Random media generation

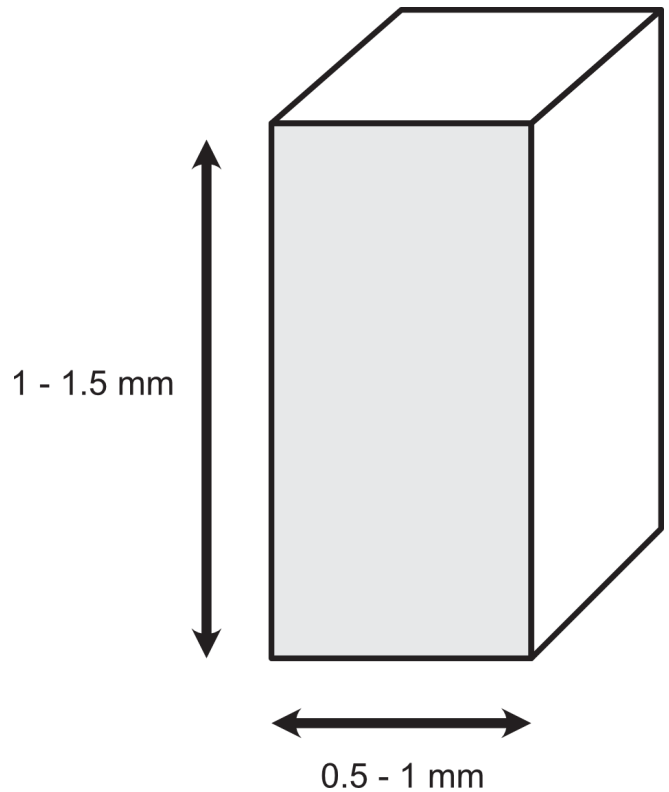
16/05/25

Introduction

Generate a random media :

- Rectangular cuboid
- Center (x, y, z)
- Size (s_x, s_y, s_z) with s_x and $s_y \in [0.5, 1]$ mm and $s_z \in [1, 1.5]$ mm
- Rotation angle $(\alpha_x, \alpha_y, \alpha_z)$

In a rectangular cuboid with fibres.



Grain model

Metropolis algorithm

Algorithm 1

1: **for** each grain g **do**
2: Determine the energy of the system E_i
3: Randomly move/rotate g
4: Determine the energy of the system E_f
5: **if** $\exp(-\beta(E_f - E_i)) < \eta$ **then**
6: Revert the move
7: **end if**
8: **end for**

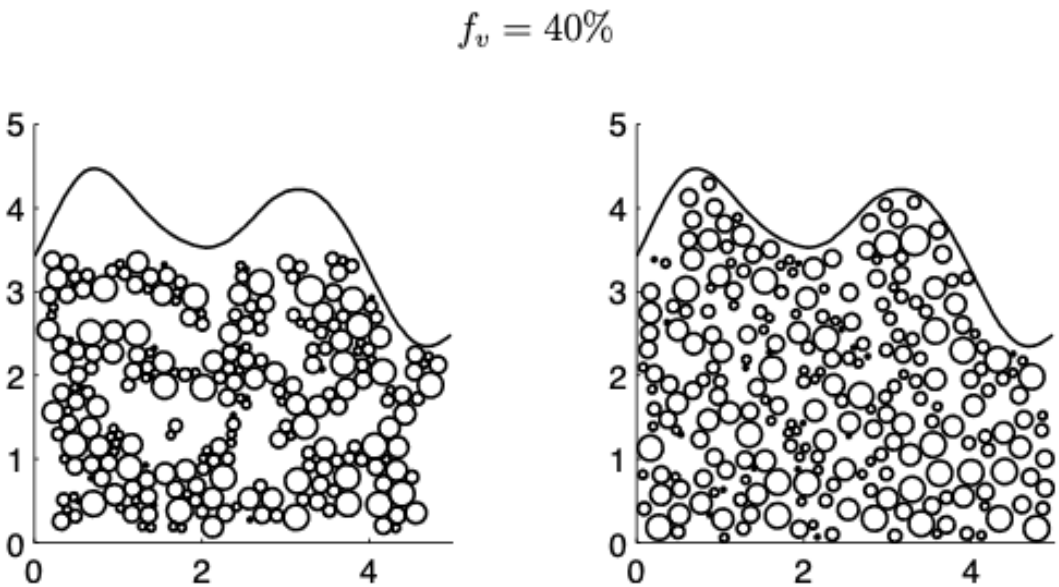
Where β is a constant and η a random number between 0 and 1.

- Allows to take into account any field, forces
- In the case of hard volumes, simplify to $E = 0$ (no intersection) or $E = \infty$ (intersection)

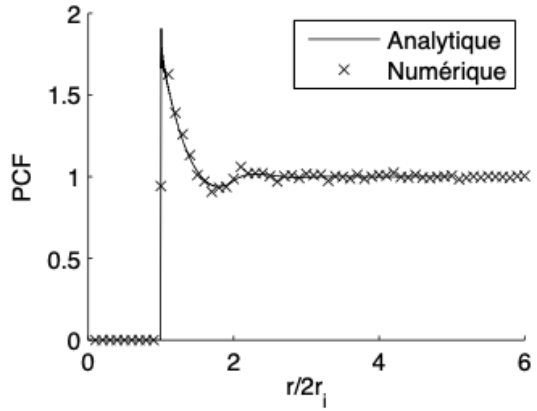
In this talk, for the sake of simplicity, the grains will be only allowed to move on the (x,y) plane but can rotate on the 3 axis.

Proof of principle

Successfully applied in "Étude de la diffusion optique par des matériaux hétérogènes rugueux : Diffusions surfacique, volumique et couplage surface/volume" by Hervé Chanal



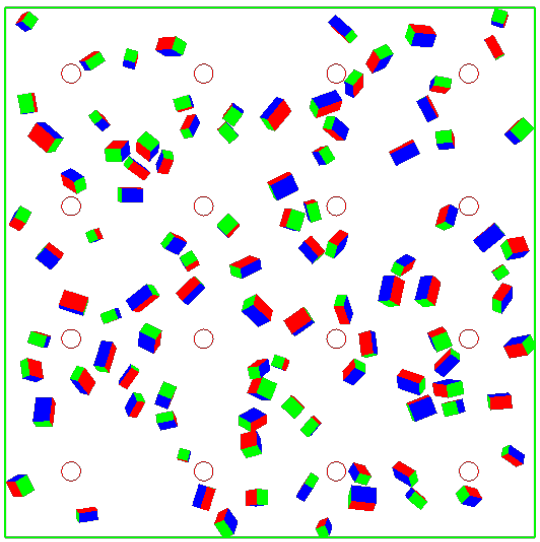
Generated media example, hard sphere model, with (left) and without (right) a $1/r^2$ field



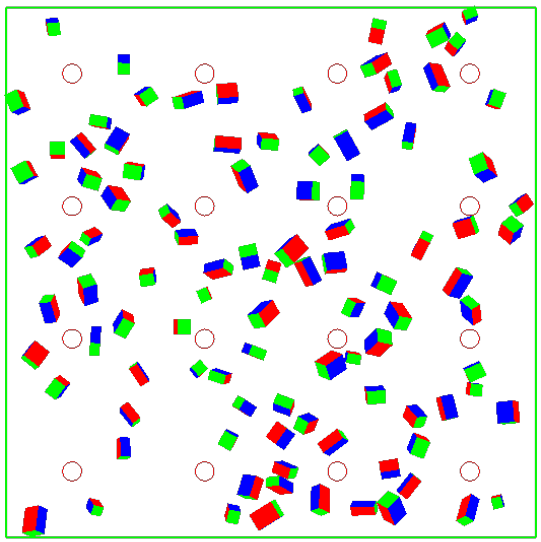
Comparison of the Pair Correlation Function for a random media between the analytical expectation and the generated media

Example 1

- Algorithm programmed in Python
- Panda3d used
- 100 grains



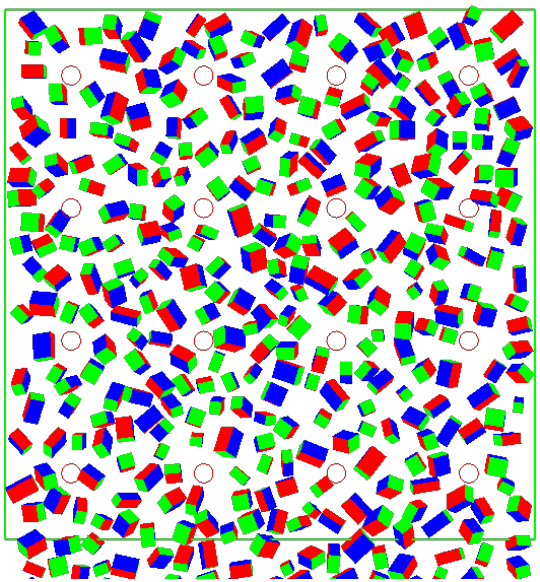
Initial step



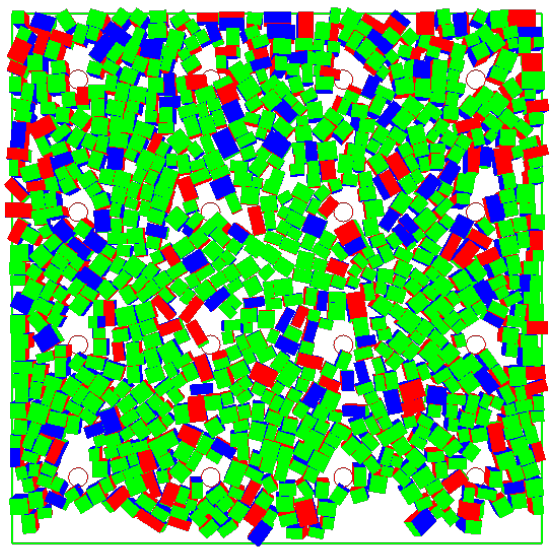
After 2000 steps

Example 2

- To reach high volumic fraction : generate boxes even outside GRAiNITA
- Use a gravity like field
- Here : 1000 grains



Initial step



After 8800 steps