Multiple Point Statistics with Multiple Training Images

Daniel A. Silva and Clayton V. Deutsch

Abstract

Characterization of complex geological features and patterns has been one of the main tasks of geostatistics. MPS simulation is an increasingly common alternative based on training images that contain and relate complex relationship between data. Reproduction of the main features results from borrowing spatial statistics from training images. The current work proposes to use multiple training images by a data integration scheme known as the linear opinion pool. An advantage of utilizing more than one training image relies on the ability to capture distinct heterogeneities, variability and patterns. The algorithm was implemented inside the snesim code. CPU time and memory requirements are proportional to the number of training images used.

Introduction

Exploration of mineral deposits and petroleum reservoirs, and later their operation, involves several branches of geosciences such as geophysics, geochemistry and geology. All contribute with valuable information that leads to construction and modeling of complex geological features important for economic evaluation. One of the challenges of Geostatistics is characterized and assets the uncertainty of these underlying features. In the past, simulation techniques based on linear relationships and the benefits of Gaussian distributions has been able to give response to a large number of geological phenomena - especially in mining. However, a statistics relationship beyond two points is needed for those deposits with complex features and patterns, as the case of petroleum reservoirs. Multiple Point Statistics (MPS) simulation was developed to simulate and reproduce complex phenomena taking borrowed high order statistics relationships of the complex features and patterns from training images. It acts as the probability distribution function of reservoir, which is completely known. The method is based on the single normal equation proposed by Guardiano and Srivastava (1992).

Several methods of MPS and important advances on practical applications have been developed since then. The **snesim** algorithm (Strebelle and Journel, 2000) popularized the use of MPS at reducing the excessive computer requirements implementing a search tree storage procedure. Also, a method that rests on simulate annealing was presented by Deutsch (1992). There are others proposed methods relied on Gibbs Sampler iteration (Srivastava, 1992; Lyster, 2007) and neural networks (Caers & Journel, 1998; Caers 2001). Ortiz (2003) introduced the integration of runs and indicator simulation to reproduce high order statistics.

All MPS simulation may divide into two general class of algorithm. First, those that scan and storage the information from training images, and calculates the conditional probability in a sequential scheme. Second, those that start from a training image and converge to the final solution. Furthermore, most MPS algorithms have been mainly focused on categorical variables. The number of categories looked upon must be lower. An increment in the use of categorical variables will increase the problem on a large scale.

MPS algorithms utilize training images for extracting specific geological patterns previously recognized. The training images replace the structural spatial tools based on two point statistics such as variogram and covariance–amply utilized in the mining industry nowadays– which considers only the linear relationship between data. Neither curvilinear nor specific patterns, involving more than two points, may be reproduced by basis algorithms on two-point statistics. As a response, the training image can summarize higher order statistics of the random function, including the variogram and covariance, allowing to MPS method to model more complex geological deposits. It reproduces the patterns and curvilinear relationship present on the training image. The advantage rests that the training image can be generated from a non-conditional simulation of an object-based algorithm or even simpler, a sketch with the main characteristic devised by a geologist.

An advantage of utilizing more than one training image, at the same time, relies on the ability to capture distinct heterogeneities, variability and patterns. Even though some MPS method can account for multiple training images simultaneously, it looks for patterns into multiple different training images, rather acting as a big training image. Thus, the conditional probability is evaluated from the total number of replicates found on all training images. There is no manner to combine the conditional probabilities of each training image under this scheme. The current work proposes to extract the information from two or more training images, but combining the conditional probabilities – calculated independently for each training image – and coming up with a new conditional probability that mixes the main patterns of each training image. It leads us to the paradigm of data integration from different sources. A weighted linear combination known as linear opinion pool is presented. It overcomes the conditional probability calculation issue from multiple training images.

Integrating Multiple Training Images

The most remarkable characteristic of Multiple Point Statistics is its simplicity. It relies on the single normal equation framework of a single-multiple point event (Journel, 1992). The purpose is to evaluate the conditional probability at location u of an attribute $\mathrm{S(u)}$ given a set of hard data $\{\mathrm{S(u_\alpha)}, \alpha = 1, \dots, \mathrm{n}\}$ using a n+1 single data event $\mathrm{D_n(u)}$.

A data event $\mathrm{D_n(u)}$ depicts the specific spatial configuration of n conditioning hard data given by the set of vectors $\{\mathrm{h_\alpha}, \alpha = 1, \dots, \mathrm{n}\}$ centered at location u. It means that the number of replicated data event found on the training image will be accounted for evaluating the conditional probability. Now, consider a variable $\mathrm{S(u)}$ that may take K values from the set $\{\mathrm{s_k}, \mathrm{k} = 1, \dots, \mathrm{K}\}$, and the data event $\mathrm{D_{m,n}(u)}$ associated to the training image m centered at location u and conditioned to hard data $\{\mathrm{S(u_\alpha)}, \alpha = 1, \dots, \mathrm{n}\}$. Then, the conditional probability at location u for a data event $\mathrm{D_{m,n}(u)}$ can be summarized as:

$$\mathrm{P}\big(S(u) = s_k \big| D_{m,n}(u)\big) = \frac{\mathrm{P}(S(u) = s_k, D_{m,n}(u))}{\mathrm{P}(D_{m,n}(u))} \tag{1}$$

Hereafter, we can drop off the notation for location u on the conditional probability. Classical two-point geostatistical methods must simplify the lack of understanding of high order statistics by the assumption of stationarity. On the contrary, a training image establishes the relationship between more than two points acting as a multiple-point statistics source. Thus, denominator and numerator probabilities of expression (1) can be evaluated directly from the training images. However, set up a training image is not a simple step. Modeller must possess a high understanding of geological process which originated complex features.

MPS is framed under sequential simulation methods. Each location is simulated conditionally to the data available and then converted in a hard data for the next simulating location. The algorithms proceed as follows:

1. Visit an unsampled node
2. Compute conditional probability of attribute s_k according with data event D_n
3. Add the simulated value as a hard data and moving to the next location
4. Repeat the process for a new realization

The size of data event $\mathrm{D_{m,n}}$ has direct implications on the reproduction of the main features of the training images over the final realization. A large data event probably will not find an enough replicates to estimate the conditional probability. On the contrary, a too specific data event will not capture the larger relationship between points. The size must ensure to capture the main features, but at the same time reduce the overload of computer requirements.

The use of training images to modeling complex geological deposits have encouraged the setting up of extensively libraries containing its main features (Pyrcz, 2004). Some of them prioritizing smooth shapes with sinuous and curvilinear characteristics; whereas, other privilege an implicit randomness component. The fact is that in so far the modeller needs to combine certain shapes from different training images, he must construct - by “some” method - a new one trying to mimic and reproduce the features needed. This current work presents an alternative to evaluate conditional probabilities (1) mixing several training images by a weighted linear combination of conditional probabilities. Also, a brief review of data integration paradigm from different sources is treated.

How to relate and incorporate the information that come from different sources for a post processing evaluation is not an easy step. On geostatistics, one of the challenges has been integrate information from categorical or continuous attributes by adding secondary information to improve the prediction and uncertainty of reservoirs (Hong, 2010). This work is focused on categorical outcomes from conditional probabilities integrating information from different training images.

Again, consider a geological attribute $\mathrm{s_k}$ taking a value k given m conditional training images with the same data event $\mathrm{D_{m,n}}$ configuration. The goal is to calculate the conditional probability:

$$\mathrm{P}\big(s_k = k \big| D_{1,n}, \dots\dots, D_{m,n}\big) \tag{2}$$

Several approaches have been developed to facilitate the calculation of conditional probabilities with different assumptions. Two of them will be treated on this work. First, the ***probability combination scheme*** based on the interdependence between sources of information making possible the conditional probability inference by the Bayes relationship. Second, the ***consensus theory*** – developed in the field of management science- considers the importance of weighting the expert opinion at combining the estimated conditional probabilities from each source of information (Winkler, 1968). There is a third avenue of data integration method known as ***multivariate density estimation***, but this has a different background relied on the complete understanding of the conditional probability function by an analytical expression.

Probability Combination Scheme Training Images Integration

Strebelle and Journel (2000) stated the option to add soft information for evaluating primary information using the Bayes relationship for conditional probabilities. Training image of primary and soft data are considered as one vectorial training images where the data event scan both training images to inference expression (1). However, this work proposes to evaluate the conditional probabilities separately and combining them to obtain an expression as (2). Thus, from probability theory the conditional probability expression (2) may be expanded by the Bayes' Law as follows:

$$\mathrm{P}\left(S = s_k \middle| D_{1,n}, \ldots\ldots, D_{m,n}\right) = \frac{\mathrm{P}\left(D_{1,n}, \ldots\ldots, D_{m,n} \middle| S = s_k\right) \times \mathrm{P}(S = s_k)}{\mathrm{P}\left(D_{1,n}, \ldots\ldots, D_{m,n}\right)} \tag{3}$$

Probability $\mathrm{P}(\mathrm{S} = \mathrm{s_k})$ is the global proportion of attribute $\mathrm{s_k}$ on all training images, $\mathrm{P}\left(\mathrm{D_{1,n}}, \ldots\ldots, \mathrm{D_{m,n}}\right)$ is the joint probability of all training images and $\mathrm{P}\left(\mathrm{D_{1,n}}, \ldots\ldots, \mathrm{D_{m,n}} \middle| \mathrm{S} = \mathrm{s_k}\right)$ is the likelihood. The solution of this would require knowing all the probabilities mentioned before. However, it is possible to assume the conditional independence between secondary data $\left\{\mathrm{D_{1,n}}, \ldots\ldots, \mathrm{D_{m,n}}\right\}$ given the primary data $\mathrm{s_k}$. Thus the likelihood may be discomposed in a linear product of $\mathrm{P}\left(\mathrm{D_{1,n}} \middle| \mathrm{S} = \mathrm{s_k}\right) \times \ldots \times \mathrm{P}\left(\mathrm{D_{m,n}} \middle| \mathrm{S} = \mathrm{s_k}\right)$, and expanding equation (3) jointly with the Bayes' Law, we have

$$\mathrm{P}\left(S = s_k \middle| D_{1,n}, \ldots\ldots, D_{m,n}\right) = \frac{\mathrm{P}\left(S = s_k \middle| D_{1,n}\right)}{\mathrm{P}(S = s_k)} \times \ldots \times \frac{\mathrm{P}\left(S = s_k \middle| D_{m,n}\right)}{\mathrm{P}(S = s_k)} \times \mathrm{P}(S = s_k) \times C \tag{4}$$

Where C is a normalizing factor, which makes sure that expression (4) satisfies the closure property. As a consequence, the conditional probability given different training images may be expressed as a multiplication of conditional probabilities of each training image. The conditional probability $\mathrm{P}\left(\mathrm{S} = \mathrm{s_k} \middle| \mathrm{D_{m,n}}\right)$ is easily extracted from the training image m given the data event $\mathrm{D_{m,n}}$.

Assume conditional probability under this scheme has some issues. Firstly, multiplication of conditional probabilities of each training images conducts to a non-convex solution. It tends to prefer the most informative conditional probability. Secondly, the final realization will be closely to the training image with the higher conditional probability. Finally, manipulation over conditional probabilities is not possible and the judgement of evaluator cannot be applied. Consensus theory proposes to overcome these issues weighting the importance of each training image according with some analytical or empirical criteria.

Consensus Theory Training Images Integration

Consensus theory assumes that each source of information, such as training images, is conditioned to the opinion of an expert, which may be subjective or forecasted from a model. The challenge is combine and weighting distinct evaluated conditional probabilities with the same level of information according with the experts's opinions. The way that the estimated probabilities are combined must be a consistent probability –values between 0 to 1 – and satisfying the closure property. The easiest approach is allocate the weights between training images using linear combination scheme, it is known as linear opinion pool (Stone, 1961):

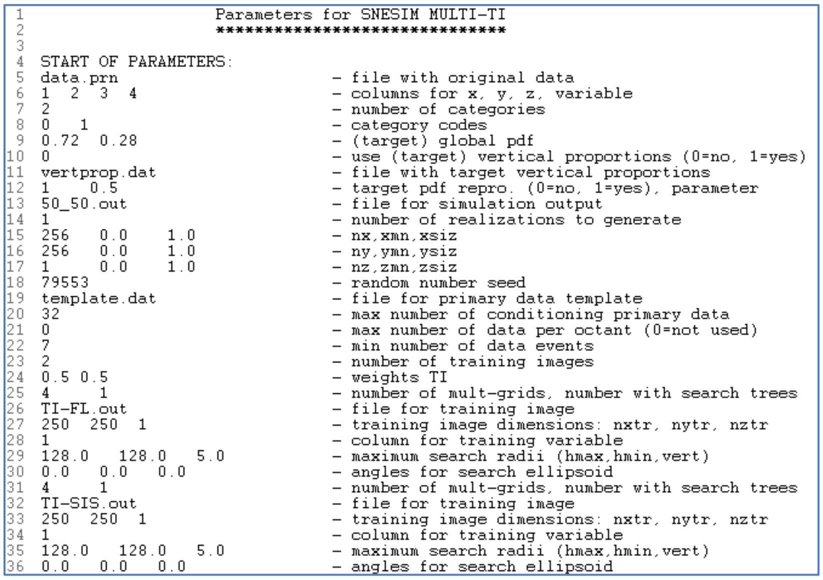
$$\mathrm{P}\left(S = s_k \middle| D_{1,n}, \ldots\ldots, D_{m,n}\right) = \sum_{i=1}^{n} w_i \mathrm{P}\left(S = s_k \middle| D_{i,n}\right) \tag{5}$$

with $\mathrm{w_i} \geq 0$ and $\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{w_i} = 1$. The problem is to quantify the weights $\mathrm{w_i}$ for each training image according with the expert opinion, which may be based on expertise of the decision maker, calibrated by some previous mathematical model from the same data or simply weighting all the estimated probabilities with the completely proportions subject to the modeller experience, and thus, getting the output desired.

The **snesim** algorithm was utilized to implant the linear opinion pool approach on the calculation of conditional probability. It is based on a dynamic data structure, called search tree, used to store the conditional probabilities evaluated from the training image. The search tree corresponds to a series of nodes linked where each node contains one specific conditioning data event. The data event must have at least one replicate to be considered a node of search tree. The main advantage of the algorithm is that the scanning process is performed only once and each repeated data event stored at the corresponding node.

Implementation of linear opinion pool modified the original snesim code and adapted the search tree for multiple training images. Each training image has associated one search tree. The algorithm is exactly than MPS sequential simulation: (1) visit each node, (2) perform and storing on the search tree the conditional probability according with the data event $\mathrm{D}_{\mathrm{m,n}}$ for the training image m, (3) combine the conditional probabilities of each training image using the linear opinion pool (5) subject to some weights previously calculated, (4) add the simulated value for be used at next location and (5) repeat the same process to generate a new realization.

The source code and the executable file are provided in parallel with the paper in the same directory. The parameters are equivalent to the original version of snesim program. A brief description is presented.



```
 1                     Parameters for SNESIM MULTI-TI
 2                     ******************************
 3
 4  START OF PARAMETERS:
 5  data.prn                          - file with original data
 6  1  2  3  4                        - columns for x, y, z, variable
 7  2                                 - number of categories
 8  0   1                             - category codes
 9  0.72  0.28                        - (target) global pdf
10  0                                 - use (target) vertical proportions (0=no, 1=yes)
11  vertprop.dat                      - file with target vertical proportions
12  1    0.5                          - target pdf repro. (0=no, 1=yes), parameter
13  50_50.out                         - file for simulation output
14  1                                 - number of realizations to generate
15  256   0.0    1.0                  - nx,xmn,xsiz
16  256   0.0    1.0                  - ny,ymn,ysiz
17  1     0.0    1.0                  - nz,zmn,zsiz
18  79553                             - random number seed
19  template.dat                      - file for primary data template
20  32                                - max number of conditioning primary data
21  0                                 - max number of data per octant (0=not used)
22  7                                 - min number of data events
23  2                                 - number of training images
24  0.5 0.5                           - weights TI
25  4     1                           - number of mult-grids, number with search trees
26  TI-FL.out                         - file for training image
27  250  250  1                       - training image dimensions: nxtr, nytr, nztr
28  1                                 - column for training variable
29  128.0   128.0   5.0               - maximum search radii (hmax,hmin,vert)
30  0.0   0.0   0.0                   - angles for search ellipsoid
31  4     1                           - number of mult-grids, number with search trees
32  TI-SIS.out                        - file for training image
33  250  250  1                       - training image dimensions: nxtr, nytr, nztr
34  1                                 - column for training variable
35  128.0   128.0   5.0               - maximum search radii (hmax,hmin,vert)
36  0.0   0.0   0.0                   - angles for search ellipsoid
```

Lines 5 up to 8 are self-explanatory. Global proportion and the use of vertical proportions for simulating the codes are set up on lines 9, 10 and 11. Line 12 does reference to the servo system correction, which adjusts the final proportion to reach the target proportion of categories. From lines 12 to 19 the code read the parameters related with the name of output file, the number of realizations, origin and size of the grid and the file with the template used up for calculating the conditional probabilities. Lines 20,21 and 22 describe the estimation parameter such as number of samples, events and octants deemed. The number of training utilized by the methodology is established on line 23, and line 24 set up the partial proportion to be employed by each training image. From lines 25 to 30 set out the name of the file and the column of the training image, the size and the search ellipsoid. For each training image involved a new set of similar lines need to be written, that is the case of the second training image called "TI-SIS.out".

Example

Synthetic examples are presented; however, it is necessary clarify that this work focuses on the paradigm of data integration between training images - relies on linear opinion pool- and not in the manner of calculating its weights. First, a fluvial reservoir example is shown in Figure (1). Two training images are used, the first associated to a smooth fluvial reservoir Figure (1.A) and the second a training image contains a higher random component generated Figure (1.B) by Sequential Indicator Simulation (SIS). Figures (1.C-1.G) show realizations obtained from the conditional probability subjects to (5) using different weights.

A second example is developed with a turbidite reservoir training images Figure (2). The procedure is exactly the same, a second training image is created using SIS based on the variogram of the first training image. Figures (2.C-2.G) show the changes on realizations at combining the weights.

To use a probability combination scheme as a source of integration of training images is a valid alternative too. Two examples show realizations using this scheme. The conditional probability is combined by (4). Although the training images are distinct, Figure (3.D) and Figure (4.D) represents a realization that is close to the smother training image. There is no option to manipulate any possible combination between training images neither by a linear combination nor by a special function calculated from the same data. It bounds the field of action of this training images integration scheme.

In terms of CPU timing performance, the time is based on the original snesim code when using single training image. Thus, the proposed algorithm is proportional to the number of training images used. The examples before used just two training images as sources to get conditional probabilities; however, the use of more than two is possible though unnecessary. For each realization are generated two search trees, one for each training image. Table – 1 shows a comparison between the time of the original snesim code and the proposed code. Using two training images the time for simulating a realization from tow training images is duplicated.

In the case of RAM memory the performance is not so different. For the original snesim code, the amount of memory necessary to storage and simulate a block model of 256x256x256 using a training image of 256x256x256 is approximately 500 MB. The fact of adding a new training image increase the use of CPU memory depending of the size of the training image; hence, use an additional training image entails to increase the memory on 130 MB associated to the storage search tree process and 330 MB as consequence to read and storage the information of the extra training images, which is almost twice the RAM used for one training image.

Discussion and Conclusion

A methodology to combine the conditional probabilities of distinct training images capturing the main features from each has been presented. A weighted linear combination of conditional probabilities known as a linear opinion pool is recommended. A second approach using a probability combination scheme was tested for a comparison. The snesim code was modified with the algorithm; however, the methodology is extensible to any scanning and storing MPS algorithm. The computer requirements increase in direct proportion to the number of training images. As future work, one of the most important things is to quantify the precise weight for each training image.

References

Caers, J. (2001) Geostatistical Reservoir Modelling Using Statistical Pattern Recognition. Journal of Petroleum Science and Engineering, Vol. 29, No. 3, May 2001, pp 177-188.

Caers, J. and Journel, A.G. (1998) Stochastic Reservoir Simulation Using Neural Networks Trained on Outcrop Data. SPE Annual Technical Conference and Exhibition, New Orleans, Oct. 1998, pp 321-336. SPE #49026.

Deutsch, C.V. (1992) Annealing Techniques Applied to Reservoir Modeling and the Integration of Geological and Engineering (Well Test) Data. Ph.D. Thesis, Stanford University, 304 p.

Guardiano, F.B. and Srivastava, R.M. (1992) Multivariate Geostatistics: Beyond Bivariate Moments. Soares, A., Editor, Geostatistics Troia ’92, Vol. 1, pp 133-144.

Journel, A.G., (1992). Geostatistics: roadblocks and chellenges. In A. Soares (Ed.), Geostatistics-Troia, 213-224. Knluwer Academic Pubnl., Dordrecht.

Ortiz, J.M., (2003). Characterization of High Order Correlation for Enhanced Indicator Simulation. Ph.D. Thesis, University of Alberta, 255 p.

Lyster, S. and Deutsch, C.V. (2008) MPS Simulation in a Gibbs Sampler Algorithm. 8th international Geostatistics Congress, 10p.

Pyrcz, M.J., (2004), The integration of geological information into geostatistical models, PhD.D. Thesis, University of Alberta, 250p.

Srivastava, M. (1992) Iterative Methods for Spatial Simulation. Stanford Center for Reservoir Forecasting, No. 5, 24 p.

Strebelle, S.B. and Journel, A.G. (2000) Sequential Simulation Drawing Structures From Training Images. Kleingeld, W.J. and Krige, D.G., Editors, 6 101-12 Th International Geostatistics Congress, 12 p.

Winkler, R. L., (1968). The Consensus of Subjective Probability Distributions, Management Science, Vol. 15, No. 2, Application Series (Oct., 1968), pp. B61-B75.

Block Model	Snesim (min)	Snesim-II (min)
64x64x64	3	6.5
128x128x128	17	36
256x256x256	119	268

Table 1: CPU time comparison

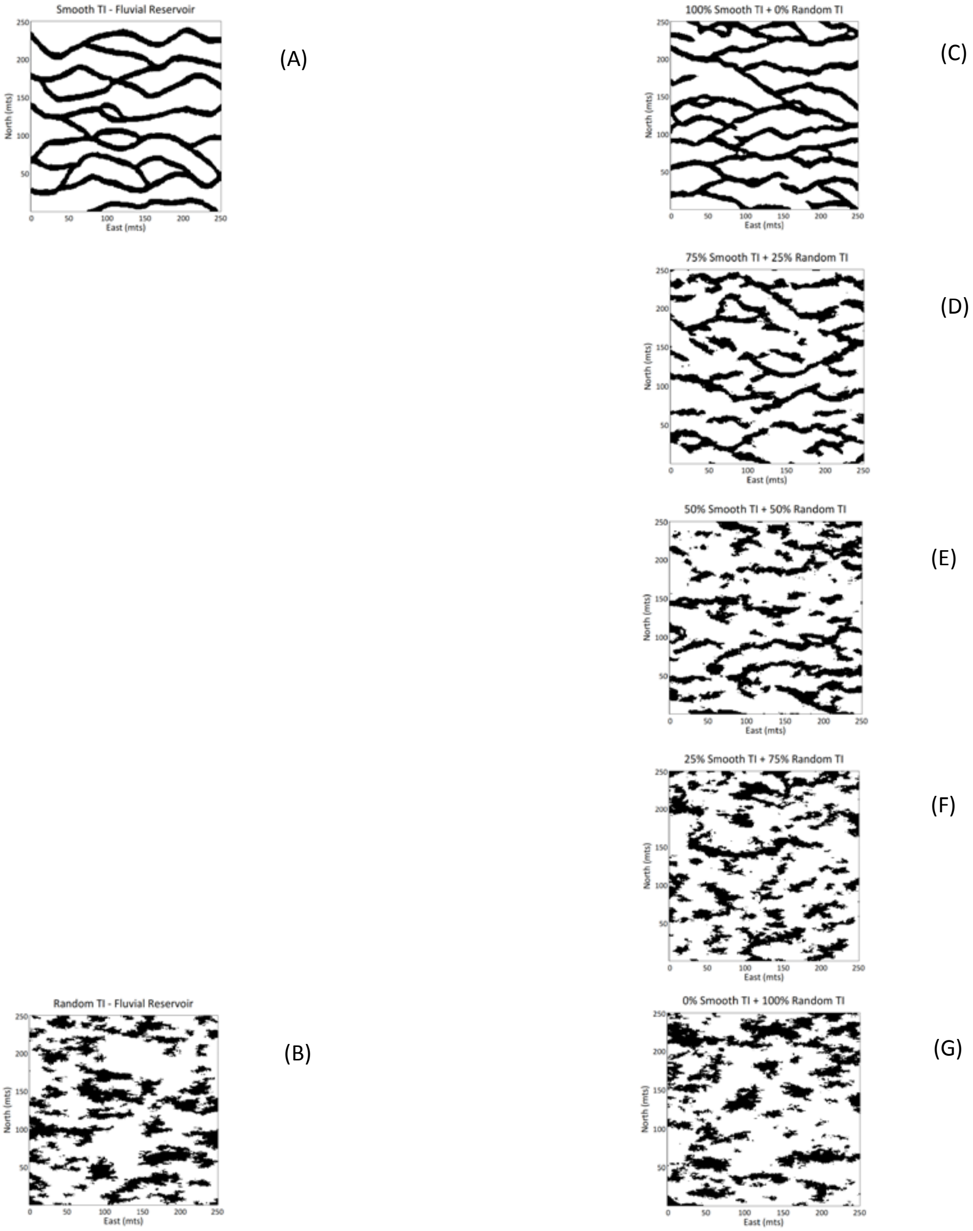


Figure 1: Realizations using the linear opinion pool data integration approach for a fluvial reservoir

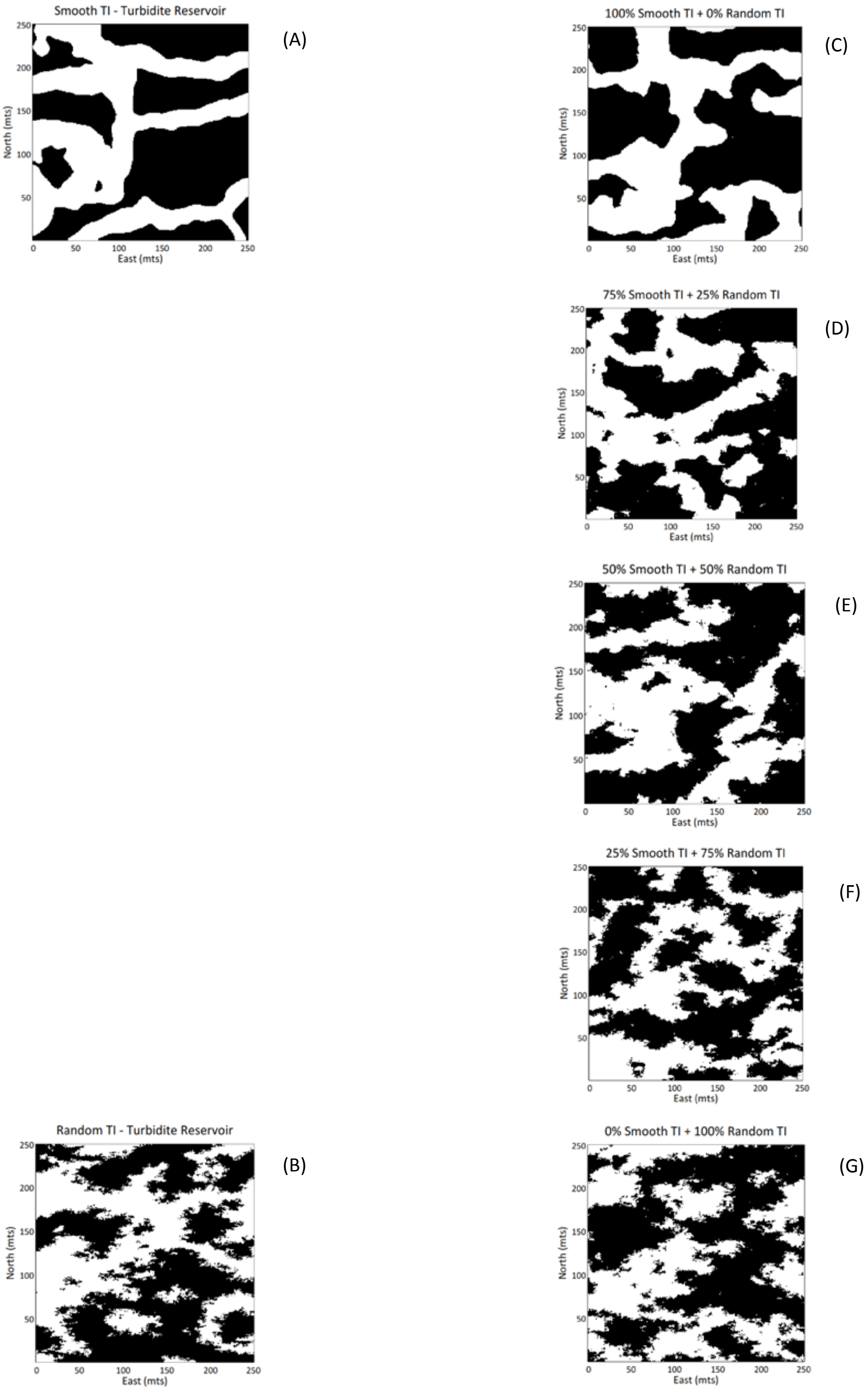


Figure 2: Realizations using the linear opinion pool data integration approach for a turbidite reservoir

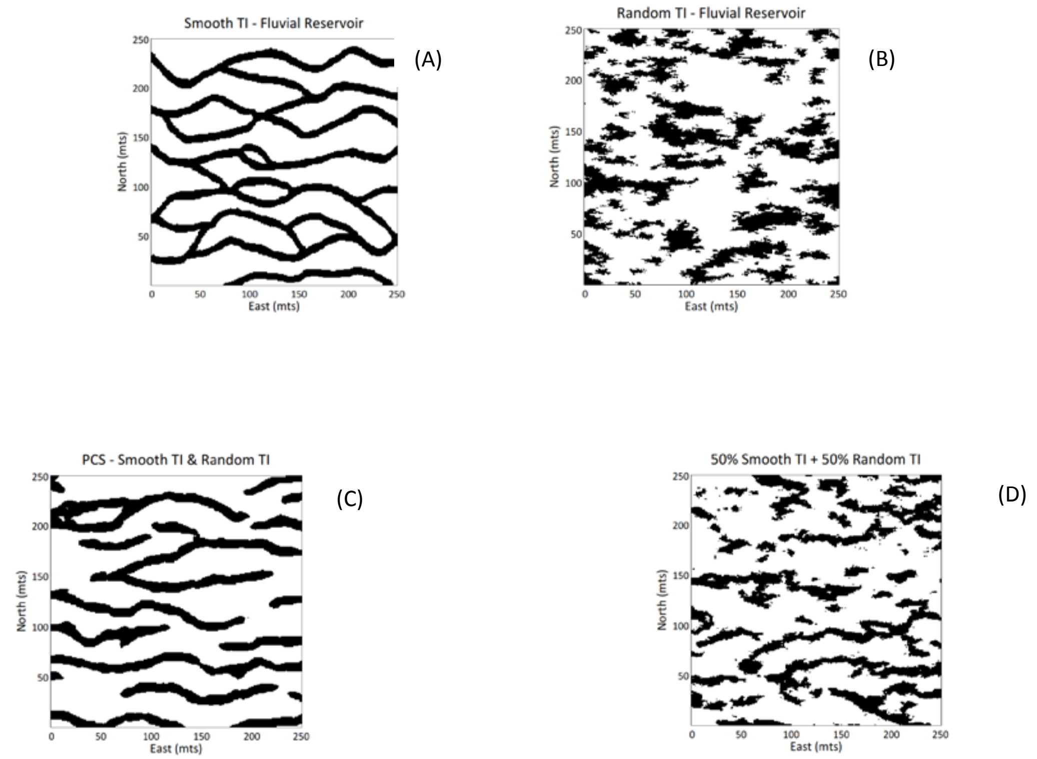


Figure 3: Realizations comparison the linear opinion pool and probability combination scheme for data integration - fluvial reservoir

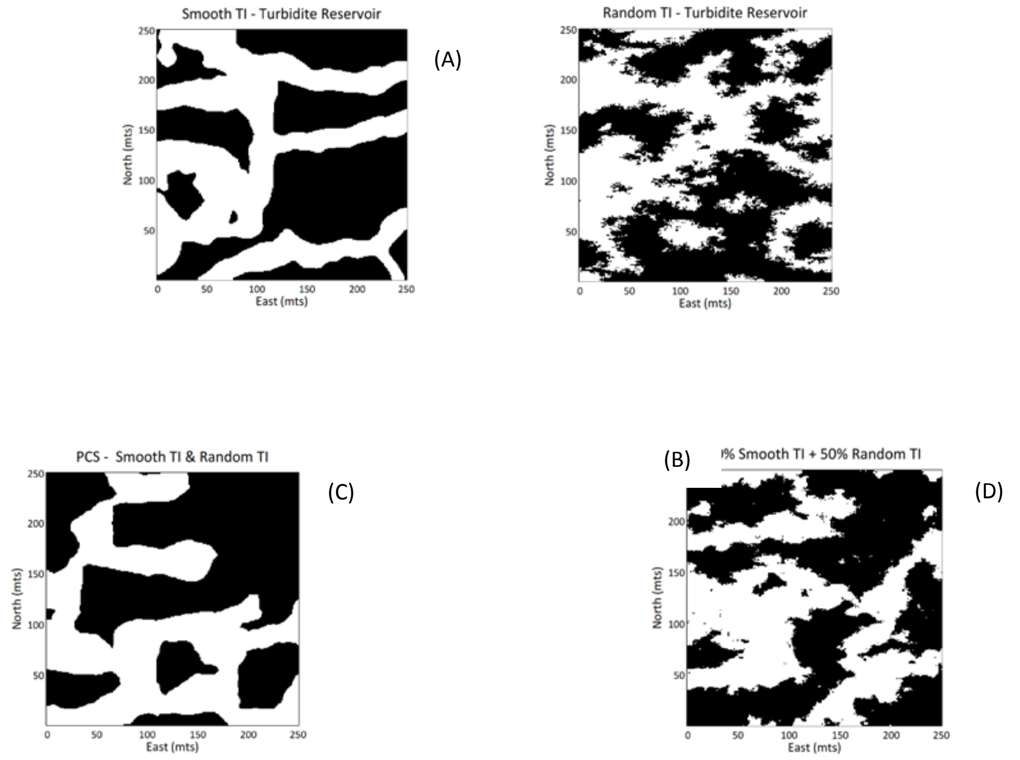


Figure 4: Realizations comparison the linear opinion pool and probability combination scheme for data integration - turbidite reservoir