This manuscript in post-review format was accepted for publication in Computational Mechanics and can be cited as follows

Karim Ehab Moustafa Kamel, Bernard. Sonon, Thierry J. Massart, An integrated approach for the conformal discretization of complex inclusion-based microstructures, Computational Mechanics, 64(4), 2019, 1049-1071

Computational Mechanics manuscript No.
(will be inserted by the editor)

An integrated approach for the conformal discretization of complex inclusion-based microstructures

Karim Ehab Moustafa Kamel · Bernard Sonon · Thierry Jacques Massart

Received: date / Accepted: date

Abstract Computational homogenization techniques nowadays are extensively used to gain a better understanding of the links between complex microstructural features in materials and their corresponding (evolving) macroscopic properties. This requires robust tools to discretize complex microstructural geometries and enable simulations. To achieve this, the present contribution presents an integrated approach for the conformal discretization of complex inclusion-based RVE geometries defined implicitly based on experimental techniques or through computational RVE generation methodologies. The conforming mesh generator extends the Persson-Strang truss analogy in order to deal with complex periodic heterogeneous RVEs. Such an approach, based on signed distance fields, carries the advantage that the level set information maintained in previously presented RVE generation methodologies [55] can seamlessly be used in the discretization procedure. This provides a natural link between the RVE geometry generation and the mesh generator to obtain high quality optimized FEM meshes exploitable in regular codes and softwares.

Keywords Conforming Meshes · Implicit geometries · Heterogeneous materials · Multi-scale analysis · Mesh Optimization · FEM

K. Ehab Moustafa Kamel
E-mail: kehabmou@ulb.ac.be
B. Sonon
E-mail: bernard.sonon@gmail.com
T. J. Massart
E-mail: thmassar@ulb.ac.be
Tel.: +32 2 650 27 42, Fax : +32 2 650 27 89
BATir, Building, Architecture & Town Planning, Université Libre de Bruxelles (ULB), CP 194/2, Brussels, Belgium

1 Introduction

1.1 Context

Materials with various degrees of randomness at the microscale are met in many different engineering problems related for instance to rocks, bones, masonry, concrete or metallic foams. For instance, heterogeneities strongly affect the permeability of rocks, thereby influencing the transient fluid transport processes in critical geo-environmental applications. In such processes, mechanical loading may alter the material microstructure, thereby impacting the fluid transfer properties [46, 18, 54].

Nowadays, multiscale analysis is used for identifying the microstructural processes responsible for a macroscopic effect, or for linking average macroscopic properties to the material properties of constituents [25, 34, 35, 44, 60]. Computational homogenization defines a general framework to model the physics of a microstructural representative volume element (RVE) and to derive the macroscopic behavior of an heterogeneous material using scale transition rules [53, 25, 58, 61]. Many computational contributions based on homogenization with periodic boundary conditions use simplified geometric microstructural representations, which can induce strong approximations for the considered materials averaged properties [24, 26].

To consider complex geometries in finite elements simulations, two main ingredients are required. First, for complex disordered heterogeneous materials, realistic RVEs incorporating the specific features of their microstructures need to be obtained or produced. This can be achieved by exploiting experimental data from modern experimental techniques, such as tomography, to characterize the spatial organization of the various phases and pore space in porous/heterogeneous materials ; or by using generation techniques reproducing available experimental features such as the size distributions of inclusions/voids, their volume fraction or the tortuosity or the connectivity in the pore space.

Secondly, these complex microstructural geometries have to be discretized by advanced techniques, able to conform to the internal material boundaries in an efficient manner. Mesh generation is a critical step in a modelling process, linking the definition of the geometry to the solution of discretized partial differential equations. Unlike for models representing the geometry of industrial parts via CAD tools, a model that aims at representing the physics of heterogeneous materials at the microscale requires incorporating complex geometries that are often defined implicitely. Such real geometries are represented by means of simple grids of points sampling the material density. Over time, idealized (virtually generated) microstructures have become more and more complex as well, with the enhancement of generation techniques and can be represented similarly to data obtained with Computed Tomography scans [42, 55]. Finally, a seamless transition between the generation and discretization steps is desirable to obtain a fully automated computational approach that does require any intervention of the analyst.

The present contribution focuses on the problematic of mesh generation in the context of multi-scale analysis of heterogeneous materials with complex microstructures.

1.2 Related work

The literature on mesh generation is way too extensive to offer an exhaustive review here. Once a description of material interfaces is obtained, two main discretization approaches can be used. Non conforming approaches, like XFEM [28, 36, 59] present the advantage to uncouple the geometrical information from the mesh generation process which can be difficult for complex geometries. This allows an easier generation of periodic structured meshes. However, this is often associated with the need to re-implement complex constitutive models in inhouse non-conforming discretization packages. Conversely, in conforming meshes, the geometry information is used in the discretization process. This makes it more complex and costly, but carries the advantage of using standard FE packages with available constitutive laws.

Most efficient conforming mesh generator softwares like GMSH [14], TetGen [51], GHS3D [15], Triangle or Netgen [45]; as well as meshing tools in finite elements packages generally are available for explicit geometries but not for implicit descriptions. Therefore an interface must be developed to provide suitable triangulated surfaces and allow the generation of usable conforming FE meshes for implicit geometries. In most of softwares, triangulation is achieved by Delaunay triangulation or by front advancing methods. The former refers to a criterion leading to a specific connectivity associated with a given set of points that can also be used in the advancing front method. The latter consists in constructing the mesh by progressively adding elements starting from the interfaces, and leading to well controlled elements sizes. For both methods, the initial node placement strategy is critical to obtain a consistent mesh [32, 5, 49, 11].

Realistic geometries are difficult to discretize in computational models. As a result, a large proportion of contributions using experimental microstructural information uses voxel-based discretization methods [4, 21, 41]. This leads to a poor representation of the material interfaces geometry by stair-case surfaces that requires the use of smoothing methods [4, 19, 41], or to inacuracies of fields derived from post-processing of the computations. This is especially true for materially non linear computations in which spurious stress concentrations significantly alter the local behavior within the RVE.

Efficient meshing methods dealing with multiple materials were developed using octree-based isocontouring in the past few years by [63, 64] with applications in medical imaging to represent the different tissues [21, 63, 64]. However, only few of them addressed finite elements simulations with periodicity, a non trivial issue for complex geometries. Some methods presented in [7, 12, 17, 41] are available but the presented results are generally either based on simplified geometries already described explicitly, or based on voxel meshing [21]. This triggered researchers to use non conforming mesh [27, 36].

In view of this, it is of interest to develop methods able to easily and robustly mesh arbitrary implicit geometries for various applications in the physics of materials.

1.3 Outline

Currently, rather few articles deal with the generation of conforming meshes on complex multi-body geometries. The present contribution proposes an integrated approach for the conformal discretization of complex heterogeneous RVEs suited for classical finite elements computations. It mainly focuses on the discretization step for geometries provided by RVE geometry generators already developed in [42, 55, 56, 57] for particulate granular media, porous media, foams or woven composites. The meshing methodology extends the truss analogy of Persson-Strang [40] in order to deal with complex (periodic) heterogeneous microstructures. The main advantage of this generator is the use of distance functions describing the microstructural geometry to be discretized for producing a conformal and periodic mesh of these complex geometries. The information obtained through the distance functions (complex geometrical description, distance to neighbours, curvatures,...) allows the process benefiting from sufficient details to handle particularly complex cases. The surfaces of the inclusions are meshed and optimized independently, simplifying the problem before producing a volume triangulation via 3D constrained Delaunay triangulation after boundary faces meshing. To this end, the mesh optimization using the extended Persson method makes it possible to generate a final result that allows to preserve the conformity and ensure periodicity while producing high-quality elements that open the way for FEM simulation on complex geometries. The approach, based on signed distance fields, carries also the advantage that the level set information used during the generation of the microstructural geometry [55] can seamlessly be used in the discretization procedure, providing a natural link between RVEs generation and the mesh generation.

The present contribution is structured as follows. Section 2 provides an overview on implicit geometries and on the specific ingredients required to implicitly define RVEs. The resulting implicit RVE geometry constitutes the input for the new developed mesh generator. Section 3 recalls the truss analogy process presented by P.O. Persson [40] to optimally mesh single body implicit geometries and its adaptation for heterogeneous geometries. It presents how boundary conformity at material interfaces can be ensured by forcing nodes to move only tangentially to these surfaces, and how an optimization algorithm can be built on the truss analogy based on a repulsive force field, to obtain high quality element shapes. Section 4 presents the extended meshing algorithm by first enlighting the difficulties related to the conforming meshing of complex periodic heterogeneous RVEs. Secondly, the meshing and optimization procedures are adressed, focusing on the issues related to the conformal meshing of materials internal boundaries. Section 5 illustrates some meshing results of complex RVEs. Finally, Section 6 discusses the results and potential further improvements of the procedure. The discretization procedure is available for both 2D and 3D discretizations. Since 3D meshing requires more specific treatments, the present contribution will focus more particularly on the 3D implementation, even though some parts are illustrated for the 2D for the sake of clarity.

2 Input Geometries

2.1 Implicit Geometries

In a microstructural heterogeneous geometry, interfaces separating phases can be described either explicitly by means of triangular facets in 3D or implicitly. A curve or a surface ϕ can be represented implicitly by means of the level sets of functions $LS(\mathbf{x}, ...)$:

$$\phi \equiv LS(\mathbf{x}, ...) = k \tag{1}$$

where $\mathbf{x}$ represents the spatial coordinates and k the iso-value of the function.

In a computational context, level set functions $LS(\mathbf{x}, ...)$ are evaluated on finite regular grids defining the domain of the geometry. The complete definition of continuous functions from those requires the association of an interpolating scheme. Due to the discrete nature of LS evaluations, the actual positions of material interfaces depend on the grid resolution and on the interpolation scheme used to define the LS function, making them not uniquely defined. Typically, the use of a linear interpolation requires the grid resolution to be fine enough with respect to the represented geometry. The same consequences hold for the computation of the spatial derivatives, the approximation of which can be obtained by finite differences.

The level-set function used to define geometries can be of different nature and may depend on various parameters. A well-kown example of level set functions is the gray-scale density map obtained by X-ray (Computed) Tomography (CT) scans in (many) research fields such as medecine, biology, mechanics or environmental applications; highlighting their ability to model and handle geometries with arbitrarily complex features. $LS(\mathbf{x})$ also gives access to some of their intrinsic properties such as the direction or curvature of the curve (resp. surface). The gradient of $LS(\mathbf{x})$ is indeed related to the normal to the level sets of $LS(\mathbf{x})$ as level sets of a function are perpendicular to its gradient. Moreover, their second derivatives give access to the local curvatures of the interface, as the divergence of the normalized gradient of LS furnishes the local mean curvature [16, 47].

A particular choice for this function is the signed distance function to ϕ [37, 47].

$$\phi \equiv DS(\mathbf{x}) = 0 \; with \; |grad(DS(\mathbf{x}))| = 1 \tag{2}$$

Given an interface ϕ dividing the RVE domain Ω in two sub-domains Ω^+ and Ω^-, the signed distance function of ϕ is a function $DS_\phi(\mathbf{x})$ with the value of the signed euclidian distance from $\mathbf{x}$ to ϕ, with by convention a negative value attributed to points included in the domain Ω^-. These functions can be easily computed from any level set function using the Fast Marching Method [37].

2.2 Implicitly defined RVEs

In the context of microstructural RVE geometry generation, a combination of level set functions can be used to represent complex and/or multi-body geometries. In addition to DS functions defining inclusions, in multi-body situation, such as in RVE illustrated in Figure 1, *global distance functions* $DN_k(\mathbf{x})$ can be used as

global descriptors of the microstructural geometry. The notation DN_k denotes the distance to the k_{th} nearest neighbour. To evaluate properly these functions, a distance function DS_{ϕ_i} is required to each inclusion i. As a consequence, the CTX scan of a heterogeneous geometry, grouping the representation of all inclusions in a single LS function, cannot directly be used to evaluate DN_k functions without specific processes. Conversely, it is possible to reconstruct any DS_{ϕ_i} based on DN_k functions. $DN_1(\mathbf{x})$ represents the first nearest neighbor distance function and can be used strictly as a distance function defining implicitly every inclusion boundary inside the RVE with a single function.

$$DN_1(\mathbf{x}) = \min_i \left[DS_{\phi_i}(\mathbf{x})\right] \tag{3}$$

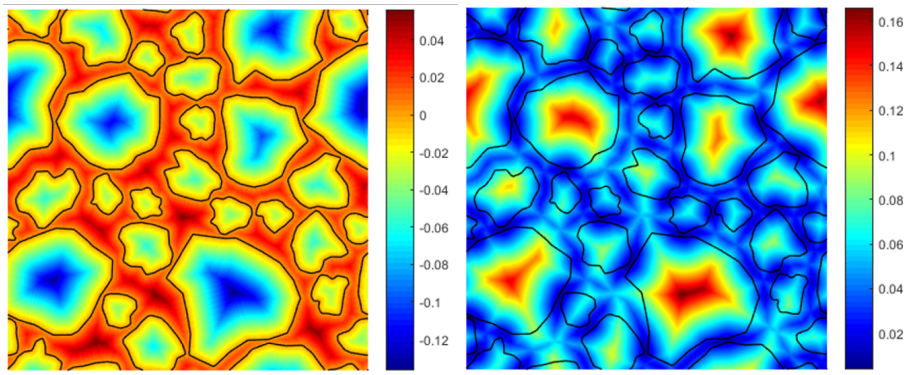

Fig. 1: *Left* : Global signed distance function $DN_1(\mathbf{x})$ for a given RVE - *Right* : $DN_2(\mathbf{x})$ function, the second nearest neighbor, for the same RVE

Further morphological information about the RVE can be obtained from $DN_2(\mathbf{x})$, the second nearest neighbor. More generally, DN_k functions partly describe the morphology of a set of inclusions. These functions will be used in the present contribution to detect some local configurations requiring a finer mesh.

3 Mesh generation for implicit geometry of homogeneous structures

A meshing tool called *distmesh* was developed in MATLAB by P.O. Persson [39, 40]. It starts from a non-conformal initial discretization transformed into a conformal mesh by using an auxiliary structural computation on a truss network in which the bars correspond to the edges of the mesh elements. This auxiliary computation makes use of a repulsive force field moving the nodes located inside the geometry domain towards the interfaces, while being constrained to remain in the domain based on the level set function that describes the implicit geometry. The code developed by P.O. Persson [39, 40] for homogeneous geometries has the aim to remain simple and public. Its central concept is explained now to allow its extension towards periodic heterogeneous complex structures in Section 4.

3.1 Global discretization process

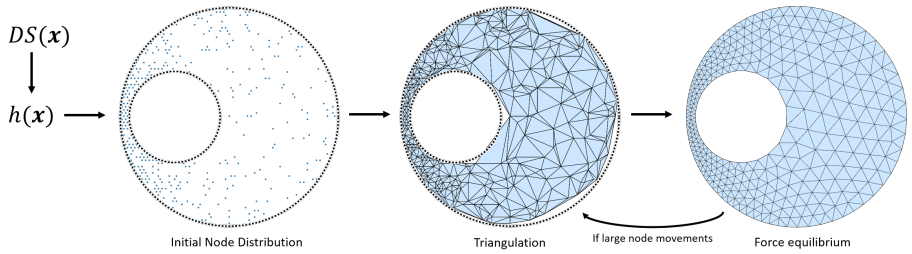

Fig. 2: Global meshing process of [40] divided in four main stages taking as input an implicit geometry. (reproduced from [40])

The meshing procedure for homogeneous structures is essentially divided in four stages as shown in Figure 2. The first stage consists in defining a size function $h(\mathbf{x})$ to be used as a space-dependent target for elements sizes, depending on geometrical or physical features. More details on this stage can be found in [39] and further in this paper. The second stage generates an initial node distribution on which a triangulation is subsequently produced. Persson proposed in [40] an efficient generation procedure for the initial node distribution, based on a probabilistic distribution called rejection method. The last stage optimizes the nodes positions according to a force equilibrium process using an auxiliary truss analogy explained in the next section to achieve optimal element shapes.

In finite element simulations, the error upper bounds depend only on the smallest angle of the mesh elements [30, 40]. Accurate numerical results are therefore obtained if 2D triangles tend to be equilateral [9, 30]. A commonly used measure to evaluate the quality of tetrahedra is therefore the ratio based on their largest inscribed and smallest circumscribed spheres. This ratio tends to a value 1/3 for a regular tetrahedron. An element quality factor q is then defined by :

$$q = 3\,\frac{r_{insc}}{r_{circ}} \tag{4}$$

Other quantification methods also exist and are addressed in more details in [9, 10, 13, 22].

3.2 Mesh quality optimization

The mesh quality optimization is based on an iterative technique that uses a simple mechanical analogy between the edges of a simplex mesh and the bars of a truss, or equivalently a structure made of springs [40]. In this analogy, the edges of tetrahedral elements and the mesh nodes correspond respectively to the bars and joints of a truss system. By assuming an appropriate force-displacement relationships for the bars, the final nodes position (p) for a set of forces (F) can be found iteratively by solving for static equilibrium according to

$$\mathbf{F}(\mathbf{p}) = \sum_i \mathbf{F}_{int,i}(\mathbf{p}) + \mathbf{F}_{ext,i}(\mathbf{p}) = 0 \tag{5}$$

where $\mathbf{F}_{int}$ and $\mathbf{F}_{ext}$ are respectively the internal forces present in the bars and the external forces stemming from boundary constraints (supports). These latter are introduced by means of the signed distance function to ensure the conformity of the mesh at an imposed boundary.

The force vector $\mathbf{F}(\mathbf{p})$ depends on the topology of the truss system. In the present case, $\mathbf{F}(\mathbf{p})$ is not continuous accross arbitrary $\mathbf{p}$ variantions because Delaunay retriangulations are performed when large node movements occur, thereby inducing some difficulties to solve the system.

A simple approach was therefore proposed by Persson to solve the system, using an artificial time-dependence. The following system of ODEs is considered (in non physical units), with initial condition $\mathbf{p}(0) = \mathbf{p}_0$ being the initial node distribution :

$$\frac{d\mathbf{p}}{dt} = \mathbf{F}(\mathbf{p}), \qquad t \geq 0 \tag{6}$$

Indeed, $\mathbf{F}(\mathbf{p}) = 0$ is satisfied if a stationary solution is reached. The latter is found by integrating (6) in time using the forward Euler method.

$$\mathbf{p}(t_{n+1}) = \mathbf{p}(t_n) + \Delta t\, \mathbf{F}(\mathbf{p}(t_n)) \tag{7}$$

Internal forces $F_{int}(\mathbf{p})$ allow steering the equilibrium state toward a configuration matching the size function $h(\mathbf{x})$. To this end, as illustrated in Figure 3, a repulsive force field $f(l, l_0)$ is defined on each bar (i.e. each tetrahedral element edge) depending on its current length l and its prescribed length l_0 interpolated from the size function $h(\mathbf{x})$ according to :

$$f(l, l_0) = \begin{cases} k\,(l_0 - l) & \text{if } l < l_0 \\ 0 & \text{if } l > l_0 \end{cases} \tag{8}$$

$$\text{where } k = \frac{l + l_0}{2\,l_0} \approx 1 \tag{9}$$

This repulsive force field tends to move internal nodes towards the boundaries of the domain. The nodal force vector $F(\mathbf{p})$ of the equivalent truss system thus contains both the internal forces from the bars ($\mathbf{F}_{int}$) ensuing from the $f(l, l_0)$ field and the reactions at the boundary nodes ($\mathbf{F}_{ext}$). Boundary nodes are themselves prevented to exit the domain based on the level set function that describes the implicit geometry of the interface. The reactions are oriented along the normal to the boundary defined by the gradient of the distance field function to this boundary :

$$F_{ext}(\mathbf{p}(t_n)) = -DS(\mathbf{p}(t_n))\, \nabla DS(\mathbf{p}(t_n)) \tag{10}$$

$$\mathbf{p}(t_n) = \mathbf{p}(t_n) + \Delta t\, \left(F_{ext}(\mathbf{p}(t_n)) + F_{int}(\mathbf{p}(t_n))\right) \tag{11}$$

The solution is considered to be obtained when the value of the maximum node movement is below a certain tolerance.

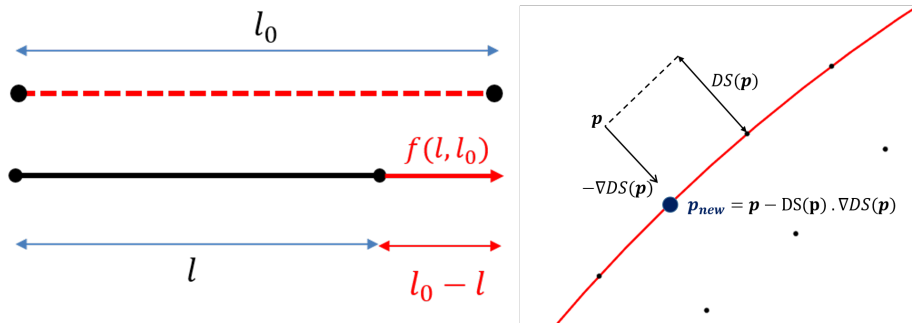

Fig. 3: *Left :* Repulsive force field in bar smaller than the targeted length l_0 - *Right :* Boundary constraints of nodes moving outside the implicitly defined boundary reproduced from [40]

In case node movements between two iterations are too large, a retriangulation is performed to modify the topology of the mesh and the optimization process is restarted until a stationnary solution is found (see Figure 2) producing very high quality meshes. There are essentially four stopping criteria in the optimization process:

- A maximum number of iterations
- A maximum number of retriangulations
- A retriangulation criterion implemented in case of significant node movements (see quantification below), and iterations number is reset to 0 after retriangulation
- A criterion also defined for stopping the optimization process when the configuration of the mesh is sufficiently stable

The criteria are left to the user discretion since they are modifiable parameters in the dashboard of the mesh generator. The quantitative input parameters values for these criteria used in the mesh generations illustrated in the manuscript are as follows: The maximum numbers of iterations and of retriangulations are set at 50 and 10 respectively. These numbers are selected in order to limit the computation time in case of non-convergence of the process. If the system does not converge, the process starts again with a five times smaller time step to make the movement of the nodes slower. If the system still does not converge, it means that the process has not found the optimal position of the nodes based on the size map set upstream, leading to areas where the elements are expected to be of poor quality. In this case, the process restarts from the beginning by modifying the initial parameters to reproduce a new more refined size map to better reflect the geometrical complexity. A retriangulation is applied when the most stressed bar undergoes a contraction or extension larger than 10% of its length. The optimization process itself is stopped when the most stressed bar of the equivalent truss undergoes a contraction or extension of less than 0.1% of its length. Note that these are parameter values that the user may want to modify. They will essentially affect the computation time and the time for the process to converge to get high quality elements according to the defined stopping criterion.

4 Extension to complex heterogeneous RVEs mesh generation

The Persson-Strang truss analogy for mesh optimization presented in the previous section is robust and effective for small and medium sized meshes based on single body geometries. However, many challenges appear in 3D when meshes have to be produced for complex heterogeneous RVEs. Among them, the presence of internal boundaries separating the inclusions from the matrix to produce conforming meshes makes the problem more intricate. Furthermore, multi-scale analysis often requires the capability to produce periodic meshes, which in 3D configurations requires specific implementations.

In the following sections, an adapted approach extending the Persson-Strang truss analogy is outlined in order to deal with complex periodic heterogeneous microstructures based on signed distance fields.

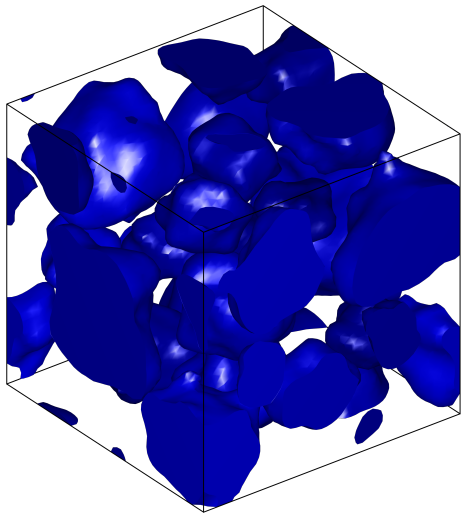

Fig. 4: Reference RVE generated by the RVE generation procedure defined in [55] to illustrate the meshing process of complex heterogeneous microstructures

4.1 Global meshing strategy

The mesh generation process for the RVE is subdivided in five steps as follows :

1. Definition of a size function $h(\mathbf{x})$ based on specific/particular geometrical features
2. Generation of an initial periodic node distribution based on an octree decomposition
3. Optimized surface meshing of the internal material interfaces, based on an initial triangulation obtained by contouring algorithms to enforce conformity
4. Optimized external boundaries meshing for periodicity

5. Optimized volume meshing of inclusions and matrix based on a Constrained Delaunay Tetrahedralization (CDT) starting from the surface meshes (produced in 3 and 4)

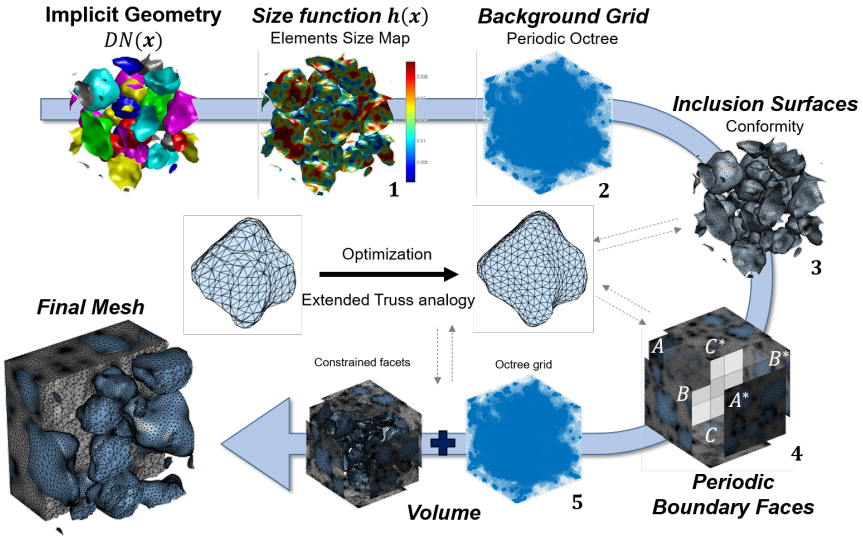

Fig. 5: Global meshing strategy based on reference RVE

The procedure starts similarly to the original Person-Strang methodology, i.e. defining a size function $h(\mathbf{x})$ to control the elements sizes. The amount of nodes is increased where complex shapes of inclusions require taking into account local features such as high curvatures, small gaps or wide size distributions of inclusions. The size function is therefore used in the optimization process and the initial distribution of nodes detailed in sections 4.2 and 4.3, in order to refine the discretization only at the positions of interest.

A key difficulty with Delaunay-based mesh generators remains the enforcement of mesh conformity at material interfaces. This issue is even more complex in 3D, and still a challenging problem both in theory and practice [52]. When applying the original Persson-Strang methodology to homogeneous domains, no specific treatment for internal boundaries is required. Therefore, the triangulation of an initial distribution of nodes and its optimization is only performed once. For heterogeneous structures, the conformity is more difficult to ensure due to the presence of nodes on both sides of the material interfaces before triangulation. Furthermore, it is not trivial to ensure periodicity of a 3D triangulation, even when starting from a periodic nodes configuration. A solution to ensure conformity and periodicity is to first mesh the inclusion interfaces and the RVE boundaries to avoid the presence of crossing tetrahedra. Once those surfaces are meshed, Boundary-conforming tetrahedral meshes are generated with a constrained Delaunay triangulation to preserve the inclusions surfaces. To do that, the Delaunay criterion is not strictly applied anymore in the neighbourhood of the constrained facets [48, 50]. This motivates

the decomposition of the procedure into several steps. Each meshing stage (surface meshing, external boundary faces meshing and 3D bulk meshing) involves two steps, namely the triangulation and the optimization process using an extended version of the principle of the Persson-Strang truss analogy.

The inclusion surfaces are first meshed set by set after extracting their zero-isosurface from the $LS(\mathbf{x})$ defining them by contouring algorithms. Then, to ensure periodicity, the RVE external boundary faces are considered as 2D meshes in which the traces of inclusions cutting the RVE boundary are discretized first, making the RVE boundary meshing process easier. The periodic enclosing surface mesh is formed by extracting, meshing, copying, translating and merging the three non-opposite faces.

Then, the entire volume is meshed by providing the optimized boundary faces and internal surface meshes as input for the Constrained Delaunay Tetrahedralization (CDT) module of the well-known meshing software TetGen [51], thereby ensuring periodicity and internal conformity of the final mesh.

Finally, a post-processing step provides mesh corrections if self-intersections of triangles in the interfaces still appear in the obtained mesh. It also attributes the elements to the inclusion and the matrix phase, and records information about external boundary nodes in order to ease the definition of finite elements boundary conditions.

The following subsections provide detailed information on how to implement each step of the procedure.

4.2 Size Function $h(\mathbf{x})$

Dealing with 3D complex geometries as sketched in Figure 4 with a uniform mesh size may induce a large numbers of elements, as stress gradient at sharp features and small gaps between inclusions then prescribe the overall mesh size. A size function defining spatially variable element sizes therefore becomes critical.

To take into account local geometrical or physical features in the meshing procedure, the narrowness between inclusions (nar), the curvature of interfaces ($curv$) and the initial interface size (iis) will be used in order to prescribe mesh refinements where needed, see Figure 6 and Figure 7. To this end, the size function $h(\mathbf{x})$ is constructed from the neighboring distance functions DN_k defined in Section 2 and readily available if the heterogeneous microstructure was computationally generated.

The initial interface size (iis) is taken as the maximum element size allowed on the interfaces to have a sufficiently accurate representation of the geometry of the RVE. This value is taken constant on all the interfaces.

$$h_{iis}(\mathbf{x}) = iis \tag{12}$$

The narrowness representing the proximity between two interfaces, is computed by taking into account the distance to the closest second neighboring inclusion $DN_2(\mathbf{x})$ from interfaces. In an inclusion-based RVE, $DN_2(\mathbf{x})$ is always positive since each inclusion is completely separated from the others in the RVE. To better control the element size refinement accross the gap between two inclusions, a parameter nb_{EL} is defined as the number of elements from the considered source

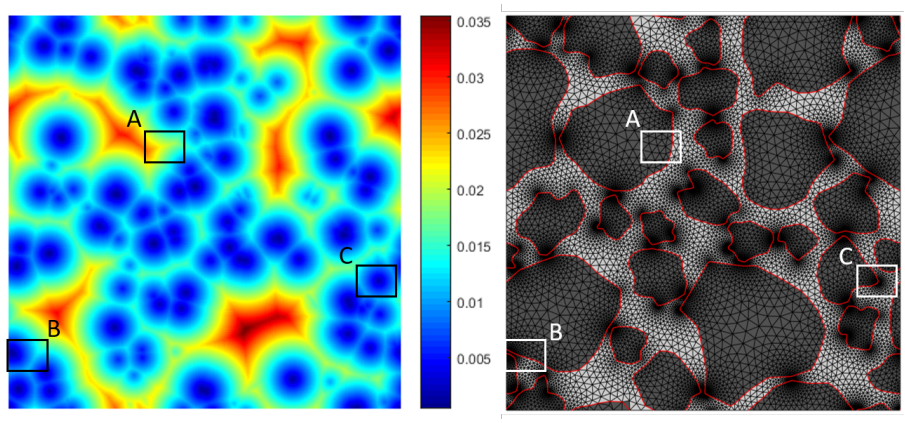

Fig. 6: *Left:* Size function $h(\mathbf{x})$ - *Right:* Corresponding optimized 2D mesh taking into account local features (initial interface size, narrownes, curvature) from $h(\mathbf{x})$

point to its respective second closest neighboring inclusion. A corresponding element size can be derived according to

$$h_{nar}(\mathbf{x}) = \frac{DN_2(\mathbf{x})}{nb_{EL}} \tag{13}$$

In 3D, the maximum principal curvature is used in order to evaluate the smallest radius of curvature of an interface [39]. Methods based on triangulated surfaces are available to evaluate curvatures as presented in [43]. However, for implicit geometries, the curvature can be directly obtained by finite differences from the implicit function. Computing the curvature from the first neighboring distance of the whole RVE (global signed distance function) DN_1 is subject to the same limitations as extracting properly the inclusion surfaces at once. For accurary reasons, the curvature computation is performed for each inclusion separately. In practice, the maximum principal curvature κ_1 is computed from the mean and gaussian curvatures K_M and K_G. K_M is computed according to relationship

$$K_M = -\nabla \cdot \nabla DN_1 \tag{14}$$

while the gaussian curvature is obtained by

$$K_G = \nabla DN_1 * H^*(DN_1) * \nabla DN_1^T \tag{15}$$

where H^* is the adjoint of the hessian matrix. This allows computing principal curvatures

$$\kappa_{1,2} = K_M \pm \sqrt{K_M^2 - K_G} \tag{16}$$

A curvature controlled element size $h_{curv}(\mathbf{x})$ is then obtained by equation (17) where κ is the maximum principal curvatures in absolute value while α is parameter allowing to adjust the sensitivity to the curvature in the size function.

$$h_{curv}(\mathbf{x}) = \frac{\alpha}{|\kappa(\mathbf{x})|} \tag{17}$$

To avoid significant element size variations over short distances in the definition of the size function that would lead to poor quality elements, the size function $h(\mathbf{x})$ is required to evolve smoothly by using a gradient limiting size variation [39]:

$$\|\nabla h(\mathbf{x})\| = g \tag{18}$$

To meet this constraint, minimum initial imposed size values h_0 depending on the three geometrical parameters *iis*, *nar* and *curv* are first computed on source points $\mathbf{x}_{sp}$ by interpolation from $DN_1(\mathbf{x})$ and $DN_2(\mathbf{x})$.

$$h_0(\mathbf{x}_{sp}) = min(\{h_{iis}(\mathbf{x}_{sp}), h_{nar}(\mathbf{x}_{sp}), h_{curv}(\mathbf{x}_{sp})\}) \tag{19}$$

For accuracy, the source points $\mathbf{x}_{sp}$ are selected as nodes located on the interfaces that are extracted using a contouring algorithm at the precision of the initial regular grid $\mathbf{x}$.

The values are then propagated smoothly over the spatial coordinates $\mathbf{x}$. In the case of a bounded convex domain [39], the following equation can be used :

$$h(\mathbf{x}) = \min_{sp}(h_0(\mathbf{x}_{sp}) + g\,|\mathbf{x} - \mathbf{x}_{sp}|) \tag{20}$$

where $h(\mathbf{x})$ is the size function evaluated on the regular grid and g is the gradient limiting factor introduced in equation (18).

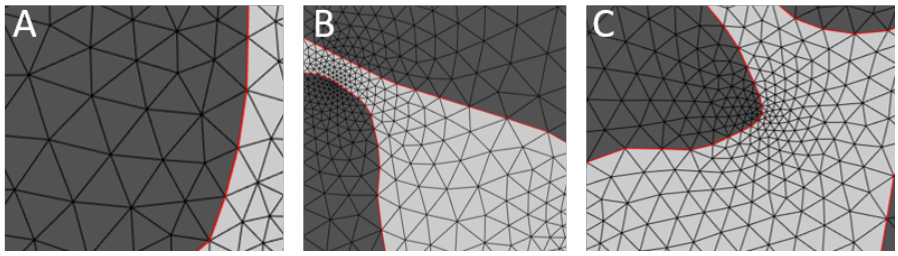

Fig. 7: Zoom on frames of Figure 6 : Initial interface size (A), Narrowness (B), Curvature (C)

Figure 8 illustrates the resulting size function for the inclusion based microstructure depicted in Figure 4 interpolated on the inclusions surfaces from the regular grid.

Following the same methodology, extending $h(\mathbf{x})$ to account for other geometrical or physical features can be achieved straightforwardly if required.

4.3 Initial nodes distribution

Several methods are available to distribute nodes inside a domain based on regularly spaced grids or on probabilistic distributions [30]. Rejection methods [39] based on probabilistic node distributions as initially proposed by Persson [40]

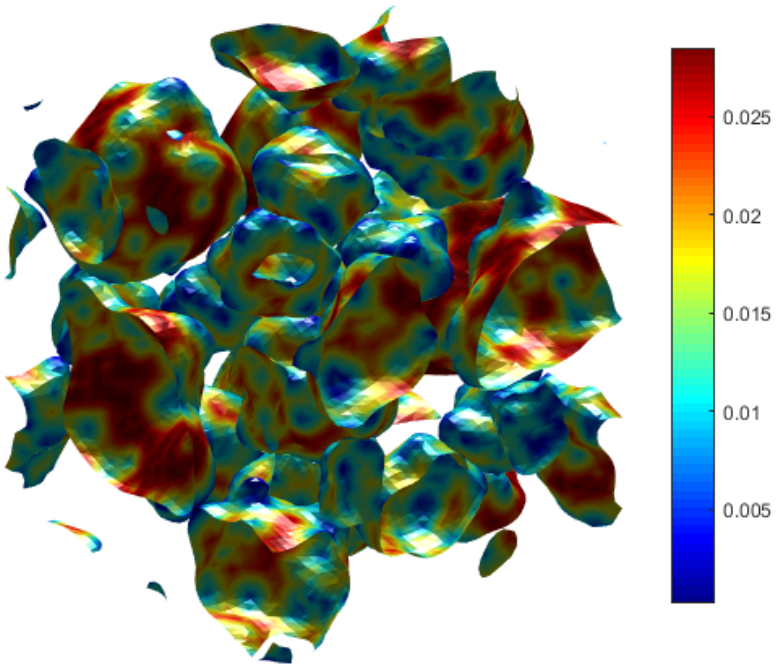

Fig. 8: Interpolated size function $h(\mathbf{x})$ on interfaces

may be interesting in view of their efficiency. However, for periodic inclusion-based RVEs enclosed in a cube (or a parallellepiped), more adapted node distributions can be obtained based on octrees linked to the size function. A periodic octree [27, 33, 38] distribution is therefore used here. The distribution starts from the 8 RVE corners and is refined recursively based on the size function $h(\mathbf{x})$ previously computed. At each recursion, the size function is interpolated at the cube mid-edges and at its geometrical center. If one of the interpolated values is smaller than the cube edges, the cube is divided into eight identical cubes. The new vertices are added to the node distribution and the process is repeated on the eight new smaller cubes. The process continues until all cubes have a satisfactory size with respect to $h(\mathbf{x})$.

This defines in a simple way a nodes distribution corresponding at best to the mesh size function $h(\mathbf{x})$, i.e. providing a higher nodes density near the interfaces and larger elements further from it, as illustrated for a 2D configuration in Figure 9. In fact, Persson's method is effective when the auxiliary truss bars are close to their equilibrium positions, i.e. the length of the initial bars from the Delaunay triangulation undergoing extension or contraction are close to the length defined by the size function. Otherwise, the resolution of the truss by the forward Euler method may introduce strong oscillations if the initial length of the bars is far from the desired length, and therefore it may not converge.

Generally, a consequence is then the poor quality of elements generated near the interfaces by a constrained Delaunay triangulation. However, thanks to the distance function, it is possible first to remove nodes too close to the interfaces.

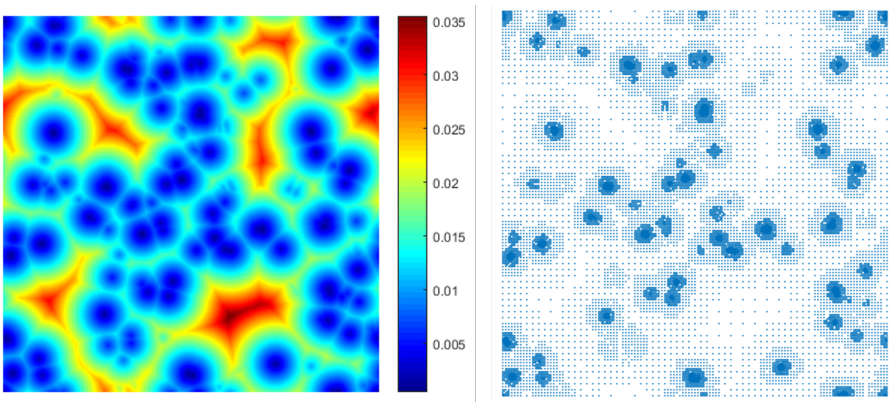

Fig. 9: *Left:* 2D size function $h(\mathbf{x})$ - *Right:* Corresponding 2D quadtree node distribution

After nodes are moved by the optimization procedure driven by the truss analogy (see Section 4.7), well shaped elements are obtained.

Using Periodic Octree distributions has some additional practical advantages making it possible to meet three objectives in the proposed generator:

- Obtaining an initial distribution of nodes between which the interdistance is close to that requested in the size function,
- Obtaining an initial triangulation with good elements quality due to the strategic positioning of nodes to feed the optimization procedure with a proper initial guess,
- Enforcing the same spatial node distribution on opposite external boundaries to allow periodic meshes generation

These objectives allow minimizing the number of operations required in the optimization process since the nodes are not very far from their equilibrium positions.

4.4 Inclusion boundaries meshing

The main meshing steps for the interfaces are illustrated in Figure 10 for the microstructure presented in Figure 4. In order to satisfy the size parameters given in input, the level-set grid $LS(\mathbf{x})$ that geometrically describes the RVE is reinterpolated on an initial 3D regular grid of points according to the iis parameter (initial interface size). This reinterpolation allows an adapted (to iis) individual or set extraction of the zero isosurfaces defining the inclusions boundaries by the Marching Cubes algorithm [31]. Then, these surfaces are selectively refined according to narrowness and curvatures. This allows conforming the surfaces mesh size to the size map generated upstream.

In spite of the simplicity, robustness and efficiency of this procedure, the resulting surface triangulation is of poor quality leading to poor 3D elements (almost-flat tetrahedra) [49], as illustrated in Figure 10.2. This initial surface triangulation is therefore optimized in surface using the truss analogy and used subsequently as input for the 3D constrained Delaunay triangulation.

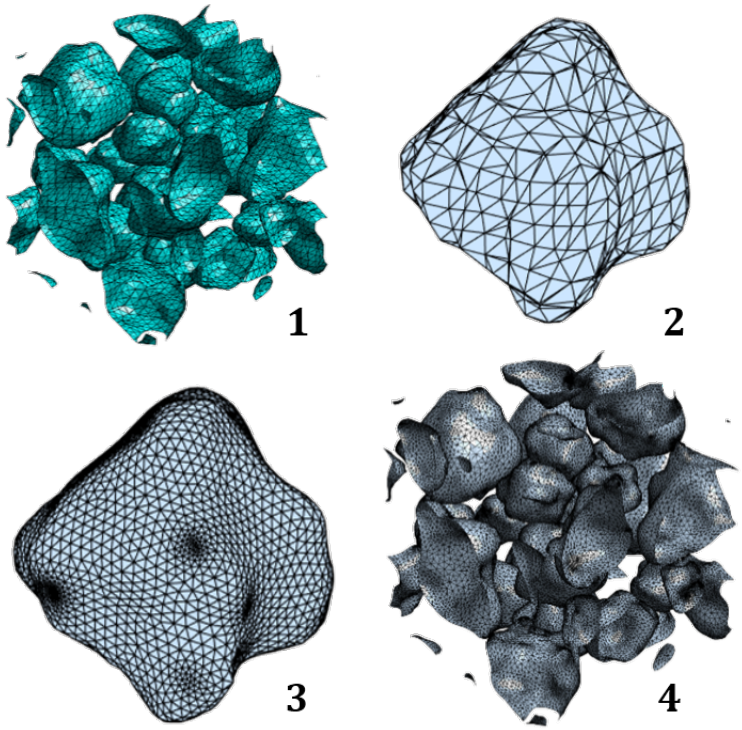

Fig. 10: Inclusions surface meshing main steps : (1) Isosurface extraction via Marching Cubes algorithm; (2) Boundary constraints; (3) Optimization via truss analogy, edge flipping, local mesh refinement; (4) Adding to the set of meshed inclusions

Another issue linked to the use of the marching cube algorithm for the construction of interfaces is the existence of ambiguous cases when the background regular grid is not refined enough. For inclusion-based RVEs with small gaps between inclusions, critical issues arise quickly, as very close inclusions tend to merge spuriously. The refinement of the background regular grid is possible but is a costly solution as it induces a cubically growing memory consumption for 3D configurations, and does not strongly solve the problem.

A solution is therefore to perform an individual extraction of each of the inclusion surfaces. This is possible if independent signed distance functions are available or can be built for each inclusion. This allows decreasing memory needs and computation time by using coarser grids for extraction and increase cache efficiency as memory chunks that are accessed are smaller.

The surface mesh optimization is analogous to a 2D problem constrained to the interfaces of inclusions. During the optimization process, the extracted triangulation is subjected to normal constraints making the movement possible only along the surfaces (see Figure 3). This is achieved using the distance function from which the interface is extracted. Based on the distance to the surface and computing its

normal at each iteration, nodes moving away from the surface are pushed back to the surface, thereby allowing only tangential movements by means of Equation 10.

The low quality initial triangulation obtained from the marching cube extraction involves a significant variation in size of the elements edges. The triangulated surface mesh of inclusions boundaries (material interfaces) is therefore further simultaneously improved by three different processes consisting of local mesh refinement, local connectivity update and node relocation by truss analogy.

Local mesh refinement is applied according to the size function $h(\mathbf{x})$. Wherever necessary with respect to $h(\mathbf{x})$, triangles are divided into four triangles keeping the same aspect ratio as shown in Figure 11 to avoid increasing the distortion of newly created triangles. Particular attention is paid to the edges intersecting the RVE boundaries to keep a periodic mesh upon refinement.

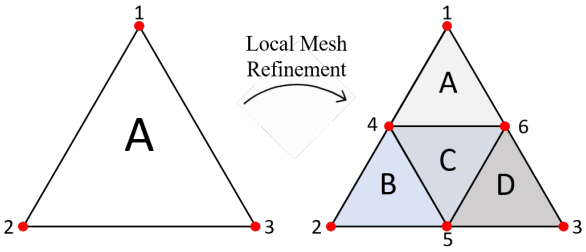

Fig. 11: Local mesh refinement : Regular division of a triangular element as a function of $h(\mathbf{x})$

Starting with bars of very different lengths from the ones desired according to the size function $h(\mathbf{x})$ in the auxiliary truss computation leads to large displacements. Therefore, it is possible to improve the topology by modifying the local connectivity via edge flipping. Considering two adjacent triangles, the edge connectivity is flipped in order to increase their quality and leave the rest of the triangulation unaffected (see Figure 12). It speeds up the optimization process significantly as it acts locally. External RVE boundary edges are however not affected to preserve periodicity.

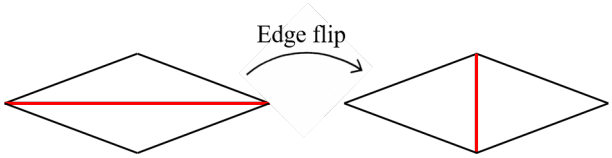

Fig. 12: Local triangular element update : Edge Flip

Node relocation allows displacing nodes by imposing internal forces in the bars based on the truss analogy [40]. This method works very well when the initial length of bars is close to the length targeted by the size function. If the initial triangulation starts far from the truss equilibrium state and involves topology

changes, movement instabilities can occur leading to self-intersections. When the implicit geometry of an inclusion described by a distance field is extracted into an explicit one using the Marching Cube algorithm to produce a facetted description of the surface of an inclusion, the initial mesh contains not only elements of poor quality (see Figure 13(a)), but also highly variable element sizes that can induce significant movement during the optimization process. It is clear that the artificial time parameter used in the resolution of the truss by the forward Euler method is important and should not be too high to avoid oscillations. On the other hand, it is possible to find areas where the elements are much too large. This can be attenuated by selective refinement illustrated in Figure 13(b). They can also be too small according to the desired size. To avoid these self-intersections, or at least to reduce them drastically, a coefficient α is introduced to control the weight of the desired length (l_0) in the internal force field.

$$f(l, l_0) = k\ (\alpha\, l_0 - l) \tag{21}$$

This force field acts similarly to a Laplacian smoothing function [8] if $\alpha\, l_0$ is taken close to zero in the first iterations unlike in [40] where a purely repulsive force field is used. It induces mainly attractive forces in the bars according to (21). The forces applied in the bars are proportional to their own current length, inducing tension in small bars and compression in longer bars in such a way that displacement variations are smoothen out. Combined with the flip edge, this will induce contraction in the long bars and extension in the small bars, thus reducing the potential risks of self-intersection while increasing the quality of the elements. Even if not entirely robust, the progressive increase of the targeted length result in an increase of the initial element shape quality without inducing self-intersections.

Progressively, as the α parameter is exponentially increased towards 1, the effective targeted length l_0 takes a larger influence in the expression of the force fields. The choice of the exponential function according to iterations is an arbitrary choice motivated by the fact that the first movement variations must be the least influenced by the desired length given by the size map until the bars reach a length close to the one defined in the size function to avoid strong movements. The force field thus evolves smoothly from a purely attractive field based only on the current length of the bars to an attractive/repulsive one based on the size function $h(\mathbf{x})$ until reaching equilibrium. Figure 13 shows on one side (c) the result of the optimization without using the α parameter and on the other side (d) with using the α parameter for the same time step ($\Delta t = 0.1$). Of course, decreasing the time step will also reduce the self-intersections shown in Figure 13(c) at the expense of computation time.

To enforce periodicity, there is a need to know which nodes are linked to each other on the opposite faces. Note that the isosurface extraction of periodic RVEs gives a periodic node distribution on opposite faces. Then, surface contours and nodes on the RVE external boundary faces are extracted from the initial surface triangulation as illustrated in Figure 14. These nodes are reordered for each inclusion in order to easily find them together with their corresponding ones on opposite faces. This step aims at providing the necessary information for node movement coordination of periodic nodes to enforce periodicity at each iteration of the optimization, and for using it as a starting point to mesh the external boundaries in the next step (cf. Section 4.5).

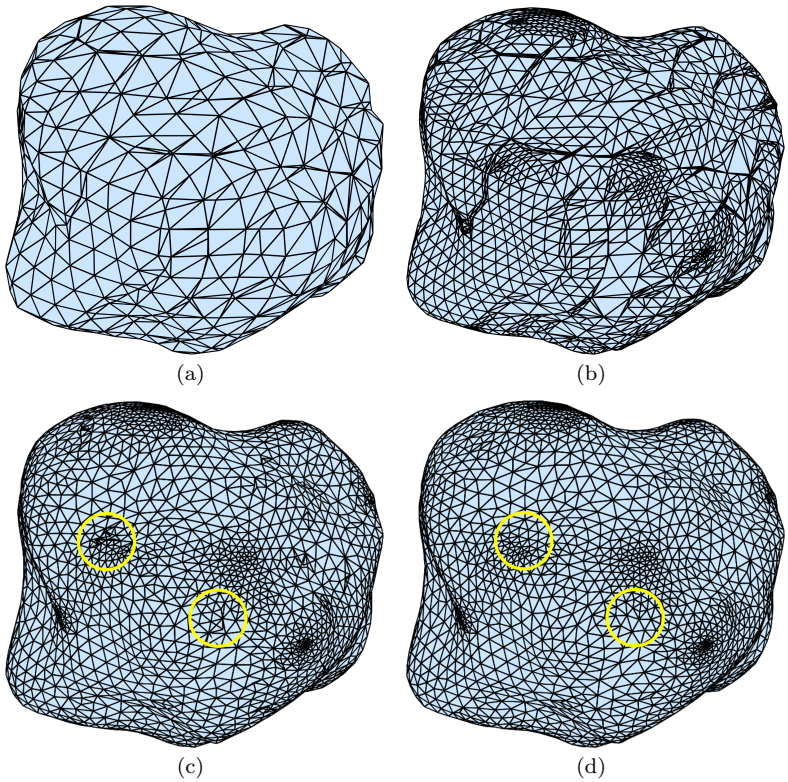

Fig. 13: (a) Extracted inclusion with Marching Cubes algorithm, (b) Selective refinement of inclusion according to size function, (c) Optimization process with α parameter equals to 1 with yellow circles showing self-intersections, (d) Optimization process with a smooth increase of parameter α from 0 to 1. Yellow circles showing clean meshes compared to (c).

In the truss analogy process, forces are applied to move nodes. To ensure that RVE boundary nodes remain on the external faces, the displacement normal to the direction of the face is prevented. Due to the interactions with adjacent nodes, the forces applied on a node and its periodic equivalent are not necessarily the same. The movement coordination ensuring periodicity is enforced by applying identical forces on both nodes of opposite faces as the average of forces determined on them separately.

Particular attention is paid on nodes located on the corners and edges of the RVE for which the displacements are constrained respectively in 3 and 2 directions.

At the end of the inclusions surface optimization process, the nodes and triangular facets are fixed on the interfaces and are not modified during the rest of the meshing procedure.

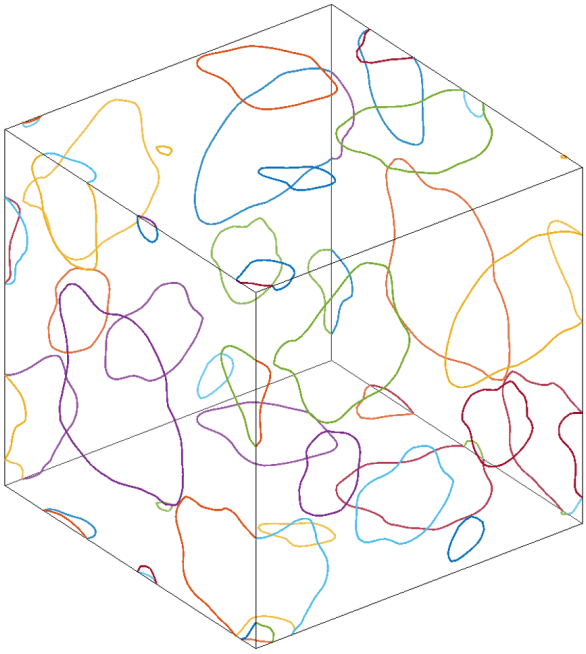

Fig. 14: External boundary edge nodes extraction and movement coordination to ensure periodicity

4.5 RVE Boundaries meshing

Three non opposite faces from the parallelepipedic RVE faces are extracted, meshed and optimized by the truss analogy process.

The three master boundary faces A, B and C are defined respectively in planes $x = 0,\ y = 0,\ z = 0$. External boundary edges and nodes are extracted from the optimized surface mesh with octree nodes lying at the external boundaries of the RVE. The problem is simplified by transforming the 3D boundary nodes on the three planes to deal with 2D problems as illustrated in Figure 15.

Then, the extracted surface contours (intersections of inclusions with the RVE face) and nodes are constrained together with corner nodes to generate an initial triangulation using a 2D constrained Delaunay triangulation (see Figure 16). The optimization of this triangulation includes in this case two steps. Iteratively, the Persson-Strang truss analogy is applied in the three planes ($x = 0,\ y = 0,\ z = 0$) separately to optimize the surface meshing of the boundary faces until the mean quality starts stagnating. Then, a retriangulation of the face is performed to update the topology in order to reach a targeted element quality (see Figure 16).

Finally, as illustrated in Figure 17, the opposite slave RVE boundary faces are created by copying the master boundary faces and translating them to their position. The end of this sub-process merges the 6 RVE boundary faces with the internal inclusions surfaces meshes. The merged surfaces (internal and external) constitute the constrained facets to ensure the internal mesh conformity and the

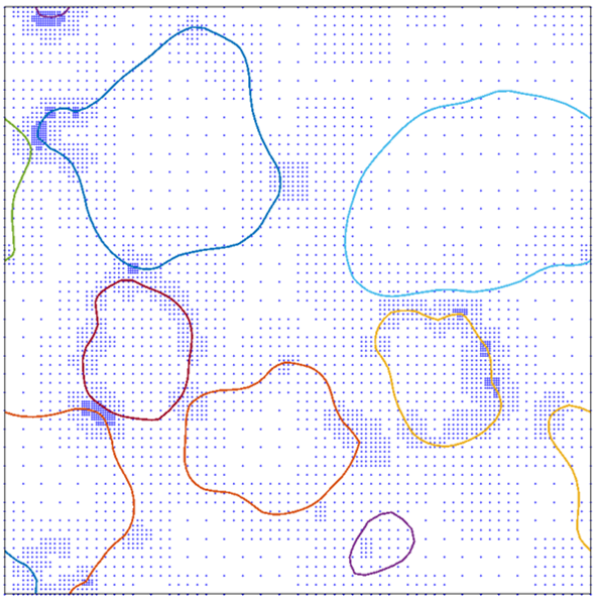

Fig. 15: 2D simplification - External boundary faces meshing : Extraction of external boundary surface edges and nodes and octree nodes lying to the considered face

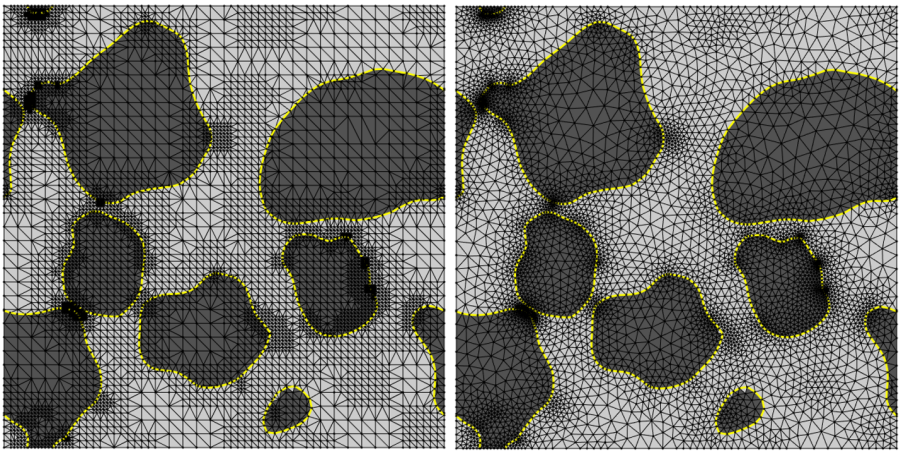

Fig. 16: 2D simplification : *Left* Initial constrained delaunay triangulation - *Right* Optimized external boundary face mesh using Persson-Strang truss analogy [40]

periodicity of the mesh, by providing a closed surface mesh to the constrained Delaunay tetrahedralization (CDT) in the next step of volume meshing.

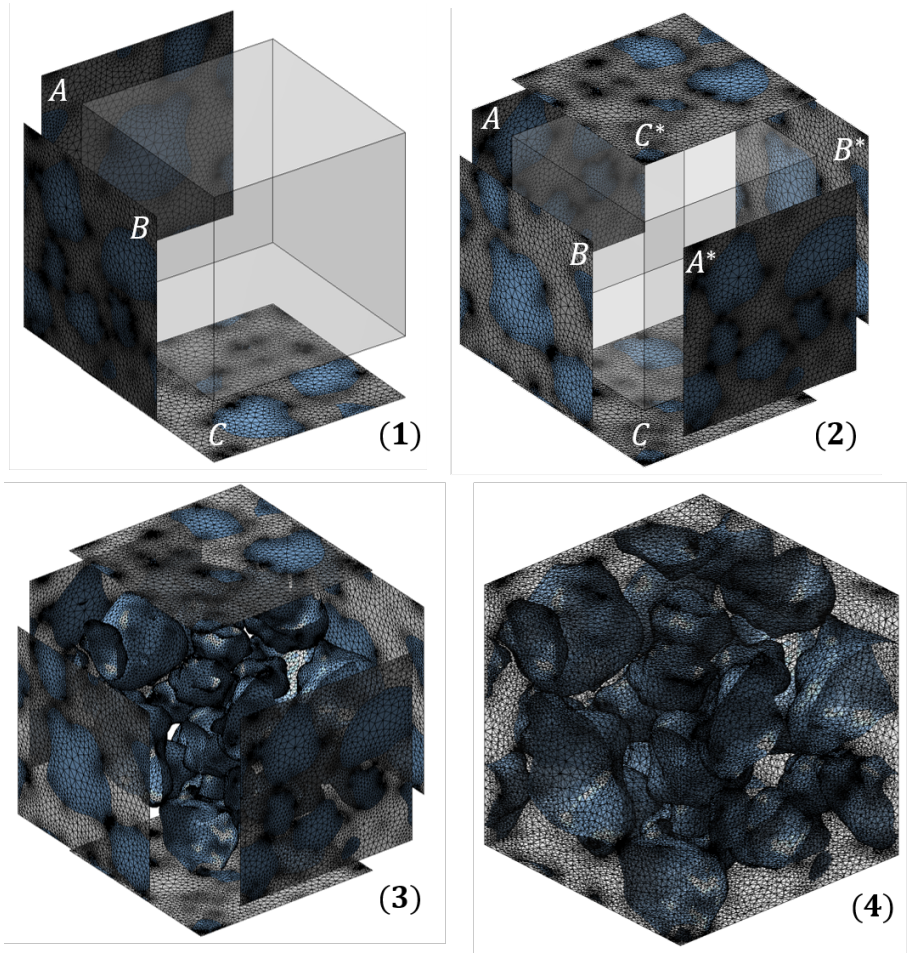

Fig. 17: 3D periodicity : (1) 2D meshing of non-opposite master faces in plan ($x = 0$,$y = 0$,$z = 0$, (2) Copy and translation of the slave faces, denoted with * to their corresponding positions, (3) Addition of the internal inclusion surfaces meshes, (4) Merge of the external boundary faces with the internal inclusions surfaces meshes and removing duplicated nodes

4.6 Volume Meshing

The final step in the meshing process consists in the generation of the volume mesh. Optimized boundary faces and internal surface meshes are used as input to the constrained Delaunay tetrahedralization while the background grid on which the

CDT is performed originates from the octree nodes distribution. The CDT module of the well-known and robust mesh generator TetGen [51] is used for this purpose. Due to the complexity of the geometries and the imposed constraints (conformity and periodicity), the resulting mesh contains low quality elements, especially close to the interfaces. The truss analogy process is therefore used in order to increase the volume elements quality. As mentioned in the previous section, this optimization includes several steps. In addition to the force equilibrium smoothing function and the retriangulations already explained, a specific treatment to get rid from very specific bad shaped elements, also called slivers or flat tetrahedra [30] is performed, as outlined in the next section.

4.7 Extended Persson-Strang truss analogy

To ensure better stability on the node displacements during the mesh optimization and to get rid from slivers which may cause finite element computation to fail [11, 39], the original truss analogy methodology is modified with an adapted ball-vertex spring method [3, 29]. This is achieved by introducing additional (linear) springs on bad-shaped elements (with quality lower than a certain threshold) as illustrated in Figure 18. The role of these additional springs is to resist the motion of a node towards its opposite faces. In a given tetrahedron, node a is selected as the node located closest to its opposite triangular face and is computed as the normal projection e of the node a on the face bcd. The method is then combined with the size function to apply repulsive forces on a in the normal direction to get it far enough from bcd. A consequence is a drastic reduction of bad-shaped elements and avoidance of nodes crossing triangles during the smoothing process.

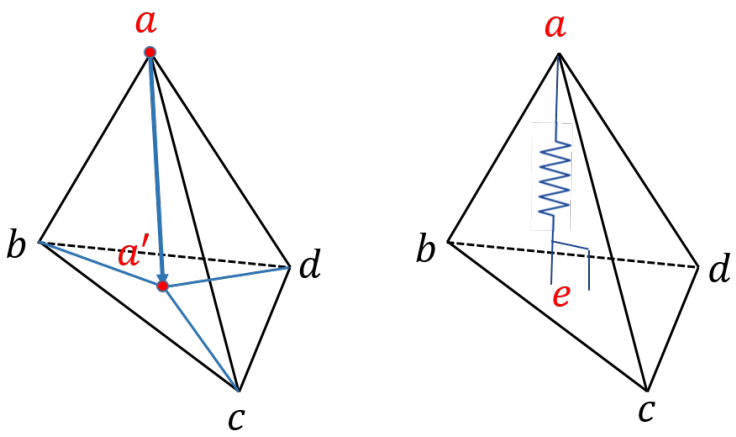

Fig. 18: *Left:* Tetrahedron collapse mechanism with the edge spring method generating slivers - *Right:* Ball-vertex spring method by connecting the closest point a with its opposite triangular face bcd to ensure stability and get rid from slivers.

Also, faster convergence can be obtained by combining attractive and repulsive force fields [20, 23]. Globally, the principle remains the same as in [40] except the expression of the force field in the bars f_{bars} used. While the initial algorithm used exclusively repulsive forces to push nodes accross the interfaces, in this heterogeneous case, extracted nodes already lie on the triangulated internal surfaces from the beginning of the 3D constrained meshing process making the optimization process more stable.

4.8 Summary

The mesh generation times and the parameters on which the subprocesses depend are summarized in Table 1. An initial fine mesh is used for the explanation of the process (Mesh fine : #366536 nodes,#2178896 elements). A coarser one is also illustrated in Figure 19 (Mesh coarse : #50839 nodes, #305892 elements). In addition, this Figure shows the effect of iis, respectively 0.05 and 0.03 for coarse and fine mesh for a cube box of length 1, on the final mesh. The computing times mentioned here are obtained for mesh generations done on a ThinkPad P50 i7 6700HQ, 32 GB RAM coded in MATLAB.

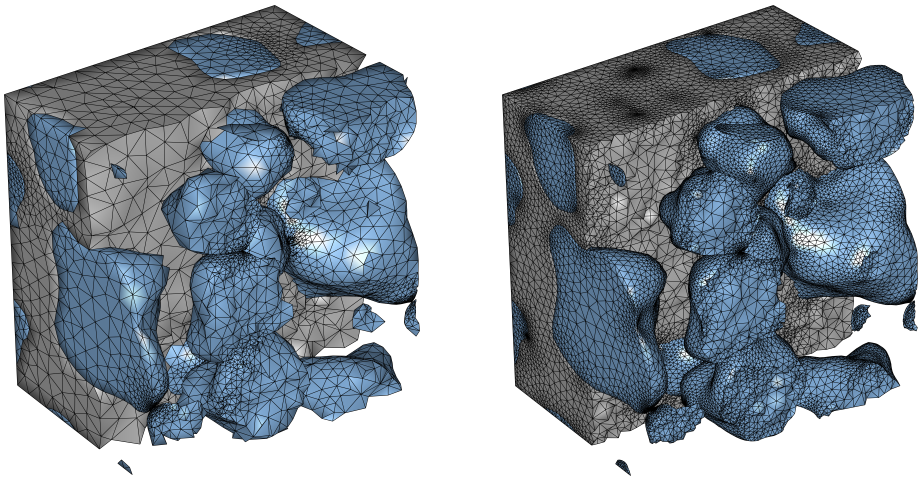

Fig. 19: Left - Cut view of coarser Mesh 1 (#50839 nodes, #305892 elements), Right : Cut view of finer Mesh 2 (#366536 nodes,#2178896 elements)

Considering that the generator was coded in an interpreted language, the evolution of the computation times should be analysed as a function of the mesh refinement and among the process relative to each other, rather than according to their absolute value. It would be easy to obtain much lower computation time by recoding the critical routines a compiled language. In addition, the surfaces are meshed and optimized one by one or set by set, which leaves the door open for easy parallelization of the process and therefore significant time savings. The routines have been designed to be vectorized and therefore optimized even if it is

Step	Mesh coarse Time (s)	Mesh fine Time (s)	Parameters
Size function	13.71	31.77	Initial interface size Narrowness Curvature Gradient limiting factor
Surface Meshing	17.25	53.07	# Inclusions Marching Cube extraction at initial interface size Selective refinement (due to narrowness and curvature) Self-intersections check Periodicity constraints Optimization process
Octree node distribution	1.26	6.19	Size function (iis,curv,nar) Density parameter
Boundaries meshing	5.89	16.06	#Extracted boundary surface constrained edges # Boundary nodes Optimization process
Volume meshing	47.56	266.52	# Constrained facets # Nodes Optimization process
Total	85.67	373.61	

Table 1: The mesh generation times and the parameters on which the subprocesses depend. The computing times mentioned here are obtained for mesh generations done on a ThinkPad P50 i7 6700HQ, 32 GB RAM coded in MATLAB. Considering that the generator was coded in an interpreted language, the evolution of the computation times should be analysed as a function of the mesh refinement and among the process relative to each other, rather than according to their absolute value.

an interpreted language. However, optimization processes require looping on the elements leading to higher computation times.

5 Applications

In order to study more realistically the influence of small scale heterogeneities on the macroscopic behavior of heterogeneous materials, complex microstructural geometries have to be obtained.

Some of these RVEs are discretized in this section in order to show the capabilities of the new mesh generator. Three types of RVEs are shown for illustration, (i) inclusion-based media [55], (ii) woven composite [57, 62], and (iii) random geometry obtained with excursion sets of random fields.

The first example is an RVE generated by DN-RSA developed by [55]. This tool is based on a distance-controlled random sequential addition algorithm. It has the capability to generate inclusion-based microstructures with large size distributions and arbitrary shapes with precise control on neighboring distances. This example illustrates a periodic inclusion-based material composed by 222 inclusions of various shapes and sizes with an inclusion volume fraction of 50,87%. The size distribution of inclusions varies between 0.05 and 0.35 for a cubic RVE of size 1. The resulting mesh shows a high quality distribution of elements with less than 1.5% of elements with quality lower than 30% (cf. Equation (4)). No slivers affecting FEM simulations are found.

The second example is an RVE of a complex periodic three dimensional textile reinforced composite RVE with very small gaps between yarns as illustrated in Figure 22. By using level set functions, the generation tool has the ability to

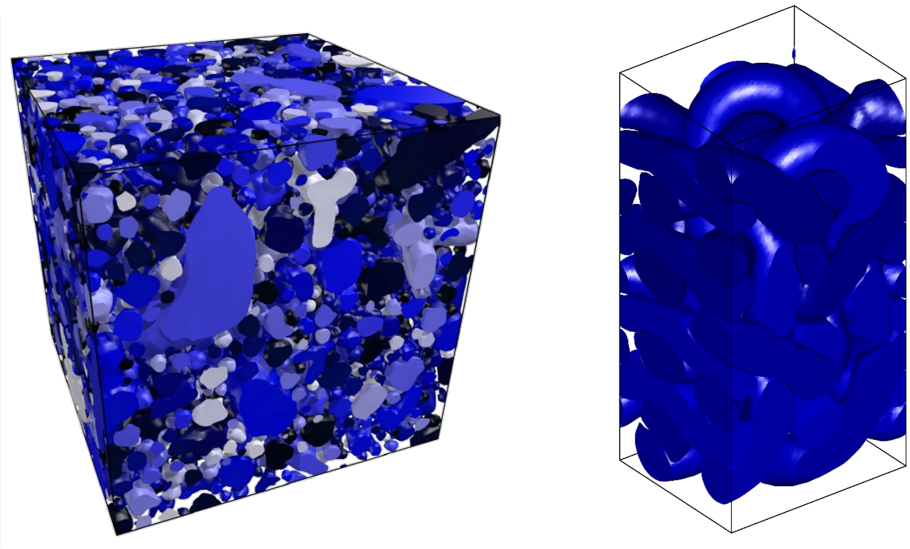

Fig. 20: Examples of different types of RVEs generated by [55, 56, 57] : inclusion-based microstructure (left) and a woven composite (right)

control the gap thickness and to remove automatically any residual interpenetration, while controlling the volume fraction of each familily of yarns in the RVE, that traditionnally other methods fail. More details about the generation of the geometry for this application can be found in [57] and [62].

The meshing methodology developed here can be naturally extended to discretize geometries obtained from RVEs generated by other techniques or from image-based CT scans [6]. Recently, RVE generation methods based on excursion sets of correlated Random Fields (RFs) with morphological control were developed in [42], producing complex randomly shaped heterogeneous material at different scales. These excursion sets, or thresholding of an RF, can be statistically controlled both geometrically (volume and surface area) and topologically (Euler characteristic) by linking analytically these to the statistical parameters of an RF, see [1] for more details. This tool leads to the generation of geometries such as matrix/inclusion morphologies or porous materials, involving opened or closed porosity as well as representing grain or pore size distributions. Figure 23 illustrates a conforming mesh of a RVE generated by excursion sets of Random Fields.

6 Discussion

The examples shown in the Applications section show the generator's ability of the generator to mesh complex geometries of very different shapes. The use of distance fields provides the necessary information to refine the areas of interest to take into account the geometrical complexity associated to rather high curvatures or narrow matrix zones between neighbouring inclusions. However, inaccuracies may appear in the evaluation of the orientation of the normal to interfaces and

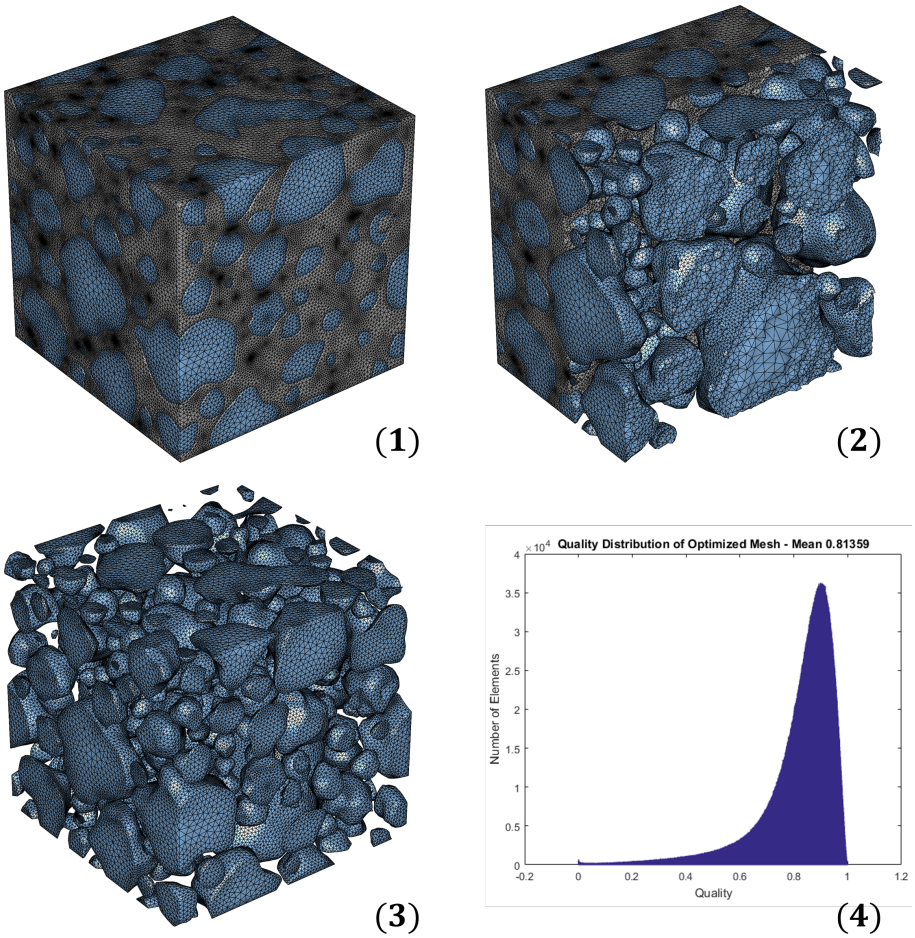

Fig. 21: Final periodic and conform mesh of the RVE (#Nodes=1 236 685/#Elements=7 501 711) : (1) Global view of the optimized mesh for both phases (inclusions/matrix), (2) Cut view of the inclusion-based RVE, (3) Inclusions meshes only, (4) Quality distribution - Mean quality : 81.36%

therefore on the nodal reaction forces on the inclusions surfaces. This may lead to self-intersections in the resulting mesh. It is a known issue addressed already by several contributions, see [2, 30]. For this case, these irregularities comes from the grid resolution of the input geometry compared with its curvature. In fact, refining locally the mesh would not solve the problem as the gradient quality can not be better than the initial grid resolution. A solution is to smoothen locally the level set function if the problem appears loosing in return locally some of the details. Another solution is to implement a predictor-corrector process to stabilize the forward euler resolution scheme.

In relative terms, with the procedure implemented here, the presence of self-intersections in particularly complex shapes is lower than 0.01 % of the total

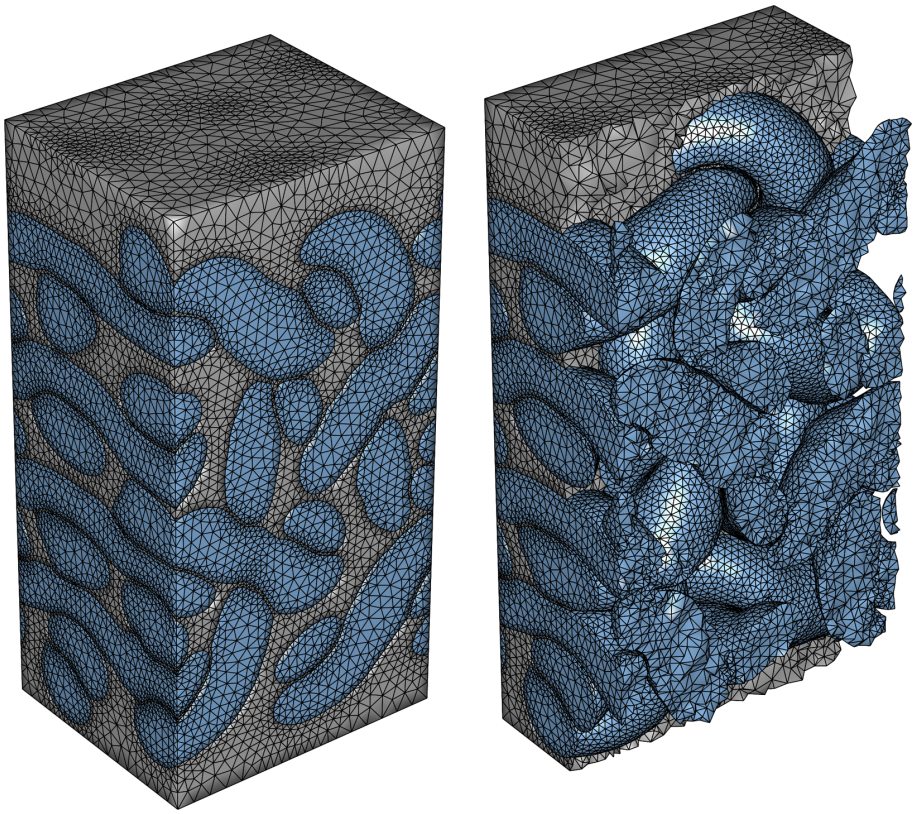

Fig. 22: Final periodic and conform mesh of the RVE (#Nodes=119 096/#Elements=729 350) - *Left* : Global view of the optimized mesh for both phases (inclusions/matrix) - *Right* Cut view of the woven composite RVE

number of surface elements. A simple solution proposed here is to remove them from the surface triangulation, which results in the opening of surfaces initially closed. When the volumic CDT is applied, some parts of the surface are therefore not constrained anymore. During, the optimization process moving the nodes to reach the targeted length size defined by $h(\mathbf{x})$, it is possible to satisfy the empty circumsphere criterion of a Delaunay triangulation and to preserve the conformity of the very small missing part by avoiding crossing elements even if CDT is not applied there.

As presented in Section 4.4, the constraint of using global level-sets functions is that they do not preserve the sharp features when an explicit facetted geometry is extracted by a Marching cubes algorithm [31]. In the context of this article, the case of sharp edges has therefore not been addressed because it requires a dedicated development. An implicit geometrical description that preserves sharp edges, as present for instance in open cell metallic foams, was proposed in [56]. It is based on the slicing of distance fields or level sets by ad hoc functions to describe sharp edges. The mesh generator proposed here could perfectly be extended to account

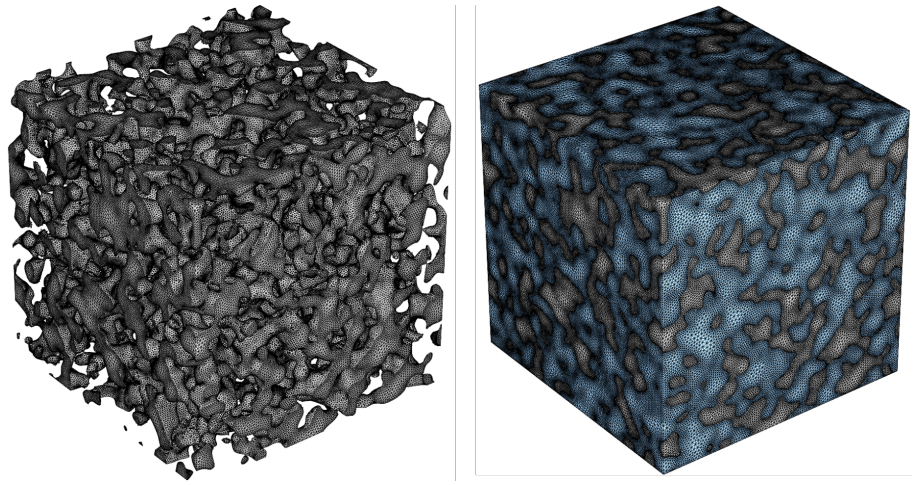

Fig. 23: Final conform mesh of the RVE based on excursion sets with a cube of size $a = 1$, $\sigma^2 = 1$, $l_c = 0.04$ (#Nodes=2 398 675/#Elements=14 601 647). The threshold is taken at 0.45 leading to a porosity of 33%. : *Left* : Matrix - *Right* Matrix/Inclusions

for such features. It would consist in resorting to the methodology proposed in [56] to extract the sharp edges.

Also, the use of an improved marching cubes for the extraction of isosurfaces from scattered datas or an octree-based dual contouring method [63, 64] could be used in future developments to produce a better initial adapted surface triangulation allowing a faster convergence of the optimization and potentially allowing preventing non-manifold triangulations.

The subsequent attribution of elements to proper material phases is done thanks to TetGen in most general cases. For inclusions with removed self-intersections, the detection of closed surfaces is no longer possible. For these particular inclusions, the attribution of elements to the material phases is done by interpolating the level set function value at the centroid of the tetrahedra.

As seen in Section 5, due to the complexity of the shapes, a few bad-shaped tetrahedra may still be present after the optimization process. Those elements are mainly located in areas with strong curvatures (requiring greater refinement) as well as at the neighborhood of the intersection between the external boundary faces of the RVE and surfaces of the inclusions. For the latter, the boundaries may exhibit sharp edges that give rise to very flat elements. Taking into account the proximity to the boundary faces through selective refinement without breaking the periodicity of the mesh would drastically reduce the number of poor quality elements. The example of inclusion based medium shows that less than 1.5% of elements have a quality lower than 30%. Even if the ratio of bad-shaped elements remains very low, this issue could become critical in simulations involving finite deformations or strong anisotropy. A solution of tetrahedral mesh improvement method presented by [22] that uses a broader set of operations such as topological

transformation could be a good option for further improvements to get rid from these few low quality elements.

7 Conclusion

The present contribution presents a new conforming mesh generation methodology for 2D and 3D periodic (or not) complex heterogeneous RVEs. The implementation is adapted and optimized for the RVEs generatorion tools developed by [55] in order to propose an integrated approach. However, a natural extension can be built for general implicit geometries obtained from other geometry generation or from experimental techniques such as CT scans.

The newly developed approach is an iterative Delaunay mesh generator based on an extended Persson-Strang truss analogy optimization process. Such an approach, based on signed distance fields, carries the advantage that the level set information used during the generation of the geometry of the microstructure by [55] can seamlessly be used in the subsequent discretization procedure.

The meshing process is hierarchical and aims at generating a triangulation, optimizing and constraining progressively interfaces, boundary faces and the volume. It offers a specific control on the inherent specificities of each part and leads to the generation of high quality FEM meshes.

On the internal surfaces, nodes are preventing from moving outside by systematically constraining their normal movement acting like boundary reactions while tension/compression forces act in the bars to reach the desired lengths defined by the size function. The latter allows optimizing the node distribution as a function of geometrical features such as curvature, nearness and narrowness. The distorsion of the elements is reduced by using a gradient limiting factor to better control the growing elements size. Periodicity is ensured by meshing independently non opposite master RVE faces before copying, translating and merging them to form the periodic enclosing box while conformity is ensured by using the Constrained Delaunay Tetrahedralization.

Acknowledgements The first author acknowledges the support of FRIA under Grant No. 29340757.

References

1. Adler, R.J.: Some new random field tools for spatial analysis. Stochastic Environmental Research and Risk Assessment **22**(6), 809 (2008). DOI 10.1007/s00477-008-0242-6. URL https://doi.org/10.1007/s00477-008-0242-6
2. Attene, M.: Direct repair of self-intersecting meshes. Graphical Models **76**(6), 658 – 668 (2014). DOI http://dx.doi.org/10.1016/j.gmod.2014.09.002. URL http://www.sciencedirect.com/science/article/pii/S1524070314000496
3. Bottasso, C.L., Detomi, D., Serra, R.: The ball-vertex method: a new simple spring analogy method for unstructured dynamic meshes. Computer Methods in Applied Mechanics and Engineering **194**(39), 4244 – 4264 (2005). DOI http://dx.doi.org/10.1016/j.cma.2004.08.014. URL http://www.sciencedirect.com/science/article/pii/S0045782504005201

4. Boyd, S.K., Mller, R.: Smooth surface meshing for automated finite element model generation from 3d image data. Journal of Biomechanics **39**(7), 1287 – 1295 (2006). DOI http://dx.doi.org/10.1016/j.jbiomech.2005.03.006. URL http://www.sciencedirect.com/science/article/pii/S0021929005001442
5. George, P.L., Borouchaki, H.: Delaunay Triangulation and Meshing. Hermès, Paris (1998)
6. Hashemi, M.A., Khaddour, G., François, B., Massart, T.J., Salager, S.: A tomographic imagery segmentation methodology for three-phase geomaterials based on simultaneous region growing. Acta Geotechnica **9**(5), 831–846 (2014). DOI 10.1007/s11440-013-0289-5. URL https://doi.org/10.1007/s11440-013-0289-5
7. Drach, A., Drach, B., Tsukrov, I.: Processing of fiber architecture data for finite element modeling of 3d woven composites. Advances in Engineering Software **72**, 18 – 27 (2014). DOI http://dx.doi.org/10.1016/j.advengsoft.2013.06.006. URL http://www.sciencedirect.com/science/article/pii/S0965997813000987. Special Issue dedicated to Professor Zdenk Bittnar on the occasion of his Seventieth Birthday: Part 2
8. Field, D.A.: Laplacian smoothing and delaunay triangulations. Communications in Applied Numerical Methods **4**(6), 709–712 (1988). DOI 10.1002/cnm.1630040603. URL http://dx.doi.org/10.1002/cnm.1630040603
9. Field, D.A.: Qualitative measures for initial meshes. International Journal for Numerical Methods in Engineering **47**(4), 887–906 (2000). DOI 10.1002/(SICI)1097-0207(20000210)47:4¡887::AID-NME804¿3.0.CO;2-H. URL http://dx.doi.org/10.1002/(SICI)1097-0207(20000210)47:4¡887::AID-NME804¿3.0.CO;2-H
10. Freitag, L.A., Ollivier-Gooch, C.: Tetrahedral mesh improvement using swapping and smoothing. International Journal for Numerical Methods in Engineering **40**(21), 3979–4002 (1997). DOI 10.1002/(SICI)1097-0207(19971115)40:21¡3979::AID-NME251¿3.0.CO;2-9. URL http://dx.doi.org/10.1002/(SICI)1097-0207(19971115)40:21¡3979::AID-NME251¿3.0.CO;2-9
11. Frey, P., George, P.: Mesh Generation: Application to Finite Elements. Hermes Science (2000). URL https://books.google.be/books?id=kwAgL0jgEXwC
12. Fritzen, F., Bhlke, T.: Periodic three-dimensional mesh generation for particle reinforced composites with application to metal matrix composites. International Journal of Solids and Structures **48**(5), 706 – 718 (2011). DOI http://dx.doi.org/10.1016/j.ijsolstr.2010.11.010. URL http://www.sciencedirect.com/science/article/pii/S0020768310004099
13. Gargallo-Peiró, A., Roca, X., Peraire, J., Sarrate, J.: Distortion and quality measures for validating and generating high-order tetrahedral meshes. Engineering with Computers **31**(3), 423–437 (2015). DOI 10.1007/s00366-014-0370-1. URL http://dx.doi.org/10.1007/s00366-014-0370-1
14. Geuzaine, C., Remacle, J.F.: Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering **79**(11), 1309–1331 (2009). DOI 10.1002/nme.2579. URL http://dx.doi.org/10.1002/nme.2579
15. TetMesh GHS3D: A powerful isotropic tet-mesher, version 4.2 (2010). URL http://www-roc.inria.fr/gamma/gamma/ ghs3d/ghs.php

16. Goldman, R.: Curvature formulas for implicit curves and surfaces. Computer Aided Geometric Design **22**(7), 632 – 658 (2005). DOI http://dx.doi.org/10.1016/j.cagd.2005.06.005. URL http://www.sciencedirect.com/science/article/pii/S0167839605000737. Geometric Modelling and Differential Geometry
17. Grail, G., Hirsekorn, M., Wendling, A., Hivet, G., Hambli, R.: Consistent finite element mesh generation for meso-scale modeling of textile composites with preformed and compacted reinforcements. Composites Part A: Applied Science and Manufacturing **55**, 143 – 151 (2013). DOI http://dx.doi.org/10.1016/j.compositesa.2013.09.001. URL http://www.sciencedirect.com/science/article/pii/S1359835X1300239X
18. Hemes, S., Desbois, G., Urai, J.L., Schrppel, B., Schwarz, J.O.: Multi-scale characterization of porosity in boom clay (hades-level, mol, belgium) using a combination of x-ray -ct, 2d bib-sem and fib-sem tomography. Microporous and Mesoporous Materials **208**, 1 – 20 (2015). DOI http://dx.doi.org/10.1016/j.micromeso.2015.01.022. URL http://www.sciencedirect.com/science/article/pii/S1387181115000347
19. Hormann, K.: From scattered samples to smooth surfaces. In: Proceedings of Geometric Modeling and Computer Graphics (2003)
20. J. Bossen, F., Heckbert, P.: A pliant method for anisotropic mesh generation. In: Proceedings of the 5th International Meshing Roundtable, pp. 63–74 (1996)
21. Klaas, O., Beall, M.W., Shephard, M.S.: Construction of Models and Meshes of Heterogeneous Material Microstructures from Image Data, pp. 171–193. Springer Netherlands, Dordrecht (2013). DOI 10.1007/978-94-007-4255-0_10. URL http://dx.doi.org/10.1007/978-94-007-4255-0_10
22. Klingner, B.M., Shewchuk, J.R.: Aggressive Tetrahedral Mesh Improvement, pp. 3–23. Springer Berlin Heidelberg, Berlin, Heidelberg (2008). DOI 10.1007/978-3-540-75103-8_1. URL http://dx.doi.org/10.1007/978-3-540-75103-8_1
23. Koko, J.: A matlab mesh generator for the two-dimensional finite element method. Applied Mathematics and Computation **250**, 650 – 664 (2015). DOI http://dx.doi.org/10.1016/j.amc.2014.11.009. URL http://www.sciencedirect.com/science/article/pii/S0096300314015148
24. Kouznetsova, V., Brekelmans, W.A.M., Baaijens, F.P.T.: An approach to micro-macro modeling of heterogeneous materials. Computational Mechanics **27**(1), 37–48 (2001). DOI 10.1007/s004660000212. URL http://dx.doi.org/10.1007/s004660000212
25. Kouznetsova, V., Geers, M.G., Brekelmans, W.M.: Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. International Journal for Numerical Methods in Engineering **54**(8), 1235–1260 (2002)
26. Kouznetsova, V., Geers, M.G.D., Brekelmans, W.A.M.: Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. International Journal for Numerical Methods in Engineering **54**(8), 1235–1260 (2002). DOI 10.1002/nme.541. URL http://dx.doi.org/10.1002/nme.541
27. Legrain, G., Allais, R., Cartraud, P.: On the use of the extended finite element method with quadtree/octree meshes. International Journal for Numerical Methods in Engineering **86**(6), 717–743 (2011). DOI 10.1002/nme.3070. URL

http://dx.doi.org/10.1002/nme.3070
28. Legrain, G., Cartraud, P., Perreard, I., Mos, N.: An x-fem and level set computational approach for image-based modelling: Application to homogenization. International Journal for Numerical Methods in Engineering **86**(7), 915–934 (2011). DOI 10.1002/nme.3085. URL http://dx.doi.org/10.1002/nme.3085
29. Lin, T.J., Guan, Z.Q., Chang, J.H., Lo, S.H.: Vertex-ball spring smoothing: An efficient method for unstructured dynamic hybrid meshes. Computers & Structures **136**, 24 – 33 (2014). DOI http://dx.doi.org/10.1016/j.compstruc.2014.01.028. URL http://www.sciencedirect.com/science/article/pii/S004579491400039X
30. Lo, D.: Finite Element Mesh Generation. Taylor & Francis (2015). URL https://books.google.be/books?id=5eeCZwEACAAJ
31. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. SIGGRAPH Comput. Graph. **21**(4), 163–169 (1987). DOI 10.1145/37402.37422. URL http://doi.acm.org/10.1145/37402.37422
32. Paul Louis George: Improvements on delaunay-based three-dimensional automatic mesh generator. Finite Elements in Analysis and Design **25**(3), 297 – 317 (1997). DOI https://doi.org/10.1016/S0168-874X(96)00063-7. URL http://www.sciencedirect.com/science/article/pii/S0168874X96000637. Adaptive Meshing, Part 2
33. Macri, M., De, S.: An octree partition of unity method (octpum) with enrichments for multiscale modeling of heterogeneous media. Computers & Structures **86**(7), 780 – 795 (2008). DOI http://dx.doi.org/10.1016/j.compstruc.2007.06.001. URL http://www.sciencedirect.com/science/article/pii/S0045794907002234
34. Massart, T.J., Selvadurai, A.P.S.: Computational modelling of crack-induced permeability evolution in granite with dilatant cracks. International Journal of Rock Mechanics and Mining Sciences **70**, 593 – 604 (2014). DOI http://dx.doi.org/10.1016/j.ijrmms.2014.06.006. URL http://www.sciencedirect.com/science/article/pii/S1365160914001506
35. Masson, R., Bornert, M., Suquet, P., Zaoui, A.: An affine formulation for the prediction of the effective properties of nonlinear composites and polycrystals. Journal of the Mechanics and Physics of Solids **48**(6), 1203 – 1227 (2000). DOI http://dx.doi.org/10.1016/S0022-5096(99)00071-X. URL http://www.sciencedirect.com/science/article/pii/S002250969900071X
36. Moës, N., Dolbow, J., Belytschko, T.: A finite element method for crack growth without remeshing. International Journal for Numerical Methods in Engineering **46**(1), 131–150 (1999). DOI 10.1002/(SICI)1097-0207(19990910)46:1¡131::AID-NME726¿3.0.CO;2-J. URL http://dx.doi.org/10.1002/(SICI)1097-0207(19990910)46:1¡131::AID-NME726¿3.0.CO;2-J
37. Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces. Applied Mathematical Sciences. Springer New York (2006). URL https://books.google.be/books?id=i4bfBwAAQBAJ
38. Paiva, A., Lopes, H., Lewiner, T., Figueiredo, L.H.D.: Robust adaptive meshes for implicit surfaces. In: 2006 19th Brazilian Symposium on Computer Graphics and Image Processing, pp. 205–212 (2006). DOI 10.1109/SIBGRAPI.2006.40

39. Persson, P.O.: Mesh generation for implicit geometries. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA (2005). AAI0807802
40. Persson, P.O., Strang, G.: A simple mesh generator in matlab. SIAM Review **46**, 2004 (2004)
41. Potter, E., Pinho, S., Robinson, P., Iannucci, L., McMillan, A.: Mesh generation and geometrical modelling of 3d woven composites with variable tow cross-sections. Computational Materials Science **51**(1), 103 – 111 (2012). DOI http://dx.doi.org/10.1016/j.commatsci.2011.06.034. URL http://www.sciencedirect.com/science/article/pii/S0927025611003764
42. Roubin, E., Colliat, J.B., Benkemoun, N.: Meso-scale modeling of concrete: A morphological description based on excursion sets of random fields. Computational Materials Science **102**, 183 – 195 (2015). DOI http://dx.doi.org/10.1016/j.commatsci.2015.02.039. URL http://www.sciencedirect.com/science/article/pii/S0927025615001408
43. Rusinkiewicz, S.: Estimating curvatures and their derivatives on triangle meshes. In: Proceedings of the 3D Data Processing, Visualization, and Transmission, 2Nd International Symposium, 3DPVT '04, pp. 486–493. IEEE Computer Society, Washington, DC, USA (2004). DOI 10.1109/3DPVT.2004.54. URL http://dx.doi.org/10.1109/3DPVT.2004.54
44. Sanchez-Palencia, E.: Non-Homogeneous Media and Vibration Theory. Lecture Notes in Physics. Springer, Berlin (1980). URL https://cds.cern.ch/record/1391389
45. Schöberl, J.: Netgen an advancing front 2d/3d-mesh generator based on abstract rules. Computing and Visualization in Science **1**(1), 41–52 (1997). DOI 10.1007/s007910050004. URL http://dx.doi.org/10.1007/s007910050004
46. Selvadurai, A.P.S., Gowacki, A.: Permeability hysterisis of limestone during isotropic compression. Ground water **46**(1), 113119 (2008). DOI 10.1111/j.1745-6584.2007.00390.x. URL https://doi.org/10.1111/j.1745-6584.2007.00390.x
47. Sethian, J.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press (1999). URL https://books.google.be/books?id=ErpOoynE4dIC
48. Shewchuk, J.R.: Constrained delaunay tetrahedralizations and provably good boundary recovery. In: Proceedings of the 11th International Meshing Roundtable, IMR 2002, Ithaca, New York, USA, September 15-18, 2002, pp. 193–204 (2002). URL http://imr.sandia.gov/papers/abstracts/Sh256.html
49. Shewchuk, J.R.: Lecture notes on delaunay mesh generation. University of California, Department of Electrical Engineering and Computer Sciences (2012)
50. Si, H.: Constrained delaunay tetrahedral mesh generation and refinement. Finite Elem. Anal. Des. **46**(1-2), 33–46 (2010). DOI 10.1016/j.finel.2009.06.017. URL http://dx.doi.org/10.1016/j.finel.2009.06.017
51. Si, H.: Tetgen, a delaunay-based quality tetrahedral mesh generator. ACM Trans. Math. Softw. **41**(2), 11:1–11:36 (2015). DOI 10.1145/2629697. URL http://doi.acm.org/10.1145/2629697
52. Si, H., Gärtner, K.: 3D Boundary Recovery by Constrained Delaunay Tetrahedralization. Preprint. WIAS (2010). URL https://books.google.be/books?id=v4zlZwEACAAJ

53. der Sluis, O., Vosbeek, P., Schreurs, P., jer, H.: Homogenization of heterogeneous polymers. International Journal of Solids and Structures **36**(21), 3193 – 3214 (1999). DOI http://dx.doi.org/10.1016/S0020-7683(98)00144-9. URL http://www.sciencedirect.com/science/article/pii/S0020768398001449
54. Song, Y., Davy, C., Troadec, D., Blanchenet, A.M., Skoczylas, F., Talandier, J., Robinet, J.: Multi-scale pore structure of cox claystone: Towards the prediction of fluid transport. Marine and Petroleum Geology **65**, 63 – 82 (2015). DOI http://dx.doi.org/10.1016/j.marpetgeo.2015.04.004. URL http://www.sciencedirect.com/science/article/pii/S0264817215001245
55. Sonon, B., François, B., Massart, T.J.: A unified level set based methodology for fast generation of complex microstructural multi-phase rves. Computer Methods in Applied Mechanics and Engineering **223**, 103 – 122 (2012). DOI http://dx.doi.org/10.1016/j.cma.2012.02.018. URL http://www.sciencedirect.com/science/article/pii/S0045782512000606
56. Sonon, B., François, B., Massart, T.J.: An advanced approach for the generation of complex cellular material representative volume elements using distance fields and level sets. Computational Mechanics **56**(2), 221–242 (2015). DOI 10.1007/s00466-015-1168-8. URL http://dx.doi.org/10.1007/s00466-015-1168-8
57. Sonon, B., Massart, T.J.: A level-set based representative volume element generator and xfem simulations for textile and 3d-reinforced composites. Materials **6**(12), 5568–5592 (2013). DOI 10.3390/ma6125568. URL http://www.mdpi.com/1996-1944/6/12/5568
58. Stroeven, M., Askes, H., Sluys, L.: Numerical determination of representative volumes for granular materials. Computer Methods in Applied Mechanics and Engineering **193**(30), 3221 – 3238 (2004). DOI http://dx.doi.org/10.1016/j.cma.2003.09.023. URL http://www.sciencedirect.com/science/article/pii/S0045782504001331. Computational Failure Mechanics
59. Sukumar, N., Chopp, D., Moës, N., Belytschko", T.: Modeling holes and inclusions by level sets in the extended finite-element method. Computer Methods in Applied Mechanics and Engineering **190**(46-47), 6183–6200 (2001). DOI 10.1016/S0045-7825(01)00215-8
60. Sun, W.C., Andrade, J.E., Rudnicki, J.W.: Multiscale method for characterization of porous microstructures and their impact on macroscopic effective permeability. International Journal for Numerical Methods in Engineering **88**(12), 1260–1279 (2011). DOI 10.1002/nme.3220. URL http://dx.doi.org/10.1002/nme.3220
61. Terada, K., Hori, M., Kyoya, T., Kikuchi, N.: Simulation of the multi-scale convergence in computational homogenization approaches. International Journal of Solids and Structures **37**(16), 2285 – 2311 (2000). DOI http://dx.doi.org/10.1016/S0020-7683(98)00341-2. URL http://www.sciencedirect.com/science/article/pii/S0020768398003412
62. Wintiba, B., Sonon, B., Kamel, K.E.M., Massart, T.J.: An automated procedure for the generation and conformal discretization of 3d woven composites rves. Composite Structures **180**(Supplement C), 955 – 971 (2017). DOI https://doi.org/10.1016/j.compstruct.2017.08.010. URL http://www.sciencedirect.com/science/article/pii/S0263822317319190

63. Zhang, Y., Bajaj, C., Sohn, B.: 3d finite element meshing from imaging data. Computer Methods in Applied Mechanics and Engineering **194**(48-49), 5083–5106 (2005). DOI 10.1016/j.cma.2004.11.026
64. Zhang, Y., Hughes, T.J., Bajaj, C.L.: An automatic 3d mesh generation method for domains with multiple materials. Computer Methods in Applied Mechanics and Engineering **199**(5), 405 – 415 (2010). DOI http://dx.doi.org/10.1016/j.cma.2009.06.007. URL http://www.sciencedirect.com/science/article/pii/S004578250900214X. Computational Geometry and Analysis