 4 shows the hallmark success of SNESIM: the simulation of a sand-shale sequence in a fluvio-deltaic environment, with locally varying directions of maximum continuity and channels that become narrower as one moves further from the paleo-shoreline.
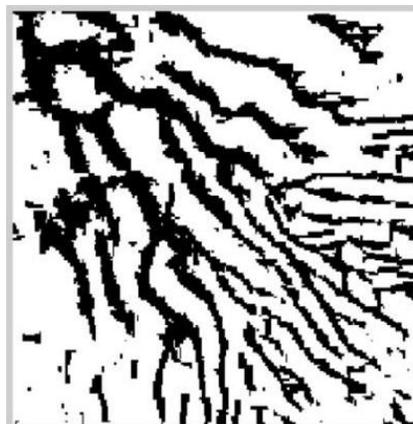

Figure 4. SNESIM simulation of sandstone (black) and shale (white) in a fluvio-deltaic environment.

Rather than treating the simulation of a fluvio-deltaic sandstone-shale as a Z(x) problem, one could approach the task as a X(z) problem. The channels have centerlines that can be simulated in much the same way as fractures were propagated in the DFN procedure described earlier. They also have widths that can be simulated with a sequential Gaussian simulation (SGS) procedure that uses a trend model.

Figure 5 shows conditioning data for a simulation exercise that aims to mimic the SNESIM image in Figure 4; the black dots are locations where a well has encountered sandstone, the white dots are locations where shale has been encountered. The goal of the simulation is to build a model of the sandstone-shale architecture that captures the spatial character of Figure 4.
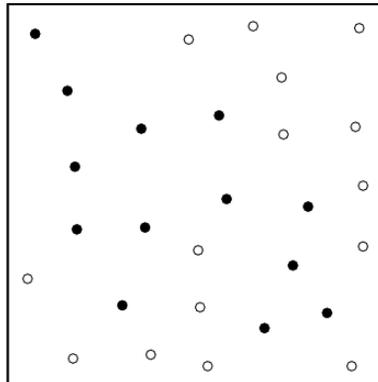

Figure 5. Conditioning data for simulation of sandstone (black) and shale (white) in a fluvio-deltaic environment.

Instead of expressing the simulation on a regular grid, as conventional sequential simulations do, we choose to express the simulation with geometric elements: a set of polylines that mark channel centerlines and a set of channel widths. Figure 6 shows an example of these elements, with the centerlines shown on the left, and the channels extended to their full width on the right.
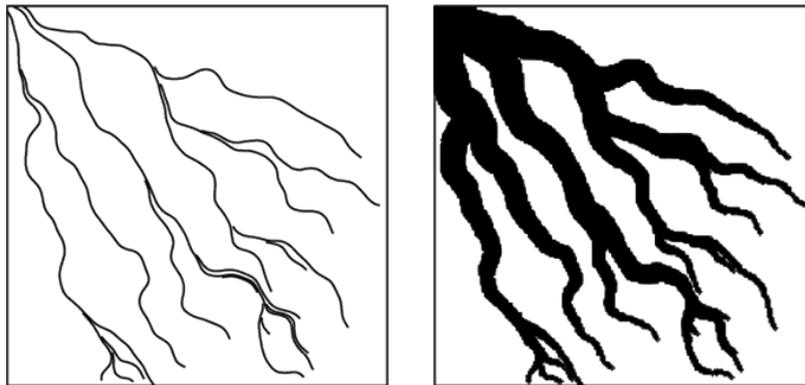

Figure 6. Simulated channels characterized by centerlines and widths.

The simulation begins by seeding a first point on each channel. As shown in Figure 7, some of these points are selected so that, with width added, a channel will cover a location where sandstone is known to occur. In addition to these, other seed points are chosen in order to have the simulation include the correct number of channels; these additional seed points must be chosen in such a way that, with width, they do not cover any location where shale is known to occur.

Each seed point is assigned a direction of propagation. These vectors are like the MCF vectors described by Maučec et al. (2011) in the sense that they describe the local direction of maximum continuity and also in the sense that they need to be defined only at the locations where they are

needed (and not on a complete grid). For the example presented here, these channel directions were extracted directly from the image in Figure 4; they could also have been derived from seismic data or, near well locations, from dipmeter data.
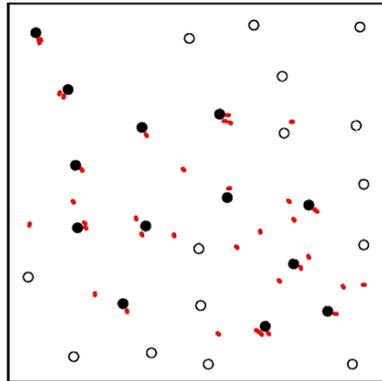

Figure 7. Initial seed points (in red) for channel centerlines.

The seed points are propagated in either direction, using kriging with a trend model to calculate a probability distribution for the azimuth; this distribution is sampled to produce a plausible direction of propagation. Nearby azimuths are used as conditioning data for the estimation of the distribution of azimuths at the tip of each growing centerline. The trend model is the same as the one used in the SNESIM examples, a grid of average azimuths that is NW-SE in the northwest corner (nearest the paleo-shoreline), and that splays, reaching E-W along the eastern edge, and N-S along the southern edge.

The variogram model used for kriging controls the sinuosity of the channels. For the fluvio-deltaic example shown here, a Gaussian variogram model was used to create a high degree of continuity in the azimuths; the range of the variogram model controls the distance over which the channels bend. In a meandering river environment, a hole effect variogram model could be used, causing the channels to swing back and forth in a roughly periodic manner.

The simulation of azimuths is actually a sequential Gaussian simulation with only a couple of minor modifications. The simulation does not follow a regular grid; instead, it is performed at the locations of the tips of the channel centerlines.

The second modification is an acceptance-rejection step in the sampling of the local distributions that is necessary to ensure that centerlines do not come too close a location where shale is known to occur. When a centerline is propagated to a new point, the channel width at that point is simulated (again, using kriging with a trend, to capture the tendency for channels to become narrower in the distal direction). At each newly simulated centerline point, the simulated channel width is added and a check is done for any nearby wells (within a radius equal to the simulated width); if any of the nearby wells is a shale, the simulated azimuth is rejected, and the probability distribution of azimuths is resampled until an acceptable combination of a new point and a new width does not contradict any of the locations where shale is known to occur.

Figure 8 shows three snapshots of the evolution of the simulation of centerlines. Figure 9 shows three examples of the final realizations. The final realizations capture the sand-shale geometry with a set of centerlines and channel widths; there is no underlying grid. Simulated values of centerline location and of channel width have been generated only where they are needed. There are two clear advantages of this style of simulation: i) it is computationally faster, and ii) it represents the channel architecture in a way that facilitates post-processing.

The speed of the computation is due to the fact that kriging and sampling of probability distributions are done only where they are needed. The creation of the realizations shown in Figure 9 involves only a

few thousand kriging/sampling calculations. Were the area to be rasterized with a grid fine enough to resolve the narrowest channels, there would be tens of thousands of pixels, each one requiring a kriging calculation and a sampling of a distribution. In a multiple-point statistics approach, replacing kriging with the acquisition of the required statistics from the training image further slows the calculation.

With channel architecture rendered as a set of centerlines and channel widths, it is easy to post-process the realizations to extract information on channel morphology. For example, it is easy to identify the inside and outside edges of a bending channel; extracting the same information from a pixelized image is more difficult. Such information is often useful when overlaying simulated porosity-permeability values onto a simulation of reservoir facies, or when refining a facies simulation to include secondary features, like overbank deposits.
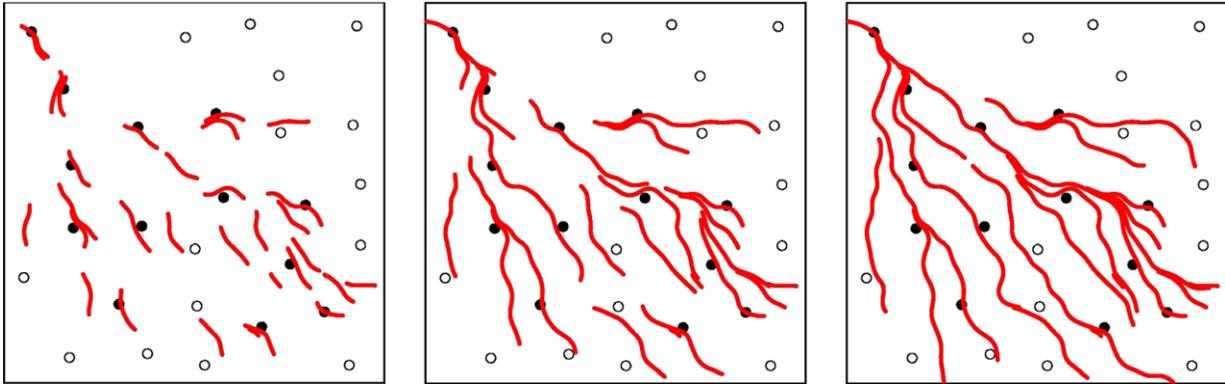

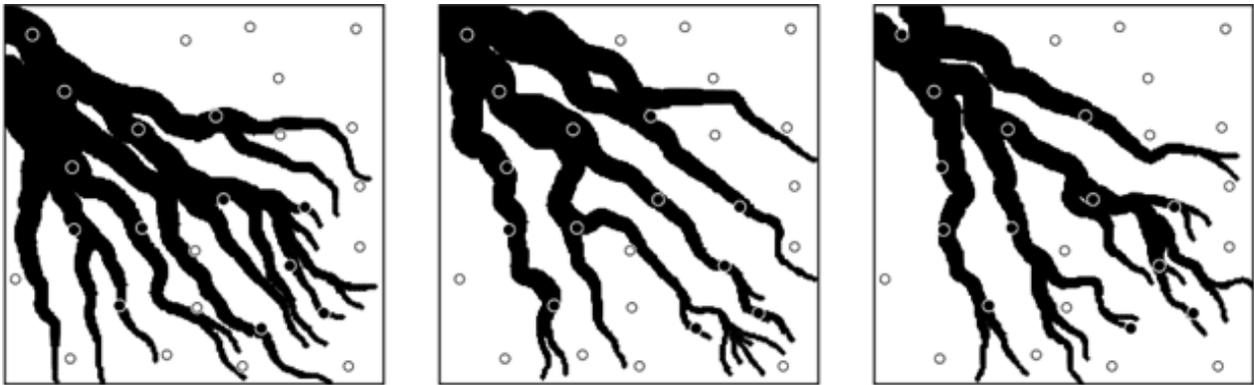Figure 8. Iterative growth of channel centerlines.

Figure 9. Three realizations of sandstone-shale geometry conditioned by the same well data.

The realizations in Figure 9 are "cleaner" than the one in Figure 4, with the channels showing clearer, sharper edges. Although it is easy to dismiss this visual clarity as a merely aesthetic advantage, a "prettier picture", there are some production planning studies in which the existence (or lack) of a continuous pathway through the sandstone is important; in such studies, pixelized images that have too much short-scale variability may underestimate the ability of reservoir fluids, and production stimulation fluids, to efficiently move large distances.

The principal parameters required for the simulations shown above are more accessible to most geologists than those required by other geostatistical simulation procedures. The two key parameters are distributions of channel orientations and widths, which could be extracted from training images, if these exist, or could be provided directly by reservoir geologists since these geometric properties are easier to comprehend than two-point statistics (variograms) and multi-point statistics. The only other parameter, the

variogram model that controls channel sinuosity, can also be extracted from training images or, if these are not available, can be calibrated by trial-and-error since the procedure can be rapidly re-run to visually check the impact of different variogram model choices.

Conclusions

Simulations of the geometry of a reservoir's major flow units do not have to be based on a predefined grid, either regular or irregular. When reservoir architecture can be characterized using simple geometric attributes, such as channel orientation and width in the previous example, simulation can proceed iteratively, taking an attribute known at one location and propagating it to a nearby location. As the fluvio-deltaic example in this paper shows, the propagation can be done using the same engine as the one used in other geostatistical procedures: kriging, with or without a trend model, to build a probability distribution that can be randomly sampled to create plausible geometries.

This grid-less approach to simulation is computationally efficient since the kriging calculations and random sampling are done only at locations where it is necessary, and these are usually far fewer than the number of cells involved in a typical grid.

References

Arpat, B., 2005, Sequential Simulation with Patterns, PhD thesis, Stanford University.

Maučec, M., Yarus, J., Chambers, R. and Shi, G., 2011, Modeling of Reservoir Properties with Grid-less Continuity Field Interpolation, Presentation at the Gussow Geoscience Conference, Banff, October 3-5, 2011.

Renshaw, C. and Pollard, D., 1994, Numerical simulation of fracture set formation: a fracture mechanics model consistent with experimental observations. Journal of Geophysical Research 99, 9359–9372.

Srivastava, R.M., 1992, Iterative Methods for Indicator Simulation, Presentation at the Stanford Center for Reservoir Forecasting Annual Meeting, Stanford, May 1992.

Srivastava, R.M., 2011, Improving Stochastic Fracture Modeling by Honoring More Conditioning Data, Presentation at the Gussow Geoscience Conference, Banff, October 3-5, 2011.

Strebelle, S., 2000, Sequential simulation drawing structures from training images. PhD thesis, Stanford. University,

Srivastava, R.M., Frykman, P. and Jensen, N., 2004, Geostatistical simulation of fracture networks, in Geostatistics Banff 2004 (Volume 2), O. Leuangthong and C.V. Deutsch (eds.), Springer, Dordrecht.