

Research Note

HISIM: Hierarchical Indicator Simulation

Julián Ortiz C. (jmo1@ualberta.ca)

Department of Civil & Environmental Engineering, University of Alberta

Hierarchical Indicator Simulation is a “new” simulation method that uses an indicator formalism. An initial grid of high values is eroded according to a sequential probabilistic formalism. The procedure works hierarchically. The idea is to fix the nodes that are above a higher threshold, so there is no need to simulate them at subsequent thresholds. There are no order relation deviations. The nodes that are drawn to be below a higher threshold constitute the “universe” for the next (lower) threshold. Only those nodes can now be simulated. The nodes are selected to be below a threshold based on their probabilities derived from indicator cokriging. The simulation switches nodes to the next threshold until the right proportion is achieved. Every time a node is switched, the probabilities are updated.

Some difficulties were found related to variogram reproduction, when the probabilities are updated using indicator kriging. Indicator cokriging should solve this problem. There are some other areas for improvement (1) CPU time requirements, (2) determining a continuous value beyond the indicator class specification.

1 Introduction

Indicator techniques have been widely used because they avoid the multiGaussian assumption that most simulation methods require. They are much easier to apply than simulated annealing, thus most practitioners have it as the only alternative approach to Gaussian simulation.

Sequential Indicator Simulation provides a simulated value at every location, drawn from a local uncertainty distribution, which is calculated by indicator kriging.

Indicator Kriging has the problem of not always satisfying order relations. This leads to the requirement of making *a posteriori* corrections of the conditional cumulative distribution function at every node.

The idea of simulating indicators hierarchically, starting at the highest threshold is to avoid order relation deviations, and (more importantly) to be able to incorporate additional multiple-points statistics.

The algorithm and some examples to illustrate the method are presented in the following sections.

2 HISIM: the algorithm

The idea behind HISIM is to simulate one threshold at a time starting with the highest, and freezing the values above the current threshold, so there is no chance of order relation

deviations when simulating the next threshold. The algorithm assigns a 0 or a 1 at every node and at every threshold.

The simulation starts with all the nodes set to be above the highest threshold. The probability of being above the highest threshold is given by the cdf value at that threshold. This probability value is then assigned to all the nodes. These probabilities are then updated by conditioning data, using Simple Indicator Cokriging, using the adjacent thresholds only. Those probabilities are normalized to construct a cdf that represents the probability of being below the current threshold. Using Monte-Carlo simulation a node is selected to be switched to be below the current threshold (a probability of 1 is assigned to it). The probabilities for the surrounding nodes are then updated by Simple Indicator Cokriging using the indicator values for adjacent thresholds (since the probability at the location selected was switched to 1). Only the nodes that can be switched are then taken into account to update the cdf of probabilities of being below the threshold. Another node is selected via Monte-Carlo simulation. This procedure is repeated until the right proportion of nodes is above the corresponding threshold, i.e. the right proportions of nodes has not been switched.

For the next threshold, all the nodes that were not switched in the previous cutoff cannot be changed since they are greater than the previous threshold and therefore greater than the current one. The same procedure is applied again. This is repeated for all thresholds until the last one is simulated.

At the end, every node falls in an indicator class. If there are n cutoffs, then there are $n+1$ classes. A value within the class could be drawn, assuming some interpolation between classes and extrapolation beyond classes, as is usual in Sequential Indicator Kriging. This has not been implemented yet.

In point form, the algorithm proceeds as follows:

1. Assign to each node the probability of the threshold cdf.
2. Update the probability with the data available, by putting the indicator value at data location, and updating the probabilities in the nearby nodes by Simple Cokriging.
3. Sum the probabilities of all nodes.
4. Draw a uniform random number in (0,1).
5. Visit nodes in order adding the probabilities until the partial sum is greater than the product of the total sum of probabilities and the random number drawn.
6. Switch this node to be below the current threshold, by assigning a probability of 1.0 at the current threshold and allowing it to be switched in the next threshold. This node can not be changed again at this threshold.
7. Update the probabilities of the nearby nodes by Simple Cokriging.
8. Update the total sum of probabilities.
9. Go to 4 and repeat until the right proportion of nodes for the current threshold has been switched.

10. Go to the next threshold. The nodes that were switched in the previous threshold, can be switched again. The nodes that were not switched are above the previous threshold and therefore, above the current one. Repeat steps 1 to 9.

3 First Approach

The original approach considered Simple Indicator Kriging (instead of Cokriging). The algorithm, although similar to the one proposed above, generated an interesting result: the nodes in adjacent classes were, in general, not next to each other in space. As shown in the following examples, for low thresholds, the variogram reproduction was not possible, since once the higher thresholds had been simulated, the probability of having a node switched right next to a node above a higher threshold was lowered by that “conditioning data” whose value was zero. One would expect to have a high probability of having that node above the current threshold, since the node next to it is high valued. This problem should be solved by using the correlation between thresholds, which is not necessary in the case of Sequential Indicator Simulation, since the values are drawn from the complete local conditional cdf.

Simple Indicator Cokriging is then proposed as a method to update the probabilities. This requires the cross variograms for adjacent thresholds, which increases the inference effort from modelling n variograms to $n(n - 1)/2$ variograms and cross variograms. To overcome this excessive inference effort, a program will be developed to calculate based on the Gaussian case, a covariance lookup table with variogram and cross variogram values for the required thresholds. This table will be called by the main program, so there will be no need to use the Linear Model of Coregionalization to model the spatial correlation of the phenomenon.

3.1 A Simple One Dimensional Example

A simple example where the probabilities were updated using Simple Kriging is shown in **Figures 1, 2, and 3**. The idea of this example is to illustrate the method. Although the updated probabilities are not correct, since Cokriging should be used rather than Kriging, the steps are the same and therefore the example is still valid to understand the procedure.

A one dimensional grid with 10 nodes is simulated hierarchically. Only one data value is available, at node 4, with a value 0.98. Three threshold have been considered assuming a uniform distribution between 0 and 1, i.e. the thresholds are equal to their corresponding cdf values: 0.25, 0.50, and 0.75.

Not all the steps are shown, but the most important are there. It can be seen the way the simulation progress, and the updating of the probabilities.

Initializing the nodes as “not simulated”, the probability corresponding to the higher threshold (0.75) is assigned to all of them. Then, the conditioning data are incorporated, and the probabilities are updated in the surrounding nodes (up to the range of correlation). In this example that updating was done by Simple Indicator Kriging, but it should be done using Cokriging.

The sum of the probabilities is calculated and a uniform random number in (0,1) is drawn to pick the node that will be switched to the lower threshold. This node is picked by standardizing all the probabilities by the sum, or equivalently, by comparing the partial

sum of probabilities with the product of the total sum, times the random number generated. Once the partial sum is greater than this product, then this node is switched, by assigning at the current threshold a probability of 1. In the example, the first node to be switched is the node 1. Then, the probabilities have to be updated again in the surrounding nodes, by Cokriging (in this example the updating was done using Kriging). This process is repeated until the right proportion of nodes has been switched, i.e. until 25% of the nodes has not been simulated. Those nodes are set to be above the current threshold. In the example, the nodes 2 and 3 are above 0.75, and therefore their probabilities for lower thresholds are also 0.

The second threshold starts with three conditioning data: the hard data and the two previously simulated nodes that are now fixed. The algorithm proceeds in the same manner: reset all probabilities to the threshold value, update by Cokriging, calculate the sum of probabilities, and find a node to switch by Monte-Carlo simulation. Repeat until the proportion of nodes above the threshold is right.

The last threshold (0.25) is simulated in the same way.

Two examples in two dimensions are presented to show the lost of correlation for low thresholds due to the use of Kriging instead of Cokriging to update the probabilities.

3.2 Example 1: isotropic indicator variograms

A two-dimensional example was performed to see the variogram reproduction. A 50 by 50 grid was simulated unconditionally. 9 thresholds were used, all with the same isotropic variogram model:

$$\gamma(h) = 0.15 + 0.85 \cdot Sph\left(\frac{h}{5.0}\right)$$

Figure 4 shows a map of the simulation. **Figure 5** shows the indicator variograms reproduction. Variograms of the simulated values showed that lower thresholds were simulated with an increasing nugget effect. This is a consequence of updating using Kriging, without accounting for the correlation *between* thresholds, that is inherent to regionalized phenomena. It seems to be a lost of degrees of freedom, while the simulation advances, leaving at the end, no choice to reproduce the variogram. In fact, the higher class appears with the correct continuity. For lower thresholds, the algorithm tries to reproduce the continuity in the space left by the previous threshold, by switching nodes away from the ones that have a probability of 0 at the current threshold, i.e. the nodes fixed by the previous thresholds. Since these probabilities are updated by Kriging, they are closer to 0 than those from nodes far away from previously simulated nodes. Adjacent classes, that is, nodes falling above adjacent thresholds, are uncorrelated with each other, not reproducing the cross correlation that should exist. This generates a purely random distribution for the lowest thresholds.

3.3 Example 2: anisotropic indicator variograms

Another example was done using an anisotropy ratio of two (N30E direction) to one (N120E direction). Ranges are 6 and 3 units respectively. The results are similar. The variogram is not reproduced for low thresholds (**Figures 6 and 7**), because of the updating technique.

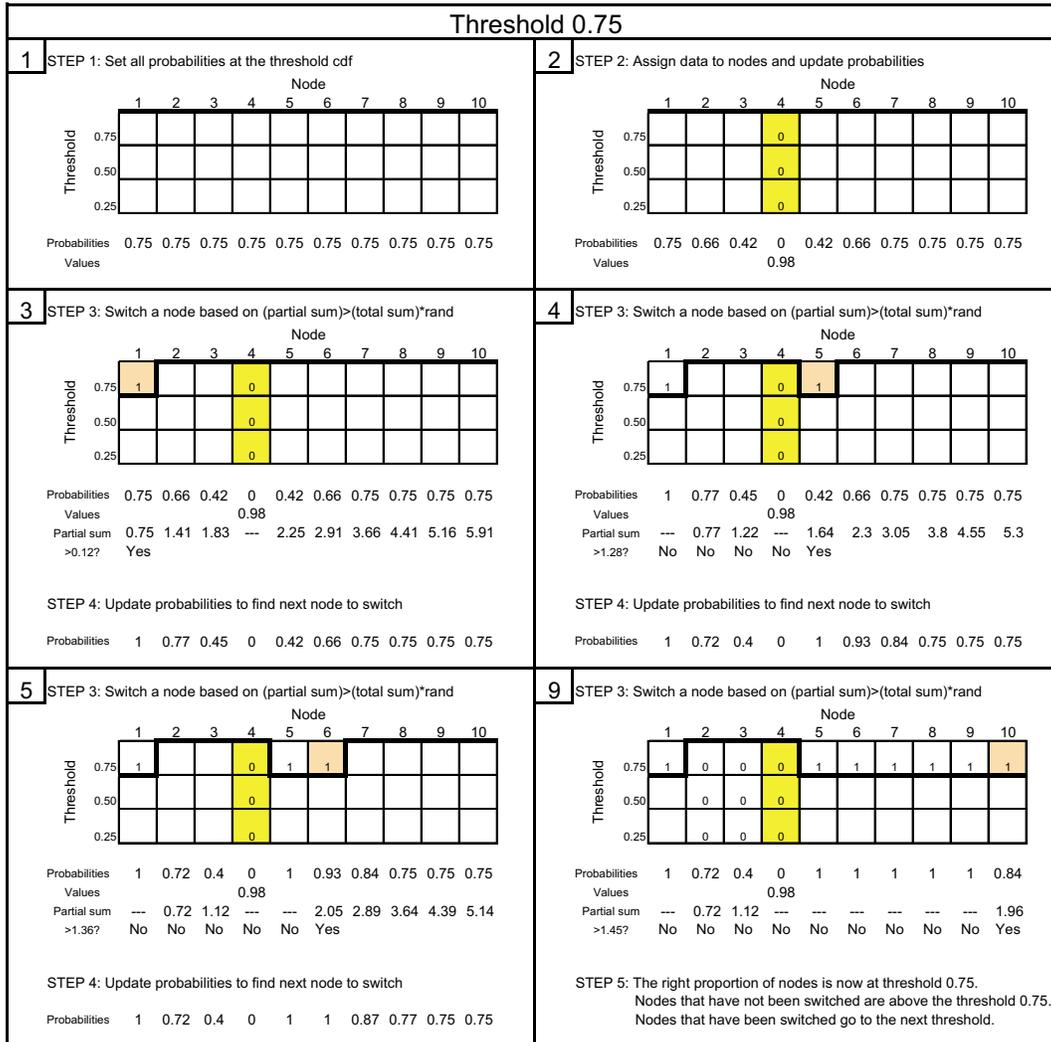


Figure 1: HISIM - Higher threshold: cdf = 0.75

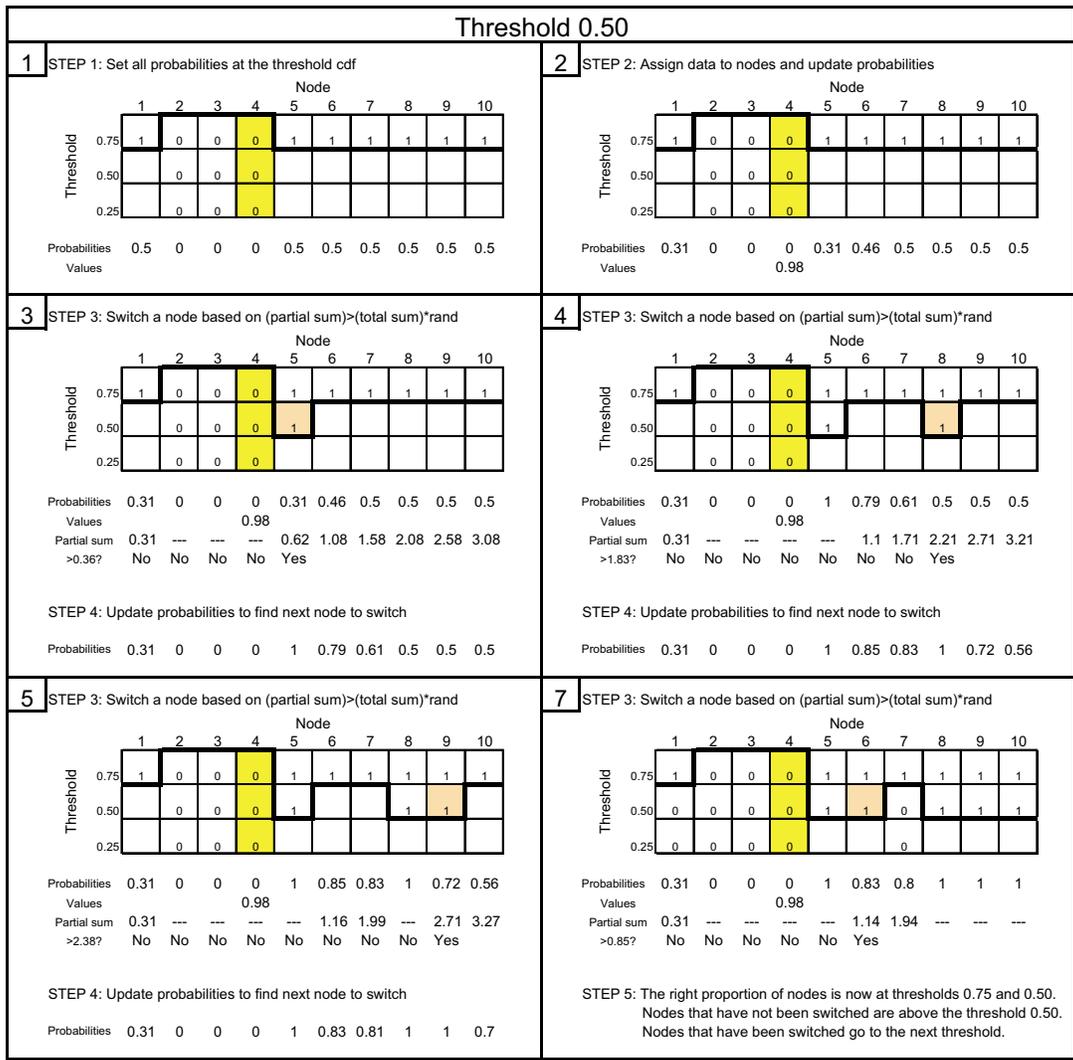


Figure 2: HISIM - Middle threshold: cdf = 0.50

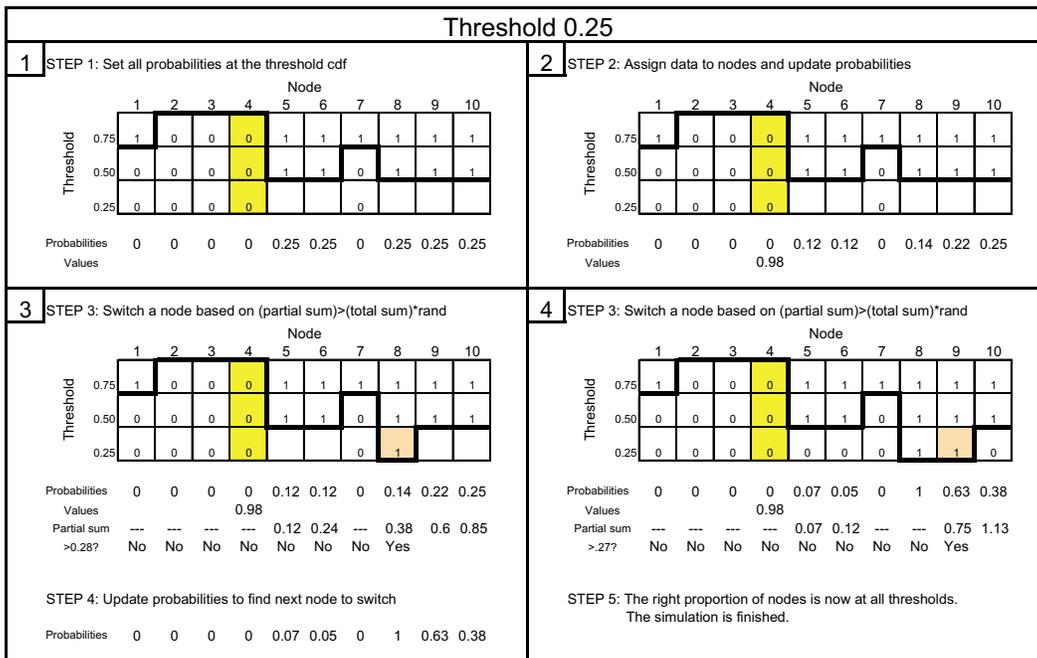


Figure 3: HISIM - Lower threshold: cdf = 0.25

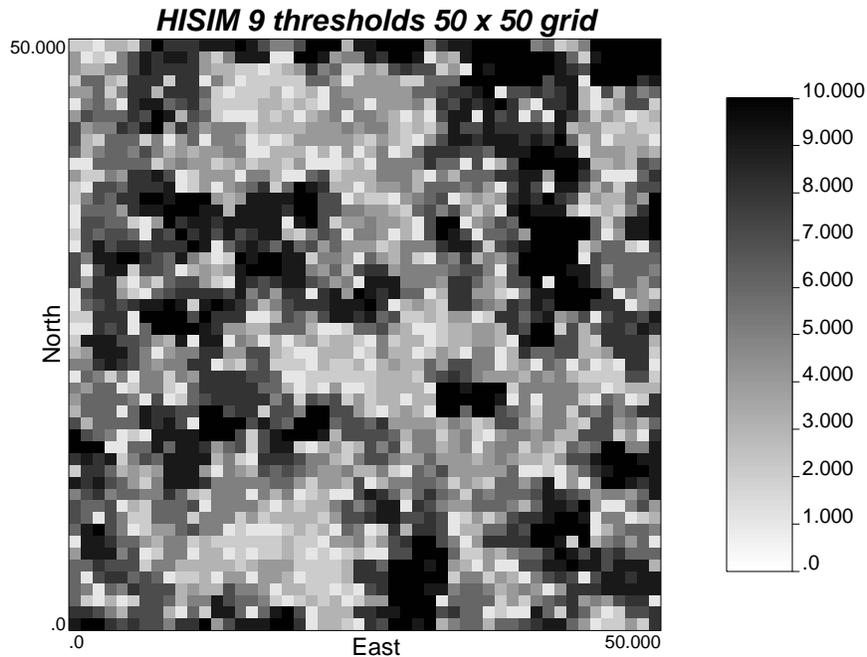


Figure 4: Map of HISIM simulated values in a 50 x 50 units grid, using 9 thresholds, all with the same variogram model (range = 5 units).

4 Discussion

HISIM is an alternative method to do Indicator Simulation. As a first approach, the method was implemented using Kriging to update the probabilities, however this resulted in a loss of structure at low thresholds because the cross correlation between thresholds was not used as input in the algorithm. This problem can be solved by using Cokriging to update the probabilities. This requires the cross variograms between adjacent threshold, which will be calculated in a cross covariance lookup table, instead of using a Linear Model of Coregionalization to model them.

CPU time is a concern, since the greater the range of the variogram, the longer it takes the simulate a grid, since more updating is required for every simulated point. this should be addressed once the algorithm itself work properly. Conditioning data are not a problem. They can be used without interfering the algorithm.

The algorithm promises to be modified to incorporate multiple point statistics in the form of *runs*, i.e. high order indicator covariances, which is the final goal of this research.

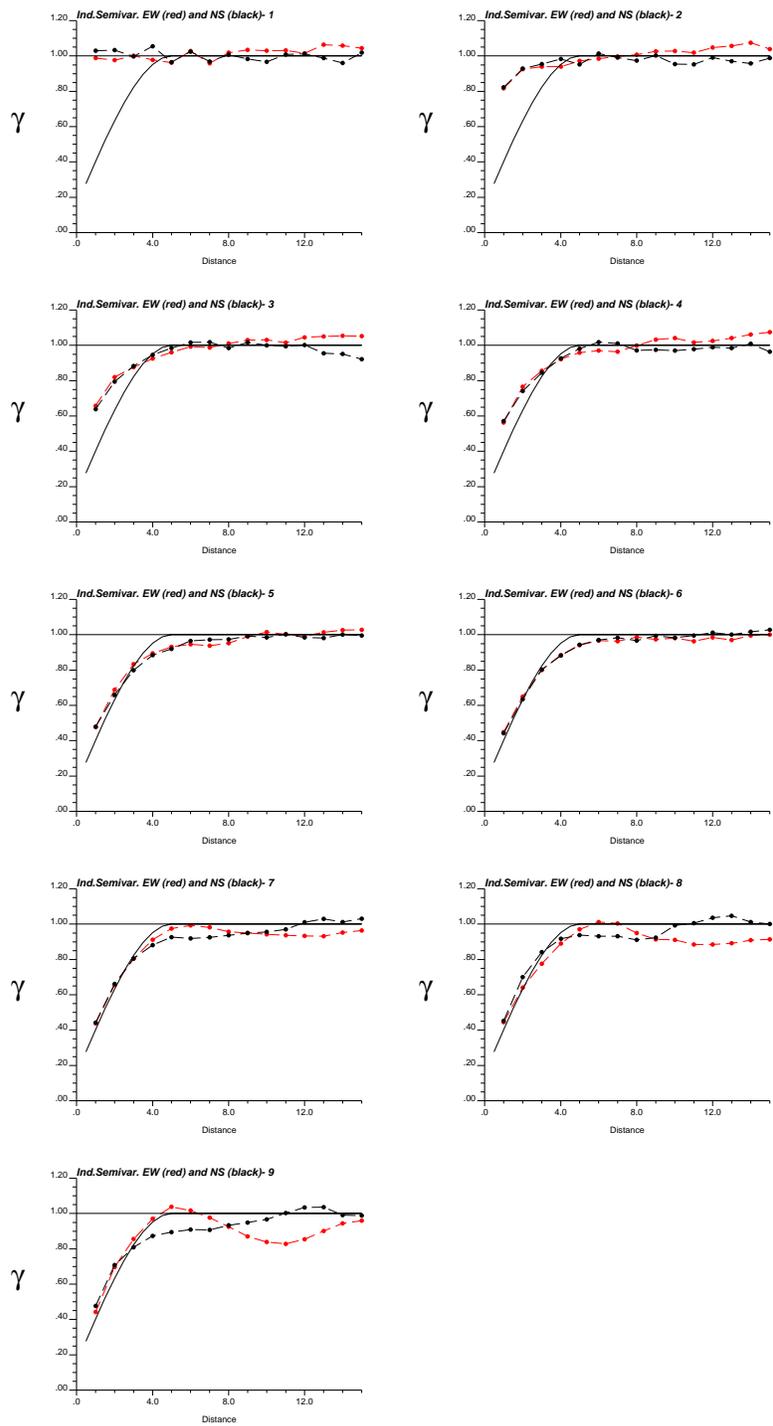


Figure 5: HISIM - Variogram reproduction for 9 threshold.

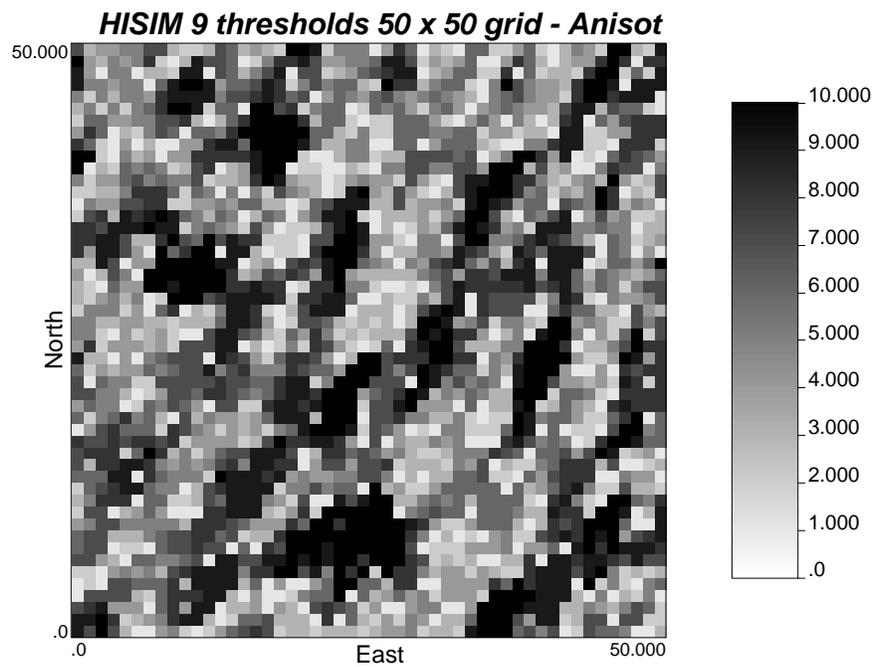


Figure 6: Map of HISIM simulated values in a 50 x 50 units grid, using 9 thresholds, variogram with anisotropy ratio (ranges = 6 and 3 units).

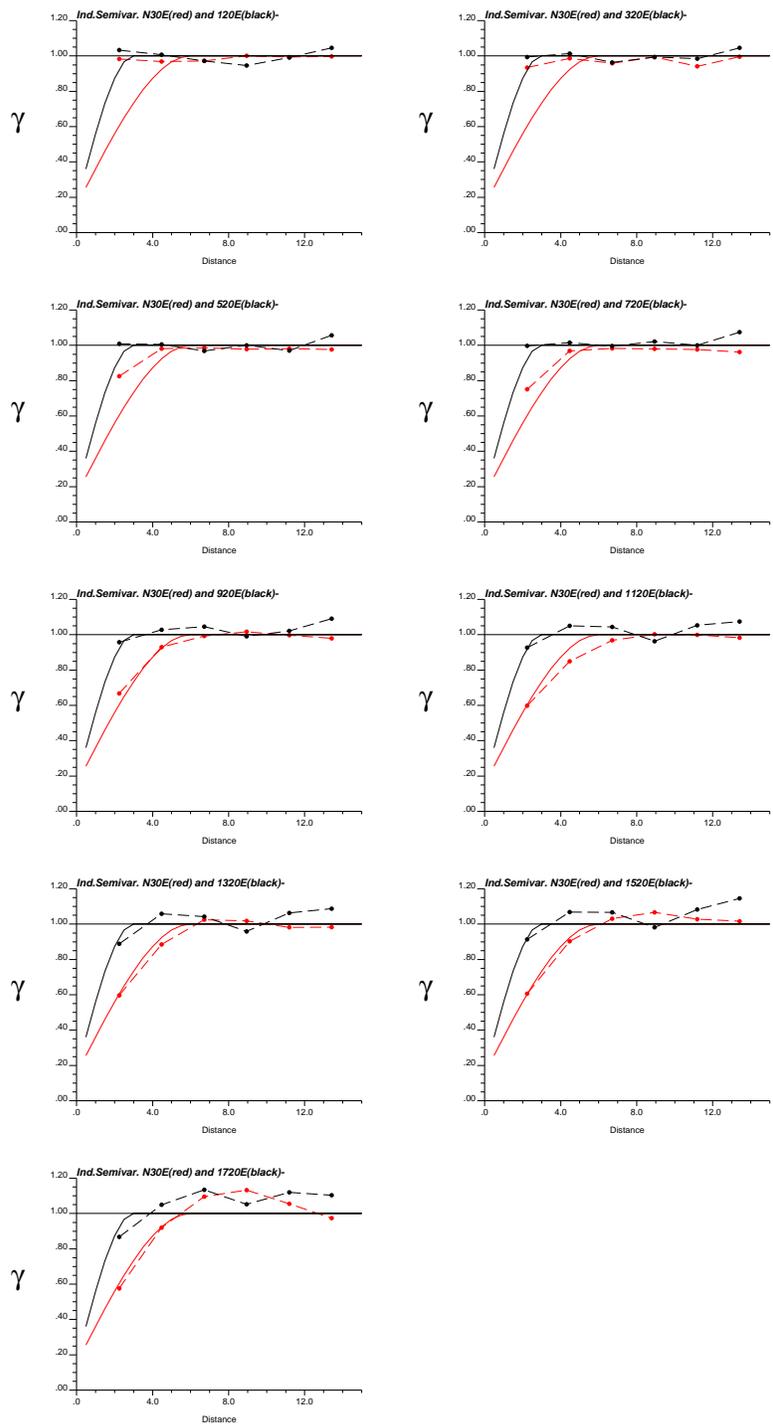


Figure 7: HISIM - Variogram reproduction for 9 threshold and anisotropy.