A new finite element strategy to simulate microstructural evolutions

S. Florez[a,*], M. Shakoor[c], T. Toulorge[b], M. Bernacki[a]

[a] *Mines-ParisTech, PSL-Research University, CEMEF – Centre de mise en forme des matériaux, CNRS UMR 7635, CS 10207 rue Claude Daunesse, 06904 Sophia Antipolis Cedex, France*
[b] *Centre de Recherche en Aeronautique, Cenaero, Rue des Frères Wright 29, 6041 Gosselies, Belgium*
[c] *IMT Lille Douai, 941 rue Charles Bourseul CS 10838, 59508 Douai Cedex, France*

Abstract

The Level-set (LS) method has been shown to be a powerful approach to model dynamic interfaces in the context of large deformations. The LS method has been applied to the simulation of microstructural evolutions as Grain Growth (GG) and Recrystallization (ReX) at the mesoscale [1]. Interfaces between grains are implicitly described in an Eulerian framework, as the zero-isovalue of the LS fields and their evolution is governed by convective-diffusive partial differential equations (PDEs). The LS approach circumvents the notoriously difficult problem of generating interface-conforming meshes for geometries subjected to large displacements and to changes in the topology of the domains.

Generally, in order to maintain high accuracy when using the LS method, moving interfaces are generally captured by a locally refined FE mesh with the help of mesh adaptation techniques. In a microstructural problem, the large number of interfaces and the fine mesh required in their vicinity make the mesh adaptation process very costly in terms of CPU-time, particularly in 3D [2].

In this work, a different adaptation strategy is used. It maintains the benefits of the classical Eulerian LS framework, while enforcing at all times the

---

*Corresponding author
  Email address:* `sebastian.florez@mines-paristech.fr` (S. Florez)

conformity of the FE mesh to implicit interfaces by means of local remeshing operations, special treatments for vacuum regions have been adopted and will be presented within the generalization of a previous adaptation algorithm presented in [3]. Source of errors will be presented and compared for different test cases. Finally, we will illustrate how the new method decreases the requirement in mesh density while maintaining the accuracy at the interfaces, hence reducing the computational cost of the simulations.

---

1. Introduction

Because most virtual polycrystalline microstructure generation tools are based on the concept of Voronoï cells or Laguerre Voronoï cells [4–6], meshing virtual microstructures does not raise any major challenge. The generation of an interface-conforming mesh usually consists in discretizing cells facets, and then the volume within each cell [7]. This procedure is commonly used if the microstructure is subjected to small deformations in the subsequent simulation. However, for real polycrystals, observed using Scanning Electron Microscopy (SEM) or 3D X-ray imaging techniques [8, 9], a Voronoï/Laguerre-Voronoï space partitioning is, of course, not accessible. While there has been much research on real 3D microstructures meshing methods [10, 11], the generalization of these methods to massively multiphase materials such as polycrystals is not straightforward [12, 13]. The main challenge is linked to multiple junctions, namely interfaces between more than two grains where obtaining both high mesh quality and fidelity with respect to experimental data can be complex.

Once a mesh has been generated, modeling large plastic strains and subsequent microstructure evolutions such as recrystallization (ReX), grain growth (GG) or solid/solid phase transformations (SSPT) in a Finite Element (FE) framework is also very challenging. As a consequence, many researchers have chosen to avoid the use of meshes where grain boundaries are explicitly meshed (*i.e.,* with a conformal mesh), and instead use implicit interface approaches such as the Level-Set (LS) [14] or the multiphase field methods [15]. While results using explicit interface methods are restricted to limited deformations, implicit interface methods have given access to the modeling of a wider range of thermomechanical phenomena. For instance, the LS method has been successfully used to simulate static or dynamic recrystallization in context of large deformations [1, 16–18]. However, the absence of a conformal mesh at grain boundaries typically seems to require a finer discretization [19] which could be a difficulty in terms of numerical cost mainly in 3D [2]. Thus, there is an interest for alternative methods with similar capabilities and robustness as the LS method, but

based on explicit and reasonable interface meshing/remeshing.

In this work, a new methodology, based on a previous work originally applied to mechanical fracture problems [20], is proposed to generate conforming FE meshes from 2D and 3D images of real microstructures through the intermediate use of LS functions. Moreover, a new mesh adaption technique is then proposed to handle large deformations and displacements of grain boundary (GB) interfaces. The ability of generating and adapting conformal FE meshes thanks to the intermediate use of LS functions is particularly useful for the modeling of topological events such as grain disappearance during GG. An alternative method based on the full reconstruction of the computational mesh at each time step is presented in [21]. Here, all proposed algorithms are based only on local mesh modifications, and not full mesh reconstruction. The present article is dedicated to the application of this new FE strategy in the context of GG mechanism and to comparisons with a classical front capturing LS-FE framework [22–25]. A detailed description of the potential sources of errors during a simulation for each approach is presented, tested and compared by means of a specific test case featuring an analytical solution when subjected to capillarity effects, namely the sphere shrinkage case. Then, comparisons involving accuracy for more demanding test cases are given, and finally comparisons of CPU time are performed for a large 2D polycrystal involving 10000 initial grains.

2. Interface treatment

2.1. Level-set modeling of polycrystals

Based on earlier developments [26], grayscale data can be transferred from a 2D or 3D image (*i.e.,* a regular grid) to an initial uniform mesh of the FE domain Ω. If these grayscale values are transferred to mesh nodes, the interface between the two components Ω_- and Ω_+ of the microstructure can be easily localized based on the iso-contours of the grayscale field $\tilde{\phi}$. If g_I is the threshold value between the two components (*i.e.,* $\Omega_- = \left\{ x \in \Omega, \tilde{\phi}(x) < g_I \right\}$, $\Omega_+ = \left\{ x \in \Omega, \tilde{\phi}(x) \geq g_I \right\}$), then the interface is defined by

$$\Gamma = \left\{ x \in \Omega, \tilde{\phi}(x) = g_I \right\}.$$

In general, the grayscale field $\tilde{\phi}$ will be transformed into a distance function ϕ thanks to the so-called LS reinitialization operation [27]. The obtained LS function is then a signed distance function to the interface:

$$\begin{cases} \phi(x) = -\mathrm{d}(x, \Gamma) & x \in \Omega_- \\ \phi(x) = +\mathrm{d}(x, \Gamma) & x \in \Omega_+ \\ \phi(x) = 0 & x \in \Gamma \end{cases}$$

where $\mathrm{d}(x, \Gamma)$ is the Euclidean distance between a point and the interface being the grayscale value at which phase transition occurs. For applications to polycrystals, multiple LS functions $(\phi_i)_{i=1...N}$ are used [1]. Independently of how grayscale values are transferred to the FE mesh, and how LS reinitialization is performed, vacuum regions between grains will usually remain. In other words, if the procedure described above is used as is, the set $\Omega_\emptyset = \left\{ x \in \Omega, \max_{i=1...N} \phi_i(x) < 0 \right\}$ will not be empty. Note that it is not possible to have mesh nodes inside $\Omega_\emptyset$, because LS functions values at mesh nodes are computed from grayscale values of the initial image, which are considered as free

---

[1]N is not necessarily the number of grains, as coloring techniques can be used to gather different grains into the same LS function and dynamically adapt this coloring throughout the simulation [17].
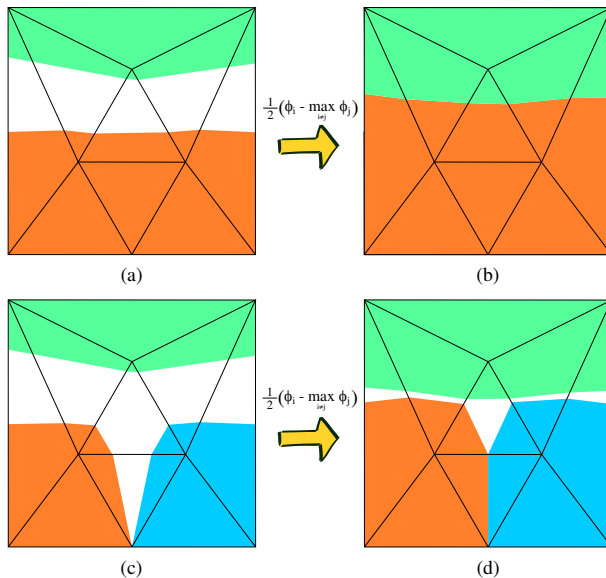
Figure 1: Global Treatment in order to eliminate non-physical vacuum regions on a FE discretization. Two colored LS (a) with no treatment of vacuum regions, (b) result after applying Eq. (1). Three colored LS (c) with no treatment of vacuum regions (d) result after applying Eq. (1).

of vacuum. Following the same reasoning, when generating the LS fields based on Voronoï cells or Laguerre-Voronoï cells over an implicit mesh, these vacuum regions will appear as well because the mesh can not ensure that some of the edges (or facets in 3D) will not be crossed by the zero isovalue of the LS functions.

The presence of non-physical vacuum regions at the multiple junctions with the LS method is well known and the following technique, proposed in [28], is classicaly used [17, 19, 23–25, 27] to treat it:

$$\hat{\phi}_i = \frac{1}{2}\left(\phi_i - \max_{j \neq i} \phi_j\right), \quad \forall i = 1 \ldots N, \tag{1}$$

where $\hat{\phi}_i$ is then used as the corrected LS function. The effect of this treatment is illustrated in Fig. 1. This equation is also valid in 3D.

As stated above, the set $\Omega_\emptyset$ does not contain mesh nodes, and if Eq. (1) is

6

applied between two LS functions as in Fig. 1(a), $\Omega_\emptyset$ is empty (see Fig. 1(b)). However, when applying the same procedure on three LS as in figure Fig. 1(c), it can be seen that this treatment does not totally remove vacuum regions (see Fig. 1(d)). Another technique is proposed in [29, 30] to overcome this problem, but it will not be considered here.

2.2. Conformal meshing of level sets

In the present work, a generalized version of the so-called interface fitting algorithm, introduced in a previous work [31], is proposed to remove the remaining vacuum regions while constructing a conform FE mesh of the interface. This algorithm is based on purely topological mesh operations, and can be easily extended to 3D. As the initial interface fitting algorithm consists simply in splitting mesh edges intersected by a LS function, and introducing this intersection as a new mesh node, it will not remove vacuum regions. Instead, entire mesh elements will be formed that do not belong to any grain. This is illustrated in Fig. 2(c) and (d).

2.3. Vaccum-less conformal meshing of grain boundaries

Let d be the number of spatial dimensions, we consider the hierarchical organization of simplicial meshes, i.e. each 3-simplex (tetrahedron) has 2-simplicial faces (triangles), which in turn have 1-simplicial edges (line segments). In order to obtain a mesh fitted to the grain boundaries without creating any vacuum region in the domain, the following general interface joining algorithm is proposed:

> **for all** $T = d \ldots 1$ **do**
> > **for all** T-simplices S of the mesh **do**
> > > Compute all intersections between edges of S and any interface
> > > Insert the barycenter of these intersections in the mesh by splitting S
> > > Set the LS function associated to any interface that intersected edges of S to zero on the inserted node
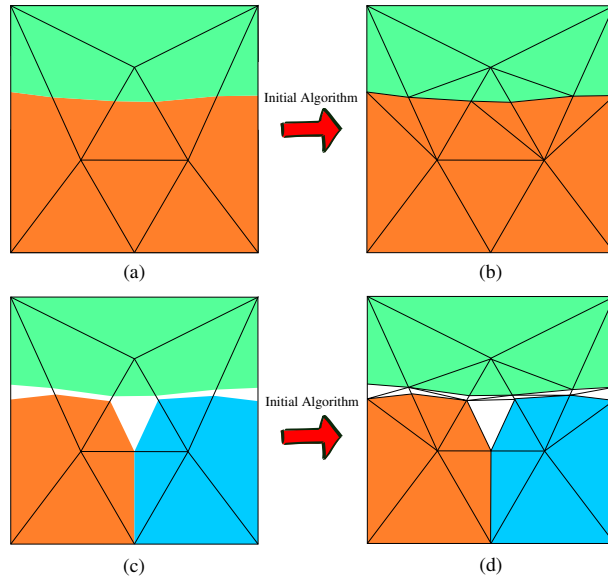> > **end for**

Figure 2: Example of FE discretization of the interfaces between two and three colored grains in 2D: (a) two implicit LS with no vacuum region, (c) three implicit LS with vacuum region and (c) and (d) results of the initial interface fitting algorithm on (a) and (c) respectively.

end for

A 2D example of the results obtained with this interface joining technique is shown in Fig. 3.

This procedure prevents elements of the mesh that are intersected by interfaces to remain partly filled with vacuum. Elements that do not contain a junction (or at least three phases) are also treated by the algorithm, which avoids the creation of elements entirely contained in vacuum. The output mesh is usually of poor quality, hence this interface joining process is generally followed by a mesh adaption step to restore a good element shape close to grain boundaries. This adaption may also follow some local mesh refinement criterion [20, 31]. An example of 2D result with a larger microstructure meshed using both the joining algorithm and mesh adaption is described in Fig. 4.
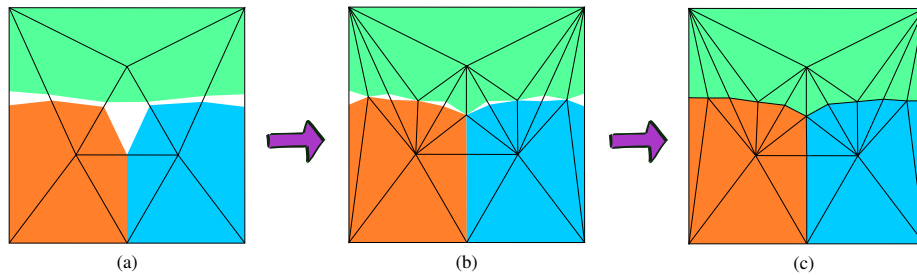
Figure 3: Solution of the three grains 2D problem: (a) result after using Eq. (1), (b) result after the first iteration of the joining algorithm, (c) result after the second and last iteration of the joining algorithm.
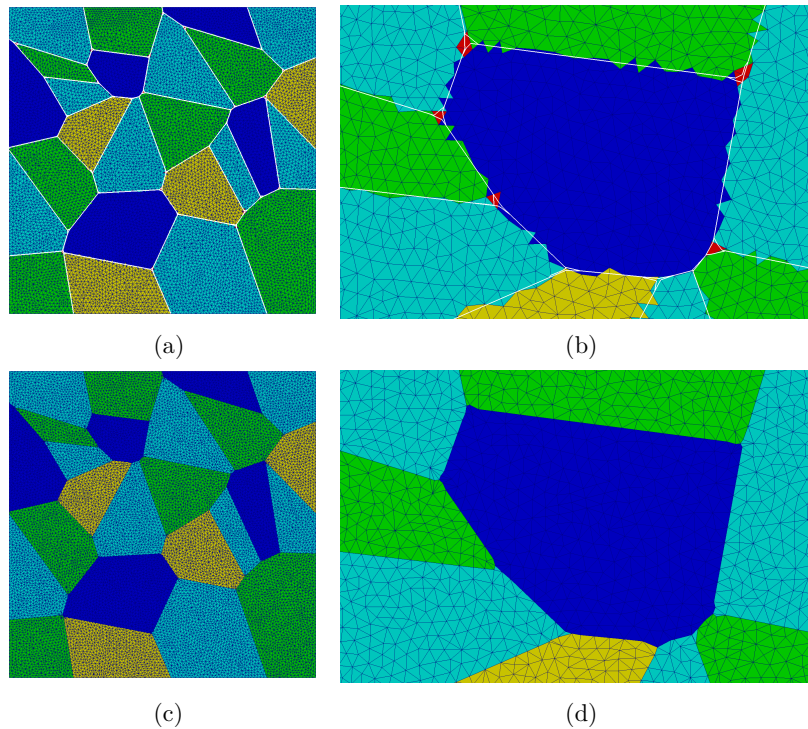

(a)

(b)

(c)

(d)

Figure 4: Example of a 2D polycrystal of 25 grains represented by 4 LS functions (blue, cyan, yellow, green). In (a) and (b), interfaces (white) are implicitly discretized and cross mesh elements, with elements containing vacuum regions (red) at some multiple junctions. In (c) and (d), interfaces are explicitly meshed using both interface joining and mesh adaption, so that vacuum regions are eliminated in the final mesh, with no significant deterioration of element shape.

9

3. Grain Growth Modeling

3.1. Model and Numerical Method

3.1.1. Governing Equations

In first approximation, GG by capillarity can be described by a pure advective process. At the polycrystal scale, the velocity $\vec{v}$ at every point on the interfaces can be approximated by the following relationship:

$$\vec{v} = -M\gamma\kappa\vec{n}, \tag{2}$$

where M is the mobility of the interface, γ the grain boundary energy, κ the local mean curvature (i.e. the curvature in 2D and sum of the main curvatures in 3D) and $\vec{n}$ the outgoing unit normal to the grain interface. The isotropy hypothesis remains here to consider M as only dependant of the temperature and γ as constant. In this context, the substitution of Eq. (2) into an advection equation for each level-set function ϕ_i results in:

$$\frac{\partial\phi_i}{\partial t} - M\gamma\kappa\vec{n} \cdot \vec{\nabla}\phi_i = 0 \tag{3}$$

Most of the methods described hereafter can be extended to the context of anisotropic grain boundary energy γ or mobility M [32], but this will not be considered here.

Eq. (3) is notoriously difficult to properly solve by usual numerical methods. The local curvature κ involves second derivatives of the level-set function ϕ_i, whose numerical estimate is very irregular, despite the large number of methods that have been proposed in the literature to obtain smooth and accurate approximations (see for instance [33–35]). Moreover, the computation of the curvature is often performed as a post-processing of the level-set function solution and not as an inherent part of the numerical scheme used to discretize Eq. (3). The time-explicit nature of this staggered approach implies the use of very small time steps because of restrictive stability conditions.

This is why it is preferred in this work to rewrite Eq. (3) by adding an additional assumption: the LS fields ϕ_i remain at all times signed distance

functions ($|\vec{\nabla}\phi_i| = 1$) around their 0-isovalues during boundary migration. The resulting diffusive equation for the LS functions reads:

$$\frac{\partial \phi_i}{\partial t} - M\gamma\Delta\phi_i = 0 \tag{4}$$

Eq. (4) is in general much more stable than Eq. (3) and avoids the direct calculation of κ. It is solved by a standard linear Finite Element method in space combined with a backward Euler scheme in time. The implementation is fully parallel and has been shown to perform very well on a large number of processors [16].

3.1.2. Global Level-Set Functions and Reinitialization

In Sec. 2, the treatment of interfaces in a polycrystal has been described assuming that each level-set function ϕ_i represents the boundary of a grain. In simulations involving thousands of grains however, which are generally necessary for physical representativity in this context, the computational cost of the method becomes prohibitive because Eq. (4) must be solved thousands of times on the whole domain.

This problem can be overcome by the use of Global Level-Set (GLS) functions, which include multiple grains in a single distance field [2, 17]. Coloring/Recoloring techniques make it possible to identify sets of non-connecting grains that can share a GLS. In the case of a GG simulation, topological events (moslty grains vanishing) occur, which often lead non-connecting phases to become adjacent. This is why dynamic re-coloring methods [2, 17], which transfer grains between GLS functions, are used in the present work to ensure that the GLS functions remain valid. With this techniques, it is possible to limit, independently of the total number of grains, the number of GLS functions (and thus the number of diffusion equations to solve) to a few tens, which is computationally affordable.

An additional consideration is that, as mentioned in Sec.3.1.1, it is necessary to *reinitialize* the LS functions, i.e. to maintain them as signed distance functions. Many methods have been proposed in the literature to perform this

task, for instance fast marching methods on regular grids [36–38] or solving a Hamilton-Jacobi equation on unstructured meshes [39]. In the present work, we prefer to recompute the distance to the iso-0 level-set in a purely geometric manner with a parallel-efficient algorithm based on spatial partitioning trees [27].

3.1.3. Interface treatment and Remeshing

Three different combinations of interface representation and mesh adaptation strategies are compared in the numerical tests performed in this work (see Sec. 4).

The first approach consists in using a Static Mesh (SM), disabling mesh adaptation so that the mesh remains the same all along the simulation. The interface is then implicitly represented by the level-set functions.

In the second approach, interfaces are still described by level-set functions only, but they are better captured by locally refining the mesh in their vicinity through Isotropic Mesh Adaptation (IMA). The local refinement makes it possible to better resolve the curvature of the grain boundaries.

The third approach consists in applying the New Fitting And Joining Algorithm (NFJA) described in Sec. 2 to track interfaces explicitly with a body-fitted mesh. With this technique, a mesh adaptation step is still required in order to improve the low mesh quality resulting from the fitting procedure and to accurately capture the interface curvature through local mesh refinement.

The mesh adaptation procedure involved in the IMA and NFJA approaches relies on local topological mesh operations, that are applied iteratively with the objective of improving a mixed criterion [40]. The criterion combines an evaluation of the local element quality and the conformance of the local edge length to a prescribed size field. Anisotropic meshes could have been generated in the vicinity of the interface through metric-based techniques [41, 42]. However, we have found highly anisotropic meshes to be of little interest in this specific context: as level-set functions are linear in the normal direction to interfaces, the main driver of the accuracy is the mesh size in the tangential direction, that

determine the resolution of the interface curvature.

Considering a mesh size h_{int} in at the interface, the mesh size field h is defined by:

$$
h = \begin{cases}
h_{\text{int}} & \text{if } |\phi| \leq 4h_{\text{int}} \\
h_{\text{int}} + \dfrac{7(|\phi| - 4h_{\text{int}})}{4} & \text{if } |\phi| \in [4h_{\text{int}}, 8h_{\text{int}}] \\
8h_{\text{int}} & \text{if } |\phi| \geq 8h_{\text{int}}
\end{cases}
\tag{5}
$$

This definition makes it possible to obtain a band of four refined elements on each side of the interface and a coarser (8 times larger) mesh in the bulk of the phase.

3.1.4. General Algorithm

The general approach can be summarized as follows:

Generate Initial State with a Coloring Method
while $Time < FinalTime$ **do**
  for all LS Field ϕ **do**
    Solve PDEs with a FE method for ϕ
  end for
  for all LS Field ϕ **do**
    Apply equation 1 to ϕ
  end for
  for all LS Field ϕ **do**
    for all Grain G in ϕ **do**
      Transport G to another ϕ if necessary (Re-Coloring)
    end for
  end for
  for all LS Field ϕ **do**
    Reinitialize ϕ
  end for
  if Remeshing Active **then**

if Body-Fitted Remesh **then**

    Remesh with the NFJA method

end if

if Interface Capturing Remesh **then**

    Remesh with the IMA method

end if

end if

end while

3.2. Source of Errors

Each one of the numeric models has a different set of sources of errors that have been classified as follows: errors given by the direct reinitialization algorithm, errors given by the remeshing and transport process and finally, errors given by the resolution of the diffusion equation in the considered FE framework (P1, unstructured FE mesh). We will detail each one of the sources of errors excluding the ones obtained by the FE solution of the EDP which are considered as a function of the convergence parameter given to the FE solver and common to the different FE remeshing strategies proposed here.

3.2.1. Direct Reinitialization Errors

Using the direct reinitialization method proposed in [27] is a very fast way of reinitializing a LS field, however, some errors will be present when the FE discretization uses linear elements. Consider the configuration of figure 5(a), in this case, the zero-isovalue of the LS field (ϕ) is obtained by interpolating the LS values whithin the elements where a change of sign of the LS was found, a sequence of segments defining the interface is identified. Then, the reinitialization algorithm will recompute the distance of each one of the nodes to the nearest part of the nearest segment. Applying this procedure to node A, B and C of figure 5(a) will result in a shrinking of the concave phase: the initial LS values are given on A, B and C by the positive norm of vector $\vec{d_a}$ and the negative norm of vectors $\vec{d_b}$ and $\vec{d_c}$ ($|\vec{d_a}|$, -$|\vec{d_b}|$ and -$|\vec{d_c}|$). The zero isovalue

14

of the interpolation of these LS values within element K will be the segment s used to reinitialize nodes A, B and C. For nodes B and C, the value of the computed distance to segment S is the same as the initial LS values for these nodes ($|\vec{d_{bs}}| = |\vec{d_b}|$ and $|\vec{d_{cs}}| = |\vec{d_c}|$). However, the nearest distance of node A to the segment S is the norm of the vector $\vec{d_{as}}$ (which is ortogonal to S). As $|\vec{d_{as}}| < |\vec{d_a}|$, when interpolating the new reinitialized LS field, its zero-isovalue will be different from the initial one (see figure 5(b)). Normally, these errors increase when the curvature of the zero-isovalue increase, and they become null if the zero-isovalue is a straight line. A way to avoid this phenomena, is by using body-fitted interfaces as the ones obtained with the help of the fitting and joining algorithm presented in this paper.

3.2.2. Remeshing and Transport Errors

It is well known that the process of transporting a numeric field from a mesh to another causes errors, mainly if the interpolation used for the transport has a low order. Figure 6 shows one example of loss of surface when a mesh using linear elements transports one LS field to another mesh. Figure 6(a) shows the initial mesh with its given zero-isovalue of the LS field. Each node of the new mesh (Figure 6(b)) will compute its LS value from the interpolation of the LS field over the old mesh. Many case scenarios can occur. For instance, if one edge of the new mesh is completely inside of one of the elements of the old mesh (as in figure 7 see edge $\vec{zy}$), the old isovalues (including the zero-isovalue) over that edge will be transported to it as they intersect the same edge. In other words, the interpolation of the old LS to the nodes of that edge will cause that the isovalues computed in the old element along the edge and the new interpolation over the edge, coincide. However, this is not the case for all the edges of the new mesh, many of them will cross the edges of the old mesh (as in figure 7 see edge $\vec{xy}$). As the interpolation over each one of the nodes of those edges comes from different elements, the isovalues on those edges will not coincide with the interpolation of the LS of the old mesh along those edges, hence causing errors.

As for the reinitialization errors, transport errors will cause the concave
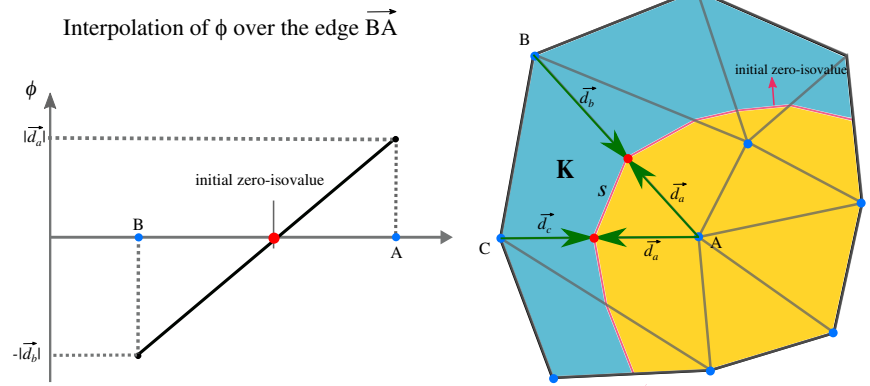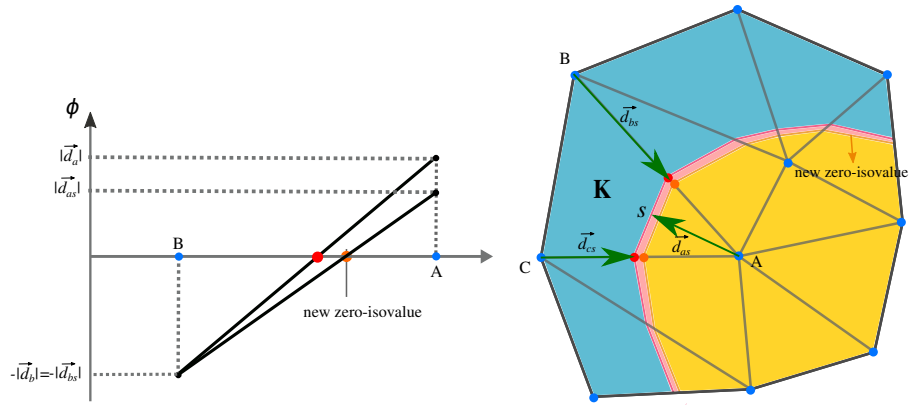
Figure 5: Reinitialization errors: a) initial configuration, interpolation of the initial LS field to its zero-isovalue, b) new interpolation after reinitializing compared to the initial one. Errors in the computation of the distance to the linear interpolation of the zero-isovalue of the LS produce a shrinkage of the concave phase.
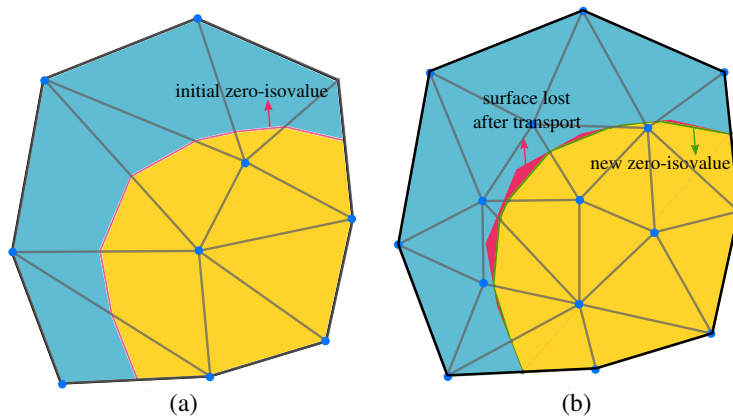
Figure 6: Transport Errors: a) Initial configuration, interpolation of the initial LS field to its zero-isovalue over the old mesh. b) New interpolation after remeshing and trasporting the LS field to the new mesh, superposition of the phase before and after the remeshing/transport process, a loss of surface is identified.

phase to shrink, producing a loss of surface. Once again, this phenomena can be avoided if the nodes of the new mesh coincide with the zero isovalue of the old mesh, which is exactly the way of producing new meshes by using the new fitting and joining algorithm.

3.2.3. Errors of the New Fitting and Joining Algorithm

One of the issues when using the new fitting and joining algorithm is that some of the elements at the interface end up with a very poor quality (in terms of shape and size) when the zero isovalue crosses the element too near to a node (see figure 8(a)). One way to avoid this problem is by pushing the interface to the node before applying the fitting algorithm to the patch of elements (see figure 8(b)). This procedure is triggered if the volume of one of the elements after fitting is smaller than an user defined value δ_v (the element colored in red in figure 8(a) left).

Once the fitting algorithm has finished, a mesh adaptation procedure begins in order to improve the quality of the mesh near to the fitted interface. Small changes on the volume of each phase are allowed under a percent value
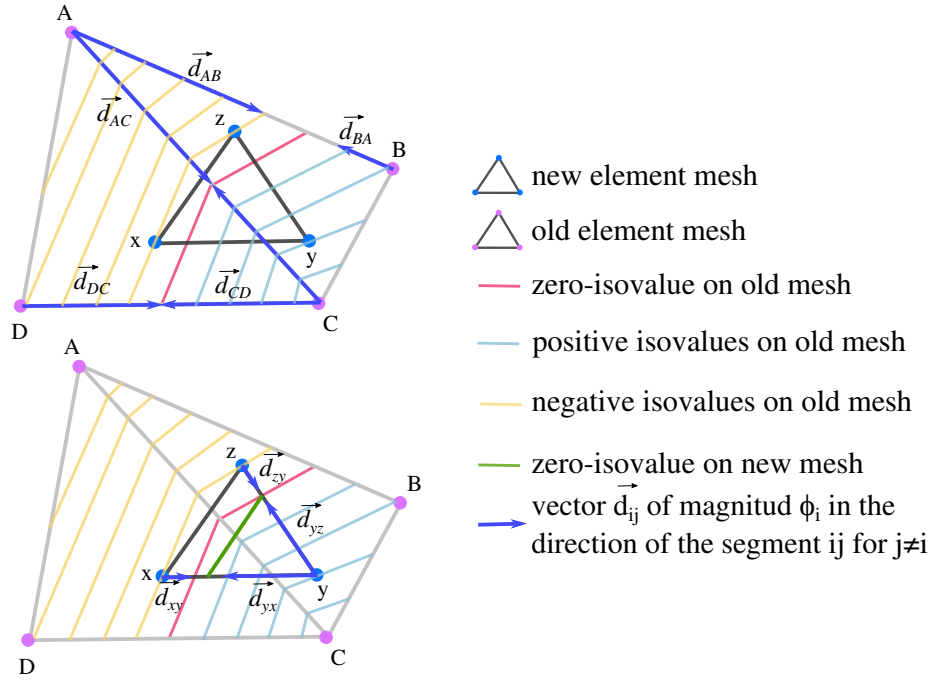
Figure 7: Transport example from two elements (ABC and ACD) to one element (xyz), a surface-loss after the transport is found on element xyz as the new linear interpolation crosses the edge $\overline{AC}$.

δ_p also user-defined. Take for instance the example showed in figure 9. An initial fitting process is done over the initial configuration (fig. 9(top-left)) giving as a result the body-fitted mesh (fig. 9(top-right)) containing potentially ill conditioned elements (see elements attatched to nodes A, B and C), then, a first adaptation process will over node A, which is going to move the same node to a position where the quality of the whole element patch of node A is improved (fig. 9(bottom-right)) till then, no change on either of the phases is registred. Then, a second attempt of adaptation is done: the patch of elements sourrounding node B in figure 9(bottom-right) contains two phases. If by applying a remeshing operation to this element patch, the quality is improved and the maximun volume change of the two phases (in this case, the yellow phase) does not go over the allowed user-defined percent (percent of the initial volume
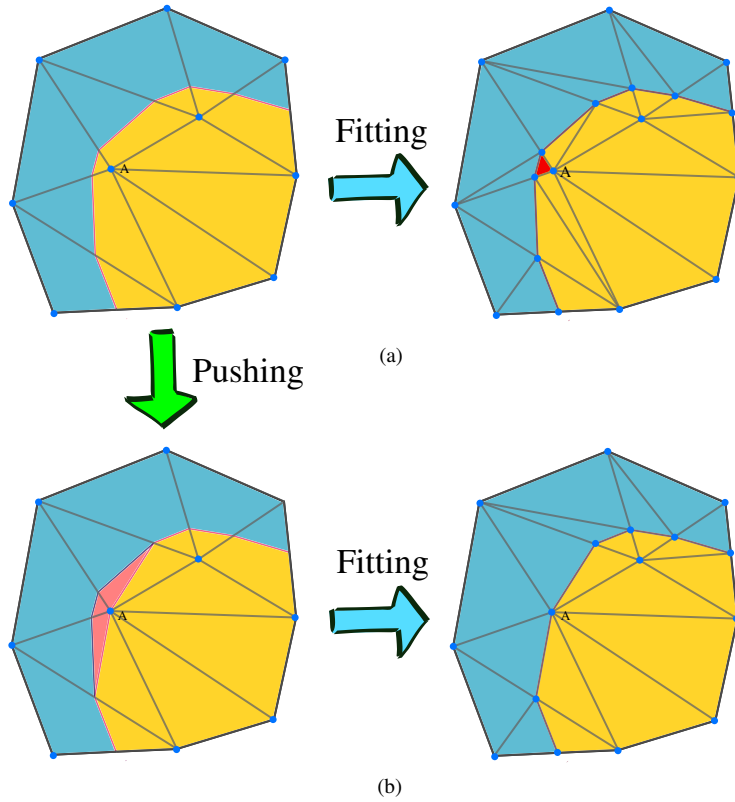
Figure 8: Fitting Relaxation: a) left: Initial configuration, right: fitting without relaxation, elements with a very small volume $< \delta_v$ are colored red. b) left: relaxation of the interfaces by pushing the boundaries to node A, the surface-loss of the yellow phase is colored pink, right: fitting after relaxation.

of the phase on the element patch) the operation is enregistred, else, the operation is discarded and no change on the patch is made. An additional constraint is added to the nodes of the boundary of the FE domain: when performing the adaptation step, the nodes of the boundary are not allowed to move, even if the movement increases the quality of the local patch. This condition enables to maintain the calculation domain boundaries during interfaces migration. Finally, point deletions are allowed in the same way as for all the other nodes (if the changes do not reduce or increase the volume of the local patch over the allowed user-defined percent).
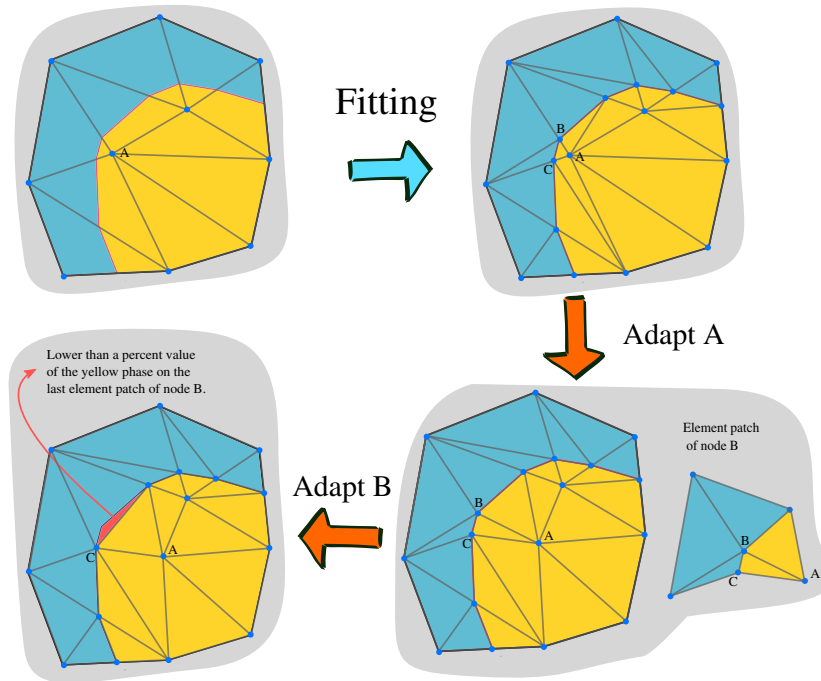
Figure 9: Adaptation Relaxation: (top-left): Initial Configuration, (top-right): Fitted Mesh, (bottom-right): adapted node A (vertex displacement), (bottom-left): adapted node B (edge $\overline{BC}$ collapsed on C)

These processes operations cause errors on the surface of the phases, producing some of them to expand or to shrink. We will measure the effect of the relaxation with the help of the following test cases. The other sources of errors explained before will be studied and compared too.

4. Numerical results

4.1. Considered geometries

Equation 4 will be considered to obtain the evolution of the LS fields for several test cases, each one responding to different interesting topological situations: sphere shrinkage, T-Junction, square shrinkage and finally a 2D Laguerre-Voronoi tesselation composed of 10000 grains. For each one of this geometrical

configurations, the three remeshing approaches (SM, IMA, NFJA) will be compared.

4.2. Circle shrinkage

The circle shrinkage test case, being the most basic of all, enables to observe the response of the interface to the instantaneous curvature and so to observe the topological disappearance of the grain. This case allows to compare the shape evolution of the phase defined by the positive values of ϕ to the analytic solution; The evolution of the analytical circle radius r is simple given by the following differential equation:

$$\frac{dr}{dt} + M\gamma \frac{1}{r} = 0 \implies r(t) = \sqrt{r_0^2 - 2M\gamma t}. \tag{6}$$

This equation can be rewrited in terms of the surface of the circle S, with a linear solution, as:

$$\frac{dS}{dt} + 2\pi M\gamma = 0 \implies S(t) = S_0 - 2\pi M\gamma t. \tag{7}$$

As mentioned before, solving Eq. 4 in a FEM context implies some errors. Here we will quantify each source of errors using the surface S_ϕ of the phase ϕ describing the "circle". In reality, the phase ϕ can not define a perfect circle but the errors given by its real shape will be neglected.

We can establish the signed difference between the numeric surface computation S_ϕ and the analytic value S as the total error E^{**}:

$$E^{**}_{(\phi,h,t,\Delta t)} = S_{\phi(\phi,h,t,\Delta t)} - S_{(t)}, \tag{8}$$

where h is the mesh size at the interfaces and Δt is the time step. Eq 8 can be rewritten in order to obtain the error per increment E^*:

$$E^{**}_{(\phi,h,t,\Delta t)} - E^{**}_{(\phi,h,t-\Delta t,\Delta t)} = S_{\phi(\phi,h,t,\Delta t)} - S_{\phi(\phi,h,t-\Delta t,\Delta t)} + S_{(t-\Delta t)} - S_{(t)}, \quad (9)$$

$$E^{*}_{(\phi,h,t,\Delta t)} = \Delta S_{\phi(\phi,h,t,\Delta t)} - \Delta S_{(t,\Delta t)}. \quad (10)$$

Finally we can express the relative error as:

$$E_{(\phi,h,t,\Delta t)} = \frac{\Delta S_{\phi(\phi,h,t,\Delta t)} - \Delta S_{(t,\Delta t)}}{\Delta S_{(t,\Delta t)}}. \quad (11)$$

The term $\Delta S_{\phi(\phi,h,t,\Delta t)}$ is actually the addition of the different contributions to the change of surface given by the different treatments done in an computational increment: solution of the EDP in the FE context, reinitialization and remeshing. The latter itself is also differentiated in the change of surface given by the processes of transport, fitting and adaption as explained in section 3.2. This can be summarized as follows:

$$\Delta S_{\phi(\phi,h,t,\Delta t)} = \Delta S_{solver(\phi,h,t,\Delta t)} + \Delta S_{reinit(\phi,h,t,\Delta t)} + \Delta S_{remesh(\phi,h,t,\Delta t)}, \quad (12)$$

$$\therefore E_{(\phi,h,t,\Delta t)} = \frac{\Delta S_{solver(\phi,h,t,\Delta t)} - \Delta S_{(t,\Delta t)}}{\Delta S_{(t,\Delta t)}} + \frac{\Delta S_{reinit(\phi,h,t,\Delta t)}}{\Delta S_{(t,\Delta t)}} + \frac{\Delta S_{remesh(\phi,h,t,\Delta t)}}{\Delta S_{(t,\Delta t)}}. \quad (13)$$

Finally, Expressions for the three principal sources of error can be stablished:

$$\xi_{solver(\phi,h,t,\Delta t)} = 100 \cdot \frac{\Delta S_{solver(\phi,h,t,\Delta t)} - \Delta S_{(t,\Delta t)}}{\Delta S_{(t,\Delta t)}}, \quad (14)$$

$$\xi_{reinit(\phi,h,t,\Delta t)} = 100 \cdot \frac{\Delta S_{reinit(\phi,h,t,\Delta t)}}{\Delta S_{(t,\Delta t)}}, \quad (15)$$

$$\xi_{remesh(\phi,h,t,\Delta t)} = 100 \cdot \frac{\Delta S_{remesh(\phi,h,t,\Delta t)}}{\Delta S_{(t,\Delta t)}}, \quad (16)$$

where the terms ξ_i represent the relative error given by the procedure i (solver, reinit ans remesh). The dependence of ϕ for the different values of ξ_i
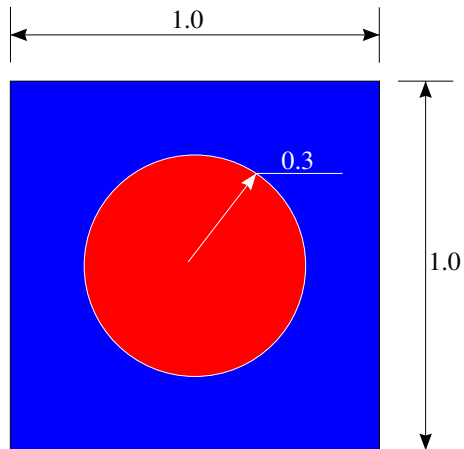
Figure 10: Circle shrinkage test case: initial state.

describes the error produced by how far the phase ϕ is to represent a perfect circle of surface S_ϕ, in this study, this dependence will be neglected and we will focus on the evolution of the error as a parameter of the mesh size h, the time t and the time step Δt. Note that we have not used an absolute value to describe each error. Indeed, we want to observe if the numeric models are quicker or slower than the analytic solution: a negative (resp. positive) value of the error would mean that phase ϕ shrinks too fast (resp. low) during the considered step.

In the following, dimensionless simulations will be considered and the value of the reduced mobility $M \cdot \gamma$ will be assumed to be unitary.
The initial radius is set as $r_0 = 0.3$ (initial surface $S_0 \approx 0.2827$) and the circle is immersed in the center of a 1×1 square as illustrated in Figure 10.

As stated before, three meshing configurations will be compared, the first one will use the interface fitting and joining algorithm to remesh every time step, the second one will use an interface capture meshing (a mesh refined only at the interfaces but not fitted) and the last one will use a static mesh with an uniform mesh size. Examples of each meshing approach are given in figure 11. Note that for the static mesh, all the domain must be refined in order to

maintain the same level of accuracy as for the other cases during the interface migration.

Multiple runs with different mesh sizes h and time steps dt were made for each one of the configurations. Eq. 4 was solved with a standard diffusion FE solver with a precision of 10^{-10} for a P1 (linear) interpolation.

One example of the obtained results for the evolution of the surface for the configuration with a static mesh are summarized in figure 12, Ks and Kr are the median values of the surface change when solving the FEM problem and when reinitializing respectively. These values can be replaced in equations 14 and 15 as ΔS_{Solver} and ΔS_{Reinit} to obtain the values of ξ_{solver} and ξ_{reinit}. Figure 13 shows the values of ξ_{solver} and ξ_{reinit} for different time steps and mesh sizes. Note that for this case $\xi_{Transport}$, ξ_{Fit} and ξ_{adapt} are equal to zero because there is no remeshing.

It is interesting to see that errors given by the reinitialization procedure are much more important than those obtained by using a FEM to solve Eq. 4. and that they tend to be bigger when the time step decreases. In fact, from our observations ΔS_{reinit} is not dependant of the time step dt used (it depends only of the mesh size h), however, a smaller dt means more increments to simulate the same time, and as the error from the reinit cumulates, the more increments the bigger the value of ξ_{Reinit}.

A similar computation can be made for the IMA and the NFJA approaches. When using the IMA approach, the value $\xi_{Transport} \neq 0$ because the remeshing makes the procedure of transport unavoidable. however, as there are no fitting, $\xi_{Fit} = 0$ and $\xi_{Adapt} = 0$. The values of ξ_{Solver} and ξ_{Reinit} are almost equal to those of the SM method displayed in figures 13 as the only difference near to the interface between these two approaches is that a procedure of remeshing is done between time steps. Figure 14 shows the results for the values ξ_{Solver}, ξ_{Reinit} and $\xi_{Transport}$ for different values of the mesh size h and the time step
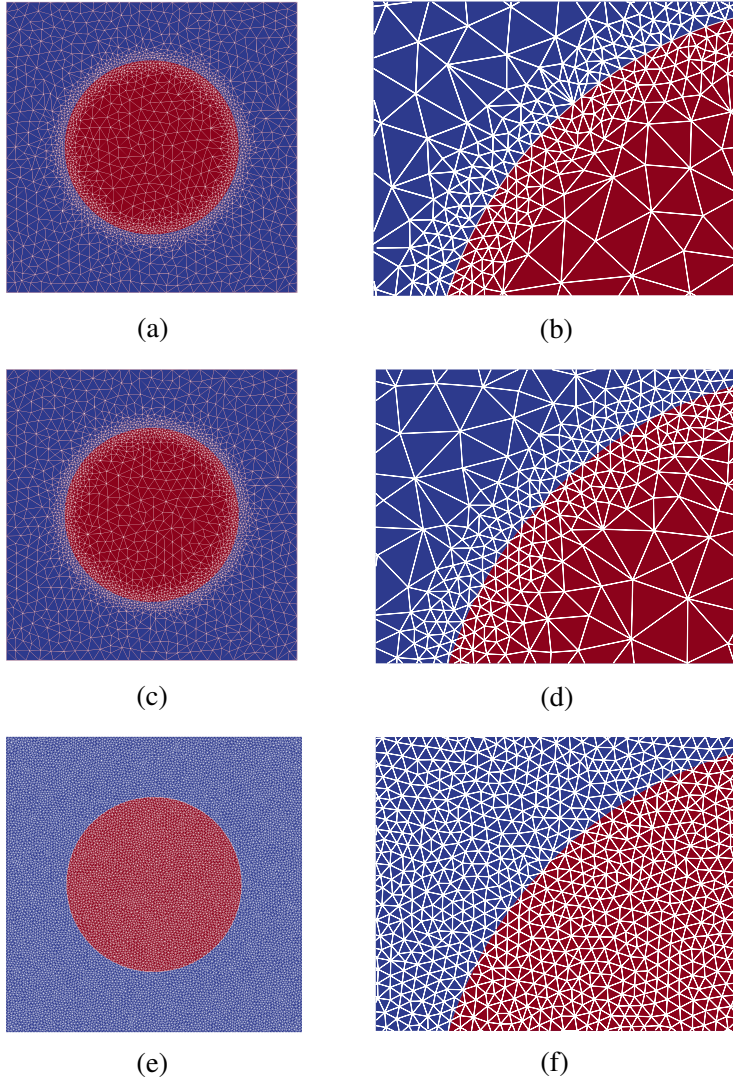
Figure 11: Meshes obtained for each one of the configurations for the circle shrinkage test at $t = 0$ and with a mesh size $h = 0.01$ at the interface. (a,b) Using the interface fitting and joining algorithm. (c,d) Using a classic interface capturing algorithm. (e,f) Static mesh.
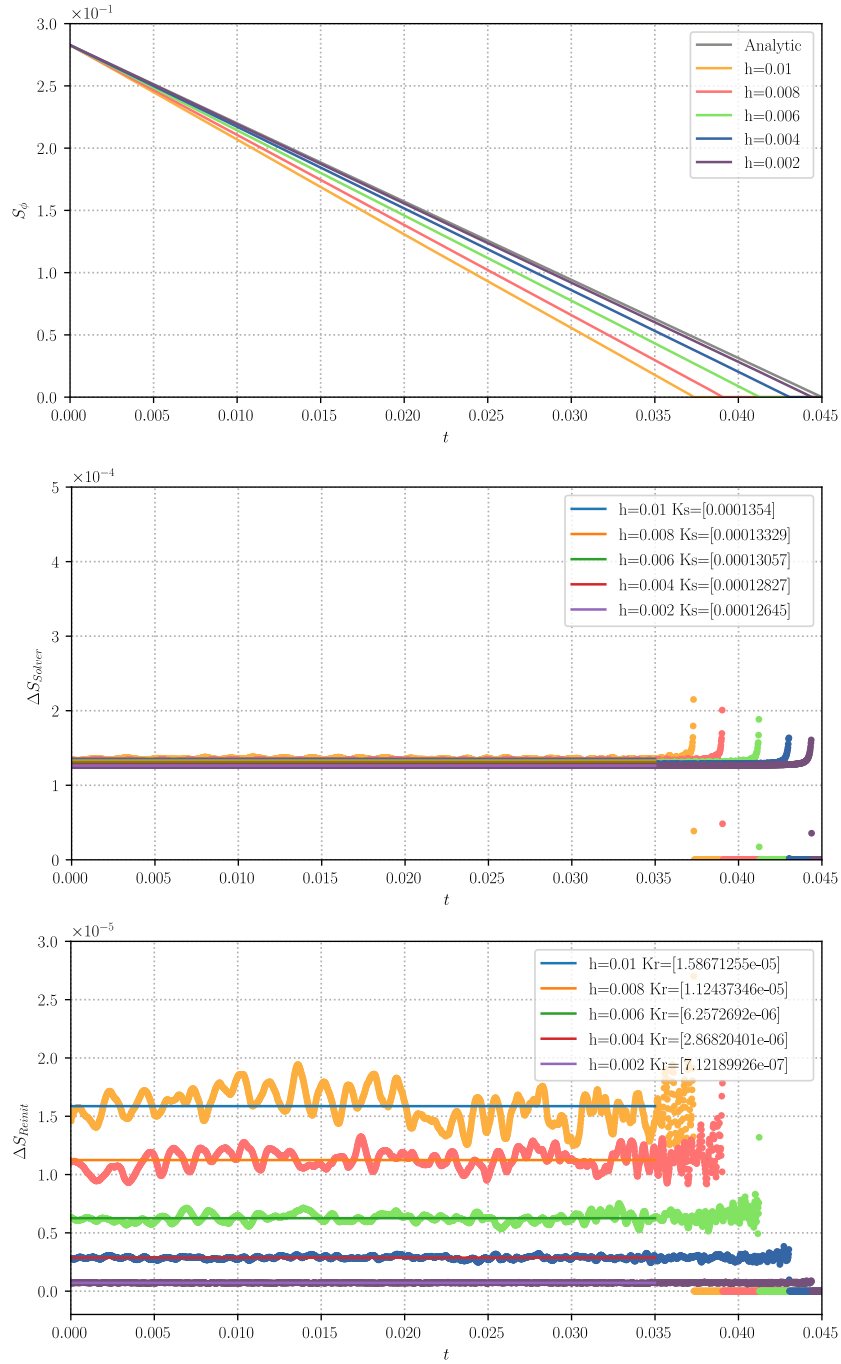
Figure 12: Results for the static mesh configuration using a $dt = 2 \cdot 10^{-5}$ and for different mesh sizes h. Top: Evolution of S_ϕ. Middle: evolution of ΔS_{Solver}, Ks is the median value for each h. Bottom: evolution of ΔS_{Reinit}, Kr is the median value for each h.

26
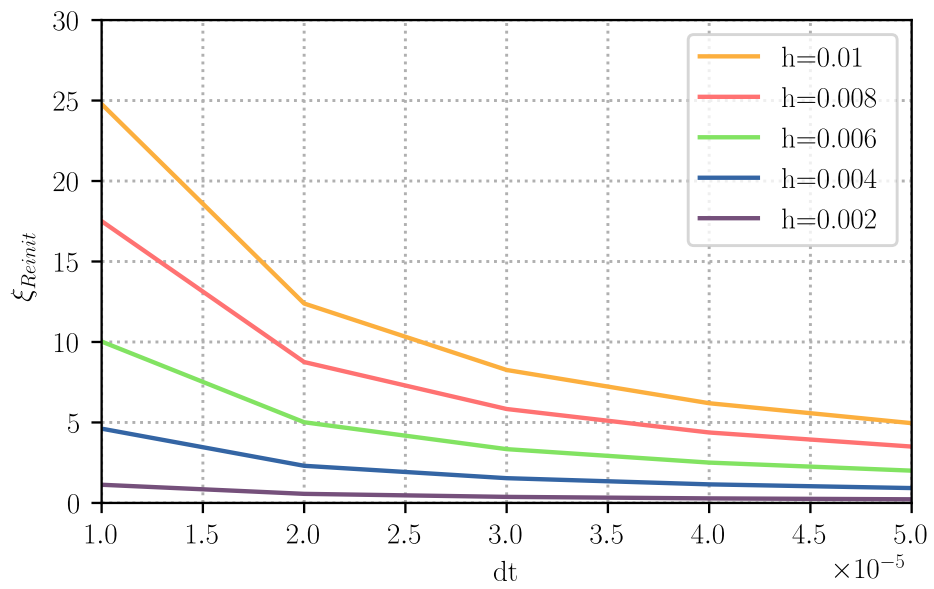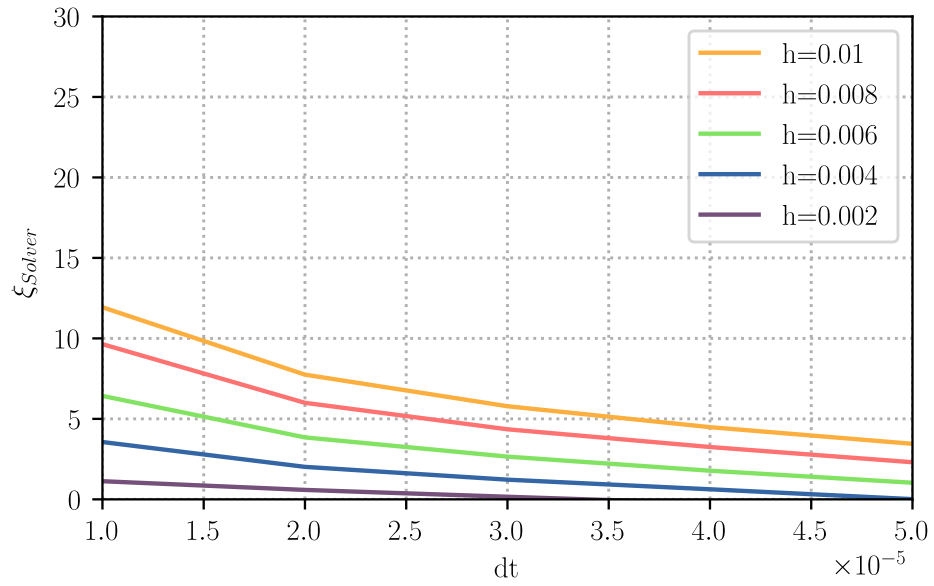
Figure 13: Results for the static mesh configuration. values of ξ_{solver} and ξ_{reinit} for different values of the time step and the mesh size.

dt in context of the IMA strategy. The values of $\xi_{Transport}$ were found to be very small compared to those of ξ_{Reinit} and ξ_{Solver} hence they can be neglected in the following.

Finally, when using the NFJA approach, as explained in section 3.2, $\xi_{Transport} = 0$ and $\xi_{Reinit} = 0$. however, $\xi_{Fit} \neq 0$ and $\xi_{Adapt} \neq 0$ as a fitting procedure is used. The NFJA allows to control the two variables δ_v and δ_p introduced in section 3.2.3, these variables control directly the values of ξ_{Fit} and ξ_{Adapt}. A very low value of δ_v will cause that the algorithm allows to get very small elements after fitting while the value ξ_{Fit} tends to zero. In the same way, a very low value of δ_p will cause that even though a very poor quality of elements were found after fitting, the algorithm could not modify the interface in order to improve that quality, while the value ξ_{Adapt} tends also to zero.

Figure 15 shows the values of ξ_{Solver} and ξ_{Adapt} for one example with $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-2}$. For these values, ξ_{Fit} are very low and can be neglected. An example of a mesh obtained with these values is showed too. In the same way, Figure 16 shows another example for the values of $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-3}$. It is interesting to see that for the latter, ξ_{Adapt} can be neglected and the remaining errors are given by the value of ξ_{Solver} and additionally, ξ_{Solver} is lower than the one from figure 15. Of course, the curvature of the interface is better preserved when δ_p is low but the mesh obtained with such values is nearly degenerated. Results shows that the FEM solver is more sensitive to the small changes in the description of the surface than to the quality of the elements, and that it will be actually more accurate to maintain the interface as it is after fitting than to try to improve the mesh quality field with remeshing operations.

Figure 17 shows the value of $\xi_{total} = \sum \xi_i$ for the IMA method and for the NFJA with the two sets of constants used in figures 15 and 16. The smallest ξ_{total} was found for the NFJA approach with the values $\delta_v = 10^{-10}$ and $\delta_p =$
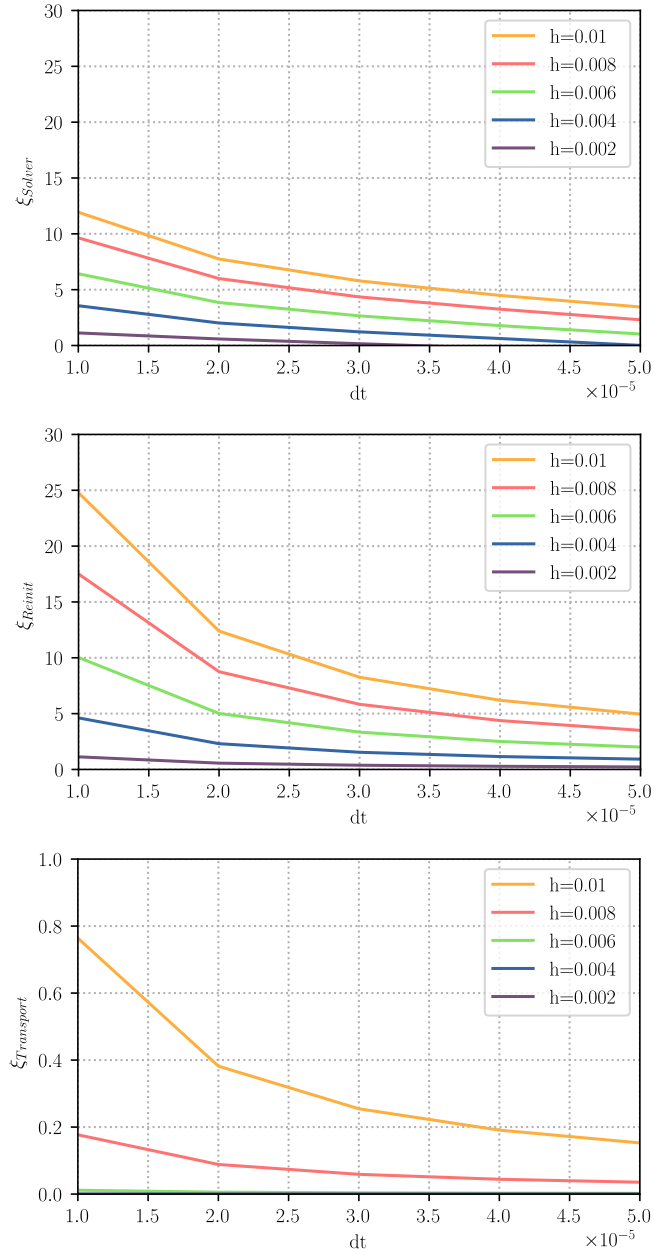
28

Figure 14: ξ_{Solver}, ξ_{Reinit} and $\xi_{Transport}$ for different mesh sizes h and time steps dt for the IMA case
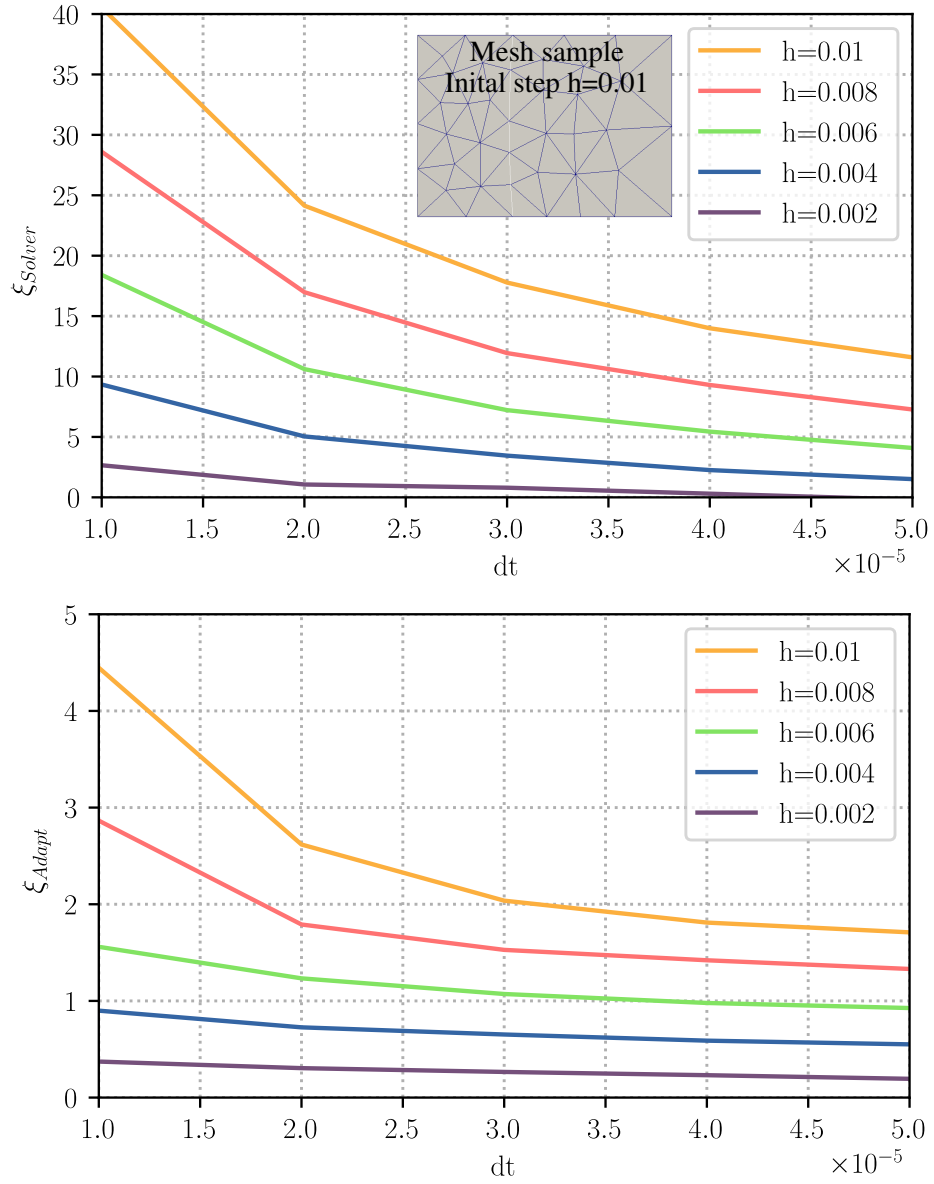
Figure 15: ξ_{Solver} and ξ_{Adapt} for different mesh sizes h and time steps dt for the NFJA case with $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-2}$
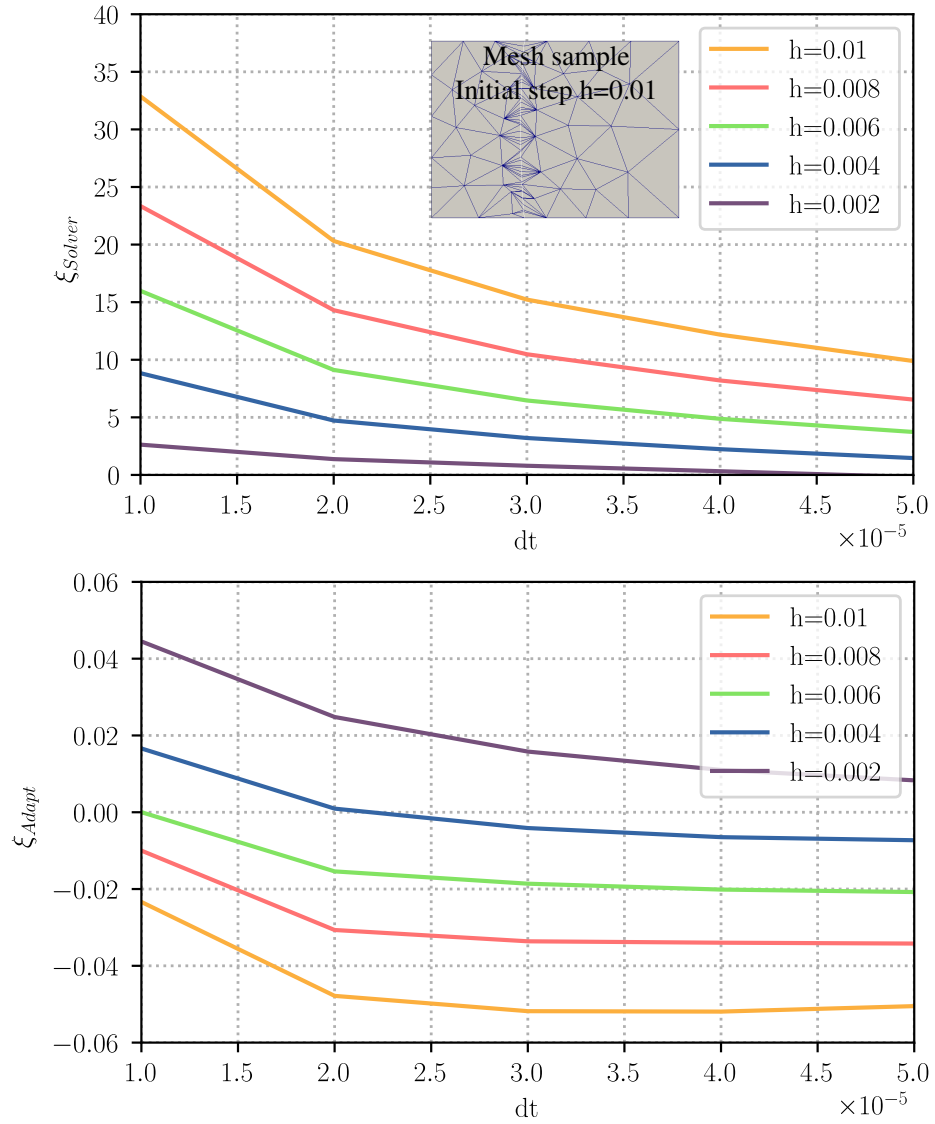
Figure 16: ξ_{Solver} and ξ_{Adapt} for different mesh sizes h and time steps dt for the NFJA case with $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-3}$

$2 \cdot 10^{-3}$. Even though the ξ_{solver} was very high for the NFJA in comparison to the one given by the IMA, results show that with the right choice for δ_v and δ_p for the NFJA approach, one can be more accurate on the prediction of the evolution of interfaces when using Eq.4 and a body fitted mesh in a FEM context.

4.3. T-Junction case

The T-junction problem is an initially unstable configuration of three interfaces (for three grains at a 90°-90°-180° initial configuration) that converge to a 120°-120°-120° quasi steady-state equilibrium as the $M\gamma$ term is assumed here isotropic. The equilibrium of the triple point is given by the Herring's equation [43]. It will bring the system to a state where the surface energy is minimized and the three lines will arrange themselves in a stable 120°-120°-120° (Young's Equilibrium) configuration evolving after with an homogeneous velocity given by the migration of the curved interfaces and the Neumann boundary conditions [44] (see figure 18).

Comparing the results from the T-junction test and the square-shrinkage is much more difficult because there is no analytic solution for these problems concerning the way to reach the quasi steady-state. However, as illustrated in the precedent study over the circle-shrinkage test, convergence is obtained when the mesh size decreased. This result will be used to obtain a reference case for the T-junction problem. Several computations where turned with different static meshes. Figure 19 shows the interfaces $\phi_1 - \phi_2$ and $\phi_1 - \phi_3$ after t=0.35 for the T-Junction configuration. At this time, the quasi steady-state is ensured. As convergence is obtained when the mesh size decreases, the simulation with 500000 elements (last one from figure 19) will be used as reference.

Figure 20 shows the ϕ_1 phase for the reference model and the geometric difference of the same phase obtained with the other models (the IMA and NFJA) at time t= 0.35. Compared to the reference model, the error on the area of the
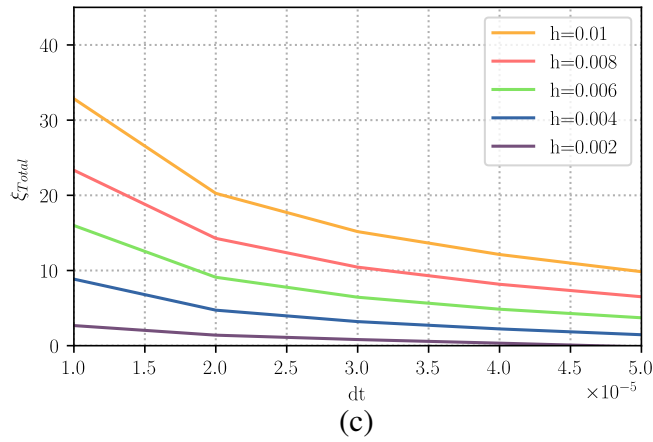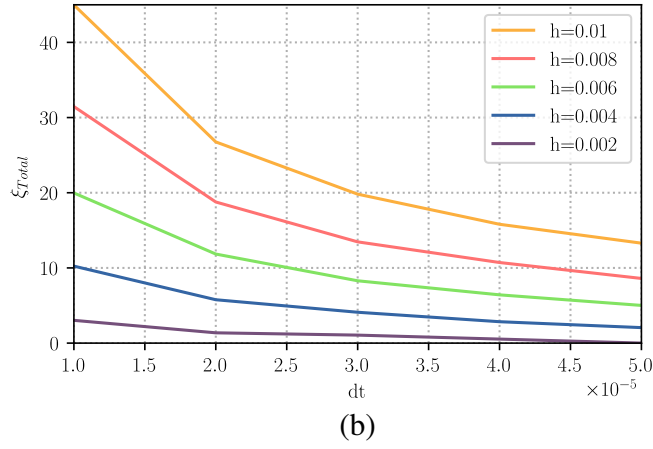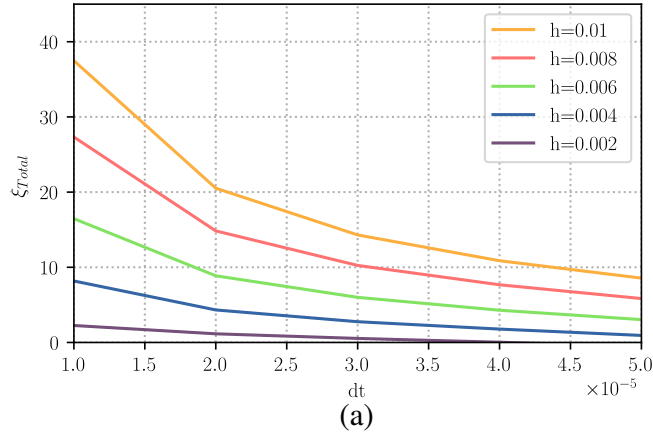
(a)

(b)

(c)

Figure 17: $\xi_{total} = \sum \xi_i$ for different mesh sizes h and time steps dt (a) ξ_{total} for the IMA method. (b) ξ_{total} for the NFJA with $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-2}$. (c) ξ_{total} for the NFJA with $\delta_v = 10^{-10}$ and $\delta_p = 2 \cdot 10^{-3}$.
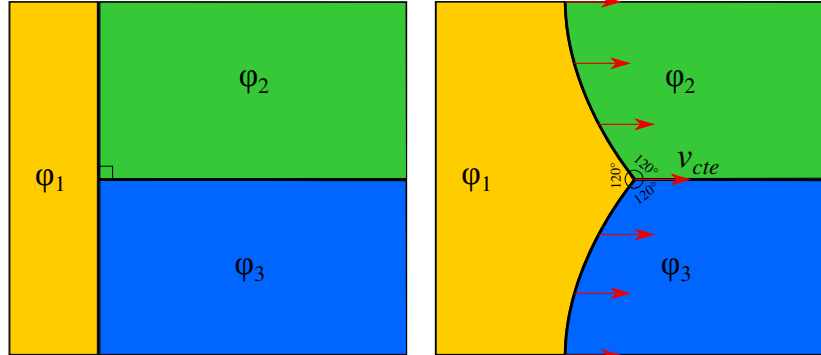
33

Figure 18: T-Junction Case. left: initial state, right: steady state.

ϕ_1 phase is $\xi_{IMA} = 10.1\%$ and $\xi_{NFJA} = 15.4\%$. Hence, the IMA approach is clearly more accurate. An acceleration of the multiple point was found for the NFJA model.

Finally, during the simulation campaign, some differences on the capabilities of the algorithms were observed. Some limits were observed when using the IMA method: the minimum number of refined elements at each side of the interface is around 4. This number of elements ensures that the numeric diffusion obtained at the remeshing step over the metric field is not too important, hence the remeshing success. Otherwise, as illustrated in figure 21 the remeshing may fail after some increments. On the other hand, the NFJA method is able to remesh successfully every time step, as some of the new nodes are fitted to the interface hence the metric field can not be numerically diffused there.

4.4. Square-Shrinkage case

The square shrinkage test makes it possible to observe the behavior of each model when four triple points converge to the same position. The instability of this configuration suggests that the 4 triple points should became 2 triple points and not 1 quadruple point. Figure 22 shows the normal behaviour of this test case: initially there are 5 phases where the central phase ϕ_1 represents a
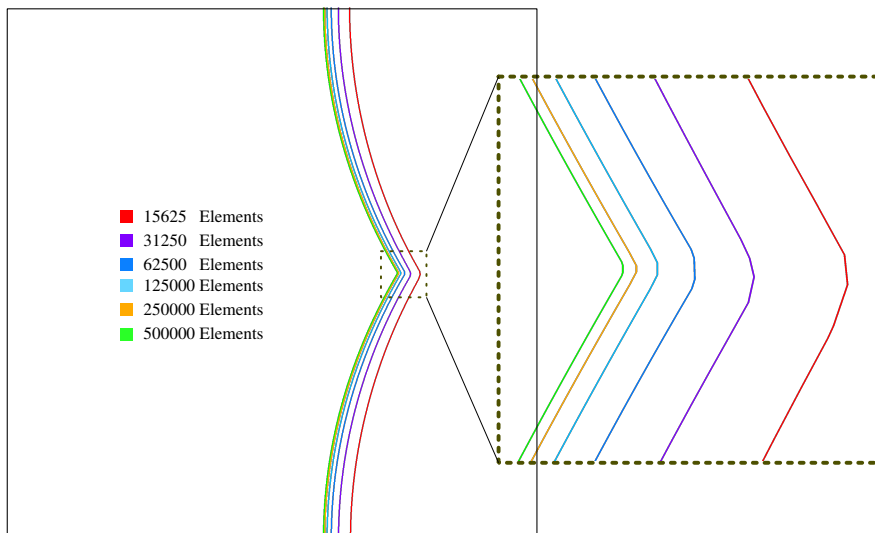
Figure 19: T-Junction Case: static Mesh, convergence analysis. Only the interfaces between $\phi_1 - \phi_2$ and $\phi_1 - \phi_3$ are shown (see figure 18 for the notation).

perfect square; then, each one of the triple points reaches a quasi steady-state similar to the one obtained for the T-Junction problem. At this stage, each one of the interfaces of the phase represented by ϕ_1 is shrinking at a constant rate. Finally, ϕ_1 disappears and a new interface is created resulting of the absolute numerical instability of the quadruple point. Here two possible configurations are possible for the creation of the new interface ($\phi_2 - \phi_4$ or $\phi_3 - \phi_5$). The choice of the created interface should be given by the difference on the surface energy of each one of the possibilities, the one with the lowest energy is the one that should be created, however, in our study isotropic grain boundary properties are considered, and either of the two decompositions are valid.

In the context of a FE level-set method, it is almost impossible to obtain a perfectly symmetric quadruple junction that will maintain its stability: a symmetric (four right angles) quadruple junction is a meta-stable state that will decompose in a lower energy state, i.e. two triple junctions, with an infinitesimal perturbation δ. As mentioned in section 3, the LS method depends on the FE mesh discretization, convergence stop criterion of the FE solver, the in-
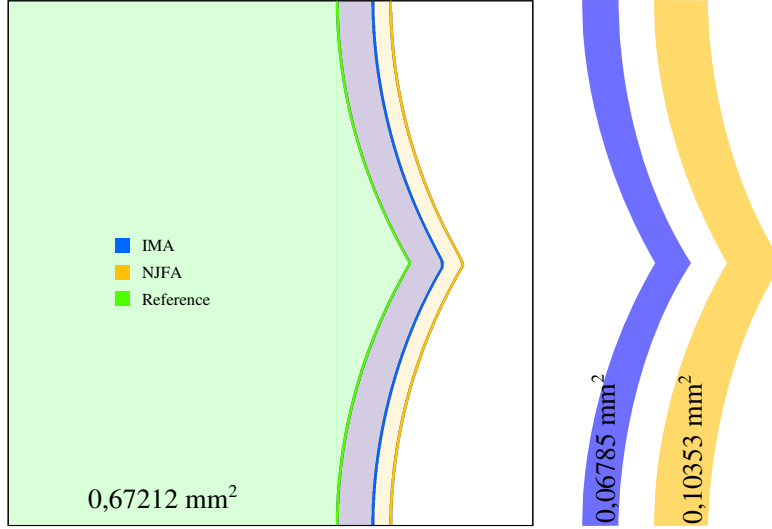
Figure 20: ϕ_1 phase for the reference model and the geometric difference of the same phase obtained with the other models (the IMA and NFJA) at time t= 0.35. values for the area of each section is given.

terpolation degree and the approximations made such as the application of eq. (1) or the reinitializing method used. Finally, by applying this method to the considered geometry (square shrinkage), the angles between the 4 interfaces at the moment when the quadruple point appears will be of $90 + \Theta$ where $\Theta >> \delta$, triggering its decomposition. In fact, It is actually highly probable that the quadruple point never really appears (two very close triple points appearing instead).

Similarly to the T-Junction test case, a convergence analysis was made using static meshes in order to obtain the reference evolution for this configuration. Convergence was obtained after using a static mesh with 1 million elements. our reference will employ a static mesh with 2 million elements.

Figure 23 shows the comparison of the NFJA and IMA methods to the reference after t=0.05. The error on the area of the ϕ_1 phase was $Err_{NFJA} = 38.4\%$ and $Err_{IMA} = 16.3\%$. Once again an acceleration on the evolution of the NFJA
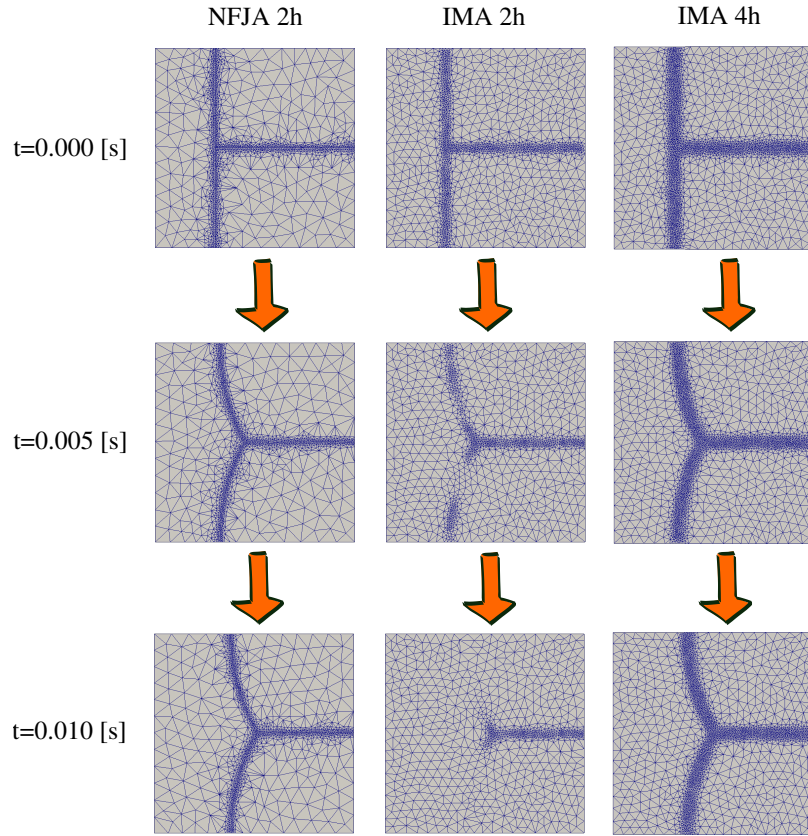
Figure 21: T-Junction Case. Numeric diffusion of the metric field used by the mesh adaptation algorithm for the IMA case. The needed thickness of refined elements of each side of the interface is around 4 (4 elements) for the IMA model and around 2 for the NFJA.

was observed. Another difference from the reference model was that both NFJA and IMA methods created interface $\phi_3 - \phi_5$ while the reference model created $\phi_2 - \phi_4$ (See Figure 24).

4.5. 2D 10000 grains case

Here, the New Fitting and Joining Algorithm is also compared with a more classic method of mesh adaptation where the interfaces are captured with a non-conform local refined mesh as detailed in section 3.1.3 and described as the
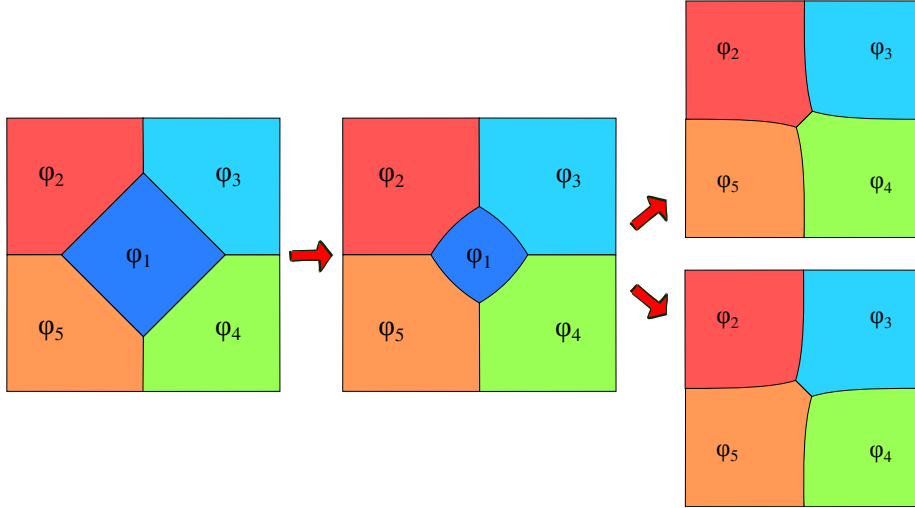
Figure 22: Square-Shrinkage. left: initial state, center: square shrinking, right: the square disappears and a new interface is made ($\phi_2 - \phi_4$ or $\phi_3 - \phi_5$).

Isotropic Mesh Adaptation (IMA) technique instead of tracked with a body-fitted mesh adaptation algorithm. Both cases will be compared to a reference case, which is the convergence of the evolution of the mean grain size (equivalent radius in number) when using an homogeneous refined static mesh. When a homogeneous static mesh is considered transport errors are not present. In addition, if the mesh size is small, errors on the reinitialization procedure become less important as the distance functions are better described and finally vacuum regions become smaller. Hence the case with the homogeneous refined static mesh will be treated as the better solution in terms of precision and errors of the two other cases will be computed thanks to its evolution.

The initial microstructure considered is composed of 10000 grains generated using the concept of Laguerre-Voronoi cells [4–6]. Fig. 25 illustrates the initial state for a square domain with surface $A = 10[mm^2]$ and a grain size lognormal distribution with a mean value of $m = 0.017[mm]$ and standard deviation $\sigma = 0.006[mm]$. The values for M and γ are chosen as representative of a 304L stainless steel at $1050°$ Celsius (with $M = M_0 * e^{-Q/RT}$ where M_0 is a constant $M_0 = 1.56e11[mm^4/Js]$, Q is the thermal activation energy
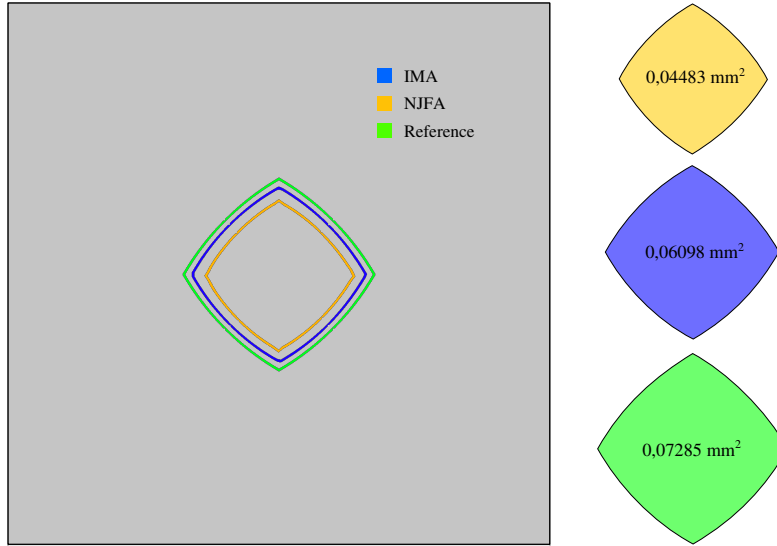
Figure 23: Square-Shrinkage. Comparison of the ϕ_1 phase at t=0.05 [s].

$Q = 2.8 \cdot 10^5 [J/mol]$, R is the ideal gas constant, T is the absolute temperature T=1323 [K] and $\gamma = 6 \cdot 10^{-7} [J/mm^2]$). The isothermal treatment is realized during 3600 seconds.

Figure 26 describes the evolution of the mean grain size (calculated in number) when different static meshes are considered. We assume that convergence is reached when the accumulated L_2 error in time remains lower than 5% at 3600s. Hence, the simulation employing a mesh size $h = 0.001[mm]$ will be considered as the reference case in the following.

Figure 27 shows the evolution of each case, the one using the IMA technique for the capturing of the interfaces and the New Fitting and Joining Algorithm (NFJA). Note that two curves are listed for the NFJA method, one (the NFJA) corresponds to a simulation which remeshing is done by employing the same metric field as in the IMA case, the other (NFJA Improved) corresponds to
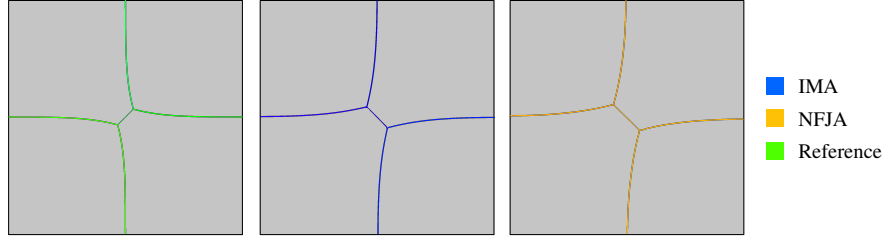
Figure 24: Square-Shrinkage. Comparison of the interfaces after the disappearance of the ϕ_1 phase. t=0.1.

another where the metric field had been improved as shown in figure 21.

Figure 28 shows the comparison of the grain size distributions weighted by surface of each model (IMA, NFJA and Static Mesh) and for the reference case with a mesh size of $h = 0.004$, for the times $t = 1800$ and $t = 3600$. Figure 29 shows the L2-Error over these grain size distributions for multiple mesh sizes $h = [0.01, 0.008, 0.006, 0.004, 0.002]$.

Finally, a summary of these results can be made in a single chart: Figure 30 shows the evolution of the $L2$ Error in function of the Total CPU-time at the end of the simulation for every case. These charts show that for a precision greater than 5% on the prediction of the mean grain size and greater than 10% on the prediction of the grain size distributions the fastest method is the one using a static mesh, meaning that the procedure of remeshing regardless which one we use seems not to given any advantage in terms of CPU-time for the considered configuration.

5. Discussion, conclusion and perspectives

Figure 27 shows that the behaviour of the CPU-time for each of the simulation is not linear, this is due to the fact that the number of grains is also changing during the simulation: the less grains there are, the smaller the zone needed to maintain with a refined mesh hence the smaller the time of computation by increment.
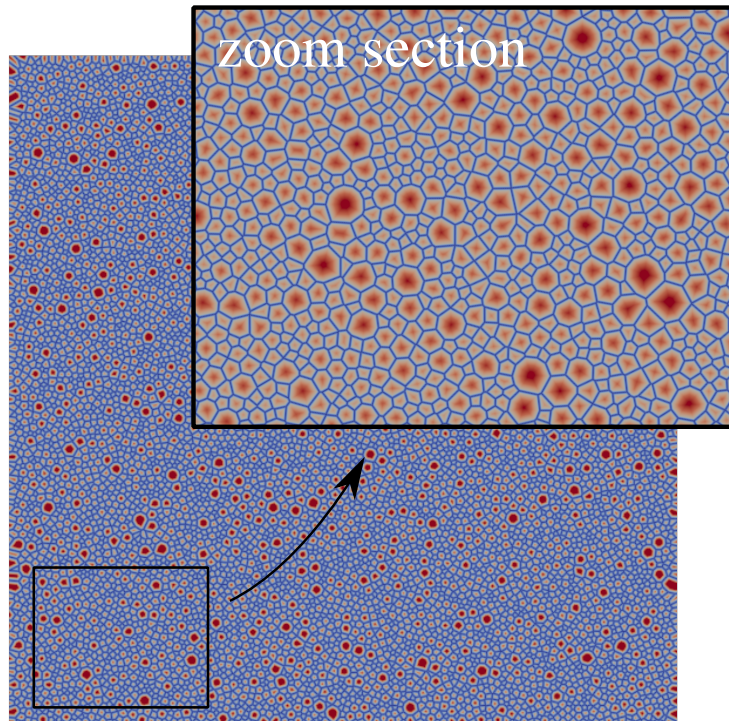
Figure 25: Initial state of the microstructure composed of 10000 grains build thanks to a Laguerre-Voronoï algorithm.
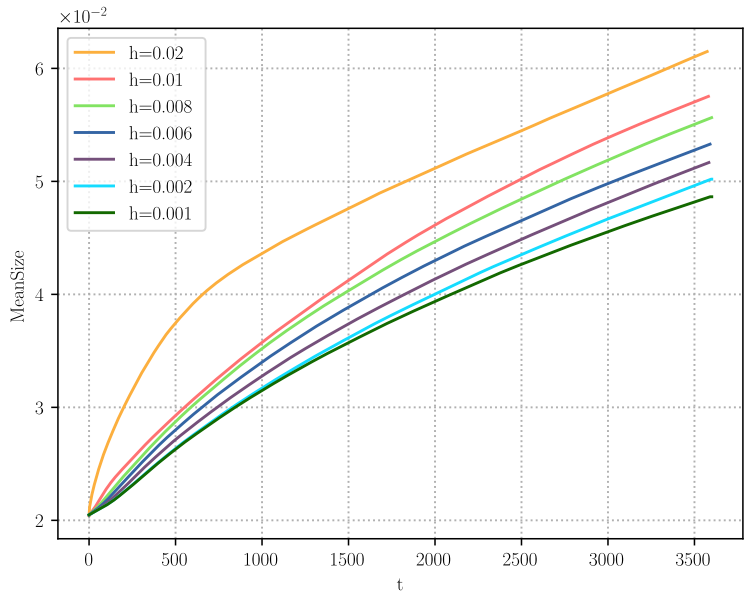
Figure 26: Convergence analysis when a homogeneous static mesh is considered: evolution of the mean grain size of the microstructure

Figure 30(a) and (b) illustrates that for IMA and the NFJA methods the accuracy range is wider that for the one using a static mesh. CPU-time for both cases (IMA and NFJA) are very near, however they represent very different processes. The time needed to remesh with the NFJA for a fixed number of elements is higher as in addition to the mesh adaptation process because the fitting process adds an extra amount of computational work. On the other hand, some of our observations showed that the NFJA needed less mesh elements to properly define the interface even if the asked metric field was the same. Thus the proposed new front-tracking approach appears already as competitive comparatively to the existing LS front-capturing approach used in the state of the art in context of unstructured FE mesh [1, 18, 25, 32]. However, when comparing the CPU-time obtained for the simulations using a static mesh, every test (except for the one with $h = 0.002$) turn out to have a lower CPU-time than for the corresponding IMA and NFJA approaches. This result in addition to the fact that the error ranges are close, take us to conclude that the actual
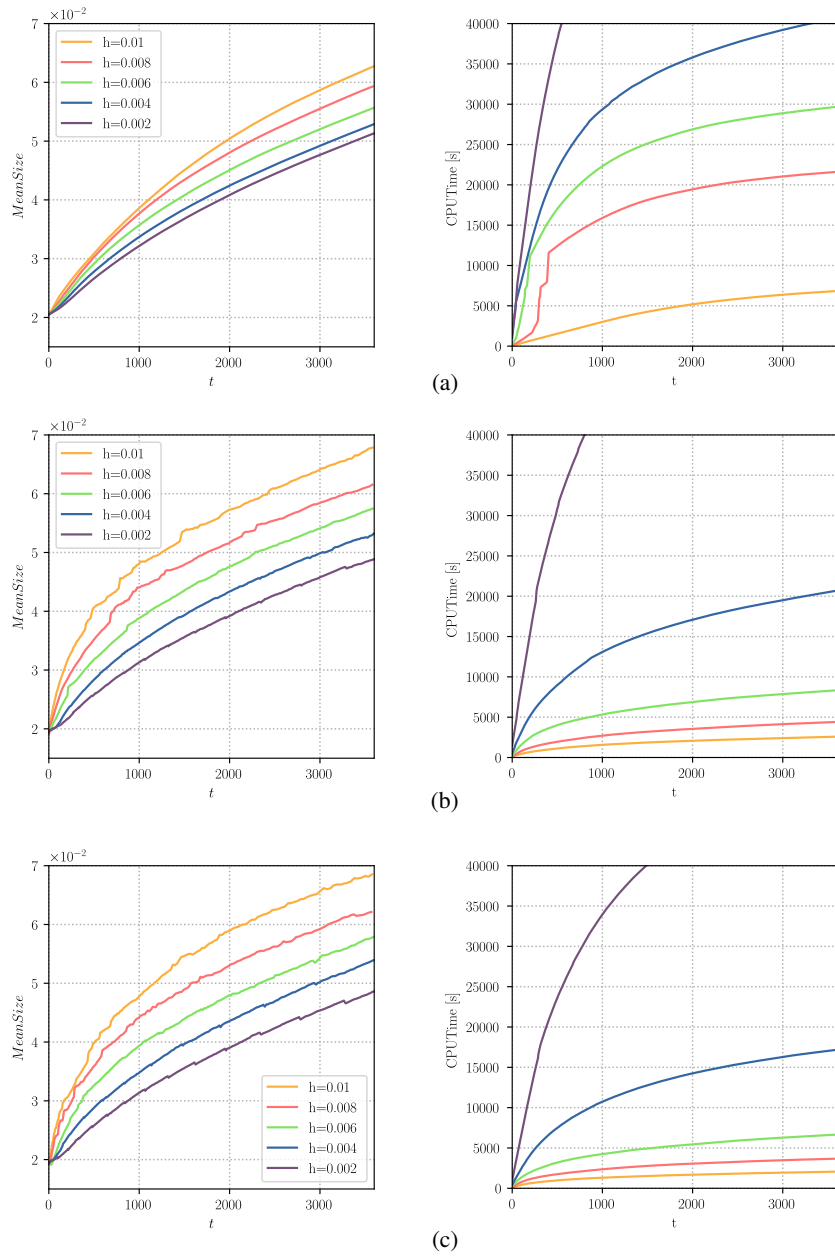
42

Figure 27: Results for each case: (a) using a classic meshing adaptation (IMA) technique, (b) using the New Fitting and Joining Algorithm (NFJA), (c) using the NFJA method with the improvements on the metric field explained in figure 21 and the reference curve. Left: evolution of the mean grain size and right: CPU-time of simulations, the reference case is not plotted.
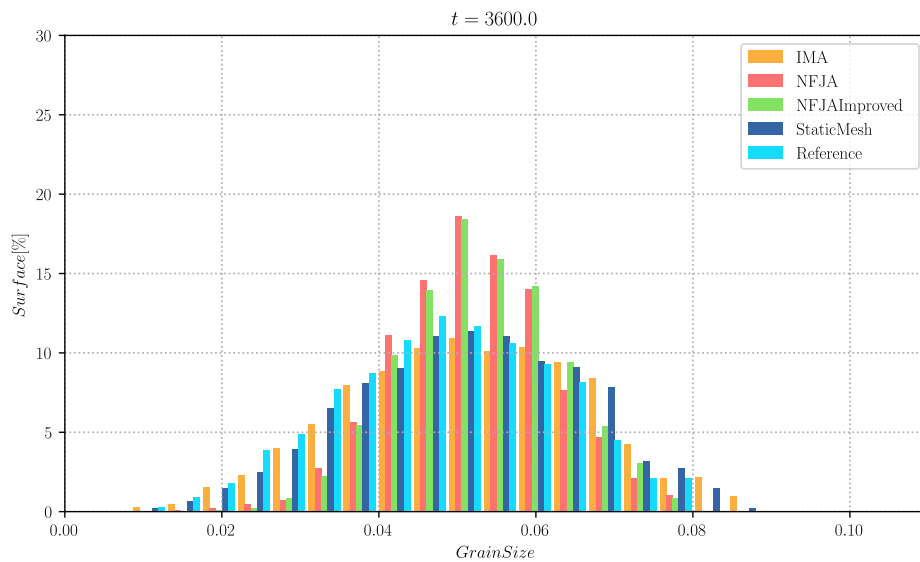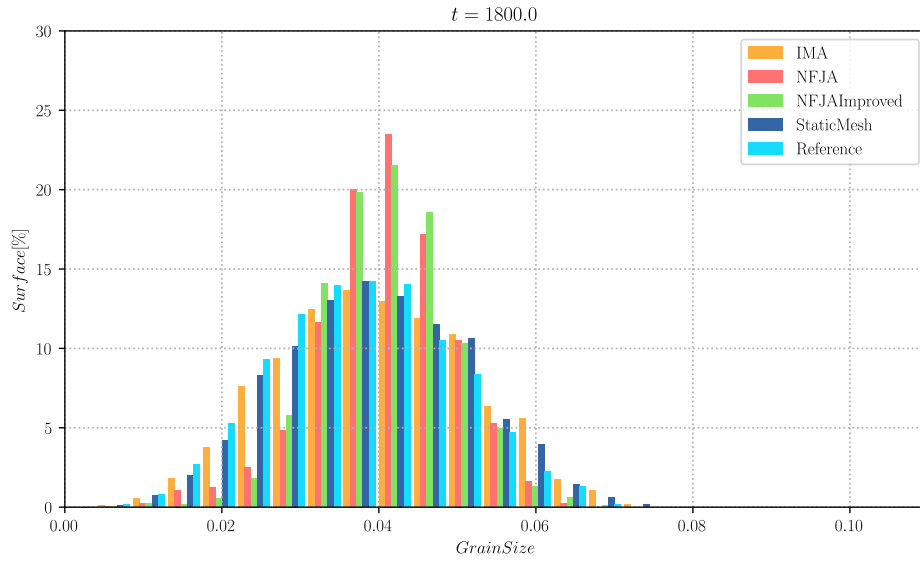
Figure 28: Grain size distributions weighted by surface for the different models and for a mesh size $h = 0.004$. (a) state at time $t = 1800$, (b) state at time $t = 3600$
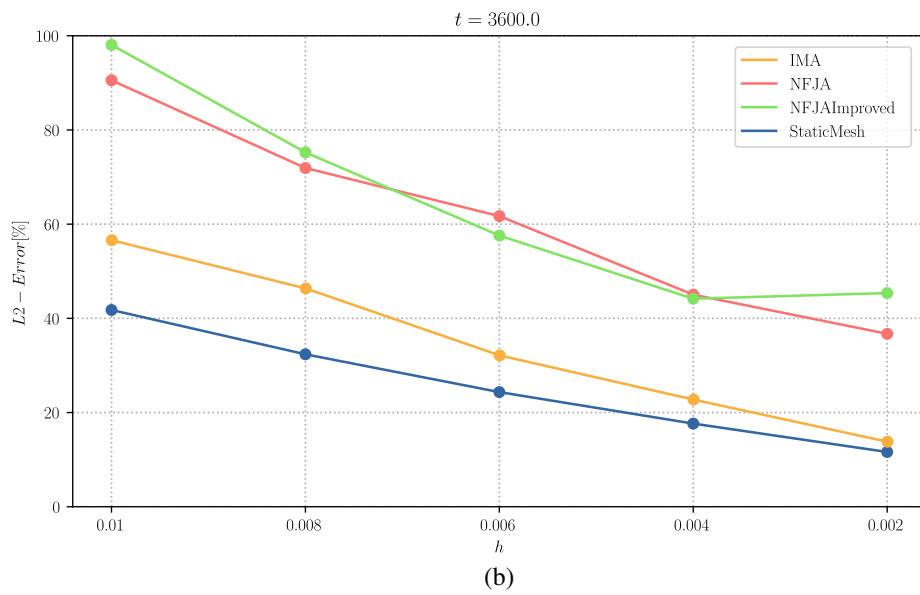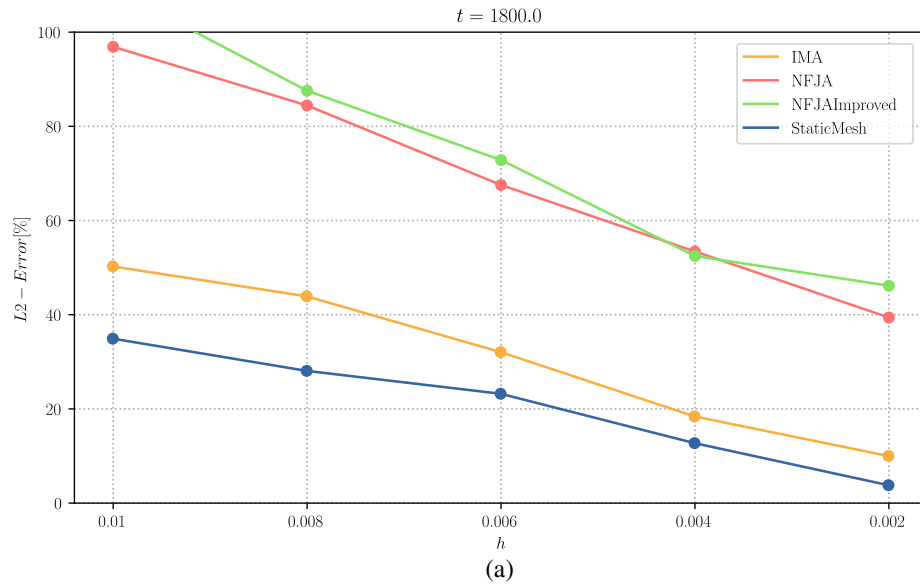
Figure 29: L2-Error over the grain size distributions vs the mesh size h. (a) state at time $t = 1800$, (b) state at time $t = 3600$
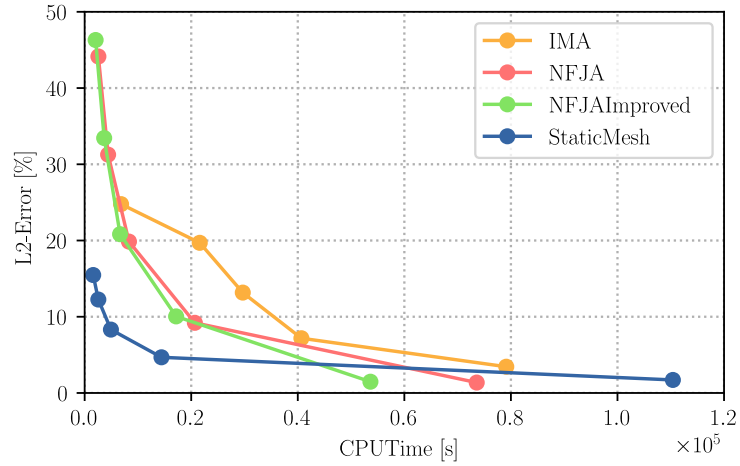
(a)

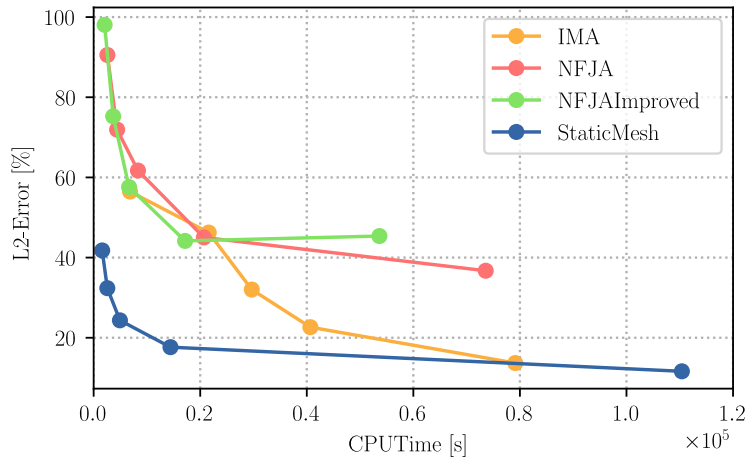(b)

Figure 30: $L2$ Error vs the Total CPU-time at the end of the simulation. Each point of the same curve represent a simulation with a different mesh size (from left to right: $h = [0.01, 0.008, 0.006, 0.004, 0.002]$) . (a) L2-Error over the mean grain size, (a) L2-Error over the grain size distributions

remeshing processes (IMA and NFJA) does not reduce the computation time of a 2D few-thousand multidomain simulation in context of the proposed recent algorithm (diffusive formulation, coloring/recoloring algorithms, optimized direct reinitialization and treatment at multiple junctions).

Figures 28, 29 and 30(b) show that in one hand the IMA case behaves better in terms of accuracy on the prediction of the grain size distributions contrary to 30(a) where the NFJA case predicts better the mean grain size. Of course, the grain size distribution gives a better description of the global state of the polycrystal hence we conclude that the IMA approach is more accurate for a fixed mesh size. On the other hand, if comparing the cpu-time in either 30(a) or (b) the computational cost needed for the IMA case is always bigger than the one for the NFJA case, but the error given by the NFJA remains too important (even for the smallest mesh size h, last point from right to left) to consider it at the optimal candidate for a multidomain simulation. In fact, comparing the grain size distributions (see Figure 28) reveals that the kinetics of the the NFJA method is clearly different from the one using an implicit description of the interfaces (IMA or Static Mesh) beeing faster on the evolution of the small grains and slower on the evolution of the big ones. This conclusion seems coherent with the results described for the T-junction and square-srinkage cases where the kinetics of the triple junctions was systematically overestimated by the NJFA approach comparatively to others. If this weakness is automatically linked to the mesh topological operations realized at the multiple junctions, solved it is not straightforward and constitutes a perspective of this work. Indeed, this approach remains clearly of interest as further improvements could be made with the use of the NFJA that could result in a suitable method to model multidomain problems with a diminution of the CPU-time and of the error that could not be possible to make by employing more classical approach. Indeed, with this approach, geometrical data such as interface normal and curvatures can be computed directly from the body-fitted mesh using the position of the interface nodes only, instead of relying on the costly and inaccurate approximation of the LS field

derivatives which could lead to the direct use of Eq. 3 in a stabilized framework. These perspectives will be described in a forthcoming publication such as the fact to confirm the conclusions of the previous paragraph for larger 2D and 3D simulations and when stored energy due to plastic deformation (involving a convective/diffusive FE formulation as in [1, 2, 45]) has to be considered.

Acknowledgements

Data availability

The raw data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study. The processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

References

[1] L. Maire, B. Scholtes, C. Moussa, N. Bozzolo, D. P. Muñoz, A. Settefrati, M. Bernacki, Modeling of dynamic and post-dynamic recrystallization by coupling a full field approach to phenomenological laws, Materials and Design 133 (2017) 498–519. doi:10.1016/j.matdes.2017.08.015.
URL http://dx.doi.org/10.1016/j.matdes.2017.08.015

[2] B. Scholtes, Development of an efficient level set framework for the full field modeling of recrystallization in 3D., Ph.D. thesis, MINES ParisTech (2016). arXiv:1410.7235, doi:10.1017/jfm.2015.566.

[3] M. Shakoor, M. Bernacki, P.-O. Bouchard, A new body-fitted immersed volume method for the modeling of ductile fracture at the microscale: Analysis of void clusters and stress state effects on coalescence, Engineering Fracture Mechanics 147 (2015) 398–417. doi:10.1016/j.engfracmech.2015.06.057.
URL https://ac.els-cdn.com/S0013794415003239/1-s2.0-S0013794415003239-main.pdf?_tid=53f3

[4] H. Imai, M. Iri, K. Murota, Voronoi diagram in the Laguerre geometry and its applications, SIAM Journal on Computing 14 (1) (1985) 93–105.

[5] K. Hitti, P. Laure, T. Coupez, L. Silva, M. Bernacki, Precise generation of complex statistical Representative Volume Elements (RVEs) in a finite element context, Computational Materials Science 61 (2012) 224–238. doi:10.1016/j.commatsci.2012.04.011.

[6] D. N. Ilin, M. Bernacki, Advancing layer algorithm of dense ellipse packing for generating statistically equivalent polygonal structures, Granular Matter 18 (3) (2016) 43. doi:10.1007/s10035-016-0646-9.

[7] R. Quey, P. R. Dawson, F. Barbe, Large-scale 3D random polycrystals for the finite element method: Generation, meshing and remeshing, Computer Methods in Applied Mechanics and Engineering 200 (17) (2011) 1729–1745. doi:https://doi.org/10.1016/j.cma.2011.01.002. URL http://www.sciencedirect.com/science/article/pii/S004578251100003X

[8] W. Ludwig, A. King, P. Reischig, M. Herbig, E. M. Lauridsen, S. Schmidt, H. Proudhon, S. Forest, P. Cloetens, S. R. du Roscoat, J. Y. Buffière, T. J. Marrow, H. F. Poulsen, New opportunities for 3D materials science of polycrystalline materials at the micrometre lengthscale by combined use of X-ray diffraction and X-ray imaging, Materials Science and Engineering: A 524 (1-2) (2009) 69–76. doi:10.1016/j.msea.2009.04.009.

[9] H. Proudhon, J. Li, P. Reischig, N. Guéninchault, S. Forest, W. Ludwig, Coupling Diffraction Contrast Tomography with the Finite Element Method, Advanced Engineering Materials 18 (6) (2016) 903–912. doi:10.1002/adem.201500414.

[10] P. G. Young, T. B. H. Beresford-West, S. R. L. Coward, B. Notarberardino, B. Walker, A. Abdul-Aziz, An efficient approach to converting three-dimensional image data into highly accurate computational models., Philosophical transactions. Series A, Mathematical, physical, and engineering sciences 366 (1878) (2008) 3155–3173. doi:10.1098/rsta.2008.0090.

[11] Y. Zhang, C. Bajaj, B.-S. Sohn, 3D Finite Element Meshing from Imaging

Data., Computer methods in applied mechanics and engineering 194 (48-49) (2005) 5083–5106. doi:10.1016/j.cma.2004.11.026.

[12] A. Rollett, D. Saylor, J. Fridy, B. El-Dasher, A. Brahme, S.-B. Lee, C. Cornwell, R. Noack, Modeling Polycrystalline Microstructures in 3D, in: S. Ghosh, J. Castro, J. Lee (Eds.), American Institute of Physics Conference Series, Vol. 712 of American Institute of Physics Conference Series, 2004, pp. 71–77.

[13] A. Brahme, M. H. Alvi, D. Saylor, J. Fridy, A. D. Rollett, 3D reconstruction of microstructure in a commercial purity aluminum, Scripta Materialia 55 (1) (2006) 75–80. doi:https://doi.org/10.1016/j.scriptamat.2006.02.017. URL http://www.sciencedirect.com/science/article/pii/S1359646206001229

[14] S. Osher, J. A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, Journal of Computational Physics 79 (1) (1988) 12–49. doi:10.1016/0021-9991(88)90002-2.

[15] I. Steinbach, Phase-field models in materials science, Modelling and Simulation in Materials Science and Engineering 17 (7) (2009) 073001.

[16] M. Bernacki, H. Resk, T. Coupez, R. E. Logé, Finite element model of primary recrystallization in polycrystalline aggregates using a level set framework, Modelling and Simulation in Materials Science and Engineering 17 (6) (2009) 64006. doi:10.1088/0965-0393/17/6/064006.

[17] B. Scholtes, M. Shakoor, A. Settefrati, P.-O. Bouchard, N. Bozzolo, M. Bernacki, New finite element developments for the full field modeling of microstructural evolutions using the level-set method, Computational Materials Science 109 (2015) 388–398. doi:10.1016/j.commatsci.2015.07.042.

[18] B. Scholtes, R. Boulais-Sinou, A. Settefrati, D. Pino Muñoz, I. Poitrault, A. Montouchet, N. Bozzolo, M. Bernacki, 3D level set modeling of static

recrystallization considering stored energy fields, Computational Materials Science 122 (2016) 57–71. doi:10.1016/j.commatsci.2016.04.045.

[19] M. Bernacki, R. E. Logé, T. Coupez, Level set framework for the finite-element modelling of recrystallization and grain growth in polycrystalline materials, Scripta Materialia 64 (6) (2011) 525–528. doi:10.1016/j.scriptamat.2010.11.032.

[20] M. Shakoor, P.-O. Bouchard, M. Bernacki, An adaptive level-set method with enhanced volume conservation for simulations in multiphase domains, International Journal for Numerical Methods in Engineering 109 (4) (2017) 555–576. doi:10.1002/nme.5297.

[21] H. Hallberg, A modified level set approach to 2D modeling of dynamic recrystallization, Modelling and Simulation in Materials Science and Engineering 21 (8) (2013) 85012. doi:10.1088/0965-0393/21/8/085012.

[22] M. Bernacki, Y. Chastel, T. Coupez, R. E. Logé, Level set framework for the numerical modelling of primary recrystallization in polycrystalline materials, Scripta Materialia 58 (12) (2008) 1129–1132. doi:10.1016/j.scriptamat.2008.02.016.

[23] A. L. Cruz-Fabiano, R. Logé, M. Bernacki, Assessment of simplified 2D grain growth models from numerical experiments based on a level set framework, Computational Materials Science 92 (2014) 305–312. doi:10.1016/j.commatsci.2014.05.060.

[24] L. Maire, B. Scholtes, C. Moussa, D. Pino Muñoz, N. Bozzolo, M. Bernacki, Improvement of 3-D mean field models for pure grain growth based on full field simulations, Journal of Materials Science 51 (24) (2016) 10970–10981.

[25] J. Furstoss, M. Bernacki, C. Ganino, C. Petit, D. Pino-Muñoz, 2D and 3D simulation of grain growth in olivine aggregates using a full field model based on the level set method, Physics of the Earth and Planetary Interiors 283 (2018) 98–109.

[26] M. Shakoor, A. Buljac, J. Neggers, F. Hild, T. F. Morgeneyer, L. Helfen, M. Bernacki, P.-O. Bouchard, On the choice of boundary conditions for micromechanical simulations based on 3D imaging, International Journal of Solids and Structuresdoi:10.1016/j.ijsolstr.2017.02.018.

[27] M. Shakoor, B. Scholtes, P.-O. Bouchard, M. Bernacki, An efficient and parallel level set reinitialization method – Application to micromechanics and microstructural evolutions, Applied Mathematical Modelling 39 (23-24) (2015) 7291–7302. doi:10.1016/j.apm.2015.03.014.

[28] B. Merriman, J. K. Bence, S. J. Osher, Motion of multiple junctions: A level set approach (1994). doi:10.1006/jcph.1994.1105.

[29] M. Elsey, S. Esedoglu, P. Smereka, Simulations of anisotropic grain growth: Efficient algorithms and misorientation distributions, Acta Materialia 61 (6) (2013) 2033–2043. doi:10.1016/J.ACTAMAT.2012.12.023. URL https://www.sciencedirect.com/science/article/abs/pii/S1359645412008816?via%3Dihub

[30] H. Hallberg, V. V. Bulatov, Modeling of grain growth under fully anisotropic grain boundary energy Modeling of grain growth under fully anisotropic grain boundary energy, Modelling and Simulation in Materials Science and Engineering 27 (045002).

[31] M. Shakoor, M. Bernacki, P.-O. Bouchard, A new body-fitted immersed volume method for the modeling of ductile fracture at the microscale: Analysis of void clusters and stress state effects on coalescence, Engineering Fracture Mechanics 147 (2015) 398–417. doi:10.1016/j.engfracmech.2015.06.057.

[32] J. Fausty, N. Bozzolo, D. Pino Muñoz, M. Bernacki, A novel level-set finite element formulation for grain growth with heterogeneous grain boundary energies, Materials & Design 160 (2018) 578–590.

[33] O. C. Zienkiewicz, J. Z. Zhu, The superconvergent patch recovery (SPR) and adaptive finite element refinement, Computer Methods in Applied

Mechanics and Engineering 101 (1-3) (1992) 207–224. doi:10.1016/0045-7825(92)90023-D.

[34] O. C. Zienkiewicz, J. Z. Zhu, Superconvergence and the superconvergent patch recovery, Finite Elements in Analysis and Design 19 (1-2) (1995) 11–23. doi:10.1016/0168-874X(94)00054-J.

[35] N.-E. Wiberg, Superconvergent Patch Recovery — a key to quality assessed FE solutions, Advances in Engineering Software 28 (2) (1997) 85–95. doi:10.1016/S0965-9978(96)00045-2.
URL https://www.sciencedirect.com/science/article/pii/S0965997896000452

[36] J. A. Sethian, A fast marching level set method for monotonically advancing fronts., Proceedings of the National Academy of Sciences 93 (4) (1996) 1591–1595. doi:10.1073/pnas.93.4.1591.
URL http://www.pnas.org/cgi/doi/10.1073/pnas.93.4.1591

[37] R. Kimmel, J. A. Sethian, Computing geodesic paths on manifolds, Proceedings of the National Academy of Sciences 95 (15) (1998) 8431–8435. doi:10.1073/pnas.95.15.8431.

[38] J. A. Sethian, A. Vladimirsky, Fast methods for the Eikonal and related Hamilton- Jacobi equations on unstructured meshes, Proceedings of the National Academy of Sciences 97 (11) (2000) 5699–5703. doi:10.1073/pnas.090060097.

[39] M. Sussman, P. Smereka, S. Osher, A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow, Journal of Computational Physics 114 (1) (1994) 146–159. doi:10.1006/JCPH.1994.1155.
URL https://www.sciencedirect.com/science/article/pii/S0021999184711557?via%3Dihub

[40] T. Coupez, H. Digonnet, R. Ducloux, Parallel meshing and remeshing, Applied Mathematical Modelling 25 (2) (2000) 153–175. doi:10.1016/S0307-904X(00)00045-7.
URL https://linkinghub.elsevier.com/retrieve/pii/S0307904X00000457

[41] C. Gruau, Metric generation for anisotropic mesh adaption with numerical applications to material forming simulation, Ph.D. thesis, MINES Paris-Tech (2004).

[42] C. Gruau, T. Coupez, 3D tetrahedral, unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric, Computer Methods in Applied Mechanics and Engineering 194 (48-49) (2005) 4951–4976. doi:10.1016/J.CMA.2004.11.020.
URL https://www.sciencedirect.com/science/article/pii/S0045782505000745?via%3Dihub

[43] C. Herring, Surface tension as a motivation for sintering, in: Fundamental Contributions to the Continuum Theory of Evolving Phase Interfaces in Solids, Springer, 1999, pp. 33–69.

[44] H. Garcke, B. Nestler, B. Stoth, A Multiphase Field Concept : Numerical Simulations of Moving Phase Boundaries and Multiple Junctions, Applied Mathematics 60 (1) (1999) 295–315.

[45] B. Scholtes, R. Boulais-Sinou, A. Settefrati, D. Pino Muñoz, I. Poitrault, A. Montouchet, N. Bozzolo, M. Bernacki, 3D level set modeling of static recrystallization considering stored energy fields, Computational Materials Science 122 (2016) 57–71. doi:10.1016/j.commatsci.2016.04.045.
URL http://dx.doi.org/10.1016/j.commatsci.2016.04.045