

SIMEA: An advanced framework for random media simulation

THIBAUD NION¹, DOMINIQUE JEULIN² and
GABRIEL FRICOUT¹

¹ Arcelor Research S.A. (ARSA), France
thibaud.nion@ensmp.fr, gabriel.fricout@arcelor.com

² Centre de Morphologie Mathématique (CMM), France
dominique.jeulin@ensmp.fr

1. Introduction

Simulation of random media is of a crucial interest and has for instance been at the heart of long standing joint research projects undertaken by Arcelor Research S.A. (Metz, France) and the Centre de Morphologie Mathématique (Mines Paris, Fontainebleau, France).

We present here SIMEA, a programming library for discrete event simulations, which is partly an offspring of those collaborations. Targeted at the development of primary grains models, its main purpose is to make it possible for researchers and engineers to develop innovative models within a robust and flexible framework.

2. A new programming framework

2.1 A glorious lineage

First of all, our library owes its name to an older set of routines developed at the CMM, with Micromorph@[11] software. This older set of routines proved to be quite useful in past studies [3, 4, 6] and still is a good basis for teaching purposes [5].

However this first framework hardly matches the needs of today’s industrial research. Our new library thus intends to be more in phase with current computing techniques.

2.2 Underpinning technologies

The first objective in building a sane framework for discrete event simulation is to ensure that all the “elements” participating in the simulation will be able to interact. This is easily ensured by using object-oriented programming language. One of these is Bjarne Stroustrup’s C++ [9] which is a very powerful language, an alleged industry standard and was chosen for the implementation of SIMEA.

The definite impulse for the renovation of SIMEA’s base code, was the maturation of the CMM’s image processing library, Morph-M [12], which leverages the power of object oriented and

generic programming. SIMEA is built in C++ over Morph-M and the peer-reviewed Boost library [10].

Though SIMEA already benefits from the extensive morphological tools implemented in Morph-M, it is also possible, for instance, to add physical simulation modules and gain the ability to easily mix primary grains models with physical models¹.

Each implemented function or object is thoroughly tested through a battery of unit-tests. This makes SIMEA a sane basis to build reliable industrial tools on. Moreover it borrowed from Morph-M the ability to prototype new models quickly, thanks to its Python interface which makes it very easy to test new ideas.

3. First realisations

3.1 Primary grain models

We will now illustrate the techniques described in previous sections. Let’s consider a very generic description of primary grains based models operating over a space (usually an image) that we will call a *simulation field*.

These models are usually decomposed into a few elementary components that belong to three distinct categories [1, 2, 5]: primary grains – each defined by a geometrical shape and a “color”–, point processes of space where grains will “appear”, and the operation – supremum, addition, etc – describing the interaction between the grains and the simulation field.

The translation of this basic scheme into object-oriented concepts is straightforward. For each category described above, we define a *class* of objects. These classes’ sole purpose is to define *interfaces* through which an element taken from one of the categories may collaborate with elements taken from the others.

3.2 Applications and perspectives

SIMEA provides a very economic way to implement all the classical models, but also the ability to extend them at nearly no cost. For instance once we have implemented the basic set of interactions and the most common point processes, we only have to implement a new family of grains, associate it with al-

¹something that formerly had to be written from scratch on a case by case basis

ready implemented point processes and interactions, and we get a brand new model.

This is the way we can simulate the surface roughness of painted steel sheets with Arcelor Research S.A. We use a Poisson point process and the interaction by addition of grains (here paint droplets) corresponding to the dilution models, and we concentrate on designing new grains. This is illustrated on figure 1, where we simulate paint deposition on steel sheets. The grains are cylinders for which we intend to fit the distributions of the height and the base's radius according to available experimental data.

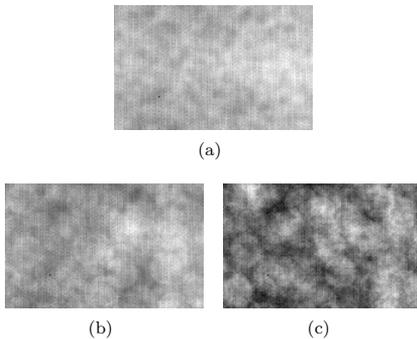


Figure 1. An attempt to simulate paint deposition with a dilution model of droplets modelled as cylinders. (a) the original topography. (b) a first layer of droplets (density of the Poisson point process: 0.005). (c) a second layer with the same density of droplets.

Such extensions of the models, though already quite useful in practice, are nearly immediate to perform with our new framework. But we also have an extensive freedom in the way we define the grain's shape, and we might think of using such an exotic grain as the amoeba structuring elements designed by Romain Lerallut [7] with Morph-M².

Another interesting feature is the ability to work with a large panel of data, thanks again to the mechanisms implemented in Morph-M and described by Raffi Enficiaud in his thesis [8]. The simulation fields we can work with can be made of nearly any data type you can think of – may it be integers, floats, complex numbers, 3D or 4D grains, or multichannel pixels. We have for instance easily produced coloured random models (see figure 2) right from the beginning of SIMEA's renewal, whereas this hadn't been done in years.

²Note that Morph-M was referred to as *Morphée* during its early development

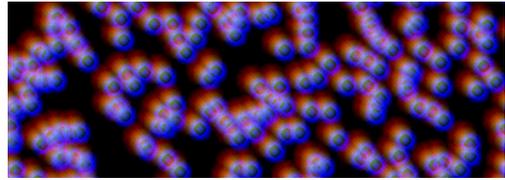


Figure 2. Colored nuclei: a simple colored boolean model with a unique grain made of three (red, green, blue) shifted cones gives this interesting structure.

4. Conclusion

We have presented here a new tool that offers a lot of flexibility and makes it possible to design innovative models of random media. Designed to match high quality standards, this framework is also a sane basis to build broader applications. Still in development, this simulation framework is already being used in an industrial research project.

References

- [1] G. Matheron, *Random Set And Integral Geometry*, J. Wiley.
- [2] J. Serra, *Mathematical Morphology and Image Analysis*, Academic Press.
- [3] Dominique Jeulin and Pascal Laurence, *Probabilistic model of rough surfaces obtained by electro-erosion*, in *Proc. ISMM'96 Conference, Atlanta, USA* (R.W. Schafer P. Maragos M.A. Butt, ed.), Kluwer Academic Publishers, 1996.
- [4] Pascal Laurence, *Analyse et modélisation du transfert de rugosité*, Thèse ENSMP-IRSID, 1996.
- [5] Dominique Jeulin, *Models of random structures*, École des Mines de Paris, 1998.
- [6] L. Decker, L. Savary, and D. Jeulin, *Models of random structures with Primary grains, Simulation with Micromorph, C-/03/MM*, Paris School of Mines Publication.
- [7] Romain Lerallut, *Modélisation et interprétation d'images à l'aide de graphes*, Centre de Morphologie Mathématique, ENSMP, 2006.
- [8] Raffi Enficiaud, *Algorithmes multidimensionnels et multi-spectraux en morphologie mathématique: approche par la méta-programmation*, Centre de Morphologie Mathématique, ENSMP, 2007.
- [9] Bjarne Stroustrup, *The C++ Programming Language.*, Addison-Wesley, Reading, MA., 1986. ISBN 0-201-12078.
- [10] *Boost Libraries*. <http://www.boost.org/>.
- [11] CMM/Ecole des Mines/ARMINES/TRANSVALOR, *Micromorph software*, 1996. <<http://cmm.ensmp.fr/Micromorph>>.
- [12] ———, *Morph-M: morphological image processing library*, 2007. <<http://cmm.ensmp.fr/Morph-M>>.