ALGES
ADVANCED LABORATORY FOR GEOSTATISTICAL SUPERCOMPUTING

amtc
ADVANCED MINING TECHNOLOGY CENTER

fcfm
Ingeniería de Minas
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS UNIVERSIDAD DE CHILE

Improving an efficient simulation of multidimensional Gaussian random fields with GPU

Daniel Baeza
Julián Ortiz
Exequiel Sepúlveda

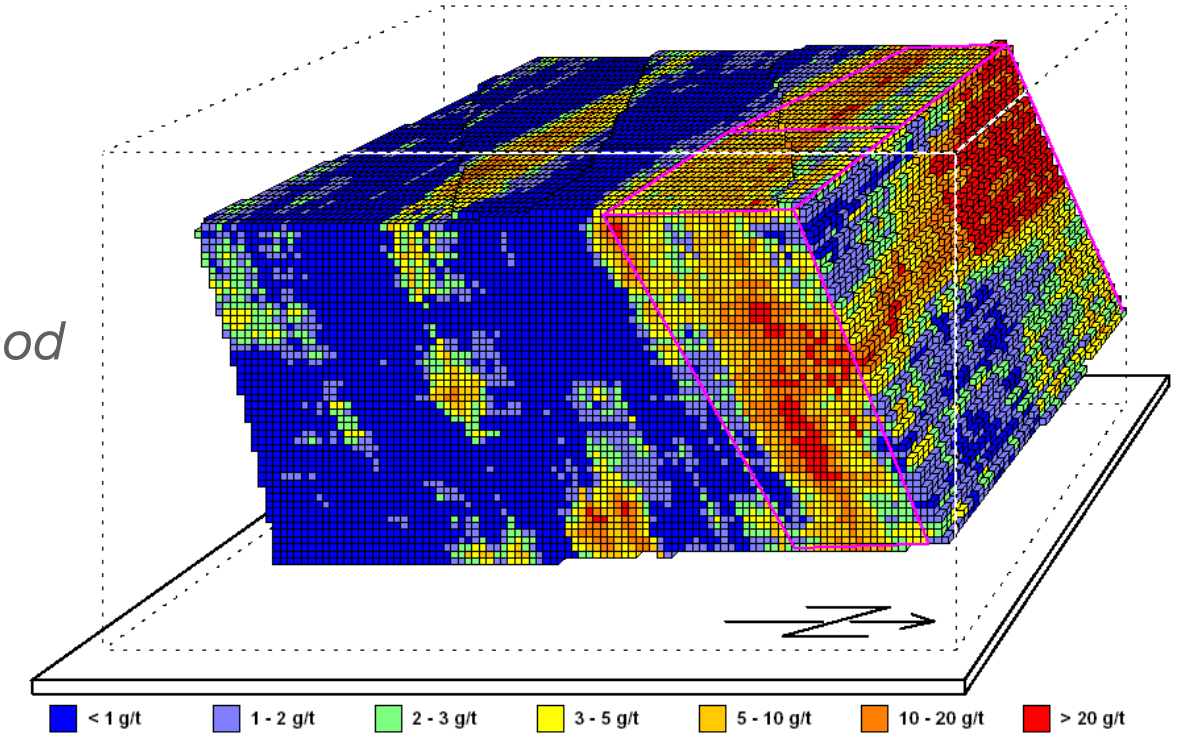
Resume

- Context
- Turning Bands Method
- The timeout problem and our proposal
- Implementation details
- Performance
- Example
- Conclusion

Context

Resource Assesment

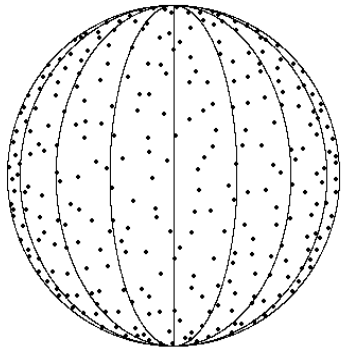
- Estimation
  Kriging
- Simulation
  Turning Bands Method

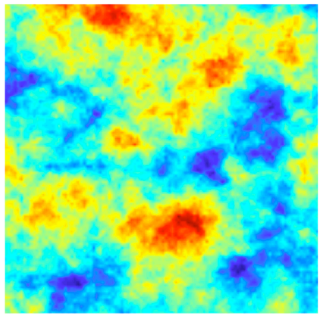


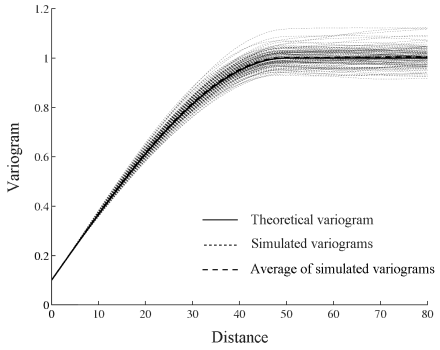
Turning Bands Method

Turning bands simplifies the complexity of the simulation with a dimensional reduction, by computing simulated values over lines, that is reducing the problem to one-dimensional simulations, and projecting them to the three-dimensional space, thereby allowing the use of very efficient simulation methods to obtain the one-dimensional simulations.

$$C_1(r) = \frac{d}{dr}[r\,C_3(r)]$$







Turning Bands Method

The method Proceeds in the following fashion:

1. Simulated values X_i are generated over lines randomly oriented over the unit sphere, with covariance function C_x
2. For every location on the three dimensional simulated grid, the simulated values generated over the line are projected orthogonally into the simulated location.
3. The final simulated value is obtained as the projection of values frommany lines:

$$Y(x) = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} X_i(< x, \mathbf{U}_i >),$$

where U_i are lines randomly distributed over the sphere S_3 , X_i is a random value with covariance function C_x and x is the location where we want to simulate.

4. Conditioning to the actual samples is done as a postprocess through the method of kriging residuals. Let $\{Y_s(x), x \epsilon \mathbf{R}^3\}$ be a non-conditional simulation. The random defined by

$$\forall x \epsilon \mathbf{R}^3,\ Y_{CS}(x) = Y_S(x) + [Y(x) - Y_S(x)]^{SK}$$

represent the conditional simulation that reproduce the distribution of $\{Y(x), x \epsilon \mathbf{R}^3\}$ conditional to the Y-data.

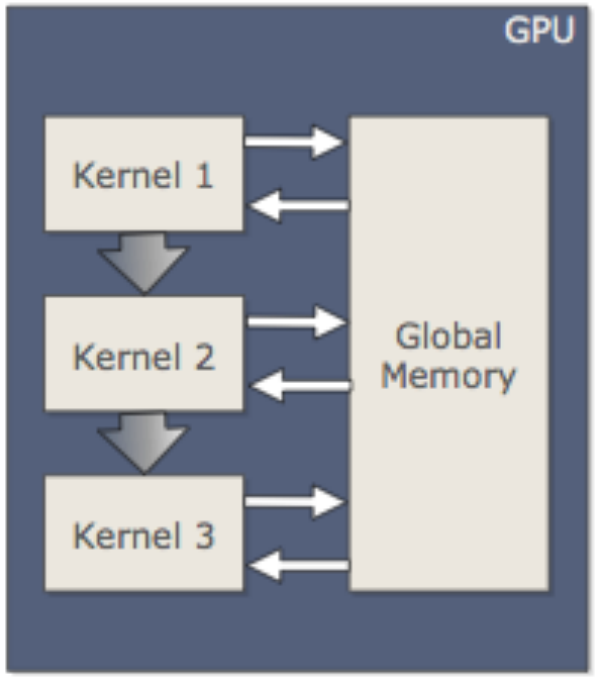
ALGES
ADVANCED LABORATORY FOR GEOSTATISTICAL SUPERCOMPUTING

The timeout problem and our proposal

- The simulation size is too large --> this method has a low performance within CPU
- Low performance implies very long processing times.
- For large resources models, computation times may be impractical.
- We use the GPU to parallelize the simulation process in the Turning Bands Method and thereby reduce the timeout.

Implementation details

- 90% of processing time in GPU is devoted to simulation computations and the other 10% is related to data transfer between GPU and CPU.
- Processing Power over 400.000 Grid Points per Second
- Unlimited grid size simulation
- The parallel algorithm involves over 98% of the computational cost involved in the serial implementation.
- There is only two data transfer between CPU and GPU memory.



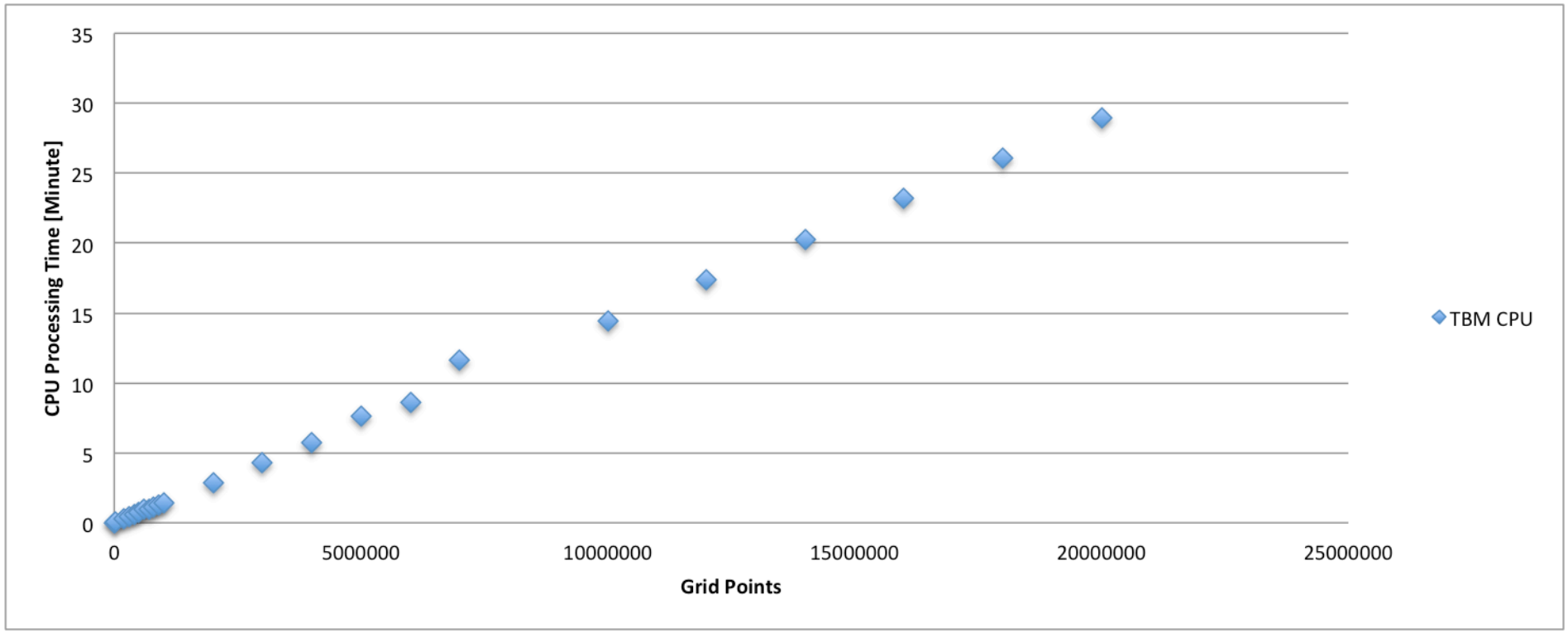
Performance

	CPU	GPU
Model Name	Intel Xeon E5405	Tesla C2050
Number of Cores	4	448
Frequency of Cores	2.67GHz	1.15GHz
RAM memory	4GB	3GB

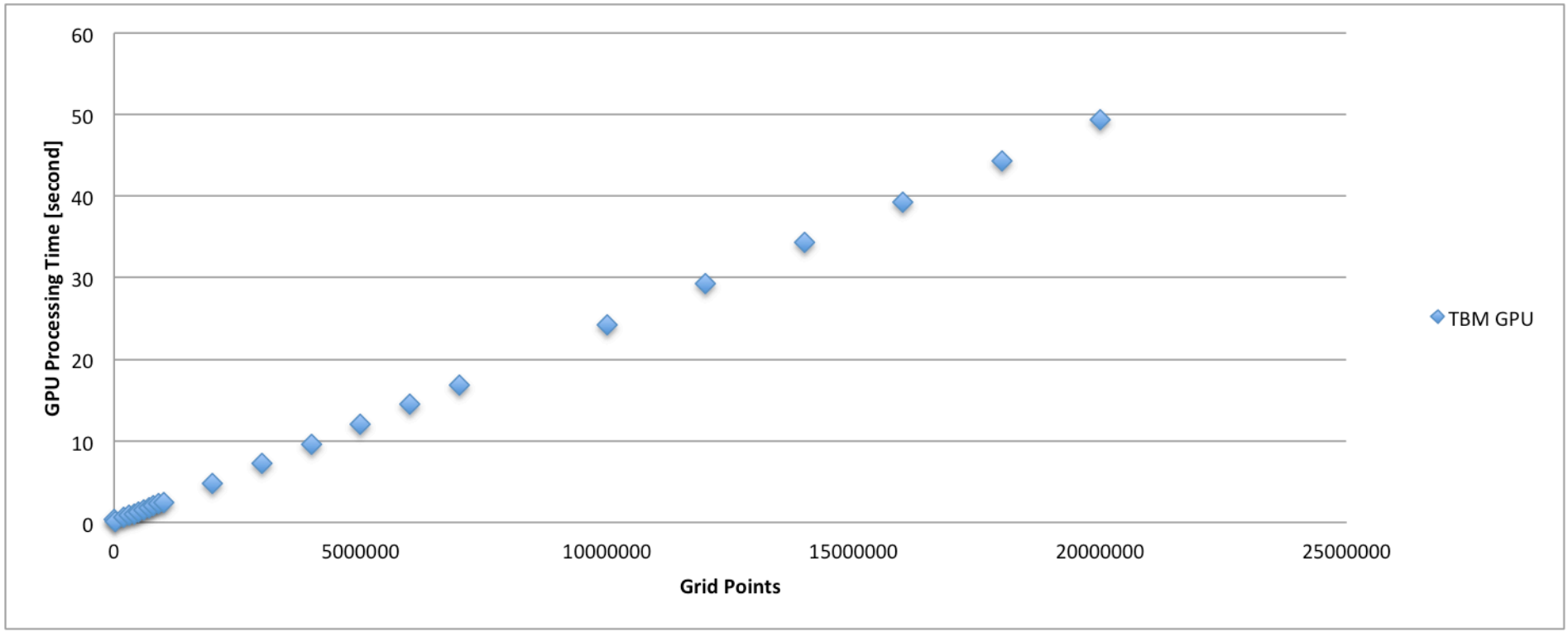
Serial and Parallel execution times are measured for runs of the implementations under different settings:

- Varying the simulated grid size from 1000 to 20 000 000 nodes.
- Running 10 to 100 realizations

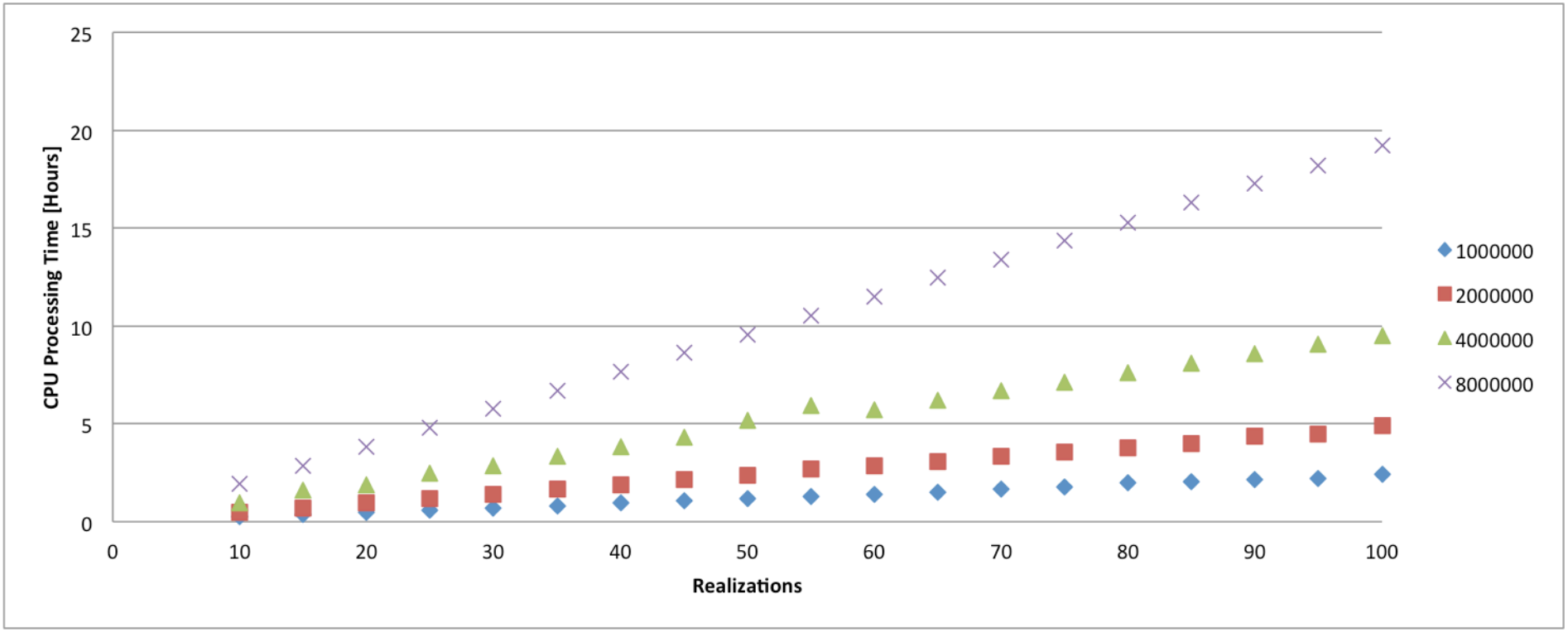
Performance
CPU vs Realizations (Serial Implementation)



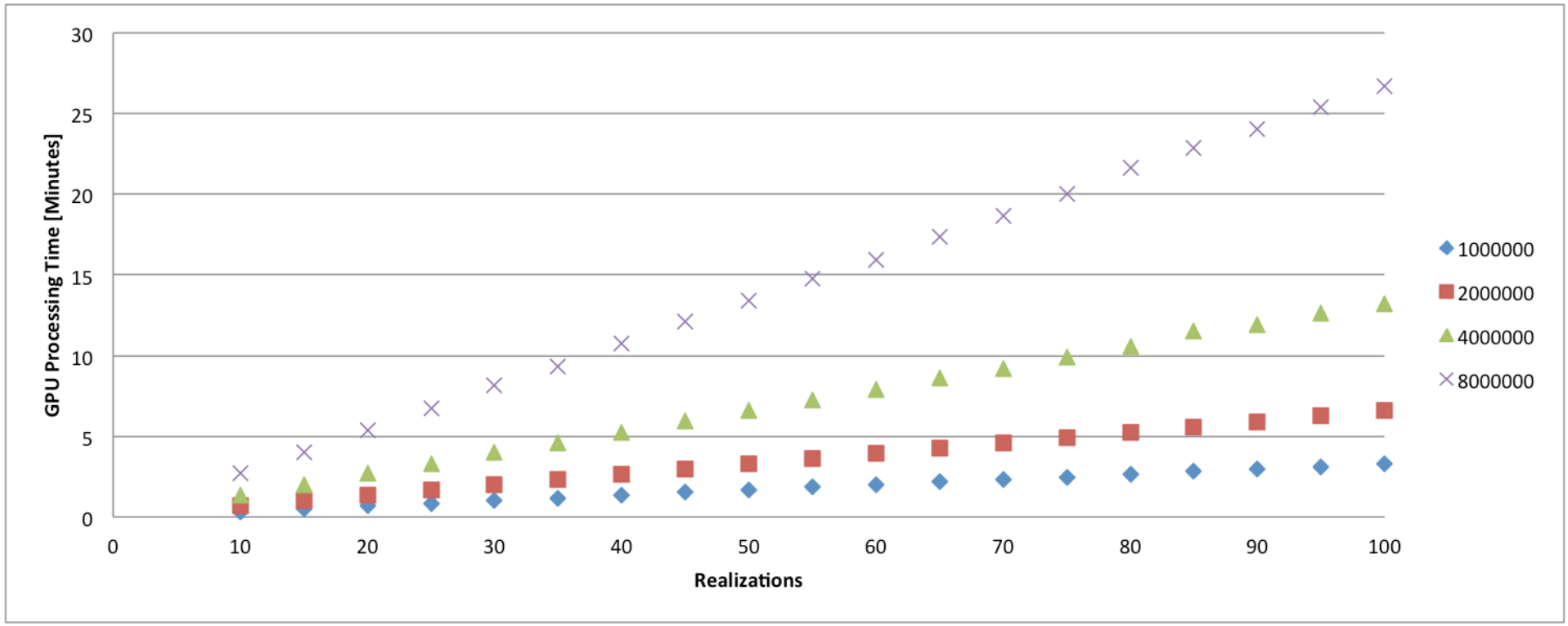
Performance
GPU vs Grid Points (Parallel implementation)



Performance
CPU vs Realizations (Serial Implementation)



Performance
GPU vs Realizations (Parallel implementation)



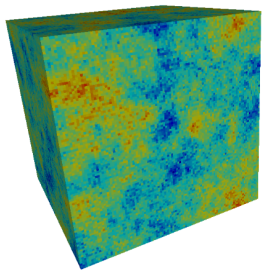
Summary of Performance

SpeedUp vs	Grid Points	Realizations
Average	**36,4x**	**43,2x**
Standard Deviation	2,1x	1,3x

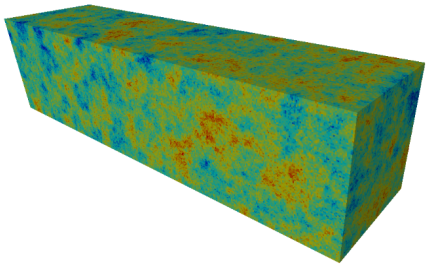
Processing Power	CPU	GPU
Average	11.560	**400.061**
Standard Deviation	518	17.171

Example

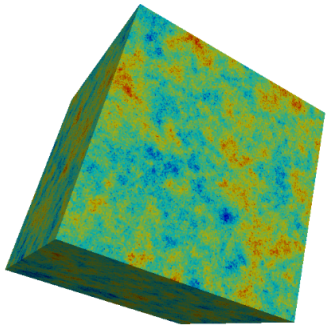
- Three different grid sizes were simulated and conditionated with a real data base.
- The data base contains 2375 samples in a copper deposit to condition the simulation process.



N=1000000



N=4000000



N=8000000

$realizations = 100$	**CPU Time** [hour]	**GPU Time** [minute]	**SpeedUp**
$N = 1000000$	2,5	6,8	21,5x
$N = 4000000$	9,8	20,8	28,2x
$N = 8000000$	19,5	34,8	33,6x

ALGES
ADVANCED LABORATORY FOR GEOSTATISTICAL SUPERCOMPUTING

Conclusions

- We have shown the application of a parallelized implementation of the turning bands simulation method using GPU.
- The optimum speedup is achieved when a large number of realizations is required.
- In a practical application, speedup reached **33x.**

- Comparisons were also made against public domain code (sgsim from GSLIB); speedup in this case reached ***100x*** (not shown here).