Parallel implementation of multiple-point simulation based on texture synthesis

Álvaro Parra and Julián M. Ortiz

Abstract Multiple-point simulation algorithms based on texture synthesis allow the reproduction of complex structures such as hierarchical categories, long connectivities and curvilinear features, honouring conditioning data. However, the use of large patterns has a deeply negative impact on the efficiency of these kinds of methodologies. This work presents a parallelization strategy and implementation at path level for multiple-point simulation based on texture synthesis. The developed implementation relies on the fixed path defined by a unilateral simulation strategy, performing assignations at several points of the realization domain concurrently. The parallelization methodology is tested over several template configurations and values for concurrent points being concurrently simulated. Speedups over one are always obtained using square templates of side at least 7 and speedups over 7 are reached using square templates of side 19.

1 Introduction

The reproduction of geological phenomena is relevant in the Earth Sciences. In particular, the spatial continuity of categorical data is an important issue since many geological processes depend on the connectivity of rock types, facies and soil. Geostatistical simulation algorithms aim to reproduce the spatial correlation of the studied variables and the relationships among them, honouring conditioning data. Examples where the study of categorical variables is important for projects are the change in flow properties depending on the spatial arrangement of sand and shale, and the continuity of a winding vein of gold ore [3].

Álvaro Parra
ALGES Laboratory, Advanced Mining Technology Center (AMTC), University of Chile, Av. Tupper 2069, Santiago, Chile, e-mail: aparra@dcc.uchile.cl

Julián M. Ortiz
Department of Mining Engineering, University of Chile, e-mail: jortiz@ing.uchile.cl

Ninth International Geostatistics Congress, Oslo, Norway, June 11. – 15., 2012

Multiple-point algorithms aim to reproduce complex structures, such as large continuities or curvilinear patterns, which are hard to characterize solely on the relations between pairs of points [2, 4].

A parallelization strategy for multiple-point simulation based on texture synthesis, proposed in [3], is developed in this work. This strategy is based on the fixed path defined for an unilateral simulation. Then, points at different rows can be simulated concurrently using a fixed pattern. The empirical speedup was measured varying the maximum amount of concurrent assignations as well as the side of square templates.

2 Multiple-point simulation based on texture synthesis

The algorithm proposed in [3] consists in visiting each node of the realization domain following a sequential unilateral path. The conditioning is achieved by extending the causal template used by Wei [5] to a square template that incorporates a non-causal region.

The algorithm starts by creating a data base of patterns retrieved from a training image. Then, the set of conditioning points are assigned to an empty grid which is filled following the unilateral path. If the visited point does not have an assigned value, neighbouring data is retrieved using a fixed template. A conditional probability distribution function for the visited point is inferred from the database using both previously simulated values and possible conditioning data captured for the non-causal region of the template.

3 Parallelization methodology

The parallelization methodology consists in simulating concurrently multiple grid positions. As a result of employing a fixed unilateral path, a node to be simulated only depends on the previously assigned points retrieved by the causal region of the template, and the possible conditioning samples scanned by the non-causal template. As illustrated by fig. 1, after reaching a suitable number l of simulated points, it is possible to start concurrently simulating points of an upper row honouring the causality of the algorithm. l is determined by the causal region of the template. As a consequence, for a square template of side a it is necessary to have at least $\lfloor a/2 \rfloor + 1$ simulated points in the $\lfloor a/2 \rfloor$ previous rows before starting to simulate a new row.

The implemented methodology ensures that the previous $\lfloor a/2 \rfloor$ rows have at least $\lfloor a/2 \rfloor + 1$ assigned values. Then, each row is simulated concurrently, though it is necessary to ensure that each concurrent row has made a simulation before assigning a new point. In addition, a maximum number of M rows being simulated concurrently is defined which defines the theoretical speedup.

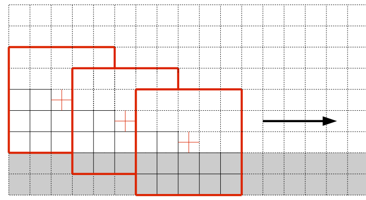
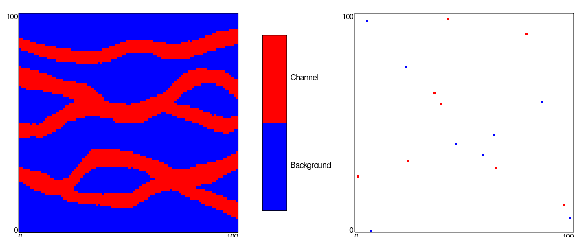


Fig. 1: Illustration of the concurrent simulation strategy.

4 Results

In order to measure the speedup of the parallel algorithm, both, conditional and non conditional simulations were tested. All tests were performed in one node of the ALGES[1] cluster (IBM BladeCenter 8852), which has two CPUs Intel Xeon Quad Core (8 cores) and 8 GB of RAM. For conditional and non conditional tests we use the training image illustrated in fig. 2a of 100×100 pixels. In the case of the conditional simulations, a set of 16 random points (illustrated in fig. 2b, equally divided in two categories) are computed as conditioning data.



(a) Training image (b) Conditioning

Fig. 2: Algorithm inputs. (a) Training image of 100×100 pixels and (b) conditioning points.

The empirical speedup was measured for configurations (M,a) where M is the maximum number of rows being concurrently simulated and a is the side of the square template. M was set for values in $\{1,2,4,6,8,12.16\}$ and a for values in $\{3,5,7,9,11,13,15,19\}$. The time for each configuration was averaged over ten realizations.

The Speedups of the unconditional and conditional simulations are reported in the table 1. The maximum speedup of 7.035 was obtained for the unconditional case using the configuration $(a = 19, M = 16)$. However, for the conditional case the maximum speedup of 7.004 was reached using the $(a = 19, M = 8)$ configura-

[1] Advanced Laboratory for Geostatistical Supercomputing, www.alges.cl

Table 1: Empirical speedups.

(a) Unconditional simulations

a	M: 2	4	6	8	12	16
3	0.289	0.358	0.384	0.420	0.443	0.453
5	0.548	0.618	0.707	0.743	0.822	0.846
7	0.955	1.178	1.339	1.435	1.676	1.891
9	1.251	1.606	1.912	2.225	2.786	3.253
11	1.275	1.914	2.306	2.770	3.593	4.348
13	1.009	1.338	1.797	2.403	3.109	2.951
15	1.506	2.997	3.530	4.360	4.745	5.014
17	1.656	4.951	4.962	5.296	4.986	5.595
19	2.343	3.494	7.036	6.974	6.983	7.011

(b) Conditional simulations

a	M: 2	4	6	8	12	16
3	0.338	0.408	0.466	0.463	0.531	0.523
5	0.567	0.664	0.735	0.774	0.852	0.914
7	1.059	1.323	1.463	1.573	1.945	2.075
9	1.241	1.662	1.952	2.164	2.681	3.148
11	1.350	1.926	2.386	2.748	3.529	4.170
13	1.000	1.316	1.752	2.171	2.684	2.537
15	1.494	2.971	3.299	4.329	4.501	4.238
17	1.663	4.982	5.014	4.971	5.013	5.348
19	1.746	3.490	6.982	7.004	6.964	7.003

tion, although similar values were obtained for $M = 12$ and $M = 16$. The empirical speedup approximates the theoretical speedup for high dimension patterns. This fact can be explained by the overhead involved in using a multi-thread implementation as well as the synchronization mechanism.

5 Conclusions

A parallelization algorithm was proposed for multiple-point simulation based on texture synthesis. The developed methodology performs multiple independent assignations without employing a complex synchronization mechanism. The developed strategy permits the reduction of at least half of the computational time for square templates of side of at least 7 pixels and over 7 times for conditional and non conditional simulations using 8 cores and concurrently simulating a maximum of 16 rows.

References

1. Daly, C.: Higher order models using entropy, Markov random fields and sequential simulation. Geostatistics Banff 2004 pp. 215–224 (2005)
2. Guardiano, F.B., Srivastava, R.M.: Multivariate geostatistics: Beyond bivariate moments. Geostatistics-Troia **1**, 133–144 (1993)
3. Parra, A., Ortiz, J.: Adapting a texture synthesis algorithm for conditional multiple point geostatistical simulation. Stochastic Environmental Research and Risk Assessment **25**, 1101–1111 (2011). URL http://dx.doi.org/10.1007/s00477-011-0489-1. 10.1007/s00477-011-0489-1
4. Strebelle, S.: Conditional simulation of complex geological structures using multiple-point statistics. Mathematical Geology pp. 1–21 (2002)
5. Wei, L.: Texture synthesis by fixed neighborhood searching. Ph.D. thesis, Stanford University (2001)