Sintering at Particle Scale: An Eulerian Computing Framework to Deal with Strong Topological and Material Discontinuities

Daniel Pino Muñoz, Julien Bruchon, Sylvain Drapier, François Valdivieso

Sintering at Particle Scale: An Eulerian Computing Framework to Deal with Strong Topological and Material Discontinuities

D. Pino-Muñoz · J. Bruchon · S. Drapier · F. Valdivieso

Abstract This work presents a numerical modelling approach of particle packing consolidation, at the particle scale, based on specific numerical methods implemented in a high-performance computing framework. Typically, the sintering process triggers several mass transport paths, thermally activated, that are driven by geometrical as well as physical gradients and laplacians. Computing precisely such major characteristics is of paramount importance but represents a real scientific challenge, which have not been fully solved yet but which must however be tackled to gain precious insights into sintering mechanisms which are seldom accessible at this scale. An Eulerian-based formulation is then proposed here to deal with the strong topological changes related to particle sintering. Also, a specific attention is paid to the precise and robust computation of high-order derivatives which are known to control the physics of surface solid diffusion, namely the surface laplacian of the curvature. Besides, the hydrostatic pressure gradient is known to control the volume diffusion path, it results from the coupled fluid-solid mechanical equilibrium, including surface tension, which must be solved precisely. Furthermore, a mesh adaptation technique allows the particles surface description to be improved, while the number of mesh elements is kept reasonable. Once verified on test-cases, this numerical approach is applied to several 3D granular packings undergoing micro-structural changes under combined surface and volume diffusion.

D. Pino-Muñoz (✉) · J. Bruchon · S. Drapier · F. Valdivieso
Lab. G. Friedel, UMR CNRS 5307, École des Mines de Saint-Étienne, Saint-Étienne, France
e-mail: daniel.pinomunoz@epfl.ch

J. Bruchon
e-mail: bruchon@emse.fr

S. Drapier
e-mail: drapier@emse.fr

F. Valdivieso
e-mail: valdivieso@emse.fr

Present Address:
D. Pino-Muñoz
Computational Solid Mechanics Laboratory (LSMS), EPFL, Lausanne, Switzerland

1 Introduction

Sintering process is, nowadays, a very important industrial process used for the manufacturing of countless materials and solid parts. Considerable attention has been paid to the understanding of the physical mechanisms responsible of the morphological changes that occur during sintering. Regarding the complexity of the parameters combination controlling sintering, it may be somehow difficult to extract pertinent information from experimental data.

Numerical simulations can help in providing meaningful information about the phenomena controlling sintering provided robust methods can be settled. Classically the macroscopical behavior is usually approximated by using phenomenological models accounting for the evolution of the microtructure. Conversely, modeling the sintering process at the microscopic scale can bring very local information. Considering new numerical approaches, in a high-performance computing framework could help bridging the gap between these two scales, provided hundreds of particles can be investigated to establish micro-macro scale transitions.

A consolidated powder is a porous packing of particles which presents a surface free energy excess, directly related

to the specific surface of the compact. Reduction of this free energy excess is achieved by diffusional transport between the particles, i.e. sintering, leading to a reduction of the surface. The complexity of the geometry of a consolidated powder combined with both the topological changes that can appear during sintering and with all the physical phenomena, make simulation of sintering at the microscopic scale a very challenging task. In the present work, the sintering of ceramic materials is modeled and investigated numerically at the microscopic scale. The main underlying physical mechanisms responsible of the microstructural evolution during solid state sintering process are surface diffusion, volume diffusion and grain boundary diffusion. Several challenges have to be addressed in order to integrate those physical mechanisms into a continuum formulation leading to a numerical approach capable of simulating the sintering process. In this work the solid state sintering by surface and volume diffusion will be studied. There are several unanswered questions about the underlying physical phenomena on sintering, this work represents a step toward a better understanding on the sintering process through surface and volume diffusions, and to a least extent the effects of the grain boundary diffusion.

This approach at the local scale is conceivable also because the frame of high performance computing was considered from the onset. This development could be the starting point for embedding microstructural evolution into macroscopic models. Moreover, a Eulerian level-set (LS) approach is used to integrate the diffusion mechanisms into a finite element continuum mechanics framework. It specially permits to deal, simultaneously, with severe topological changes which characterize powder sintering. All this is possible because of the power of computers has dramatically been increased by parallel computing techniques allowing the simulation of larger systems.

The generalities of the sintering process will be presented along with the Fick's laws used to describe the diffusion phenomena in Sect. 2. Concerning the sintering simulation at the particles scale, different numerical approaches are available in literature. Among those models there are some analytical laws which allow to predict the growth of the neck between two particles. Those analytical models as well as more elaborated numerical approaches will be presented in Sect. 3. The lack of a numerical strategy able to handle simulation of sintering in 3D by different diffusion mechanism is evident. For this reason a numerical approach for the simulation of sintering at the particles scale will be proposed. The computational framework, based on the LS method, will be set in Sect. 4. Next, a method allowing to simulate sintering by surface diffusion will be presented in Sect. 5. In order to introduce the volume diffusion into the numerical approach proposed, the pressure field inside the particles has to be computed. Taking into account this pressure computation, the numerical strategy for the sintering simulation for volume diffusion will be

presented in Sect. 6. Finally the coupling between the volume diffusion and the surface diffusion will be presented in Sect. 7, as well as the bases for the grain boundary diffusion.

2 Sintering and Diffusion Phenomena

Even if the use of powder metallurgy has started from the beginning of human civilization, it was only after the 1940s that sintering phenomena were studied scientifically. Since then, a large number of scientific publications have appeared, leading to remarkable developments in the sintering science along with advances from the practical point of view. However, understanding in depth the underlying physical phenomena is still a bottle neck, meanwhile today's challenge is to produce sintered parts with reproducible and controlled microstructure through the control of the process variables.

In Sect. 2.1 a short general introduction to the manufacturing process by sintering of consolidated powder is presented. This manufacturing process presents several stages, but the present work will be concerned essentially with sintering stage which will be shown to be controlled by matter diffusion phenomena, the theory of which is developed in Sect. 2.2. Finally the conclusions are presented in Sect. 2.3.
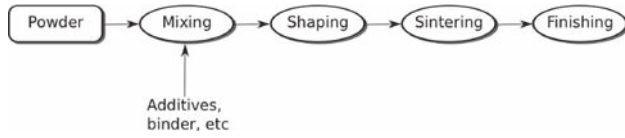
2.1 Manufacturing Process by Sintering

From the physical point of view, sintering is a thermally activated phenomenon driven by the excess of free energy of the system and which allows the passage from a compact powder to a coherent material, where the solid body acquires some specific mechanical or functional properties. Today, sintering is a widely used manufacturing process and its applications are widespread: high performance structural parts, porous materials for multiple applications, carbides for cutting tools, biocompatibles materials and medical devices are some examples of such applications.

Different types of sintering technics exist, such as hot isostatic pressing sintering, microwave sintering, spark plasma sintering, free sintering, …. Their applications as well as their modelling are very different. The present research is concerned by the free sintering. Even if in this work the emphasis will be placed on the study of the sintering stage itself, it is important to know the preliminary stages. Indeed, several studies in literature show that the preliminary operations have an important impact on the properties of the finished part or material [73].

In order to produce a sintered part it is necessary to follow the stages reported in the flow chart of Fig. 1:

1. *Powder preparation* The powder characteristics of greatest interest are the particle size and size distribution, particles shape, degree of agglomeration, chemical compo-

Fig. 1 General flow chart of the sintering process

Fig. 2 Packing density during the sintering stages

Fig. 3 Neck between the particles created during sintering ([33], p. 71)

sition and purity, which all depend on the preparation method used (mechanical or chemical).

2. *Mixing* The powder is dispersed into a liquid to create a colloidal suspension to improve the homogeneity of the material. Some additives (solvents, dispersants, binders, plasticizers, etc.) are introduced during this step, and are usually removed by heating the green body at temperatures around 500 °C or by dissolution in a solvent.

3. *Shaping* This stage, also known as *forming*, is used to shape the powder into a particular geometry, resulting in the *green body* of the part. One of the most important variables to be controlled in this stage is the packing density. A high packing density is favorable to the fabrication of fully dense materials associated with reduced sintering durations. This packing density depends on multiple parameters such as process parameters (forming pressure, temperature, etc.) and morphological parameters (particle size, etc.). It can be defined as

$$\text{Packing density} = \frac{\text{Volume of the particles}}{\text{Total volume : particles + voids}} \quad (1)$$

4. *Sintering* The main goal of this stage is to heat the consolidated powder to obtain the desired packing density and microstructure. Raise on temperature triggers some diffusion mechanisms that are responsible for the microstructural evolution of the structure. Two main sintering categories can be identified wether one phase, at least melts: solid state and liquid phase sintering. The present work is dedicated to the study of the solid state sintering at the particle scale and will focus on this specific stage of sintering of consolidated powders.

5. *Finishing* According to the application, additional operations are performed to get the final product.

2.1.1 Sintering Step

As a function of the packing density, it is possible to identify three sequential stages during the sintering process (see Fig. 2):

1. In the first stage (Fig. 2—green), the neck bridging the particles is rapidly created and the particles of the powder system are still distinguishable (see Fig. 3). This stage is supposed to last until the radius of the neck between the particles has reached a value about 0.4–0.5 of the
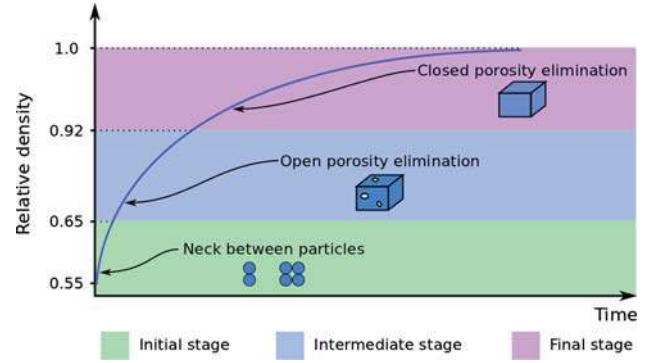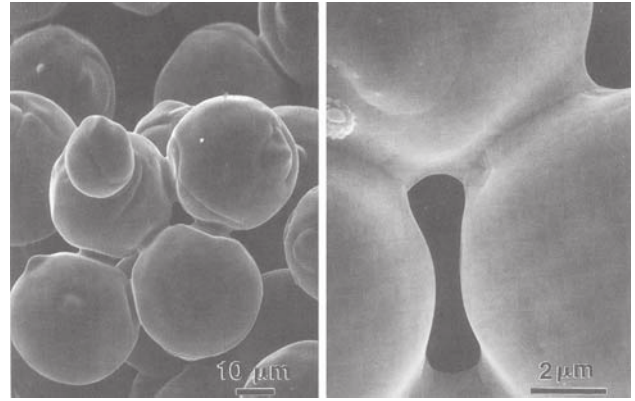
particle radius. From the macroscopic point of view, this is equivalent to a packing density of about 0.65 [2,73].

2. At the beginning of the intermediate stage the porosity is still open, i.e. pores are interconnected and their shape is roughly cylindrical. As matter diffusion takes place porosity shrinks (Fig. 2—blue), some isolated pores appear and the packing density continues raising up to a value of about 0.92. This stage covers the major part of the sintering process.

3. The final stage (Fig. 2—magenta) leads to the final microstructure of the material. Pores, which have become isolated and spherical keep on shrinking to finally end up by almost disappearing. The final relative density of the material can raise up to 0.999.

It is possible to distinguish at least six mechanisms leading to the necks growing and/or the densification of the solid [2]. Those six matter transport paths shown in Fig. 4, have a common driving force: the reduction of the total free surface area which is directly related to the surface free energy of the system—of the order of 0.1–100 J/mol for coarse to fine powders [28,39]. These six diffusion mechanisms shown in Fig. 4 correspond to surface diffusion (1), volume diffusion
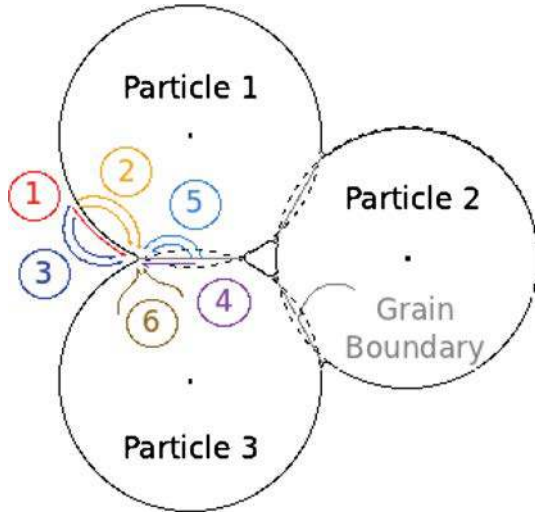
**Fig. 4** The six different diffusion paths, from [2]

Table 1 Matter transport paths

Diffusion path		Source	Sink	Densif.
1	Surf. diff.	Surface	Neck	
2	Vol. diff.	Surface	Neck	
3	Vapor transp.	Surface	Neck	
4	Grain bd. diff.	Grain bd.	Neck	✓
5	Vol. diff.	Grain bd.	Neck	✓
6	Vol. diff.	Dislo.	Neck	✓

(2, 5 and 6), grain boundary diffusion (4) and vapor transport (3). During diffusion the total free surface is reduced, but in the meantime the grain boundary surface—the solid-solid interfaces—increases. The grain boundary energies usually are lower than surface energies. However the matter transport can stop because of the establishment of local equilibrium between grain boundary and free surface energy [28,42].

All the paths presented in Fig. 4 contribute simultaneously to the neck growth, but only some of them lead to densification. Table 1 shows the sources and the sink of matter for each diffusion path presented in Fig. 4. The "Densifying" column indicates if the diffusion path leads whether or not to densification.

At the particle scale, densification is defined as the rate at which the particle centres approach each other. Considering an atom or a mole of material, according with its position in/on the solid (surface, volume or grain boundary) an associated *chemical potential* can be defined and the matter flux is shown to be proportional to the gradient of this chemical potential. This will be further detailed in Sect. 2.2.
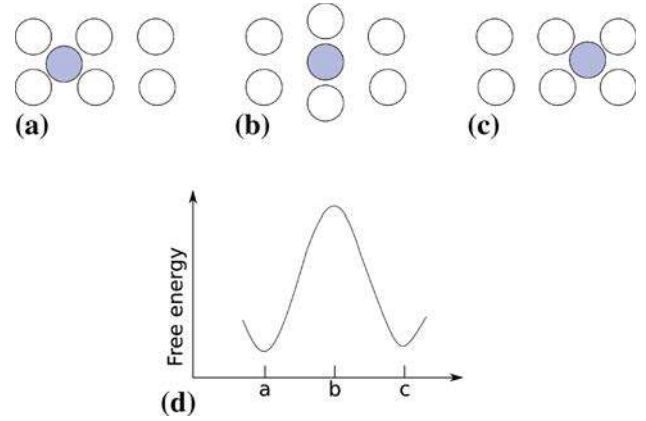
Fig. 5 Schematic diagram showing the configuration of a set of atoms when one of them changes its position (**a**–**c**) and the corresponding free energy of the lattice (**d**)

2.2 Diffusion Phenomena

The *Fick's laws* describe the movement of chemical species as a function of the gradient of concentration. In cases where the concentration is independent on time, the diffusion process can be described by using Fick's first law [44], which states that the flux of diffusing species is proportional to the gradient of concentration as follows:

$$\mathbf{j} = -D\nabla \mathbf{C} \tag{2}$$

where D is the diffusion coefficient and is assumed to be independent on concentration [73]. This expression can be extended easily to orthotropic cases, and diffusion is in this case characterized by a second order diffusion tensor $\mathbf{D}$.

It is very important to recall that diffusion is thermally activated. This means that an *activation energy* should be supplied to the system to trigger diffusion. For example, consider the three (a, b, and c) states presented in Fig. 5. In order to induce the change in position of the interstitial atom shown in Fig. 5a–c, the lattice should be distorted in the intermediate position (Fig. 5b). As the energy in the intermediate state is higher than in the initial one, this distortion can only be achieved by supplying some supplementary energy to the system. As the level of energy of the atoms is different, only a certain fraction of atoms will have sufficient energy to be able to move from one position to another. Considering this fact, the diffusion coefficient D is expected to depend on the temperature as follows [44,73]:

$$D = D_0 \exp\left(\frac{-Q}{RT}\right) \tag{3}$$

where Q is the activation energy, R is the gas constant and T is temperature.

As outlined earlier, the diffusion phenomena occuring during sintering can be studied by using Fick's first law (Eq. 2) where the matter flux is proportional to the concentration gra-

dient. However in literature [2,28,39,44,53,73], sintering is considered as a *chemical diffusion process* and it is usually studied in terms of *chemical potential*. Instead of using the concentration gradient for modeling the matter flux, this flux is considered to be proportional to the chemical potential gradient as detailed below in Sect. 2.2.1.

2.2.1 Chemical Potential

The chemical potential measures the tendency to diffuse of a substance, and the coresponding matter flux goes downward the chemical potential gradient, described through a classical *first gradient law.*

In order to introduce the *chemical potential*, let us consider a pure solid substance in which vacancies are the only kind of point defect present. If the local equilibrium between vacancies and atoms is assumed, the *Gibbs-Duhem* relation holds [39,73]:

$$G = \mu_a N_a + \mu_v N_v \tag{4}$$

where μ_a is the chemical potential of the atoms, μ_v the chemical potential of the vacancies, N_a is the number of atoms, N_v is the number of vacancies and G is the Gibbs free energy which is defined as a function of temperature T, pressure p and internal energy U as follows:

$$G = U + pV - TS \tag{5}$$

with V the volume and S the entropy of the system.

The chemical potentials which represent the energy brought by an atom, μ_a, and by a vacancy, μ_v, can be derived from the previous equation:

$$\mu_a = \left(\frac{\partial G}{\partial N_a}\right)_{N_v, T, p} \tag{6}$$

$$\mu_v = \left(\frac{\partial G}{\partial N_v}\right)_{N_a, T, p} \tag{7}$$

The change in internal energy U can then be written as a function of the change of the extensive quantities[1] S, V, N_a and N_v:

$$dU = TdS - pdV + \mu_a dN_a + \mu_v dN_v \tag{8}$$

Temperature T, pressure p, chemical potential of the atoms μ_a and vacancies μ_v, are the intensive quantities of the system (that do not depend on the amount of material).

Equation (8) tells that the energy of a system can change in different ways: by changing its entropy S, its volume V, the amount of atoms N_a or the amount of vacancies N_v. The intensive quantities (T, p, μ_a, μ_v) determine the magnitude of the energy change related to the change of the corresponding extensive quantities (S, V, N_a, N_v). For example, given a change of the system entropy dS, the energy

---

[1] That depend on the amount of material of the system.
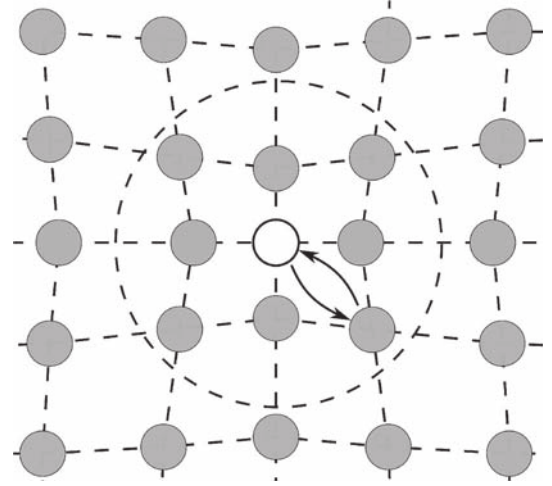
Fig. 6 Schematic diagram showing the rearrangement of the lattice around a vacancy

increase is large if the temperature is high, and is small if the temperature is low [51]. Indeed, the change of the free energy of the system is given by the sum of two contributions, the change in internal energy δU and the change of the surface free energy.

Finally a set of thermodynamical expressions that will be useful for future reference can be introduced:

$$\left(\frac{\partial \mu_a}{\partial p}\right)_{N_v, T} = \left(\frac{\partial^2 G}{\partial p \partial N_a}\right)_{N_v, T} = \Omega \tag{9}$$

$$\left(\frac{\partial \mu_v}{\partial p}\right)_{N_a, T} = \left(\frac{\partial^2 G}{\partial p \partial N_v}\right)_{N_a, T} = f\Omega \tag{10}$$

where the volume of a vacancy is supposed to be the fraction f of the atomic volume Ω [30,39]. A physical sense of the dependence of these chemical potentials upon mechanical state, namely pressure here, is easily seized in Fig. 6 which shows how the lattice is distorted because of the swap between a vancancy and an atom, i.e. atom diffusion.

In addition, the change in chemical potentials μ_a and μ_v with respect to the change in vacancies concentration C_v is mainly due to the entropy mixing of atoms and vacancies [39]. The vacancies concentration is defined as a function of the number of atoms N_a and the number of lattice sites N_L in a given volume: $C_v = (N_L - N_a)/N_L$. Since $C_v \ll 1$, it is possible to write:

$$\left(\frac{\partial \mu_v}{\partial C_v}\right)_{p, T} = \frac{kT}{C_v} + \mathcal{O}(1) \tag{11}$$

$$\left(\frac{\partial \mu_a}{\partial C_v}\right)_{p, T} = -kT + \mathcal{O}(C_v) \tag{12}$$

The diffusional transport can take place only by migration of interstitial atoms or lattice vacancies and neither of these processes changes the number of lattice sites in the region.
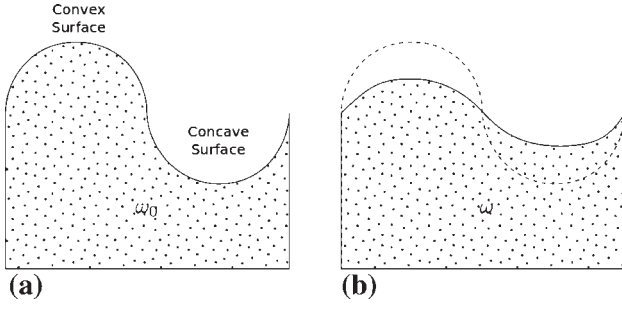
Fig. 7 Schematic diagram showing a region ω of a solid Ω_s with a concave and a convex surface. **a** Initial shape ω_0. **b** Final shape ω (the *dashed line* represents the initial shape)

In other words the number of atoms N_a and the number of vacancies N_v change by equal and opposite amounts [39]. Under those conditions, the change of the free energy of a region is therefore equal to the number of atoms entering or leaving it multiplied by the difference of chemical potential involved in the atom-vacancies switching $(\mu_a - \mu_v)$. Moreover, the diffusional flux is given by:

$$\mathbf{j} = -\frac{D}{kT\Omega}\left(\nabla \boldsymbol{\mu_a} - \nabla \boldsymbol{\mu_v}\right) \quad (13)$$

The previous expressions will be used in the following sections in order to establish the expression of the chemical potential beneath a surface (2.2.2), on the volume (2.2.3) and over the grain boundary (2.2.4).

2.2.2 Surface Diffusion

In order to introduce the chemical potential beneath a curved surface, consider Fig. 7a, where a part ω of solid Ω_s with a concave and a convex surface is presented.
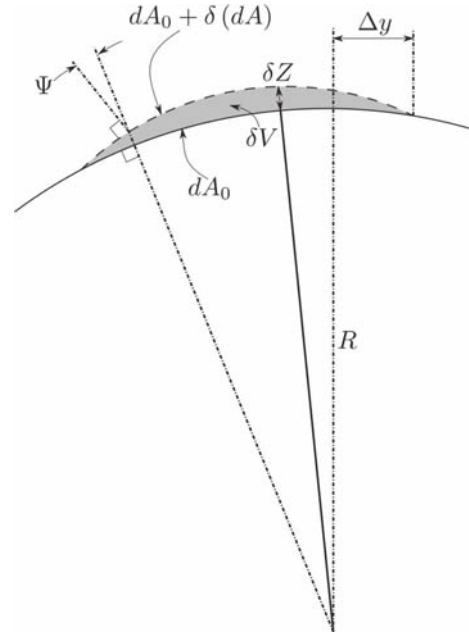
The *specific surface free energy*[2] (γ_{sf}) of a crystalline solid is defined as the increase of energy when the area of the free surface $\partial\omega$ of the crystal is increased by a unit amount. The surface free energy E_s of the system is then given by:

$$E_s = \int_{\partial\omega} \gamma_{sf}\, dA \quad (14)$$

In this work γ_{sf} is considered to be constant along the surface.

By considering Eq. (14), it is possible to show that the solid in Fig. 7a has a higher surface free energy than the solid presented in Fig. 7b. Therefore a way to reduce the energy is to reduce the total surface of the solid, and this can be achieved by transporting matter from the convex region to the concave region or globally to tend toward a surface of lower curvature.

An expression of the chemical potential can be found by establishing that the free energy is a minimum with respect

---

[2] Sometimes also known as *surface tension coefficient* or for the sake of simplicity just *surface tension*.

Fig. 8 Infinitesimal hump formed building up a curved surface. *Full line* original surface. *Dashed line* built-up surface [39]

to any infinitesimal virtual change in which the local shape of the surface is altered by removing atoms from the interior and placing them on the surface [35,39], or vice versa (see Fig. 8).

Considering a smoothly curved surface as shown in Fig. 8, the change of the surface free energy is given by:

$$\delta\left(\int_{\partial\omega} \gamma_{sf}\, dA\right) = \int_{\partial\omega_0} \delta\left(\gamma_{sf}\right) dA_0 + \int_{\partial\omega} \gamma_{sf}\delta\left(dA\right) \quad (15)$$

where δ represents a small change of the quantity, dA_0 is the area of an element of the original surface and $dA_0 + \delta dA$ the area of the element of the final surface. The term $\delta\gamma_{sf}$ vanishes since γ_{sf} is considered to be constant.

Considering Fig. 8, $\delta\left(dA\right)$ is given by:

$$\delta\left(dA\right) = \left(\frac{1}{\cos\Psi} - 1\right)dA_0 + \delta Z\left(\frac{1}{R_1} + \frac{1}{R_2}\right)dA_0 \quad (16)$$

where R_1 and R_2 are the two principal radii of curvature at dA_0. A principal radius of curvature is considered to be positive for convex surface, e.g. $R_1 = R_2 = R$ for a sphere of radius R. If Δy (Fig. 8) is small, then R_1 and R_2 may be taken constant in the hump and $(1/\cos\Psi - 1) \simeq 0$. Finally Eq. (15) can be simplified as follows:

$$\delta\left(\int_{\partial\omega} \gamma_{sf}\, dA\right) = \gamma_{sf}\left(\frac{1}{R_1} + \frac{1}{R_2}\right)\delta V \quad (17)$$

with δV the volume of the hump: $\delta V = \int_{\partial\omega} \delta Z\, dA_0$.

From Equ. (8) and considering an isothermal process where only a vacancy is brought (no change on the number of atoms), the change of the volume term of the free energy due to the creation of the hump is:

6

$$\delta U = -p\delta V + \mu_v \frac{\delta V}{\Omega} \tag{18}$$

where p is the hydrostatic pressure beneath the surface element being considered and $\frac{\delta V}{\Omega}$ represents the number of atoms which must be brought into this portion of the crystal to build up the hump. As the shape of the surface should be in equilibrium with respect to the creation or annihilation of small humps, the internal and external energies variations [sum of Eqs. (18) and (15)] must vanish [35,39,73]:

$$\delta U + \delta \left(\int_{\partial \omega} \gamma_{sf} dA \right) = 0$$
$$\Rightarrow -p\delta V + \mu_v \frac{\delta V}{\Omega} + \gamma_{sf} \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \delta V = 0 \tag{19}$$

From Eq. (19), the vacancies chemical potential beneath a curved surface can be written as:

$$\mu_v = p\Omega - \gamma_{sf} \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \Omega = p\Omega - \gamma_{sf} \kappa \Omega \tag{20}$$

where κ is the mean curvature. In order to get an expression of the diffusional flux on surface $\mathbf{j^s}$, the atom chemical potential should be computed. Since the vacancy concentration C_v is always $\ll 1$, Eq. (12) shows that variations in μ_a due to variations of C_v will be negligible compared to the corresponding variation of μ_v (Eq. 11) [39]. Therefore by using Eq. (9) one can write:

$$\mu_a = \mu_{a_0} + p\Omega \tag{21}$$

where μ_{a_0} is the atoms chemical potential in a free-stress state.

Finally, the surface atom flux $\mathbf{j^s}$ is given by:

$$\mathbf{j^s} = -\frac{D_s \gamma_{sf}}{kT} \nabla_{\mathbf{s}} \kappa \tag{22}$$

where D_s is the surface diffusion coefficient. The surface diffusion path will then depend mainly on the geometrical representation of the particles, and mostly on their curvature κ. And since the chemical potential is defined along the surface of the particles, the gradient in this Fick's law has to be computed along the surface of the particles (∇_s).

2.2.3 Volume Diffusion

Consider the solid presented in Fig. 6 (page 8) where the lattice is deformed because of the presence of a vacancy. The deformation of the lattice due to the presence of a vacancy is supposed to be isotropic. This means that a vacancy that has been brought into a portion of the crystal will induce a virtual change of the volume δV. The number of vacancies required to generate that volume change is given by $\delta V / f\Omega$.

Furthermore, as the system should be in equilibrium with respect to this virtual volume change, the change of the internal energy δU must equate zero:

$$\delta U = -p\delta V + \mu_v \frac{\delta V}{f\Omega} = 0 \tag{23}$$

From previous Eq. (23), it is possible to establish an expression for the vacancy chemical potential μ_v inside the volume:

$$\mu_v = f\, p\Omega \tag{24}$$

Like in previous Sect. 2.2.2, the atom chemical potential μ_a is considered not to depend on the vacancy concentration C_v as $C_v \ll 1$. Therefore the atom chemical potential μ_a will be, again, given by Eq. (21). And the volume atom flux can be written as follows:

$$\mathbf{j^v} = -(1-f)\frac{D_v}{kT} \nabla\, \mathbf{p} \tag{25}$$

where D_v is the volume diffusion coefficient. Conversely to surface diffusion, the volume diffusion route will depend on the local mechanical equilibrium of the particle, including its surface in contact with the surrouding medium.

2.2.4 Grain Boundary Diffusion

Even if this work is mainly concerned by surface and volume diffusion, the grain boundary diffusion is a very important path of sintering and the theory related to the matter transport by this mechanism is presented. As in Sect. 2.2.2, the change of the free energy of the system is given by the sum of two contributions, the change in internal energy δU and the change of the surface free energy. The main difference is related to the internal energy that can be lowered by the migration of matter from one particle to its neighbor. If the internal energy of one of the particles forming the grain boundary is higher than the internal energy of its neighbor, matter will migrate and the particle of lower energy will grow at the expense of the other particle.

By making the same kind of development as those presented in Sects. 2.2.2 and 2.2.3, it is possible to show that the chemical potential at the grain boundary between the particles can be written as a function of the normal stress σ_{nn} [39,64,88]:

$$\mathbf{j^{gb}} = -\frac{D_{gb}}{kT} \nabla_{\mathbf{gb}}\, \sigma_{\mathbf{nn}} \tag{26}$$

where D_{gb} is the grain boundary diffusion coefficient. Like in the surface diffusion, ∇_{gb} corresponds to a gradient computed along the grain boundary.

2.2.5 Diffusion Induced Velocity

According to the path followed by the diffusion flux, the induced velocity should be computed in a different way. In the case of the surface and grain boundary diffusions, matter is transported along the interfaces. For the surface diffusion, the flux follows the free surface and for the case of the

(a) Surface or Grain boundary diffusion
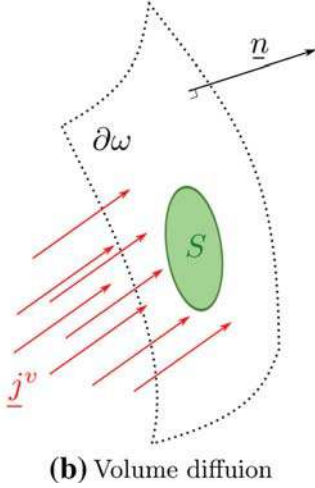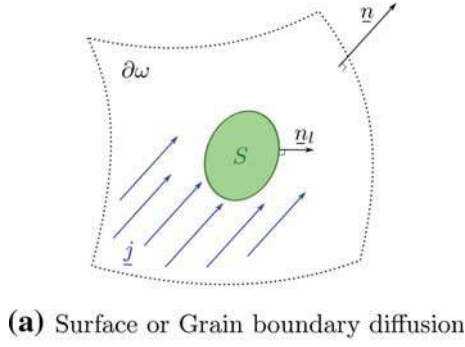
(b) Volume diffuion

Fig. 9 Elementary surface S under a matter flux due to: **a** Surface diffusion $\mathbf{j} = \mathbf{j}^s$ or grain boundary diffusion $\mathbf{j} = \mathbf{j}^s$. **b** Volume diffusion $\mathbf{j}^v$

grain boundary, the flux appears along the grain boundary between the particles. Those transport paths result in a deposition or removal of material over the corresponding interface (whether it is a free surface or a grain boundary), which gives raise to a displacement rate assumed to be normal to the interface: $\mathbf{v_n} = v_n \mathbf{n}$.

In order to find an expression for the normal velocity v_n, consider either a surface diffusion flux or a grain boundary diffusion flux $\mathbf{j}$. A matter balance is considered over the region $S \subset \partial\omega$ shown in Fig. 9a. Since the density of the material is constant, the amount of matter transported by the induced velocity $\mathbf{v_n}$ is $\mathbf{v_n}/\Omega$. Then mass balance is given by:

$$\int_{\partial S} \mathbf{j} \cdot \mathbf{n}_l dL = -\int_S \frac{1}{\Omega} \mathbf{v_n} \cdot \mathbf{n} dS$$
$$\int_{\partial S} \mathbf{j} \cdot \mathbf{n}_l dL = -\int_S \frac{v_n}{\Omega} dS \tag{27}$$

where $\mathbf{n}_l$ is the outward-pointing normal to ∂S (e.i. $\mathbf{n}_l$ is tangent to the surface S, see Fig. 9a). By using the divergence theorem, Eq. (27) becomes:

$$\int_S \left(\nabla_{\mathbf{s}} \cdot \mathbf{j} + \frac{1}{\Omega} \mathbf{v_n} \cdot \mathbf{n} \right) dS = 0$$
$$\int_S \left(\nabla_{\mathbf{s}} \cdot \mathbf{j} + \frac{v_n}{\Omega} \mathbf{n} \cdot \mathbf{n} \right) dS = 0 \tag{28}$$

where $\nabla_{\mathbf{s}} \cdot$ is the surface divergence operator. Since Eq. (28) holds for any arbitrary surface $S \subset \partial\omega$, the integrand must be equal to zero. In this way, expressions for the diffuion induced velocity for surface diffusion $\mathbf{v^s}$ and grain boundary diffusions $\mathbf{v^{gb}}$ are:

$$\mathbf{v^s} = v^s \mathbf{n} \qquad ; \qquad v^s = -\Omega \nabla_{\mathbf{s}} \cdot \mathbf{j^s} \tag{29}$$
$$\mathbf{v^{gb}} = v^{gb} \mathbf{n} \qquad ; \qquad v^{gb} = -\Omega \nabla_{\mathbf{gb}} \cdot \mathbf{j^{gb}} \tag{30}$$

with $\nabla_{\mathbf{gb}} \cdot$ is the surface divergence operator over the surface defined by the grain boundary.

In contrast with the surface and grain boundary diffusions, matter flux due to volume diffusion is not transported along the interfaces, but matter comes from the lattice (see Fig. 9b). The matter balance writes:

$$\int_S \mathbf{j^v} \cdot \mathbf{n} = \int_S \frac{1}{\Omega} \mathbf{v^v} \cdot \mathbf{n} dS \tag{31}$$

where $\mathbf{v^v}$ is the volume diffusion velocity.

Again, as Eq. (31) holds for any arbitrary surface $S \subset \partial\omega$, the normal velocity induced by volume diffusion $\mathbf{v^v}$ is given by:

$$\mathbf{v^v} = v^v \mathbf{n} \qquad ; \qquad v^v = \Omega \mathbf{j^v} \cdot \mathbf{n} \tag{32}$$

2.3 Conclusions

From a macroscopic point of view, it is very difficult to develop a model taking into account all the different variables that have an impact on the evolution of the structure during sintering. The complexity of the sintering process makes very difficult the understanding of the underlying physical phenomena and experimental approaches very often do not allow to gather enough information.

The diffusion phenomena are responsible for the microstructural evolution of the structure. The *Fick*'s laws, which relate the matter flux to the gradient of the chemical potential, can be used for the modeling of these phenomena. The different expression for the chemical potential in surface, volume and grain boundary were developed and their corresponding velocities were also found.

As a summary, regarding the three diffusion paths considered, the driving force for matter diffusion is related either to geometrical changes for surface diffusion, or to surface-tension related mechanical equilibria for volume and grain boundary diffusion. The study of the sintering at the particle scale is of great interest to enhance the understanding of the underlying physical phenomena. At this point, numerical

approaches become mandatory to cope with both geometrical and mechanical changes and discontinuities.

3 Sintering Modeling

In this section a general overview of different numerical simulations of the sintering process at the particle-size scale is presented. The first computer simulations of sintering appeared between 1955 and 1965. Those early attempts to simulate the sintering process were developed to predict the evolution of the neck and the shrinkage of two particles as a function of time. With the development of computers, more complex models of sintering appeared and the simulation of more realistic powder packing became possible.

First in Sect. 3.1, some power laws used to study the growth of the neck between two particles are presented. Even if this kind of laws is limited to the study of the sintering between only two particles, it still represents a very useful tool concerning the validation of more complex models. Furthermore, kinetics of the different diffusion mechanisms is considered to be well represented by this kind of models (at least during the first stages of sintering). In Sect. 3.2, more elaborated numerical approaches are presented into two main categories: stochastic and deterministic approaches. Finally the conclusions are discussed in Sect. 3.3

3.1 Analytical Models

As presented in the previous section, matter transport by diffusion is the underlying physical phenomenon during sintering. The main idea behind the analytical models established in the 1950s is to solve a differential equation of matter transport established through some hypotheses mainly related to the geometry of the particles and the stress distribution.

Usually those models are developed to study the diffusion phenomena between two spherical particles or between a spherical particle and a plane by just one diffusion path. Even in those simple cases, the exact quantitative description of the geometry of the contact area presents some analytical difficulties, therefore multiple geometrical hypotheses should be made. In general, the geometry of the bodies is assumed to remain unchanged and the real shape of the contact area is replaced by a geometry were the curvature of the neck is constant. The most used geometrical parameters needed to develop this kind of models are shown in Fig. 10. All those geometrical approaches are complemented with some hypotheses related to the diffusion path that is being modeled. Nevertheless, those simplifications are generally accepted [85].

Using the approximations underlying these simple approaches of sintering, expressions for the neck growth can be obtained as a function of time for the three main diffu-
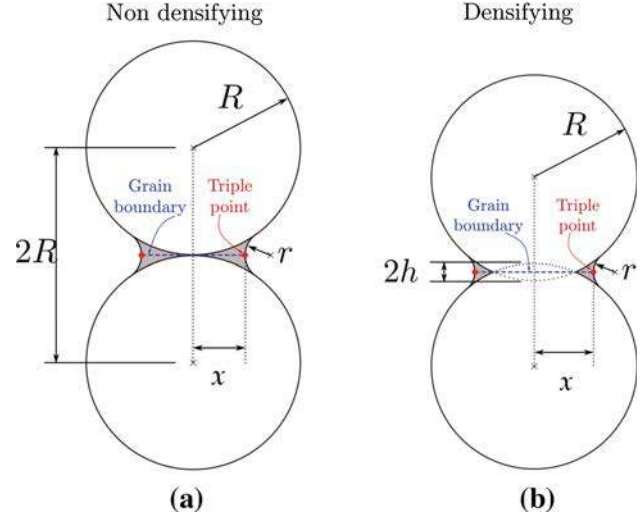
Fig. 10 Geometrical parameters of two particles. **a** Geometry for two particles without densification. **b** Geometry of two particles with densification

sion paths. According to the hypotheses made in dedicated literature, different results for the same diffusion mechanism can be found. However those models are generally of the power-law type:

$$\left(\frac{x(t)}{R}\right)^{n} = B R^{-m} t \tag{33}$$

where R is the radius of the particles, t is time and n, m and B are constants. The value of those constants depends on the hypotheses used to obtain the model. Table 2 lists the range of values for m and n that can be found in literature and a plausible set of values for the three constants presented by [28].

The previous equations are based on some strong simplifications and their validity is limited to neck radii $X < 0.3 R$, and limited to the study of the sintering of two particles. As the computational capabilities increased, the interest in performing more elaborated sintering (particle packing) simulation became an important research field. A trend toward the simulation of more realistic powders sintering is presented in Sect. 3.2.

3.2 Numerical Modeling of the Sintering Process

Concerning the numerical modeling of the sintering process at the particles scale, it is possible to distinguish two main categories which tend toward more realistic sintering simulations but mostly in 2D at the moment. On one hand, the deterministic models aim at studying the evolution of the compact powder under some specific conditions by modeling the underlying physical phenomena. On another hand, stochastic models are based on probabilistic considerations

Table 2 Constants appearing in Eq. (33) and a set of plausible values [28]

Diffusion path	Values range		Plausible set of values from [28]		
	n	m	n	m	B[a]
Surface diffusion	3–7	2–4	7	4	$\dfrac{23\gamma_{sf}\,D_s\delta_s\,\Omega}{kT}$
Volume diffusion from surface	4–5	3	4	3	$\dfrac{20D_v\gamma_{sf}\,\Omega}{kT}$
Vapor transport	3–7	2–4	3	2	$3\sqrt{\dfrac{2}{\pi}}\,\dfrac{\upsilon\gamma_{sf}\,p_g\Omega^{3/2}\rho^{1/2}}{(kT)^{3/2}}$
Grain boundary diffusion	6	4	6	4	$\dfrac{96\,\Omega\,D_{gb}\delta_{gb}\,\gamma_{sf}}{kT}$
Volume diffusion from grain boundary	4–5	3	5	3	$\dfrac{16D_v\gamma_{sf}\Omega}{kT}$
Volume diffusion from dislocations	2	1	2	1	$\dfrac{3\gamma_{sf}}{2\eta}$

[a] Where the symbols in the expression of B: D_s, D_v and D_{gb} are the diffusion coefficients for surface, volume and grain boundary, respectively. γ_{sf} is the surface tension of the solid-vapor interface, δ_s and δ_{gb} are the thickness of the surface and grain boundary diffusion layers, Ω is the molar volume, k is the Boltzmann's constant, T is the absolute temperature, υ is an accommodation constant for gas transport, p_g is the gas pressure, ρ is the specific density and η is the viscosity

yielding several different solutions out of the same initial set of conditions.

3.2.1 Stochastic Approaches

This kind of approaches are based on probabilistic considerations and the Kinetic Monte-Carlo method is the main method used. This method has been used to study the grain growth and the microstructural evolution of structures [9,18,38,43,72,83,84,90,92]. Among those works, the Potts' model [92] is used to simulate sintering of particles.

The idea is to create a grid of grain sites which contains the particles and the surrounding medium (pores). A particle is built-up of several grid sites and each site can assume any of Q distinct states. The state of a given site of a particle is given by a value of the $q : q_{\text{particle}} = [1, 2, 3, \ldots, Q]$ and a grid site corresponding to the surrounding medium is given by a value of $q : q_{\text{pore}} = -1$.

The computational domain is given by a square in 2D or by a cube in 3D. The particles are mapped on the grid sites, each particle corresponding to a single state q. Contiguous grid sites of the same state q $(q > 0)$ form a particle and contiguous empty sites $(q = -1)$ form a pore. Grain boundaries exist between neighboring particle sites of different states, q, and pore-grain interfaces exist between neighboring pore and particle sites. The total energy of the system is the sum of the surface and grain boundary contributions. For a system of N grid sites, the total energy is given by:

$$E = \frac{1}{2} \sum_{i}^{N} \sum_{j}^{N} J(q_i, q_j) w_{i,j} [1 - \delta(q_i, q_j)] \qquad (34)$$

where the term $\delta(q_i, q_j)$ is the Kronecker delta function such that $\delta(q_i, q_j) = 1$ if $q_i = q_j$ and $\delta(q_i, q_j) = 0$ if $q_i \neq q_j$.

This means that the term $[1 - \delta(q_i, q_j)]$ is only different from 0 where an interface is present (a free surface or a grain boundary). $J(q_i, q_j)$ is the energy between the states q_i and q_j. $w_{i,j}$ is a weighting term for the nearest neighbors and the next-nearest neighbors:

$$
\begin{aligned}
w_{i,j} &= w_f = 1 && i, j \text{ nearest neighbors} \\
w_{i,j} &= w_s && i, j \text{ next-nearest neighbors} \\
w_{i,j} &= 0 && \text{otherwise}
\end{aligned}
\qquad (35)
$$

in some works [9,83,84], the weigth of the next-nearest neighbors is not considered ($w_s = 0$).

The value of the states energy $J(q_i, q_j)$ is given by:

$$
J(q_i, q_j) =
\begin{cases}
J & \text{if } q_i\, q_j < 0 \\
\varrho J & \text{if } q_i\, q_j > 0
\end{cases}
\qquad (36)
$$

where J corresponds to the energy of the free surface interface and ϱJ corresponds to the energy of the grain boundary.

The effect of different phenomena is taken into account in different ways. Concerning the grain-growth, a particle site $(q > 0)$ is chosen at random from the computational domain and then a new state q is chosen from the Q possible states in the system. Before the new state q was assigned the total energy of the system (Eq. 34) was E_i, after the state q was changed, the total energy of the system can be different (E_f). The energy change is given by: $\Delta E = E_f - E_i$. Next the standard Metropolis algorithm is used to perform the grain growth step based in Boltzmann statistics. A random number between 0 and 1 is generated. The transition probability (P) is calculated using the following equation:

$$
P(\Delta E) =
\begin{cases}
\exp \frac{-\Delta E}{k_B T} & \text{if } \Delta E > 0 \\
1 & \text{if } \Delta E \leq 0
\end{cases}
\qquad (37)
$$

where k_B is the Boltzmann constant and T is the temperature. If a random number R is lower than $P(\Delta E)$, then the new state q is accepted otherwise the original state is restored.

Other physical phenomena such as the surface diffusion is considered by performing another procedure, this time an empty grid site ($q = -1$) is chosen randomly, if this site has a neighbor with a state $q \neq 0$, then the sites are temporarily exchanged. The exchange can lead to a change of the total energy of the system. Again, the Metropolis algorithm is used and the corresponding transition probability is computed by using Eq. (34)[3]. After generating a random number R between 0 and 1, if $R \leq P$ the exchange is accepted otherwise the origial states are restored.

By using similar methods, other physical phenomena are introduced into the Monte-Carlo simulation. Time (t) is usually set in terms of Monte-Carlo steps (MCS) such that 1 MSC corresponds to N attempted exchanges or changes, where N is the total number of grid sites of the computational domain.

Figure 11 shows the sintering of a set of about 20 particles of different size at different time steps. The significant feature of this result is the disappearance of some of the initially presented particles and the increase in the average size of the remaining particles.

The main drawback of this kind of approaches is related to the high number of parameters that has to be set and very often comparisons with experimental results are required. Additionally, it is very complex to modify the parameters of the model in order to take into account different mechanical behaviors or different materials.

3.2.2 Deterministic Approaches

Within the deterministic approaches, one will find the finite element methods, finite difference methods, phase field methods, etc. Very often, those methods are used to simulate the sintering of a set of particles by multiple diffusion mechanisms at the same time [19,64,88]. Even if almost all the approaches developed are supposed to be used over an arbitrary number of particles, many of them are only used within the framework of the sintering of two particles. Furthermore, most of them are limited to simulations in 2D.

These numerical methods have been used to study the underlying physical phenomena during the sintering process by many research team around the world. An early attempt to simulate the sintering process by coupled surface and grain boundary diffusion at the particle scale was presented by Bross and Exner [12] in 1979. This work, based on the model developed by Nichols and Mullins [58], uses the finite difference method. The results obtained in [12] opened the way

toward the simulation of the sintering of more complex powder compacts, and at the same time, greatly enhanced the understanding of sintering processes. However, this model was limited to 2D simulations and to rather simple geometries.

Until recently, the simulation of processes involving microstructural evolutions, such as sintering and recrystallization, of large systems was not possible because of the computational power limitations. Nowadays, the power of computers has significantly increased and with the development of parallel computing techniques, the simulation of this kind of problems is becoming possible.

As an important remark, the chemical potential associated with the volume and grain boundary diffusions is related to stress, specifically pressure p and normal stress σ_{nn}. However most of the studies concerning the sintering simulation by either of those mechanisms do not take into account the mechanical behavior of the particles. Instead, the stress is supposed to evolve following a known distribution [76,88] which is then used to compute the matter flux. To the author's knowledge, there exists only one study where the matter flux is computed by explicitly taking into account the mechanical behavior of the material [26,27].

Some important works concerning the sintering simulations will be presented in the following Sections.

Pan & Cocks model [64] In that work, the surface and the grain boundary diffusions are coupled to study the microstructural evolution involving these two diffusion paths. A rigorous treatment of the continuity conditions at the junction between the surface and the grain boundary is presented. This method has been successfully applied to simulate the microstructural evolution in 2D systems.

The computational domain is shown in Fig. 12. Grain boundaries are assumed to be formed by straight lines and the grains are supposed to be rigid. The matter flux is modeled by using the Fick's law presented in Sect. 2 (Eqs. 22 and 26 for surface and grain boundary diffusions, respectively).

The mass conservation at a triple point formed by two free surfaces and a grain boundary (as in Fig. 13) is written as a function of the grain boundary matter flux $\mathbf{j_{gb}}$ and matter fluxes at the free surfaces $\mathbf{j_s}$:

$$\mathbf{j_{gb}} = -(\mathbf{j_s^+}) + (\mathbf{j_s^-}) \tag{38}$$

where $+$ and $-$ indicate the two free surfaces at the triple point and the sign convention for these fluxes is defined in Fig. 13.

If the triple point is formed by n grain boundaries, the mass conservation imposes:
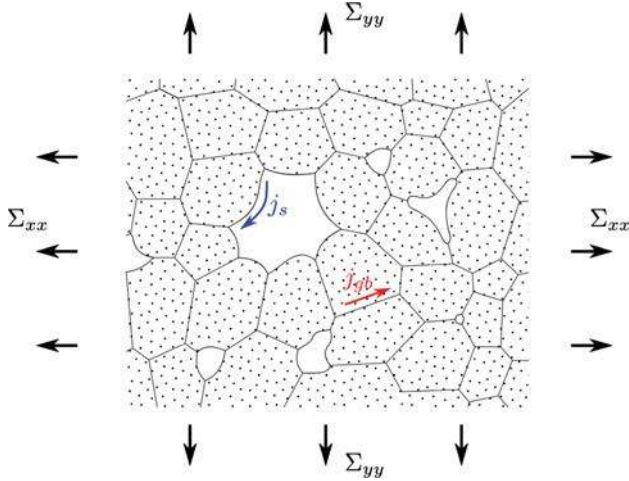
$$\sum_{i=1}^{n} \mathbf{j_{gb,i}} \cdot \mathbf{t_i} = 0 \tag{39}$$

---

[3] The Boltzmann constant (k_B) present in this equation can be different for every physical phenomenon, therefore it is a parameter that has to be set.
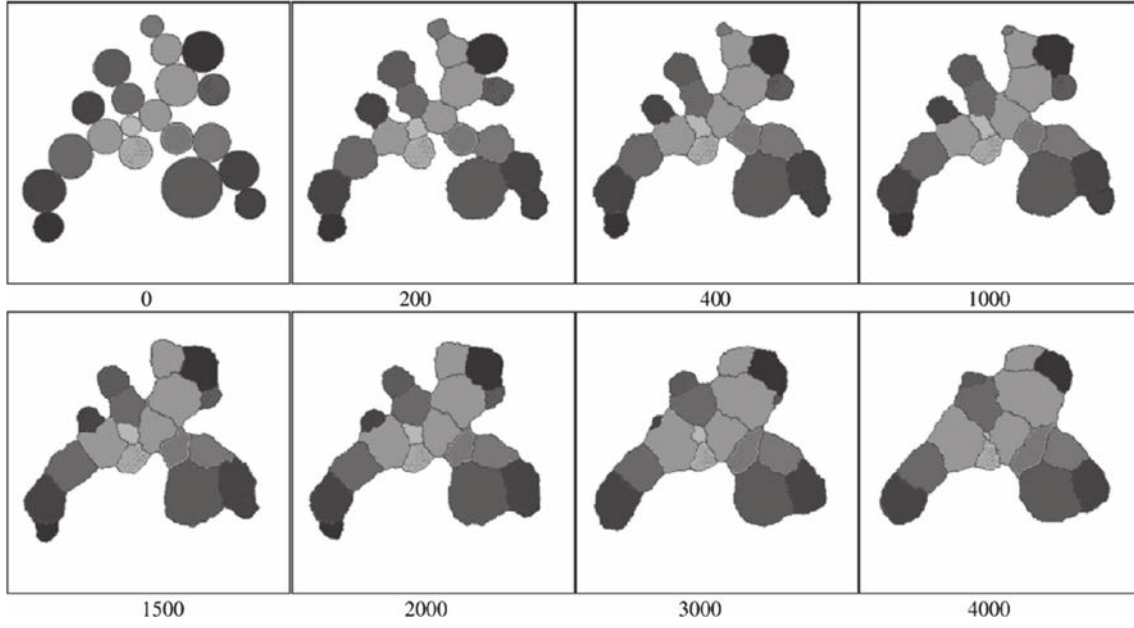
Fig. 11 Sintering of a set of particles at different time steps. Using: $k_B T/J = 0.5$, $K_B T/\varrho J = 1$ and $w_s = 0.5$ [72]


Fig. 12 Polycrystalline material with pores [64]

where t_i is a unit vector pointing away from the i*th* grain boundary.

Concerning the momentum conservation, another condition can be written. The surface tension γ_{sf} and the grain boundary tension γ_{ss} lead to a discontinuity of the tangent at the triple point which is referred to as the dihedral angle 2θ (Fig. 13):

$$\cos \theta = \frac{\gamma_{ss}}{2\gamma_{sf}} \tag{40}$$

Finally, the chemical potential should be continuous and therefore a relationship between the normal stress σ_{nn} and
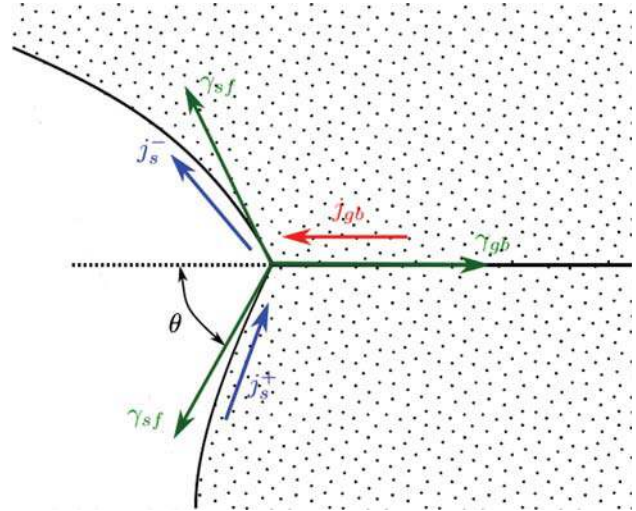
Fig. 13 A triple point: junction between a grain boundary and a free surface [64]

the "curvature" at the triple point κ^{tp} is established:

$$\sigma_{nn}^{tp} = \gamma_{sf} \kappa^{tp} \tag{41}$$

It is important to say that the curvature κ is undefined at the triple point. Therefore, it is considered as an additional unknown of the problem.

Grain boundary matter diffusion is discretized using finite elements while surface diffusion flux is computed by using a finite differences approach. In order to compute the matter flux by grain boundary diffusion, consider the following functional Π:

$$\Pi = \Pi_0 - \sum_{tp}\left(\lambda\sum_{i=1}^{n}\mathbf{t^i}\cdot\mathbf{j_{gb}^i}\right)$$

$$\Rightarrow\quad \Pi = \frac{1}{2}\int_{\Gamma_{gb}}\mathbf{j_{gb}}\cdot\mathbf{j_{gb}}\,dA - \int_{S_T}\mathbf{T}\cdot\mathbf{V}\,dS$$

$$+\sum_{tp}\sigma_{tp}\,\mathbf{t}\cdot\mathbf{j_{gb}}$$

$$+\sum_{tp}\gamma_{sf}\parallel\mathbf{v_{gb}}\parallel\sin\theta - \sum_{tp}\left(\lambda\sum_{i=1}^{n}\mathbf{t^i}\cdot\mathbf{j_{gb}^i}\right) \qquad (42)$$

where $\sum_{tp}$ is a summation performed over all the triple points, $\sum_{i=0}^{n}$ corresponds to the n grain boundaries that meet at a given triple point, $\mathbf{t}$ is the unit vector along the grain boundary toward the triple point, S_T is the surface where the traction $\mathbf{T}$ is applied, $\mathbf{V}$ is the velocity of S_T, θ is the dihedral angle and λ represents a set of Lagrange multipliers introduced to ensure the mass conservation at the triple points. Among all the kinematically admissible fields $\mathbf{j_{gb}}$ the true field makes the functional Π_0 minimum [64,65].

Concerning the surface diffusion, since all the simulations performed were in 2D, the free surface is discretized by using straight segments. From the coordinates of the nodes of the surface and from Eq. (41), the curvature on each node is computed. Next the matter flux and the associated velocity is computed in the same way.

The coupling between the grain boundary and the surface diffusion is established by using the mass conservation at the triple points (Eq. 38). Figure 14 shows the evolution of two particles during sintering.

Several works using this approach have been presented by the team of Pan [20,21,52,63,65,66]. In [65], a fully finite element formulation is used to solve the variational principle presented in Eq. (42). All the simulations are performed in 2D, therefore the computational domain is discretized by using straight line elements. Concerning the surface diffusion and in contrast with the finite differences approach used in [64], a finite element approach is used to compute the surface flux and the induced velocity. In this case, special elements are used to establish the junction between the free surfaces and the grain boundary.

An enhancement to the coupled finite element formulation presented in [65] is developed in [20]. The main idea is to represent the structure by using classical cubic spline elements. One of the advantages of this method is related to the smoothness of the interface that is enforced in such a way that second order derivatives are continuous at any point of the interface. This smoothness allows to reduce the high frequency oscillations of the interface during their migration and focuses the numerical solution on the global evolution of the microstructure.

Phase field model [89] When using finite element methods or finite difference methods within a Lagrangian continuum
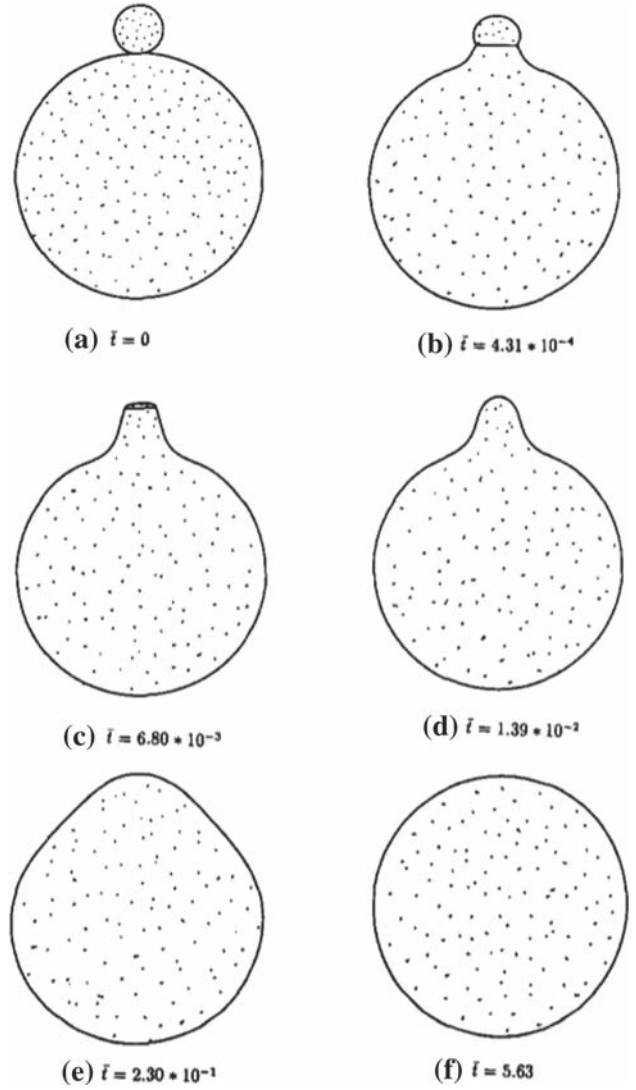

(a) $\bar{t} = 0$

(b) $\bar{t} \approx 4.31 * 10^{-4}$

(c) $\bar{t} \approx 6.80 * 10^{-3}$

(d) $\bar{t} \approx 1.39 * 10^{-2}$

(e) $\bar{t} \approx 2.30 * 10^{-1}$

(f) $\bar{t} \approx 5.63$

Fig. 14 Sintering of two cylinders of different radii [64]

mechanics framework, the free surfaces of the system as well as the grain boundaries are used to apply boundary conditions to the problem. This kind of approaches are very useful when dealing with 2D problems and very often are used to validate the hypotheses made. Nevertheless, it is very complicated to enhance that kind of models in order to deal with 3D problems over complex geometries.

Alternative methods allowing to describe microstructural evolutions over complex geometries have been developed. The phase field methods is among those alternative methods and has been used by Wang in [89] within the context of sintering simulation.

The phase field model uses several field functions (the so-called phase fields) which correspond to well-defined physical parameters such as the composition. In the case of the sintering simulation, the field functions take specific values in each particle and change smoothly but rapidly across the

13
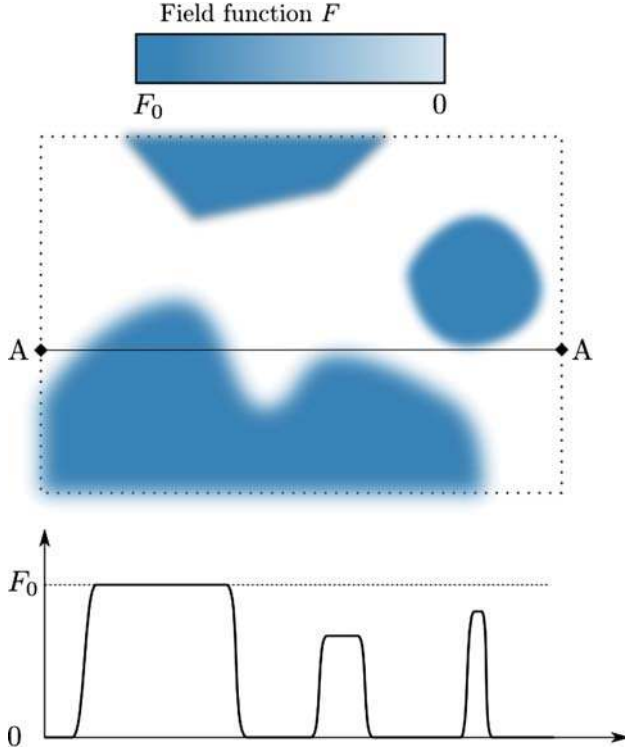
Fig. 15 Diffuse interface of the field function F: on *top* a field function F is plotted over a square domain and at bottom the value of the field function across the line *A–A* is shown

interfaces (the so-called diffuse interfaces) [19,89]. Figure 15 shows a field function F, equal to F_0 inside the particles and equal to 0 outside the particles. The bottom of Fig. 15 shows the evolution of the field function along the horizontal line A–A shown in the top. As it can be seen, there is a smooth transition of F across the interfaces.

The total free energy of the system is a functional of the field functions and the microstructural evolution is driven by the reduction of this total free energy. In [89], a parameter field η_i is defined to describe each particle, η_i is equal to 1 inside the ith particle and equal to 0 outside. An additional mass density field ρ is defined, this field allows to identify the particles from the surrounding media.

The change in the structure is given by the mass conservation equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{43}$$

where $\mathbf{v}$ is the local instantaneous velocity. Let define a mass flux density $\mathbf{j} = \rho \mathbf{v}$. The flux can be written as a sum of the contribution of two distinct processes:

$$\mathbf{j} = \mathbf{j}_{dif} + \mathbf{j}_{adv} \tag{44}$$

where $\mathbf{j}_{dif}$ and $\mathbf{j}_{adv}$ are the diffusion and advection[4] flux densities, respectively. The advection flux is supposed to be induced by a rigid body motion characterized by a translation and a rotation of each particle, computed in such a way that the mass is conserved. All the details of its computations are presented in [89].

The diffusion flux takes into account all the diffusion paths at the same time and is computed as a function of a global chemical potential:

$$\mathbf{j}_{dif} = -D_{gl}\nabla\boldsymbol{\mu_{gl}} = -D_{gl}\nabla\frac{\delta F}{\delta \rho} \tag{45}$$

where D_{gl} is the diffusion coefficient function of the phase fields ρ and η_i because its value depends on the region of the solid being considered (surface, grain boundary, volume, etc.). F is the total free energy of the system also computed by using the field functions ρ and η_i as follows:

$$F = \int_\omega (f(\rho, \{\eta_i\}) \dots$$
$$\dots + \frac{1}{2}\beta_\rho \parallel \nabla\boldsymbol{\rho} \parallel^2 + \sum_i \frac{1}{2}\beta_\eta \parallel \nabla\eta_\mathbf{i} \parallel^2 \Bigg) dV \tag{46}$$

where β_ρ and β_η are constant coefficients of the model, $f(\rho, \{\eta_i\})$ corresponds to the energy associated with the volume and the last two terms correspond to the energy at the free surface and the grain boundaries, respectively.

Equations (43)–(46) are solved by using a finite difference scheme. Figure 16 shows the evolution of a compact powder during sintering.

3.3 Conclusions

Sintering is a very complex process and several challenges should be handled in order to simulate the sintering process at the particle scale. However, important developments have been proposed and today it is possible to think about the simulation of the sintering of a packing of particles in 3D.

The analytical models presented at the beginning of this section are limited to very simple cases. Nevertheless, they represent the most useful way to validate the results of more complex numerical approaches because, it is still very tricky to use experimental setups to validate those results. Furthermore, analytical models allow to study the kinetics of the process that is mainly based on the driving force selected for a given sintering mechanism.

More complex approaches allowing to simulate the sintering of more complex particle packs are usually limited to 2D and even if the numerical methods should work in 3D,

---

[4] This kind of flux path is considered in order to ensure the mass conservation; if only the diffusion paths were considered, the mass could, in some cases, not be conserved [88,89].
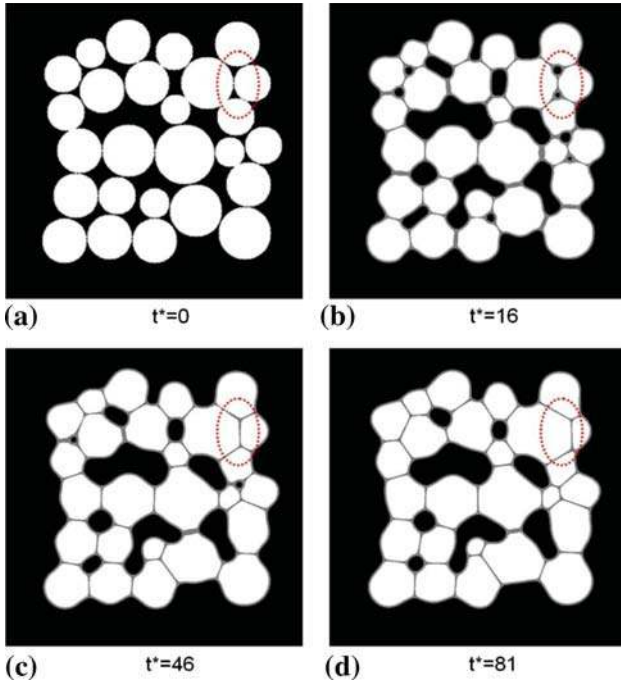
Fig. 16 Simulated microstructure evolution in a powder compact during sintering. **a** initial green compact. **b–d** Typical snapshots of the simulated sintering process. The *highlighted area* shows the removal of two pores and the subsequent grain boundary migration [89]

none of them are used to perform simulations of sintering over realistic sets of particles in 3D.

As it has been presented in Sect. 2, the chemical potential in volume and grain boundary is computed as a function of the state of stress state of the particles. However, the mechanical response of the material to the external loading and the surface tension acting on free surface is usually neglected. A numerical approach taking into account the mechanical behavior of the particles and allowing to perform simulations in 3D has to be developed.

4 Numerical Strategy

The numerical simulation of processes involving strong topological changes is a major field of research, and different approaches have been developed to cope with these processes. Sintering, as it was presented in the previous sections, is one of these processes and considering its simulation at the particle scale remains a challenging task. Considering this, a classification of the numerical methods can be made depending on the nature of the computational grid for spatial discretization: (1) deformable grids and (2) fixed grids, where an additional strategy is needed to describe the internal change in the structure.

The use of deformable grids leads to an explicit description of the compact powder, and therefore the boundary con-

ditions at the surface of the particles can be represented in a more accurate way. In counterpart, when representing the strong topological changes that can appear during the process, large deformations of the grid can occur. To deal with those large deformations, complete re-meshing is required which is very complex from the computational point of view and is usually quite expensive, especially for 3D problems. For those reasons, the evolution of the structure will be preferably handled by using a fixed grid.

The use of a fixed grid usually requires a surrounding medium if the first phase is not physically bounded. In this way, the interface between the two phases (the compact powder and the surrounding medium which in this case is the air) should be described in a separate way. According to the method used to describe the interfaces, the numerical approaches can be divided into two different categories. The "Front tracking" methods and the "Front capturing" methods. Front tracking methods are based on a Lagrangian description of the interface where some markers are used to locate and follow the interface over the time. On the other hand, in front capturing methods the interface is implicitly represented by a phase function discretized on the fixed grid. This phase function allows to identify to which phase a given point belongs and the interface is defined by using the phase function.

In this section, some front tracking and front capturing methods are presented in Sects. 4.1 and 4.2, respectively. The choice of the LS method is also supported in Sect. 4.2. All the generalities of the LS method are discussed in Sect. 4.3. The LS approach used in this work is slightly different from the classical LS method. The modified LS approach as well as a mesh adaptation strategy backed on this modified LS approach are presented in Sect. 4.4. Finally the conclusions of the section are discussed in Sect. 4.5.

4.1 Front Tracking Methods

Front tracking methods are based on the use of markers which make a Lagrangian description of the interface. The main advantage of this kind of approaches is the straightforward interface definition. A high degree of accuracy can be achieved by extracting the interface geometry with high-order polynomial interpolations [23,71]. Nevertheless, as for deformable grids, front tracking methods require regular redistribution of the markers on the interface to ensure a proper representation of the moving front.

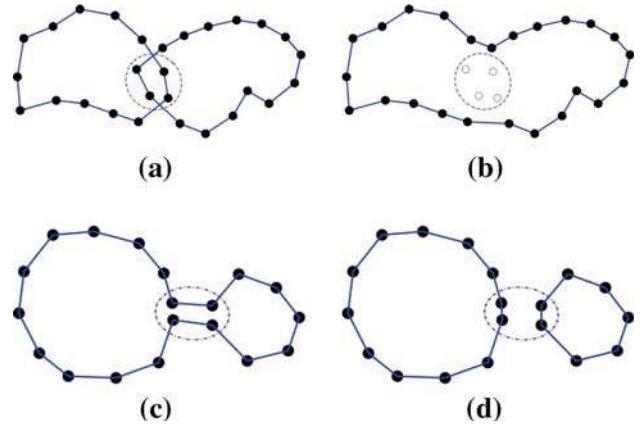The earliest numerical technique developed to deal with problems involving two phases was the well-known marker and cell (MAC) "volume tracking" method and was initially used within the framework of free surface flow [37]. This technique uses marker particles that are convected with the local fluid velocity and the markers distribution allows the current fluid configuration to be known. Figure 17a shows a
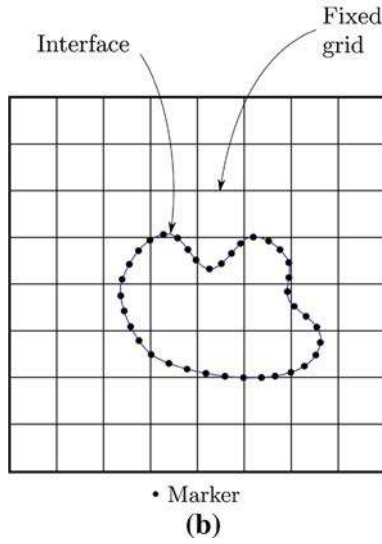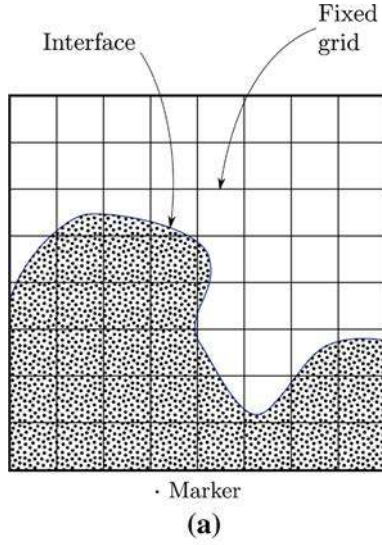
Fig. 17 Front tracking methods in a fix grid. **a** MAC technique [37]. **b** Front markers method [77]

Fig. 18 Surface markers coalescence (**a**, **b**) and detachment (**c**, **d**)

set of markers distributed over a fixed grid where each black point is a marker.

Although the MAC method allows to represent an arbitrary region over a fixed grid, heavy computations are induced to properly use it. First to enforce boundary conditions on a "staggered" interface [77], and second to redistribute the markers which will localise in highly sheared regions. To avoid this latter phenomenon, Daly and Pracht in 1968 [24] proposed to use markers only over the interface instead of having them all over the concerned region, a so-called "Surface tracking" method. The motion of the interface is simply computed by moving the marker according to the local velocity interpolated from the fixed grid velocities. The interface is explicitly defined by interpolating a curve (2D) or a surface (3D) across the markers (Fig. 17b), but there still remain

some issues of markers concentrations which must be redistributed.

More generally, the fixed grids approaches are supposed to handle topological changes with ease. However, in this case the coalescence or detachment of a section of the surface can be very difficult to represent as illustrated in Fig. 18. Figure 18a shows two different interfaces that should be merged, markers then should be removed (dashed gray ellipse) to obtain a single interface as in Fig. 18b. In the same way, Fig. 18c shows an interface that should be split into two separated interfaces (Fig. 18d), inducing a strong modification of the connectivity between the markers to obtain the detachment of a part of the interface. Additionally, those operations become a burden for 3D problems.

4.2 Front Capturing Methods

In this category of methods the interface is implicitly described within a fully Eulerian approach. Here an additional phase function is required and the motion of the interface is studied by solving the convection problem under a given velocity field (this will be further discussed in the next Sect. 4.3). The main advantage of these approaches is that all the topological changes are taken into account naturally by the numerical technique. However, the computation of integrals over the interface is usually more complex compared with the front tracking methods, because the interface is neither defined by markers nor by the computational grid. The most common numerical methods of front capturing are the volume-of-fluid (VOF) and the LS methods.

4.2.1 Volume-of-Fluid Method

This method, introduced in the early 1980s by Hirt and Nichols [40] relies on the main idea that a fractional volume, or "color" function C, will be used to indicate the fraction of a mesh cell that is filled with a particular phase. In particular,
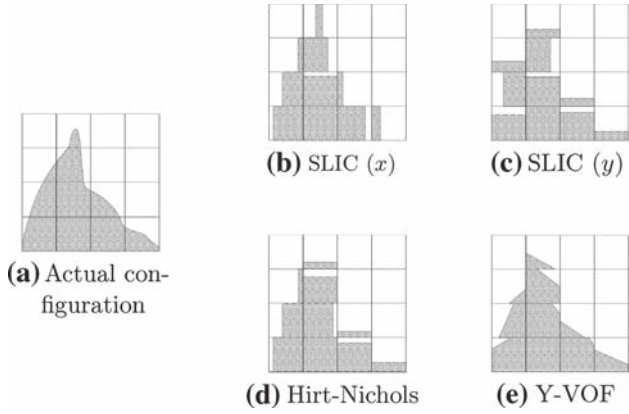
Fig. 19 Schematic representation of interface reconstructions of the actual phase configuration shown in **a**; **b**, **c** SLIC (x- and y-sweep respectively); **d** Hirt-Nichols' VOF; **e** Y-VOF method [77]

VOF methods have been developed to solve the advection Eq. (47) in such a way that interfaces remain sharp [77]:

$$\frac{\partial C}{\partial t} + \nabla \cdot (\mathbf{v}C) = 0 \tag{47}$$

Geometry of the interface can be reconstructed from the values of the color function C and there are many schemes allowing to perform this interface reconstruction. Several of these reconstruction schemes have been reviewed by Rider and Kothe in [75]. The most basic schemes are : the "Simple Line Interface Calculation (SLIC)" presented by Noh and Woodward [60] where interface is reconstructed using straight lines in 2D or planes in 3D aligned with one of the coordinate directions, the "Hirt-Nichols" method presented in [40] where neighbors are considered during reconstruction, and the "Y-VOF" method presented by Youngs in [93] where a local direction is computed which corresponds to a minimum gradient of the color function $\nabla \mathbf{C}$. Figure 19 shows a schematic representation of different interface reconstructions of the actual geometry shown in Fig. 19a.

The main advantages of VOF approaches are related to the simplicity of the method and the volume conservation capability. Considering their application to the sintering simulation at the particles scale, the main drawback concerns the description of the interface which is not reconstructed in a very accurate way. Also the computation of the curvature of the interface and high order derivatives can be difficult and, very often, introduces numerical noise as the color function is constant within every cell.

4.2.2 Level-Set Method

The LS method was first introduced by Osher and Sethian in 1988 [62]. Initially the method was presented to study fronts propagating with curvature-dependent velocity. Since its introduction it has been used in wide range of applica-

tions such as multiphase flow, Stefan problems, kinetic crystal growth, etc [61]. The main idea behind the method is to represent the interface Γ as the zero iso-value of a smooth function $\alpha(\mathbf{x}, t)$:

$$\Gamma(t) = \{\mathbf{x} \in \mathbb{R}^n, \alpha(\mathbf{x}, t) = 0\} \tag{48}$$

α is usually computed as a signed distance function to the interface Γ (positive from one side of the interface and negative from the other one). One of the main advantages of the method is related to the ease of computation of geometrical quantities such as the curvature and the normals. Additionally, it has been shown that the approach allows to deal easily with problems in 3D and its implementation is simple [61,68]. The aim of the LS method is to represent the motion of an interface Γ under a velocity field $\mathbf{v}$ which can depend on position, time, geometry of the interface, and external physical laws [61]. Its main drawback is related to the volume conservation which can not be ensured just by transporting α (see Sect. 4.3 for further details).

4.2.3 Choice of the Method

As it has been presented previously in Sects. 2 and 3, during the simulation of the sintering process at the particle scale strong topological changes must be handled. Therefore fixed grid methods are good candidates since they are more likely to handle properly this kind of structural evolution.

Matter diffusion play the main role in the structural changes of the system. The direct simulation of these phenomena require a very accurate interface representation as the result highly depends on geometrical quantities such as the normal and, even more importantly, the curvature for the computation of the matter flux by surface, volume and grain boundary diffusion. Additionally, one of the goals of this work is to carry out simulations in 3D, hence the numerical method must allow to perform simulations either in 2D or in 3D with equal ease.

VOF approaches do not allow to have a precise description of the interface, moreover, it is difficult to have a good estimation of the curvature and the normals and 3D simulations seem to be carry out. On the other hand, the LS method allows to have an accurate description of the interface and both curvature and normal can be directly computed with this approach. For these reasons, the LS method represents a better option. A general introduction to this method is presented in the next Sect. 4.3.

4.3 Classical Level-Set Method

The classical LS method will be discussed in this section. The LS function α will be first presented in Sect. 4.3.1, its exploitation (Sect. 4.3.2) and motion description (Sect. 4.3.3) will be discussed then. The necessity of the reinitialization

step as well as the equations used to perform this operation will be presented in Sect. 4.3.4.

4.3.1 Level Set Function

Let Ω be a bounded region of $\mathbb{R}^n$ (the computational domain, and et Γ be the boundary of a subdomain $\Omega_s \subset \Omega$ (the set of grains) which can deform along the time. At every time t, the description of Ω_s and its boundary Γ is achieved through a function $\alpha : \mathbb{R}^n \times \mathbb{R}^+ \rightarrow \mathbb{R}$ which has the following properties:

$$
\alpha(\mathbf{x}, t) \begin{cases} < 0 & \text{if } \mathbf{x} \in \Omega_s \\ > 0 & \text{if } \mathbf{x} \notin \Omega_s \\ = 0 & \text{if } \mathbf{x} \in \Gamma \end{cases} \tag{49}
$$

with $\mathbf{x}$ the point in R^n where α is being evaluated and t is time.

In the classical LS method, the LS function α is a smooth function given by Eq. (50).

$$
\alpha(\mathbf{x}, t) = \begin{cases} -\text{dist}(\mathbf{x}, \Gamma) & \text{if } \mathbf{x} \in \Omega_s \\ \text{dist}(\mathbf{x}, \Gamma) & \text{if } \mathbf{x} \notin \Omega_s \\ 0 & \text{if } \mathbf{x} \in \Gamma \end{cases} \tag{50}
$$

where $\text{dist}(\mathbf{x}, \Gamma)$ is the Euclidean distance from a point $\mathbf{x}$ to the interface Γ. Contrary to the color function used with a VOF-like method, the LS function α is smooth: at least continuous and with $\| \nabla \alpha \| = 1$ where this gradient exists. These properties allow the use of continuous finite elements for solving the transport equation (see Sect. 4.3.3), which represents an advantage in terms of numerical developments.

Figure 20 shows the LS function corresponding to a circle of radius $R = 0.3$ centered in $x = 0.5$ and $y = 0.5$. The Z axis represents the value of α.

4.3.2 Level-Set Features

One of the main advantages of this method is its capability of computing some geometrical quantities such as the curvature κ and the normal $\mathbf{n}$. It is also possible to compute with ease other functions that will be very useful regarding the sintering simulation. The outward normal $\mathbf{n}$ and curvature κ can be computed by using Eqs. (51) and (52), respectively.

$$
\mathbf{n}_\alpha = \frac{\nabla \alpha}{\| \nabla \alpha \|} \tag{51}
$$

$$
\kappa_\alpha = \nabla \cdot \mathbf{n} \tag{52}
$$

As presented previously, when a front capturing method (e.g. the level method) is used, a second phase is introduced into the problem modeling. In the case of the sintering simulation, the computational domain Ω will be composed of two different phases: the compact powder and the surrounding
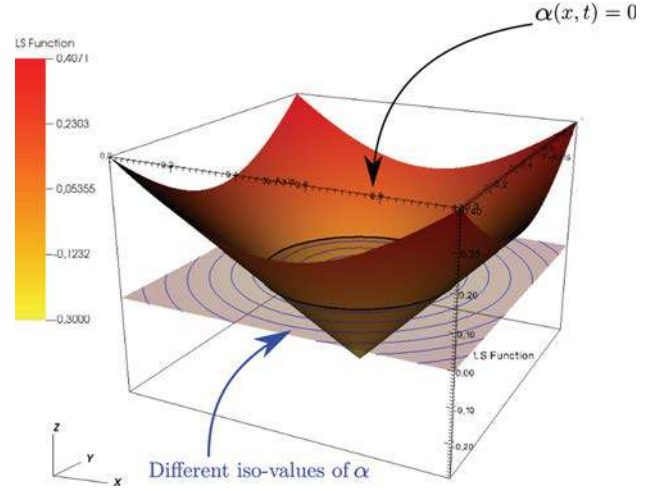
Fig. 20 LS function α of a circle. The *black line* represents the zero iso-value of α

medium. It is also possible to compute the Heaviside functions[5] H corresponding to each phase (H^s for the compact powder and H^f for the surrounding medium):

$$
H^s(\alpha) = \begin{cases} 1 & \text{if } \alpha \leq 0 \\ 0 & \text{if } \alpha > 0 \end{cases} \quad ; \quad H^f = 1 - H^s \tag{53}
$$

Those Heaviside functions H are used to compute volume (3D problems) or surface (2D problems) integrals over just a region of the computational domain Ω. For example the integral of function $f(\mathbf{x}, t)$ over the solid phase (the compact powder) can be computed as follows:

$$
\int_\Omega f(\mathbf{x}, t) H^s(\alpha) dV \tag{54}
$$

note here, that the computational domain Ω contains both the compact powder and the surrounding medium, but by introducing the Heaviside function H^s into the integral, the above integral corresponds to the integral of a function $p(\mathbf{x}, t)$ over the compact powder only.

4.3.3 Convection

As stated previously, the goal of the LS method is to represent the motion of an interface Γ under a velocity field $\mathbf{v}$. The motion of the interface Γ (defined by the zero isovalue of α) is given by the result of the convection equation:

---

[5] Also known as characteristic functions.

18

$$\frac{\partial \alpha}{\partial t} + \mathbf{v} \cdot \boldsymbol{\nabla}\boldsymbol{\alpha} = 0 \tag{55a}$$

$$\alpha(\mathbf{x}, t = 0) = \alpha_0(\mathbf{x}) \tag{55b}$$

$$\alpha(\mathbf{x}, t) = g^{\text{inflow}} \qquad \text{if } \mathbf{x} \in \partial\Omega^- \tag{55c}$$

where Eq. (55) corresponds to the inflow boundary. This equation sets the value of the LS function α over the inflow boundary $\partial\Omega^-$ to be equal to g^{inflow}. The inflow boundary is defined as: $\partial\Omega^- = \{\mathbf{x} \in \partial\Omega, \mathbf{v}(\mathbf{x}) \cdot \mathbf{n} < 0\}$.

The solution of Eq. (55) does not ensure that the norm of the gradient of α remains equal to one: $\| \boldsymbol{\nabla}\boldsymbol{\alpha} \| = 1$ [61, 62,68]. In fact, according to the velocity $\mathbf{v}$ the LS function α can become very flat or very steep at the interface Γ. A procedure, usually called *reinitialization*, is used to reset the LS function α to be a signed distance function to Γ, this procedure will be presented in the next section.

4.3.4 Reinitialization

The reinitialization procedure can be described simply as the process of replacing the function $\alpha(\mathbf{x}, t)$ by another function $\tilde{\alpha}(\mathbf{x}, t)$ that has the same zero iso-value but behaves better. Then the new function $\tilde{\alpha}(\mathbf{x}, t)$ is used until the next round of reinitialization.

A way to find this new function $\tilde{\alpha}(\mathbf{x}, t)$ is to find the location of the interface Γ with some interpolation technique and then compute a signed distance function out from the interpolation, as presented in [56]. Some drawbacks of this approach are related to the computational cost and the noise that is introduced during the reinitialization that can have an important impact on some geometrical quantities such as the curvature [68]. An alternative strategy has been presented by Sussex et al. in [82]. A Hamilton–Jacobi equation is implemented to reconstruct iteratively the LS function from the zero iso-value of $\alpha(\mathbf{x}, t)$. A virtual time τ is introduced ($\tau < \tau^*$):

$$\frac{\partial \tilde{\alpha}}{\partial \tau} + \text{sgn}(\tilde{\alpha}_0)(\| \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} \| - 1) = 0 \tag{56a}$$

$$\tilde{\alpha}(\mathbf{x}, 0) = \tilde{\alpha}_0(\mathbf{x}) = \alpha(\mathbf{x}, t) \tag{56b}$$

where the signed function $\text{sgn}(\alpha)$ is defined below, but usually approached by $\hat{\text{sgn}}(\alpha)$:

$$\text{sgn}(\alpha) = \begin{cases} 1 & \text{if } \alpha > 0 \\ -1 & \text{if } \alpha < 0 \\ 0 & \text{if } \alpha = 0 \end{cases} \quad ; \quad \hat{\text{sgn}}(\alpha) = \frac{\alpha}{\sqrt{\alpha^2 + \epsilon^2}} \tag{57}$$

where ϵ is a parameter related to the spatial discretization size.

For practical purposes Eq. (56) can be rewritten as a convection equation and can be solved by using the same numerical method used to solve Eq. (55) (i.e. using the SUPG finite

element technique described in Sect. 4.4.3):

$$\frac{\partial \tilde{\alpha}}{\partial \tau} + \mathbf{v}_r \cdot \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} = \hat{\text{sgn}}(\tilde{\alpha}_0) \tag{58a}$$

$$\tilde{\alpha}_0(\mathbf{x}) = \tilde{\alpha}(\mathbf{x}, 0) = \alpha(\mathbf{x}, t) \tag{58b}$$

since $\boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} \cdot \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} = \| \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} \|^2$, the reinitialization velocity $\mathbf{v}_r$ is given by:

$$\mathbf{v}_r = \hat{\text{sgn}}(\tilde{\alpha}_0) \frac{\boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}}}{\| \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} \|} \tag{59}$$

In contrast with the transport Eq. (55), there is no inflow boundary condition. When using the LS method, the computational domain is usually given by a cube (3D) or a square (2D) which contains the interface that is being tracked. In this case, the reinitialization velocity is always pointing out of the computational domain, and therefore the inflow boundary is empty: $\partial\Omega^- = \emptyset$.

To summarize, the standard LS method resolution scheme is presented in Algorithm 1.

Algorithm 1 Classical LS method

$t \leftarrow 0$
$\alpha_0(\mathbf{x}, t = 0) \leftarrow$ Eq. (50) from a given geometry Ω_{s0} at $t = 0$
while $t < t_f$ **do**
    $\mathbf{v} \leftarrow$ from physical phenomena, e.g. Eqs. (29), (30) and (32)
    $\alpha(\mathbf{x}, t + \Delta t) \leftarrow$ from Eq. (55)
    $\tilde{\alpha}(\mathbf{x}, \tau) \leftarrow$ from Eq. (58) with $\tilde{\alpha}_0(\mathbf{x}, \tau = 0) = \alpha(\mathbf{x}, t + \Delta t)$
    $\alpha(\mathbf{x}, t + \Delta t) \leftarrow \tilde{\alpha}(\mathbf{x}, \tau^*)$
    $t \leftarrow t + \Delta t$
end while

Equation (58) should be solved iteratively until the steady state is reached (corresponding to the time τ^*). When the steady state is reached $\partial\tilde{\alpha}/\partial\tau = 0$ and therefore $\| \boldsymbol{\nabla}\tilde{\boldsymbol{\alpha}} \| = 1$. The terms involving $\hat{\text{sgn}}(\tilde{\alpha}_0)$ are introduced to ensure the zero isovalue of $\tilde{\alpha}$ to be exactly $\alpha(\mathbf{x}, t)$.

An important point should be highlighted concerning the values of the physical velocity $\mathbf{v}$: this velocity is only required in the vicinity of the interface, to transport the zero-isovalue of the LS function. Far from this interface, the LS function is mainly set by the reinitialization velocity. In fact, the reinitialization procedure is an additional problem that should be solved which leads to an increase of the computational cost. For this reason an alternative procedure allowing to perform both convection and reinitialization in a single step will be presented in the next section.

4.4 Local Level-Set Approach and Mesh Adaptation Strategy

One of the main drawbacks of the LS method is related to its computational cost. By embedding the interface as the zero iso-value of a higher dimensional function, a one dimensional interface problem is transformed into a two dimen-

sional problem. In three dimensions, considerable computational labor is required per time step [1]. Additionally, the reinitialization procedure is computationally expensive. For those reasons, in this sections some numerical approaches aiming to reduce the computation time are presented.

4.4.1 Convective Reinitialization

The main idea is to couple the reinitialization procedure with the convection step [22]. Let define a parameter λ relating the virtual time τ and the real time t such that:

$$\lambda = \frac{\partial \tau}{\partial t} \tag{60}$$

Additionally, it is possible to write:

$$\frac{\partial \tilde{\alpha}}{\partial t} = \frac{\partial \tilde{\alpha}}{\partial \tau} \frac{\partial \tau}{\partial t} = \lambda \frac{\partial \tilde{\alpha}}{\partial \tau} \tag{61}$$

By replacing Eq. (61) into the reinitialization Eq. (58), the following expression can be found:

$$\frac{\partial \tilde{\alpha}}{\partial t} + \lambda \, \mathbf{v}_r \cdot \nabla \tilde{\alpha} = \lambda \, \hat{\mathrm{sgn}} \, (\tilde{\alpha}_0) \tag{62}$$

The previous Eq. (62) corresponds to the reinitialization step over the real time. Now if $\tilde{\alpha}$ is considered to evolve under the physical velocity $\mathbf{v}$ (Eq. 55) and the initial condition of the reinitialization step establishes that $\tilde{\alpha}(\mathbf{x}, \tau = 0) = \alpha(x, t)$, then Eq. (62) can be rewritten as the convection-reinitialization equation for α:

$$\frac{\partial \alpha}{\partial t} + \mathbf{v} \cdot \nabla \alpha + \lambda \, \mathbf{v}_r \cdot \nabla \alpha = \lambda \, \hat{\mathrm{sgn}} \, (\alpha) \tag{63}$$

The term $\mathbf{v} \cdot \nabla \alpha$ comes from the fact that α lives within an Eulerian frame.

Now, consider a time marching scheme of physical time step Δt associated with the virtual time step $\Delta \tau$ evaluated as $\| \mathbf{v}_r \| \Delta \tau \approx h$, with h being a typical characteristic of the discretization. Since the gradient of the reinitialization velocity $\mathbf{v}_r$ (Eq. 59) is close to one ($\| \nabla \mathbf{v}_r \| \approx 1$), the parameter λ will be chosen equal to $h/\Delta t$. Finally, the convected reinitialization equation can be written as follows:

$$\frac{\partial \alpha}{\partial t} + (\mathbf{v} + \lambda \, \mathbf{v}_r) \cdot \nabla \alpha = \lambda \, \hat{\mathrm{sgn}} \, (\alpha) \tag{64a}$$

$$\alpha(\mathbf{x}, t = 0) = \alpha_0(\mathbf{x}) \tag{64b}$$

As stated previously, in practice the physical velocity $\mathbf{v}$ is only necessary over a region close to the interface (the zero isovalue of the LS function α) since it is responsible for the motion of the interface. Away from the interface, the physical velocity $\mathbf{v}$ is no longer required because the value of the LS function is controlled by the reinitialization velocity $\mathbf{v}_r$ which also ensures that the gradient of α remains equal to one: $\| \nabla \alpha \| = 1$.

4.4.2 Local Level Set Function

One of the drawbacks of the LS method stems from the required computational effort [1]. Considering that all the geometrical useful information (the interface itself, the curvature and the normal) of the LS function is present in a narrow band close to the interface, the convection of the LS function is not necessary over all the computational domain Ω. Furthermore, it could be the cause of numerical instabilities [86]. A way to reduce the computational cost and avoid numerical instabilities is to cut off the LS function at a thickness E using for example a sinusoidal filter [22,86].

A sinusoidal filter is applied to the LS function α given by Eq. (49) to obtain the filtered LS function $\breve{\alpha}(\alpha)$:

$$\breve{\alpha}(\alpha) = \begin{cases} \frac{2E}{\pi} & \text{if } \alpha > E \\ \frac{2E}{\pi} \sin \frac{\pi}{2E} \alpha & \text{if } -E \leq \alpha \leq 0 \\ -\frac{2E}{\pi} & \text{if } \alpha < E \end{cases} \tag{65}$$

The advantage of using this function is that its derivative is continuous:

$$\frac{\partial \breve{\alpha}}{\partial \alpha} = \cos \left(\frac{\pi}{2E} \alpha \right) \tag{66}$$

and thus, the reinitialization condition that has to be satisfied is not any more $\| \nabla \alpha \| = 1$, but:

$$\| \nabla \breve{\alpha} \| = \sqrt{1 - \left(\frac{\pi}{2E} \breve{\alpha} \right)^2} \tag{67}$$

For simplicity, the notation α will be used from now on instead of $\breve{\alpha}$. By using this modified LS function, the reinitialization equation presented in Sect. 4.3.4 must be also modified. As a result the convective-reinitialization Eq. (64) is transformed into the following expression:

$$\frac{\partial \alpha}{\partial t} + (\mathbf{v} + \lambda \, \mathbf{v}_r) \cdot \nabla \alpha = \lambda \, \hat{\mathrm{sgn}} \, (\alpha) \sqrt{1 - \left(\frac{\pi}{2E} \alpha \right)^2} \tag{68a}$$

$$\alpha(\mathbf{x}, t = 0) = \alpha_0(\mathbf{x}) \tag{68b}$$

It is important to highlight that Eq. (68) is non-linear since the reinitialization velocity $\mathbf{v}_r$ is a function of the LS function α and the right hand side term also depends on α. But it is linearized by computing $\mathbf{v}_r$ at the previous time step. This linearization is valid as the aim of the convection-reinitialization step is to perform the convection of the LS function α under the physical velocity $\mathbf{v}$, while keeping the smoothness of α. However, the time step must remain small in order to ensure the numerical stability of the method. Fortunately, this condition is often satisfied because the time step needed to compute the physical velocity is small enough to guarantee the stability of the convection-reinitialization step [86].

Fig. 21 Two different filters of the signed distance LS funcion α over a thickness $E = 0.1$. The blue dotted line corresponds to a sinusoidal filter (Eq. 65) and the red dashed line corresponds to a hyperbolic tangent filter (Eq. 69)
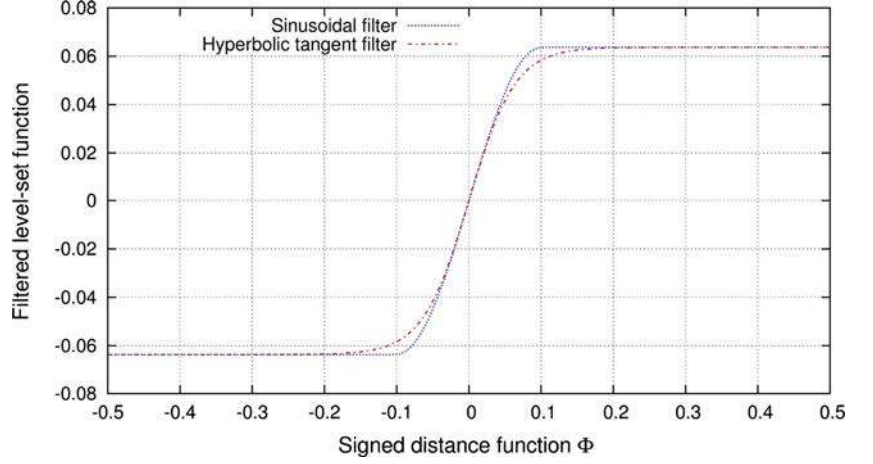
The sinusoidal filter applied in Eq. (65) is one of the filters that can be used. For example a hyperbolic tangent filter can be used, and in this case the filter would be given by:

$$\breve{\alpha}(\alpha) = E \, \tanh\left(\frac{\alpha}{E}\right) \tag{69}$$

It is important to recall that the reinitialization Eq. (58) has to be modified accordingly to the LS filtered function gradient $\nabla\breve{\alpha}$. As a consequence, the convective-reinitialization Eq. (64) also has to be modified. Figure 21 shows a comparison between a sinusoidal filter (Eq. 65) and a hyperbolic tangent filter (Eq. 69). Thickness E is 0.1 for both filters.

4.4.3 Finite Element Discretization

In a finite element frame, this LS has to be implemented. Let us consider the computational domain Ω discretized by using an unstructured mesh $\mathcal{T}_h(\Omega)$ built up of simplex elements K (triangles in 2D and tetrahedra in 3D). The discretized domain is given by Ω_h:

$$\bar{\Omega}_h = \bigcup_{K \in \mathcal{T}_h(\Omega)} K \tag{70}$$

Equation (68) is a first-order hyperbolic equation and consequently can not be discretized by using a Galerkin approximation. Indeed, Galerkin approximation based methods are not stable for purely convective equations. Here, in a finite element framework, the LS function is approximated by a continuous piecewise linear function α_h:

$$\alpha_h \in \mathcal{W}_h = \{w_h \in \mathcal{C}^0(\Omega), w_h \in \mathrm{P}^1(K), \forall K \in \mathcal{T}_h(\Omega)\} \tag{71}$$

where $\mathcal{C}^0$ is the space of continuous functions over Ω.

Equation (68) is discretized by using the well-known "Streamline Upwind/Petrov-Galerkin" (SUPG) method, presented in 1982 by Brooks and Hughes [11] (see also [45]) which leads to a stable formulation. The main idea of this method is to add a diffusion term acting along the direction of the convection velocity $\mathbf{v}$. This is achieved by choosing the weighting functions in a functional space different from

the one of the shape functions (Petrov-Galerkin method). The stabilization is performed by using the SUPG method and the weighting functions $\tilde{w}_h$ are given, on a mesh element K, by:

$$\tilde{w}_h|_K = w_h + \tau_{SUPG} \, \mathbf{v} \cdot \nabla \mathbf{w}_h \tag{72}$$

where the coefficient τ_{SUPG} is given by [86]:

$$\tau_{SUPG} = \frac{1}{M \, |\mathbf{v} \cdot \nabla \mathbf{w}_h|} \simeq \frac{1}{2} \frac{h_K}{\| \mathbf{v} \|} \tag{73}$$

with M the number of nodes per element ($M = n + 1$ in R^n), $| \bullet |$ the absolute value operator and h_K the element size.

The discrete weak formulation of Eq. (68) reads

$$\int_{\Omega_K} \frac{\partial \alpha_h}{\partial t} \tilde{w}_h dV + \int_{\Omega_K} (\mathbf{v}_h + \lambda \, \mathbf{v}_{hr}) \cdot \nabla \boldsymbol{\alpha}_{\mathbf{h}} \tilde{w}_h dV$$
$$= \int_{\Omega_K} \lambda \, \mathrm{s\hat{g}n}\,(\alpha_h) \sqrt{1 - \left(\frac{\pi}{2E}\alpha_h\right)^2} dV \tag{74}$$
$$\alpha_h(\mathbf{x}, t = 0) = \alpha_0(\mathbf{x})$$

It is important to highlight that a validation of the method and its implementation in the finite element library *CimLib*® [25] has been performed by Ville et al. in [86]. In that work, several benchmark problems were considered, including the convection of a circle in 2D and a sphere in 3D, the Zalesak's problem in 2D and 3D, and multiple applications to the jet buckling problem also in 2D and 3D.

4.4.4 Mesh Adaptation Strategy

The LS approach presented in Sects. 4.4.1 and 4.4.2 should be combined with an appropriate mesh adaptation strategy in order to have a better description of the interface Γ. In this work an anisotropic mesh adaptation is used. This strategy has been developed by the team of Coupez [57]. The idea is to create an orthotropic mesh, with different element sizes in each spatial direction.

Very often, discontinuities must be handled across the interface: different mechanical properties, different mechanical behaviors, normal stress discontinuities, etc. Regarding

Fig. 22 Mesh adaptation strategy: **a** Geometry description; **b** classical LS function over the initial mesh; **c** filtered LS function over the initial mesh; **d** inter-particular region from **c**; **e** filtered LS function over the adapted mesh; **f** inter-particular region from **e**

(a) Initial geometry

(b) Classical LS function (Initial mesh)

(c) Filtered LS function (Initial mesh)

(d) Inter-particular zone from c.

(e) Filtered LS function (Adapted mesh)

(f) Inter-particular zone from e.
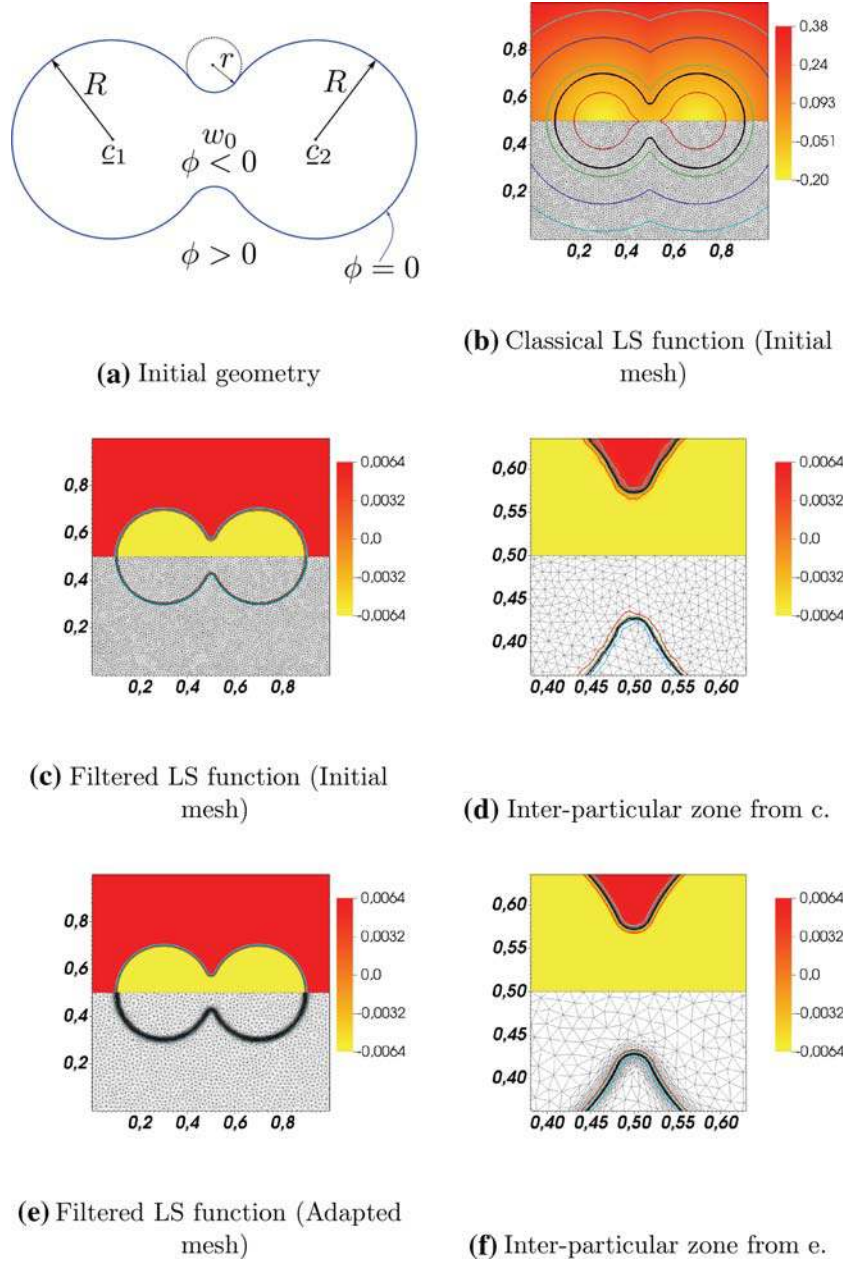
those discontinuities, a way to cope with those problems is to solve them by using a mesh adapted in such a way that an error estimation is minimized. The mesh adaptation is performed by refining and coarsening the mesh based on a metric which is a symmetrical positive defined tensor of order n in $\mathcal{R}^n$. This metric specifies the stretching in the space directions and is computed by using *a posteriori* error estimation. The error estimation is based on the Hessian (second-order spatial derivatives) of a given field. This mesh adaptation strategy allows to capture discontinuities in a more accurate way while keeping a -rather- reasonable number of nodes and elements [57].

In the present work the filtered LS function (Eq. 65) is chosen to compute the metric used for the mesh adaptation.

In fact, by using the filtered LS function for the computation of the metric (error estimation), the obtained mesh is adapted with respect to the geometry of the interface, allowing to describe the interface in a very precise way. Furthermore, the framework for the treatment of the discontinuities for mechanical properties and normal stress (due to the surface tension) is set.

As an example, consider two particles connected with a neck, embedded in a computational square domain of side 1 in 2D. The radius R of both particles is equal to 0.2 and the initial neck radius between them is 10 % of R. The particles are centered respectively in $\mathbf{c}_1 = [0.3, 0.5]$ and $\mathbf{c}_2 = [0.7, 0.5]$. This geometry is shown in Fig. 22a.

The classical LS function is initialized from Eq. (50) and is shown in Fig. 22b along with the initial mesh. The solid black line corresponds to the zero iso-value of the LS function and the other iso-values shown allow to see how the classical LS function changes over all the computational domain. Then, the classical LS function is filtered by using Eq. (65) with $E = 0.01$, it is shown in Fig. 22c. As previously, the black line corresponds to the zero iso-value, but this time all the other iso-values are packed together near to the interface (narrow band of width $2E = 0.02$). It is important to highlight, that the interface is not modified when the filter is applied. The mesh adaptation strategy presented is applied using the filtered LS function to compute the metric. From this metric the mesh is adapted and the result can be seen in 22e. Finally, a close-up of the inter-particular region is shown in Fig. 22d, f.

The initial mesh is made up of 19,800 triangles and the adapted mesh has 17,654 triangles. Even if the adapted mesh has less elements than the initial one, from Fig. 22d, f it is clear that the interface representation is much more smooth when the LS function is computed over the adapted mesh. This more precise representation is obtained because the element size near to the interface on the adapted mesh is about $h_K \approx 0.001$ compared with $h_K = 0.006$ on the initial mesh. If an isotropic mesh was to be built with the equivalent element size of the adapted mesh, it would have about one million of elements. Even if the remeshing process may induce extra computational costs, it will be shown in the next sections that the total computational efficiency (cost and convergence rate) is improved by using mesh adaptation.

4.5 Conclusions

Different numerical strategies allowing to deal with problems involving topological changes have been presented. Among those numerical strategies, fixed grids are more suitable to be used for the sintering simulation at the particles scale. Within the fixed grids approaches, the LS method and multiple VOF techniques were considered. Eventually, the LS method has been chosen over the VOF approaches due to its capability of handling strong topological changes while having a good description of the interface. Also, the computation of some geometrical quantities is easier within a LS context. The main drawback of the LS approach is related to its computational cost, therefore some strategies allowing to reduce the computational time were presented.

First, a modified LS approach, which allows to couple the convection to the reinitialization step, is used. Furthermore, a filtered LS function is used instead of a classical distance signed function which avoids unnecessary convection of the LS function away from the interface. Second, a mesh adaptation strategy is combined in order to have a good accuracy of the solution while keeping a reduced number of nodes.
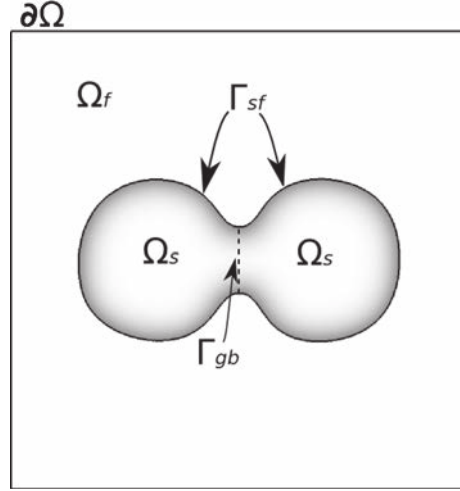

Fig. 23 Computational domain $\Omega = \Omega_s \cup \Omega_f$

In the next section, the simulation of sintering at the particles scale by surface diffusion is presented.

5 Sintering by Surface Diffusion

It was concluded in Sect. 2.3 that the mathematical description of sintering by surface diffusion can reduce to expression 29 of the velocity $\mathbf{v^s}$ associated with this route, defined over the grain free surface Γ_{sf} (Fig. 23) in terms of surface quantities (normal vectors and curvature). However, it has been seen in previous section that within a LS framework, surfaces are embedded into a higher dimensional space (Eq. (57)). Consequently, surface quantities and especially the surface velocity, have to be expressed into a LS framework, in order to be defined over the whole computational domain Ω. It is only under this condition that the advection Eq. (68) can be numerically solved. This issue, as well as subsequent stability considerations, are detailed in Sects. 5.1 and 5.2. Next, comparison between two-grain geometrical model and numerical simulations is presented in Section 5.4.

5.1 Level-Set Formulation of Surface Diffusion

Since surface diffusion velocity is normal to the free surface, and regarding expressions (51)–(52), we can write

$$\mathbf{v_{s\alpha}} = (C_s \Delta_s \kappa_\alpha)\mathbf{n}_\alpha \qquad (75)$$

where $C_s = (1 - f)\Omega_0^a \frac{D_s \delta_s}{kT}$ is assumed to be constant in every particle. According to [16], the surface Laplacian operator can be expressed in the following LS form

$$\Delta_s \kappa_\alpha = \frac{1}{\| \nabla \alpha \|} \nabla \cdot (\| \nabla \alpha \| \mathcal{P}_\alpha (\nabla \kappa_\alpha)) \qquad (76)$$

where

$$\mathcal{P}_\alpha = \mathcal{I} - \mathbf{n}_\alpha \otimes \mathbf{n}_\alpha \tag{77}$$

is the projection matrix operator onto the planes tangential to the isosurfaces of α. Let $\mathbf{a}$ and $\mathbf{b}$ be two vectors of components a_i and b_j respectively, $a_i b_j$ are the components of the second order tensor $\mathbf{a} \otimes \mathbf{b}$.

Equations (75) and (76) give a complete description of the surface diffusion phenomenon in a LS framework. More precisely, these equations allow the velocity $\mathbf{v_{s\alpha}}$ to be defined over the whole computational domain, and to correspond to the surface diffusion velocity in the vicinity of the free surface Γ_{sf}.

5.2 Mixed $\kappa_\alpha / C_s \Delta_s \kappa_\alpha$ Formulation

Regarding Eqs. (75) and (76), the normal component of $\mathbf{v_{s\alpha}}$, that is $v_{s\alpha} = C_s \Delta_s \kappa_\alpha$ is a function of the second-order derivatives of the curvature κ_α, and consequently depends on the fourth-order derivatives of the LS function. However, since the LS function is approximated by a piecewise linear function, its first derivatives are piecewise constant, and all its derivatives of higher order are null. Consequently, $v_{s\alpha}$ can not be calculated as a function of α in a straightforward way.

A first solution to this issue is to compute κ_α and $v_{s\alpha}$ successively in a decoupled way. More precisely, κ_α is first computed as a piecewise linear quantity by considering Eq. (52) in a weak sense. After that, the normal velocity can be computed as a piecewise linear field by taking the weak formulation of (76). However, it is shown in [13,15,69] that such a scheme is not stable and leads to spurious oscillations of the interface. It is possible to regularize the scheme by adding a diffusion term in curvature equation as well as in velocity equation. But this artificial diffusion leads to a lost of accuracy and involves consistency errors.

An alternative, proposed in [13,15,69] is to couple the computation of curvature and normal velocity in a mixed system. A stabilization term is then introduced, based on following considerations. Let us consider the first-order Taylor's expansion

$$\alpha^{n+1/2} \stackrel{\text{def}}{=} \alpha^n + \Delta t \frac{\partial \alpha^n}{\partial t} \tag{78}$$

Assuming the LS is transported by transport Eq. (55) with the surface diffusion velocity (75), one gets

$$\begin{aligned}
\alpha^{n+1/2} &= \alpha^n - \Delta t \mathbf{v_{s\alpha}^n} \cdot \nabla \alpha^n \\
&= \alpha^n - \Delta t \| \nabla \alpha^n \| v_{s\alpha}^n
\end{aligned} \tag{79}$$

since $\nabla \alpha^n \cdot \nabla \alpha^n = \| \nabla \alpha^n \|^2$. The mixed method consists in defining the curvature κ_α^n not with respect to α^n any more, but with respect to $\alpha^{n+1/2}$:

$$\begin{aligned}
\kappa_\alpha^n &= \nabla \cdot \frac{\nabla \alpha^{n+1/2}}{\| \nabla \alpha^{n+1/2} \|} \\
&\approx \nabla \cdot \left(\frac{1}{A^n} \left(\nabla \alpha^n - \Delta t \, \nabla v_{s\alpha}^n \right) \right)
\end{aligned} \tag{80}$$

where

$$A^n = \| \nabla(\alpha^n - \Delta t v_{s\alpha}^{n-1}) \| \tag{81}$$

Hence, in this previous expression of the curvature (80), two approximations are done: the norm of $\nabla \alpha^{n+1/2}$ is evaluated explicitly with the normal velocity taken at the previous time step, in order to keep a linear property of the equation; the norm $\| \nabla \alpha^n \|$ has been removed from the equation: we assume $\| \nabla \alpha^n \| = 1$, which is true in the vicinity of the interface. In fact, not removing this term implies to compute its gradient, which is not an easy task, and creates numerical noise when transporting the LS function, as it has been verified by various numerical tests [13].

To summarize, the mixed formulation in curvature κ_α / normal component of surface diffusion velocity $v_{s\alpha}$ (or, alternatively surface Laplacian of the curvature) writes, at time t_n

$$\begin{aligned}
\kappa_\alpha^n + \Delta t \, \nabla \cdot \left(\frac{1}{A^n} \nabla v_{s\alpha}^n \right) &= \nabla \cdot \left(\frac{1}{A^n} \nabla \alpha^n \right) \\
C_s \, \nabla \cdot \left(\| \nabla \alpha^n \| \mathcal{P}_\alpha (\nabla \kappa_\alpha^n) \right) - \| \nabla \alpha^n \| v_{s\alpha}^n &= 0
\end{aligned} \tag{82}$$

where A^n is defined in Eq. (81)

The second term of the first equation in system (82) is a coupling term between κ_α^n and $v_{s\alpha}$, introduced by $\alpha^{n+1/2}$. It involves the second-order derivatives of $v_{s\alpha}$, and plays the role of a stabilization term, adding a kind of numerical diffusion controlled by the time step. The weak formulation of system (82) is obtained by multiplying equations by scalar weight functions ψ, integrating over Ω, and using the divergence theorem:

$$\begin{aligned}
\int_\Omega \kappa_\alpha^n \psi \, dv &+ \Delta t \int_\Omega \frac{1}{A^n} \nabla v_{s\alpha}^n \cdot \nabla \psi \, dv \\
&= \int_\Omega \frac{1}{A^n} \nabla \alpha^n \cdot \nabla \psi \, dv C_s \int_\Omega \| \nabla \alpha^n \| \mathcal{P}_\alpha (\nabla \kappa_\alpha^n) \cdot \nabla \psi \, dv \\
&- \int_\Omega \| \nabla \alpha^n \| v_{s\alpha}^n \psi \, dv = 0
\end{aligned} \tag{83}$$

for any ψ smooth enough. We can note that κ_α^n, $v_{s\alpha}^n$ and ψ belong to the same functional space: these functions, as well as their first-order derivatives must be square-integrable, and consequently they belong to the $H^1(\Omega)$ Sobolev space. Furthermore, integrals over the computational domain boundary (involved by the divergence theorem) do not appear in (83), and are therefore equal to zero. Consequently, κ_α^n and $v_{s\alpha}$, solution of (83) satisfy $\mathcal{P}_\alpha(\nabla \kappa_\alpha^n) \cdot \mathbf{n}$ and $\nabla v_{s\alpha}^n \cdot \mathbf{n} = 0$ on $\partial\Omega$, where $\mathbf{n}$ is the unit vector normal to $\partial\Omega$.

Let us now consider the finite element discretization of (83). The curvature κ_α and the normal velocity $v_{s\alpha}$ are approximated by continuous piecewise linear functions. Approximating α by a piecewise linear function is now allowed, since no derivatives of α of order higher than one are involved in (83). Once $v_{s\alpha}$ is known, the surface diffusion velocity can be computed as a continuous piecewise linear function

$$\mathbf{v_{s\alpha}}(\mathbf{x}) = v_{s\alpha}(\mathbf{x})\mathbf{N}_\alpha(\mathbf{x}) \tag{84}$$

where $\mathbf{N}_\alpha$ is the piecewise linear vector obtained as the normalized nodal average of the $\mathbf{n}_\alpha$ vectors (which are constant per element).

5.3 Time-Stepping Strategy

Time-stepping strategy for the simulation of sintering by surface diffusion is summarized in Algorithm 2. It consists in initialising the LS A^0 (Eq. 81), to solve for curvature κ_α^n and normal surface velocity $v_{s\alpha}^n$ and then update configurations (convection-reinitialisation of the level-sets).

Algorithm 2 Time-stepping for surface diffusion

$n \leftarrow 0, \alpha \leftarrow \alpha_0$ (Eq. 49), $A^0 \leftarrow \| \nabla \alpha_0 \|$ (Eq. 81)
Initial mesh adaptation
while $t_n \leq \Theta$ **do**
    $\left(\kappa_\alpha^n, v_{s\alpha}^n \right) \leftarrow$ from Eq. (83)
    $\mathbf{v_{s\alpha}}^n \leftarrow$ from Eq. (84)
    $\alpha^{n+1} \leftarrow$ from Eq. (55) with $\mathbf{v} \leftarrow \mathbf{v_{s\alpha}}^n$

    if necessary **then**
        mesh adaptation step
   **end if**
    $n \leftarrow n + 1$
end while

5.4 Sintering Between Two Grains

The case under consideration involves two spherical grains of equal radii $R = 0.2$, as depicted in Fig. 24 (see references[13–15]). One single LS function α is used to represent both grains, meaning that the grain boundary is not taken into account. This function is initialized at each mesh node, as the maximum of the LS functions associated with each spherical grain [because of the sign convention adopted in Eq. (49)]. As shown in Fig. 24a, the grains are initially nearly tangential. It has to be pointed out that there is no special algorithm to deal with either the contact surface nor the singularities of the LS function which is not differentiable on the triple junction. However, despite the initial "roughness" of the area of contact between the grains, and due to the matter diffusion, this area becomes quickly smooth as shown in Fig. 24 b-c. This phenomenon is specifically outlined in
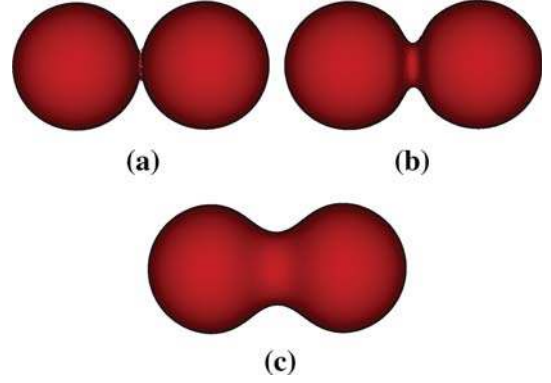
Fig. 24 Change in the free surface $\Gamma_{sf} = \{\alpha = 0\}$ during the sintering by surface diffusion between two grains of equal size (radius $R = 0.2$). **a** $t = 0$; **b** $t = 10^{-2}$; **c** $t = 3$
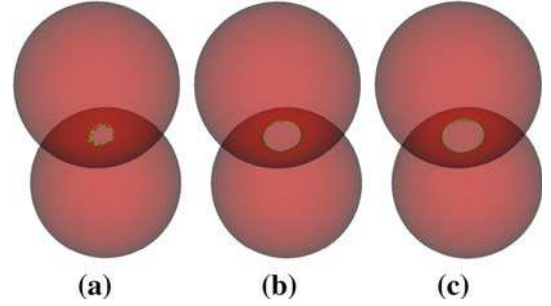
Fig. 25 Triple junction (contour of the contact surface) at times $t = 0$, $t = 10^{-2}$ and $t = 2 \times 10^{-2}$ respectively

Fig. 25: the triple junction (green line) appears to be very irregular when the computation starts (its shape depends on the mesh size), while it has been greatly smoothed after only 10 time steps, and it has become a perfect circle within 20 increments. Furthermore, the flow under Laplacian of curvature has a physical meaning only if the curvature and its second-order derivatives exist, that is, only if the triple junction is smooth enough. Hence, this early stage of the simulation allows us to obtain a triple junction in a "natural" and easy way. Our simulations and the study presented below prove that this stage does not affect the subsequent evolution of the grain cluster. Furthermore, the rate of change in grain volume is presented in Table 3. Of course, this rate depends on the mesh size (and on the time step). However, the results reported in Table 3 show that the formation of a neck between the grains (for a time $t < 0.05$) do not change the grain volume. Indeed, let us recall that surface diffusion is a non-densifying mechanism, as stated earlier in Sect. 2 (Table 1). Here, the grain volume is well-preserved, with a variation of 0.8 % in 1,000 increments. As expected, no shrinkage phenomenon occurs since the grain centres do not move.

As proposed in Sect. 3.1, well-known geometrical models provide analytical relationships for the neck radius growth

Table 3 Surface diffusion between two grains of radius $R = 0.2$: rate of change in volume of grains ($C_s = 10^{-6}$, $\Delta t = 10^{-3}$, $h_{min} = 8.0 \times 10^{-3}$)

Time	0	0.02	0.06	0.1	0.4	0.6	1
Change (%)	0	+0.023	+0.0005	−0.011	−0.18	−0.44	−0.79

upon time under the form of power laws (Eq. 33), with plausible coefficients such as presented by [28] (Table 2). One of these well-known geometrical models of the literature [73] states

$$\left(\frac{x(t)}{R}\right)^n = A\frac{\delta_s D_s \gamma_{sf} \Omega_0^a}{kT R^4} t = A t^* \tag{85}$$

where R is the grain radius, A is a constant and n is a parameter depending on the diffusion route, which can easily be related to other constants such as reported in Table 2. Figure 26 presents, in logarithmic scales, the growth of the dimensionless neck radius x/R versus the dimensionless time $t^* = \frac{\delta_s D_s \gamma_{sf} \Omega_0^a}{kT R^4} t$, obtained by finite element simulation for different grain radii, ranging from 0.1 to 2.5. The best curve fitting these data, obtained by a least-square approximation of the numerical results, is $x/R = 1.3 t^{*1/7}$ and is referred to as "Simulation, 1/7" in Fig. 26. The value $n = 7$ for the surface diffusion route corresponds to the value provided by the analytical model developed by Kuczynski in 1949 [53]. However, it has to be underlined that this value corresponds here to a sort of average value which takes into account the different stages of sintering. The same remark is addressed in [32]. To illustrate that, a curve corresponding to a 1/6-power-law, and referred to as "1/6 law", has been plotted in Fig. 26. It can be shown that this 1/6-law provides a better approximation of the first stage of the sintering ($0.025 \leq x/R \leq 0.053$, obtained with $R = 2.5$). Furthermore, it can be observed in this figure a type of undercutting effect in the early stage of each simulation. This effect, first described in [59], can be explained in the present case by the fact that when the simulation starts, the neck between two grains is defined with an accuracy which depends on the mesh size. When the neck size becomes "reasonable" compared to the mesh size, this effect vanishes and no longer affects the subsequent neck growth.

The next case study is dedicated to the neck growth by surface diffusion between two spherical grains of different radii, say R_1 and R_2. References [66] and [55] state that the neck growth obtained in this case is equal to the one obtained for two grains of same equivalent radius $\overline{R}$ defined by

$$\overline{R} = \frac{2R_1 R_2}{R_1 + R_2}$$

A simulation involving two grains of different sizes (for example $R_1 = 0.1$ and $R_2 = 0.2$ in Fig. 27) proves the relevancy of this definition: the neck radius obtained by sim-
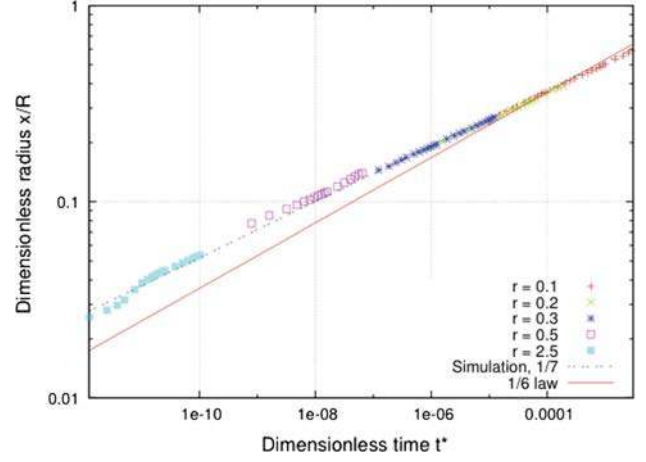
Fig. 26 Change in the dimensionless neck radius x/R over dimensionless time t^* (logarithmic scale) for different values of R, and with $C_s = 10^{-7}$
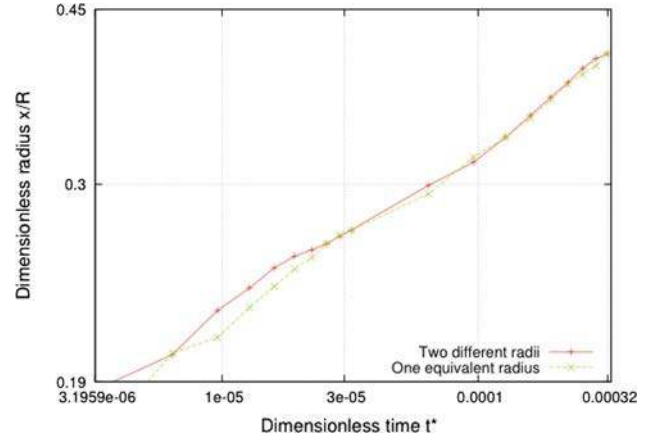

Fig. 27 Growth by surface diffusion of the dimensionless neck radius $x/\overline{R}$ over dimensionless time t^* (logarithmic scale) for two spherical grains of different radii ($R_1 = 0.1$ and $R_2 = 0.2$) and for two spherical grains of same equivalent radius ($\overline{R} = 0.133$)

ulation with two grains of equivalent radius (here $\overline{R} = 0.133$) is shown to be in good agreement with the one computed for two grains of radii 0.1 and 0.2 in Fig. 27.

5.5 Particle Packing Sintering

A sintering simulation by surface diffusion of a set of 154 spherical particles is presented here. As verified in the previous section, surface diffusion does not lead to shrinkage of the compact powder. Figure 28a shows the initial geometry of a powder compact of 154 particles. It can be seen that a mesh adaptation strategy is used and the element size near to the particle surface is smaller ($h_{min} = 0.001$) than the element size far from the interface ($h_{max} = 0.1$). After $170\Delta t$ the neck between the particles has developed up to 50 % of the

(a) $t = 0$

(b) $t = 170\Delta t$.
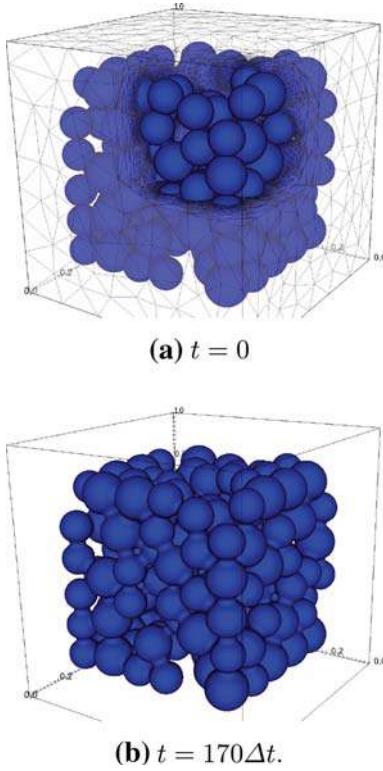
Fig. 28 Sintering simulation by surface diffusion of a set of 154 particles at the initial time (**a**, $t = 0$) and after $170\Delta t$ (**b**, $t = 105\Delta t$)

radius of some particles, however it can be seen in Fig. 28b that the cluster of particles has not shrunk.

One of the main advantages of numerical approaches is that they give access to some variables which would not be accessible though experiments. More precisely, concerning the sintering at the particle scale, the total surface or the specific surface[6] are really complex to assess experimentally during sintering for a compact powder. Fig. 29 shows (dashed green line) the change over time in both simulated total volume and total free surface for the compact powder.

6 Sintering by Volume Diffusion

The velocity associated with the volume diffusion route depends on the gradient of the hydrostatic pressure as established in Sect. 2, (see Eq. 25). In turn, pressure is solution of the classical mechanical problem of momentum balance within the solid grains : $\nabla \sigma = \mathbf{0}$ (Eq. 86). For example using the properties of the isotropic solid, G the shear modulus and K the bulk modulus, the internal equilibrium writes

---

[6] The specific surface is the ratio between the total free surface of the compact powder and its mass.

$$2G \nabla \cdot \varepsilon(\mathbf{u}) - \left(1 - \frac{2G}{3K}\right) \nabla p = 0. \tag{86}$$

where $\mathbf{u}$ is the displacement field. In the small strain framework adopted here, the strain tensor is defined by $\varepsilon(\mathbf{u}) = (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)/2$.

However, within the Eulerian context presented here, the mesh boundary does not correspond to the grain free boundaries, and consequently a surrounding medium has to be taken into account. Thus, as described in Fig. 23, the computational domain is filled with two immiscible phases, Ω_s (the set of solid grains) and Ω_f, the surrounding medium. However, an accurate description of the dynamics of this fluid is not required for sintering simulations: the surrounding media modelling just aims at transmitting the external normal stress applied over the computational domain boundary to the grain free surface. Hence, this surrounding medium is modelled as a incompressible Newtonian fluid with low viscosity. Section 6.1 describes this solid–fluid interaction problem, and the associated numerical methods. A special care has to be paid to the treatment of the surface tension term, the presence of which is a specificity of sintering simulations. Once the pressure field is known, the computation of its gradient is not straightforward, and is explained in Sect. 6.4. Finally, the case of sintering by volume diffusion between two grains is investigated in 6.6.

6.1 Mechanical Problem: Elastic solid—Low Viscous Fluid Coupling

Taking into account the surrounding medium, the mechanical problem in the solid phase (86) is to be completed with a Stokes system in Ω_f (see Fig. 23). Assuming a newtonian fluid ($\sigma^f(\mathbf{v}) = 2\eta\dot{\varepsilon}(\mathbf{v}) - p\mathcal{I}$), the whole system to be solved for the mechanical equilibrium writes:

$$
\begin{array}{lll}
2G \nabla \cdot \varepsilon(\mathbf{u}) - \left(1 - \frac{2G}{3K}\right) \nabla p = 0 & \text{in} & \Omega_s \\
\nabla \cdot \mathbf{u} + \frac{p}{K} = 0 & \text{in} & \Omega_s \\
2\eta \nabla \cdot (\dot{\varepsilon}(\mathbf{v})) - \nabla p = 0 & \text{in} & \Omega_f \\
\nabla \cdot \mathbf{v} = 0 & \text{in} & \Omega_f \\
[\sigma \mathbf{n}]_{\Gamma_{sf}} = \gamma_{sf}\kappa\mathbf{n} & \text{over} & \Gamma_{sf} \\
[\sigma \mathbf{n}]_{\Gamma_{gb}} = \gamma_{gb}\kappa\mathbf{n} & \text{over} & \Gamma_{gb} \\
(2\eta\dot{\varepsilon} - p\mathcal{I})\mathbf{n} = -p_{ext}\mathbf{n} & \text{over} & \partial\Omega
\end{array}
\tag{87}
$$

where η is the fluid viscosity, $\mathbf{v}$ the fluid velocity, and p_{ext} the pressure applied on the boundary of the computational domain. The strain rate tensor is defined by $\dot{\varepsilon}(\mathbf{v}) = (\nabla \mathbf{v} + (\nabla \mathbf{v})^T)/2$.

In order to solve system (87) by a finite element method, the variational formulation of this system is established by summing up variational formulations obtained in fluid domain and in each solid grains. Integrating by parts (divergence theorem), the terms of jump of the stress vector over Γ_{sf} (solid-fluid) and Γ_{gb} (grain-grain) are implicitly taken

27

Fig. 29 Sintering simulation by surface diffusion of a set of 154 particles. Variation of the volume (*green dashed line*) and total surface reduction (*blue line*). (Color figure online)
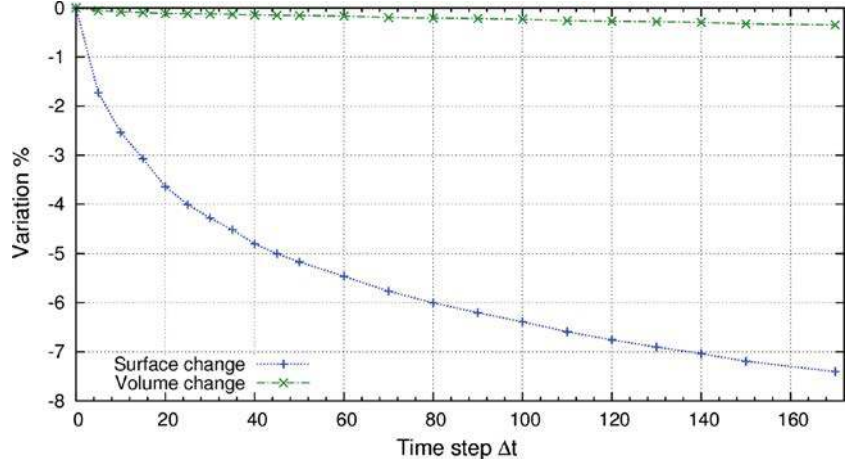
into account in the final variational formulation through integrals defined over Γ_{sf} and Γ_{gb} [69]. Let us recall that the Laplace's law which expresses this normal stress vector jump writes, for an assumed constant surface tension over the solid-fluid interface:

$$[\,(\boldsymbol{\sigma} \cdot \mathbf{n})\,]|_{\Gamma_{s/f}} = \gamma_{sf}\kappa\mathbf{n} \tag{88}$$

and a similar form can be written for interface Γ_{gb}. Hence, the mixed coupled variational formulation consists in finding $(\mathbf{u}, \mathbf{v}, p)$ solutions of:

$$\int_{\Omega_s} 2G\varepsilon(\mathbf{u}) : \varepsilon(\boldsymbol{\Psi})\,dv + \int_{\Omega_f} 2\eta\dot{\varepsilon}(\mathbf{v}) : \dot{\varepsilon}(\boldsymbol{\Psi})\,dv$$
$$- \int_{\Omega_s}\left(1 - \frac{2}{3}\frac{G}{K}\right) p\,\nabla\cdot\boldsymbol{\Psi}\,dv - \int_{\Omega_f} p\,\nabla\cdot\boldsymbol{\Psi}\,dv$$
$$= \int_{\Gamma_{sf}} \gamma_{sf}\kappa\mathbf{n}\cdot\boldsymbol{\Psi}\,da + \int_{\Gamma_{gb}} \gamma_{gb}\kappa\mathbf{n}\cdot\boldsymbol{\Psi}\,da$$
$$+ \int_{\partial\Omega} -p_{ext}\mathbf{n}\cdot\boldsymbol{\Psi}\,da \tag{89}$$

$$\int_{\Omega_s} \psi\,\nabla\cdot\mathbf{u}\,dv + \int_{\Omega_s} \psi\frac{p}{K}\,dv = 0 \tag{90}$$

$$\int_{\Omega_f} \psi\,\nabla\cdot\mathbf{v}\,dv = 0 \tag{91}$$

for any scalar and vector weighting functions, ψ and $\boldsymbol{\Psi}$, smooth enough. Pressure p and the corresponding weighting function ψ just need to be square-integrable: they belong to the $L^2(\Omega)$ functional space. Each component of $\mathbf{u}$, $\mathbf{v}$ and $\boldsymbol{\Psi}$, as well as their first-order derivatives, need to be square-integrable over their domain of definition: they belong to the $H^1(\Omega_i)$ Sobolev space, with $i = s, f$. Finally, note that Eqs. (90) and (91) will be summed up only after the time discretization step.

The pressure field is well-defined in the fluid part as well as in the solid grains, with a discontinuity at the interface. The velocity is well-defined in the fluid part, through the Stokes' equations, while the displacement is well-defined in the solid part, through the relation between strain and dis-

placement. In order to remove to remove the displacement from Eq. (89), we simply set $\mathbf{v} = d\mathbf{u}/dt$ in the solid part. This relation involves a material derivative. However, as dealing with solids, the convective term of this relation is neglected. Using an Eulerian scheme, the displacement is then related explicitly to the velocity. At a time t_n, we have

$$\mathbf{u}^n = \mathbf{u}^{n-1} + \Delta t\mathbf{v}^n \tag{92}$$

Introducing this time discretization into the variational formulations (89), (90) and (91), leads to the mixed velocity–pressure variational formulation of the mechanical problem (87), which is: at a time t_n, assuming the displacement $\mathbf{u}^{n-1}$ to be known, find $(\mathbf{v}^n, p^n)$ solution of the system

$$\int_{\Omega} 2(G\Delta t H_s + \eta H_f)\dot{\varepsilon}(\mathbf{v}^n) : \dot{\varepsilon}(\boldsymbol{\Psi})\,dv$$
$$- \int_{\Omega}\left(\left(1 - \frac{2}{3}\frac{G}{K}\right) H_s + H_f\right) p^n\,\nabla\cdot\boldsymbol{\Psi}\,dv =$$
$$- \int_{\Omega} 2GH_s\varepsilon(\mathbf{u}^{n-1}) : \varepsilon(\boldsymbol{\Psi})\,dv + \int_{\Gamma_{sf}} \gamma_{sf}\kappa\mathbf{n}\cdot\boldsymbol{\Psi}\,da$$
$$+ \int_{\Gamma_{gb}} \gamma_{gb}\kappa\mathbf{n}\cdot\boldsymbol{\Psi}\,da + \int_{\partial\Omega} -p_{ext}\mathbf{n}\cdot\boldsymbol{\Psi}\,da \tag{93}$$

$$\int_{\Omega} \psi\,\nabla\cdot\mathbf{v}^n\,dv + \int_{\Omega} \frac{H_s}{K\Delta t}\psi p^n\,dv =$$
$$- \int_{\Omega} \frac{H_s}{\Delta t}\psi\,\nabla\cdot\mathbf{u}^{n-1}\,dv \tag{94}$$

for any weighting functions ψ and vector $\boldsymbol{\Psi}$ smooth enough. In fact, the velocity field solution of system (93)–(94) is not unique, and velocities corresponding to rigid bodies motions have to be removed. For example, normal velocity can be enforced to vanish on two (in two-dimensional cases) or three plans of the computational domain boundary. In previous system, H_s is a presence function equal to 1 in Ω_s and to 0 elsewhere, and $H_f = 1 - H_s$.

Note that the time discretization scheme (92) is the simplest scheme that can be used. In particular, it involves a minimum of additional terms in mixed variational formu-

lations (93)–(94). However, this scheme is of first order in time and is not so accurate. Consequently, when the elastic body is not in a quasi-equilibrium state and deforms under the velocity (or the displacement) solution of the mechanical problem (93)–(94), a more accurate scheme should be considered, such as the Crank-Nicolson scheme used in [67]. However, the simulations presented in the following, have been carried out by using the Euler's scheme (92), because in our particular case the motion induced by the mechanical problem can be neglected regarding the one induced by the matter diffusion.

6.2 Numerical Strategy

The mixed variational formulation (93)–(94) holding over the whole domain is discretized by using a mixed finite element method. Two issues occur for the choice of the mixed finite element pair. First, it is well-known that the finite element pair chosen for discretizing Stokes problem or elasticity problems (or any mixed problem), must satisfy a compatibility condition, which can be expressed as an $Inf - Sup$ condition, the so-called Brezzi - Babuška condition [10]. If this condition is not satisfied, the algebraic system is ill-conditioned, and the method will not converge. Second, two kinds of discontinuities at the interfaces are involved in Eqs. (93), (94): discontinuities of material parameters, due to the fluid–solid transition; discontinuities of the pressure field, due to Laplace's law which expresses the normal stress jump across an interface separating two media (Eq. 88). These discontinuities can lead to spurious oscillations of the solution fields, especially of the pressure, in the vicinity of the interfaces if no attention is paid.

6.2.1 Stabilized Finite Elements

Still using a mesh made up of simplexes, regarding previous considerations, and because it is suitable with our mesh adaptation strategy, velocity and pressure are approximated by continuous piecewise linear functions, referred to as $P1/P1$ approximation. The $P1/P1$ pair does not fulfil the Brezzi–Babuška condition for the Stokes problem and the quasi-incompressible elasticity problem: the solution of the discretized problems is not unique in pressure. However, this stability condition can be circumvented by adding stabilization terms in the discretized variational formulation.

A theoretical framework to this stabilized finite element approach, has been developed during the last two decades, and is known as variational multiscale (VMS) methods [46–48]. The basic idea of VMS methods, consists in splitting the unknowns of a variational problem in resolvable terms [the finite element (FE) solution] and unresolvable terms (the fine-scale terms, which can not be captured by the finite element mesh). Fine-scale terms are not accounted for by usual

FE methods, which possibly leads to spurious oscillations in the FE solution. Consequently, VMS methods propose to approximate the effects of the fine-scales on the resolvable terms by adding terms in the discretized variational formulation. These additional terms act as stabilization terms.

In concrete words, let $\mathcal{V} = H^1(\Omega)^d$ be the velocity space. Then, we write $\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}$, where $\mathcal{V}_h$ is the space of continuous piecewise linear vectorial functions. Consequently, velocity and associated weighting functions can be split as $\mathbf{v} = \mathbf{v}_h + \tilde{\mathbf{v}}$, $\mathbf{\Psi} = \mathbf{\Psi}_h + \tilde{\mathbf{\Psi}}$, with obvious notations. In the stabilization method proposed here, only the velocity is split, although methods splitting both velocity and pressure could also be considered. For further details, we refer to [69,70]. Here, we just give some essential features. The discretized velocity fine-scale problem is obtained by considering Eq. 93 with $\tilde{\mathbf{\Psi}}$ as weighting functions. Since at the elementary level (K) $\mathbf{v}_{|K}$ is linear, its second-order derivatives are equal to zero. Consequently, assuming that functions of $\tilde{\mathcal{V}}$ vanish over each element edge ∂K, the integration by parts of the velocity fine-scale formulations is reduced to

$$\sum_K \int_K \nabla \cdot (a_1 \dot{\varepsilon}(\tilde{\mathbf{v}})) \cdot \tilde{\mathbf{\Psi}} \, dv$$
$$= \sum_K \int_K a_2 \nabla p_h \cdot \tilde{\mathbf{\Psi}} \, dv \tag{95}$$

where $\sum_K$ stands for the summation over all the mesh elements, $a_1 = 2(G \Delta t H_s + \eta H_f)$ and $a_2 = (1 - 2G/3K)H_s + H_f$. Equation (95) can be interpreted as a projection relation, since it is equivalent to

$$\tilde{\mathcal{P}} \nabla \cdot (a_1 \dot{\varepsilon}(\tilde{\mathbf{v}})) = \tilde{\mathcal{P}} (a_2 \nabla p_h) \tag{96}$$

where $\tilde{\mathcal{P}}$ is the operator of projection onto $\tilde{\mathcal{V}}$. Note that the right-hand side term of Eq. (96) is nothing but the projection of the momentum balance evaluated with the FE solution. Next, the differential operator $\frac{1}{a_2} \nabla \cdot (a_1 \dot{\varepsilon}(\cdot))$ in (96) is approximated, on each element, by an algebraic expression, $\tau_K^{-1} \mathcal{I}$, where τ_K is a stabilization parameter. This approach has been developed in [5,6,17] for example. Finally, taking $\tilde{\mathcal{P}}$ as the identity when applied to ∇p_h leads to the Algebraic Sub-Grid Scale (ASGS) method:

$$\tilde{\mathbf{v}}_{|K} = \tau_K \nabla p_{h|K} \tag{97}$$

This stabilization term, $\tau_K \nabla p_{h|K}$, does not appear in the finite element scale Eq. (93) (which corresponds to $\mathbf{\Psi} = \mathbf{\Psi}_h$), since as already mentioned, the products between the first derivatives of $\mathbf{v}_h$ and the ones of $\mathbf{\Psi}_h$ are equal to zero.

In Eq. (94), the subscale velocity appears through the divergence term:

$$\int_\Omega \nabla \cdot (\mathbf{v}_h + \tilde{\mathbf{v}}_h)\psi \, dv = \int_\Omega \nabla \cdot \mathbf{v}_h \psi \, dv - \sum_K \int_K \tilde{\mathbf{v}} \cdot \nabla \psi \, dv$$

Consequently, the ASGS methods consists in adding the stabilization term

$$\sum_K \int_K \tau_K \nabla p_h \cdot \nabla \psi_h \, dv \qquad (98)$$

to the mass conservation Eq. (94). In the present case (Stokes and elasticity equations), the derived term acts as a pressure diffusion term. Such a stabilization term has also been derived from a bubble stabilization in [79] and is used in [4].

Finally, the stabilization parameter τ_K can be approximated through an adequate Fourier analysis. Using the same procedure as in [6], τ_K can be related to the mesh size h_K and to the material properties:

$$\tau_K = \frac{a_2}{a_1} h_K{}^2 = \frac{(1 - 2G/3K)H_s + H_f}{2(G\Delta t H_s + \eta H_f)} h_K{}^2 \qquad (99)$$

6.2.2 Discretization and Computation of the Surface Tension Integral

The Laplace's law (88) that expresses the fluid - solid coupling appears in Eq. (93) as an integral over the interface $\Gamma_{s/f}$, in a weak form : $\int_{\Gamma_{sf}} \gamma_{gb}\kappa \mathbf{n} \cdot \mathbf{\Psi} \, da$. However, the implementation of this surface integral is not straightforward within a Eulerian context. Indeed, surface $\Gamma_{s/f}$ is not known explicitly (it is not a set of mesh element faces), but it is defined by the zero LS of the function α_h, which passes through the mesh elements. To overcome this situation, two strategies are proposed here: the Continuum Surface Force (CSF) method and the surface local reconstruction (SLR) method. Additionally, two methods for the computation of the curvature are discussed.

Continuum Surface Force (CSF) Method The surface tension term in Eq. (93) has been widely studied [3,7,8,31,36,49, 50,54,74,81,91]. Most of those works uses the CSF method, where the surface integral is approximated by a volume integral. The transformation from the surface integral into a volume integral is achieved by multiplying the original function by a *Dirac* function $\delta(\alpha)$:

$$\int_{\Gamma_{s/f}} \gamma_{sf}\kappa\mathbf{n} \cdot \mathbf{\Psi} \, dS = \int_\Omega \gamma_{sf}\kappa\mathbf{n} \cdot \mathbf{\Psi}\,\delta(\alpha) \, dV \qquad (100)$$

The Dirac function $\delta(\alpha)$ is computed by differentiating a smooth characteristic function $\hat{c}(\mathbf{x})$ that allows to identify the phase which the point $\mathbf{x}$ belongs to:

$$c(\mathbf{x}) = \begin{cases} c_1 & \text{if } \mathbf{x} \in \Omega_s \\ c_2 & \text{if } \mathbf{x} \in \Omega_f \\ \frac{c_1 + c_2}{2} & \text{if } \mathbf{x} \in \Gamma_{s/f} \end{cases}$$

$$\hat{c}(\mathbf{x}) = \mathbf{K} * c(\mathbf{x}) = \int_{\Xi_K} c(\mathbf{x}')\mathbf{K}(\mathbf{x}' - \mathbf{x})d\mathbf{x}' \qquad (101)$$

where Ξ_K is the compact support of the kernel $\mathbf{K}$ (see [91] for further details). The performance of this kind of methods highly depends on the choice of the kernel $\mathbf{K}$ and its support Ξ_K. The numerical computation of this function requires many integration points in order to guarantee the accuracy of the computation of Eq. (100).

The characteristic function used in the CSF version considered in the present work is an approximated Heaviside function that is used also for approximating H_s and H_f, the presence functions in the monolithic formulation (93)

$$\hat{H}_s(\alpha_h)$$

$$= \begin{cases} 1 & \text{if } \alpha_h < -E \\ \frac{1}{2E}\left(E - \alpha_h - \frac{E}{\pi}\sin\frac{\pi\alpha_h}{E}\right) & \text{if } -E \le \alpha_h \le E \\ 0 & \text{if } \alpha_h > E \end{cases}$$

$$(102)$$

$$\hat{H}_f(\alpha_h) = 1 - \hat{H}_s(\alpha_h) \qquad (103)$$

where α_h is the discrete piecewise linear LS function, and E is the width of the transition zone where the mechanical behavior is a mixture of the mechanical behavior of both phases, solid and fluid. The Dirac function in Eq. (100) can be found by differentiating $\hat{H}_f(\alpha_h)$:

$$\delta(\alpha_h) = \frac{\partial \hat{H}^f(\alpha_h)}{\partial \alpha_h}$$

$$= \frac{1}{2E}\left(1 + \cos\left(\frac{\pi\alpha_h}{E}\right)\right) \qquad (104)$$

The corresponding support is given by the region $\Xi_K : -E \le \alpha_h \le E$. One of the disadvantages of this method is that the kernel $\mathbf{K}$ depends on a numerical parameter, here the width of the transition zone E, which has to be chosen with respect to the mesh size [91]. The performance of the method highly depends on this choice.

Surface Local Reconstruction (SLR) method An alternative method to integrate the surface tension term in Eq. (93) is proposed in [70]: here, the interface $\Gamma_{s/f}$ is given by the zero isosurface of the LS function, $\{\alpha = 0\}$. Since it allows to locally reconstruct the interface, then the surface tension term can be explicitly computed over this reconstructed surface.

This method consists, for each element cut by the interface, in approximating linearly this surface by a segment in 2D or a plane in 3D. This approximation is possible thanks to the metric properties of the LS function. Once the interface has been locally reconstructed, the contribution of the

element to the surface integral can be explicitly computed by Gaussian integration[7]: the integration points used to compute the integral are placed over this surface. It has to be pointed out that this approach is local to each element, and can be implemented in the assembly loop of a finite element code. In particular, the whole surface is never reconstructed. The first advantage of this method is that it does not require to choose any numerical parameter contrary to the CSF method. Furthermore, the accuracy of the numerical results are shown to be slightly improved using this technique, as described in Sect. 6.3.

From a practical point of view, the linear approximation of the surface is given at the element level by the plane (or the segment in 2D) where the LS function vanishes. The points where the LS function is equal to zero are referred to as vertices in the following. These vertices can then be placed over the edges of the element, when the LS function α is positive in one node of this edge and negative in the other one; in special cases, they correspond to a node of the element (when $\alpha = 0$ on that node). The different possibilities of intersections between the interface and an element are given in Fig. 30 for the 2D case, and in Fig. 31 for the 3D case. In the general case, the coordinates of the vertex are function of the coordinates of the two nodes forming the edge as well as the value of the level-let function on these nodes. The coordinates of the vertex $\mathbf{x}_{vertex}$ placed on the edge formed by the nodes m and n are given by:

$$\mathbf{x}_{vertex} = \mathbf{x}_m + (\mathbf{x}_n - \mathbf{x}_m) \frac{\| \alpha(\mathbf{x}_m) \|}{\| \alpha(\mathbf{x}_m) \| + \| \alpha(\mathbf{x}_n) \|} \qquad (105)$$

where $\mathbf{x}_m$ and $\mathbf{x}_n$ are, respectively, the coordinates of nodes m and n, and $\alpha(\mathbf{x_m})$ and $\alpha(\mathbf{x_n})$ are the values of the LS function at these nodes, which should have opposite signs.

Note that in 3D cases, the intersection of the interface with an element can be a quadrilateral (Fig. 31g) if the LS function is positive in two nodes and negative in the two other nodes. In this case, the quadrilateral is divided into two triangles, and the contribution of the element to the surface integral is the sum of the contribution of both triangles. Furthermore, special attention has to be paid to particular cases described in Figs. 30 and 31. First, in the cases presented in Figs. 30a and 31a, b, the element is not intersected by the interface, and its contribution to the surface integral is therefore zero. Next, Figs. 30b and 31c show configurations where the interface can be shared by two neighboring elements. The contribution of each of these elements to the surface integral consequently to be divided by two. Finally, in Figs. 30c and 31d, e, the LS function is equal to zero in at least one node of the element. In this case, the integration is carried out as for a regular element cut by the interface.

---

[7] An analytical integration can also be carried out.
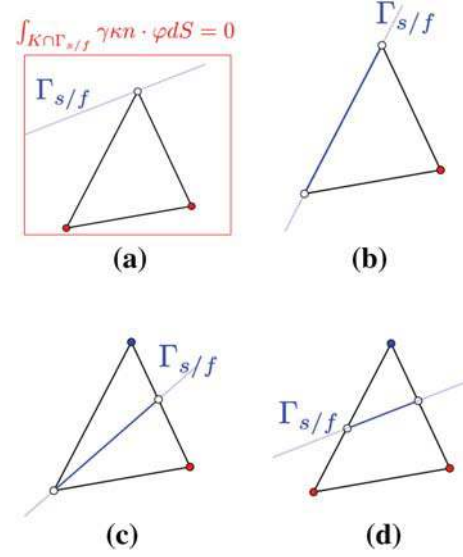
Fig. 30 Intersection between the interface $\Gamma_{s/f}$ and an element in 2D. **a** Point. **b** Segment (edge). **c** Segment (point). **d** Segment
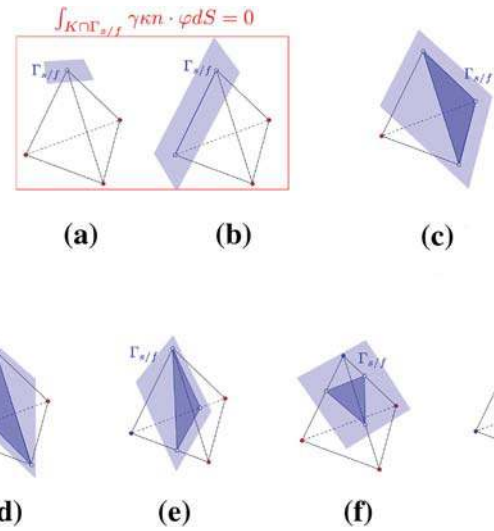
Fig. 31 Intersection between the interface $\Gamma_{s/f}$ and an element in 3D. **a** Point. **b** Edge. **c** Triangle (face). **d** Triangle (edge). **e** Triangle (point). **f** Triangle. **g** Quadrilateral

Eventually, the SLR method also serves as a "preconditionning" for the computation of the volume integral which make up the system to be solved, for elements intersected by the interface. This procedure is then equivalent to consider exact Heaviside functions (Eq. 53) instead of approximate functions, such as Eq. (102), to separate fluid and solid regions.

Curvature computation strategies As indicated previously, the performance of the present method highly depends on the accuracy of the computation of the curvature κ. The first method has been introduced in the previous Sect. 5 within the context of sintering by surface diffusion. By construction, the mixed 'curvature-surface Laplacian' formulation ($\kappa / \Delta_s \kappa$)

presented in Sect. 5.2 [13] is solved for the curvature κ. Then this method can also be used here, since the simulation of sintering by multiple diffusion mechanisms constitutes, the aim of this work, can rely on the curvature obtained from the surface diffusion step and then be introduced into the mechanical solver to obtain the pressure field needed for the volume diffusion. This explicit curvature computation is referred to as "Direct" method.

An alternative method can be adopted to avoid the explicit computation of the curvature. Let $\boldsymbol{\Psi}$ be a vector in $\mathcal{R}^n$. The surface gradient of $\boldsymbol{\Psi}$ with respect to $\Gamma_{s/f}$ is defined as the tangential part of the gradient of $\boldsymbol{\Psi}$: $\nabla_{\mathbf{s}}\,\boldsymbol{\Psi} = \nabla\boldsymbol{\Psi} - (\nabla\boldsymbol{\Psi} \cdot \mathbf{n}) \otimes \mathbf{n} = (\mathcal{I} - \mathbf{n} \otimes \mathbf{n}) \cdot \nabla\boldsymbol{\Psi}$, where $\mathbf{n}$ is the outward normal to $\Gamma_{s/f}$. The surface divergence of $\boldsymbol{\Psi}$ is defined as the trace of the surface gradient operator:

$$\nabla_{\mathbf{s}} \cdot \boldsymbol{\Psi} = \nabla \cdot \boldsymbol{\Psi} - (\nabla\boldsymbol{\Psi} \cdot \mathbf{n}) \cdot \mathbf{n} = (\mathcal{I} - \mathbf{n} \otimes \mathbf{n}) : \nabla\boldsymbol{\Psi} \quad (106)$$

Furthermore, since $\mathcal{I} - \mathbf{n} \otimes \mathbf{n}$ is the operator of projection onto the tangent space of $\Gamma_{s/f}$, the vector $(\mathcal{I} - \mathbf{n} \otimes \mathbf{n}) \cdot \boldsymbol{\Psi}$ is tangent to $\Gamma_{s/f}$. Consequently, and because $\Gamma_{s/f}$ is a closed surface, the following relation holds [29]:

$$\int_{\Gamma_{s/f}} \gamma_{sf} \nabla_{\mathbf{s}} \cdot [(\mathcal{I} - \mathbf{n} \otimes \mathbf{n}) \cdot \boldsymbol{\Psi}]\, dS = 0 \quad (107)$$

where γ_{sf} is assumed to be a constant.

By using the definition of the surface divergence (Eq. 106), the above relationship (Eq. 107) writes:

$$\int_{\Gamma_{s/f}} \gamma_{sf} [\mathcal{I} - \mathbf{n} \otimes \mathbf{n}] : \nabla\boldsymbol{\Psi}\, dS$$
$$= \int_{\Gamma_{s/f}} \gamma_{sf} (\mathcal{I} - \mathbf{n} \otimes \mathbf{n}) : \nabla(\mathbf{n} \otimes \mathbf{n} \cdot \boldsymbol{\Psi})\, dS$$

Finally, it can be shown that the right hand side term in the previous equation is equal to the surface tension term that is to be computed in Eq. (93). The mean curvature can then be defined as the divergence of the outward unit normal along the surface because $\nabla \cdot \mathbf{n} = \nabla_s \cdot \mathbf{n}$. Therefore the last expression in the previous equation is equal to the surface tension term in Eq. (93).

$$\int_{\Gamma_{s/f}} \gamma_{sf} [\mathcal{I} - \mathbf{n} \otimes \mathbf{n}] : \nabla\boldsymbol{\Psi}\, dS = \int_{\Gamma_{s/f}} \gamma_{sf} \kappa \mathbf{n} \cdot \boldsymbol{\Psi}\, dS \quad (108)$$

By substituting the surface tension term of Eq. (93) by the left hand side term of Eq. (108), the explicit computation of curvature κ can be avoided. This approach was inspired by [29], and also discussed in [54].

6.3 Numerical Validation

Again, all the simulations presented in the forthcoming section have been performed by using the finite element library *CimLib*, a fully parallel C++ scientific code, mainly developed at the CEMEF, MinesParisTech [25].

6.3.1 Parasitic Current (Two-Phase Incompressible flow)

As a first validation of the approach, a benchmark case is considered that has been widely studied to check the accuracy of the different methods introduced so far. The test consists in computing the mechanical equilibrium of an incompressible fluid bubble placed inside another incompressible fluid, with the stress vector set to $-p_{out}\mathbf{n}$ over the boundary of the computational domain. The analytical solution of this problem is a zero velocity field and a pressure field p equal to p_{out} outside the bubble and jumping to $p_{in} = p_{out} + \gamma_{sf}\kappa$ inside the bubble.

It has to be highlighted that the formulation of the mechanical problem (Eqs. 93 and 94) holds for a problem where the phase 1 (Ω_s) presents a linear elastic behavior. Hence, in order to perform this two-fluid test case, the formulation has to be slightly modified. Both phases will be assumed to respond as Newtonian fluids. Even if the analytical solution of this problem is a zero velocity field, some nonphysical *parasitic currents* are generated as a result of the numerical simulation [3,7,8,31,36,49,50,54,74,81,91]. These parasitic currents are used to assert the convergence and the performance of the different approaches presented below to calculate the surface tension integral.

The dependence of the maximum velocity $\mathbf{v}_{max}$ to the surface tension / viscosity ratio (γ_{sf}/η) is first investigated. Table 4 shows the value of $C = \|\mathbf{v}_{max}\|/(\frac{\gamma_{sf}}{\eta})$ obtained for a wide range of values of γ_{sf}/η, from 10 to 10^5. C is found to be constant. This result confirms the proposition in [54] and [81]: $\|\mathbf{v}_{max}\|$ depends linearly on γ_{sf}/η.

Figure 32 shows how the parasitic currents are mainly concentrated around the interface $\Gamma_{s/f}$. The simulations have been performed over a structured mesh with an element size of 1/64 (Fig. 32 left), using the direct computation of the curvature, i.e. from the mixed formulation $\kappa/\Delta_s\kappa$. Complementary tests showed that the maximum velocity $\mathbf{v}_{max}$ does not depend on the element size, and consequently, the results quality depends mostly on the method used for prescribing the surface tension. Additionally, those tests showed that the result did not either change when the surface integral was computed using CSF or SLR.

Next, the dependence of the finite element error on the bubble mean pressure is evaluated versus the mesh size. The simulation is performed in a structured 2D mesh of N nodes over a square domain of side length 1 m. The parameters of the simulation are a bubble radius of $R = 0.2\,\mathrm{m}$, a surface tension coefficient $\gamma_{sf} = 0.9\,N/m$ and a viscosity of $\eta = 1000\,\mathrm{Pa} \cdot \mathrm{s}$. Pressure at the outer boundary is set to zero ($p_{out} = 0$). Therefore the theoretical value of the pressure inside the bubble is $p_{in}^{th} = 4.5\,\mathrm{Pa}$. Figure 33 shows the relative error on the mean pressure inside the bubble ($(p_{in}^{th} - p_{in})/p_{in}^{th}$) as a function of the square root of the num-

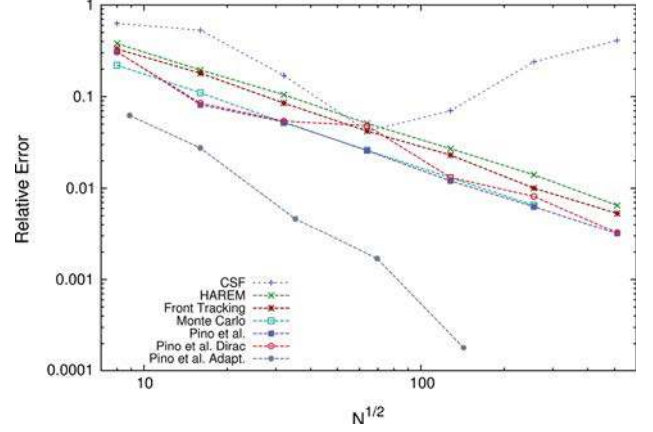**Table 4** Values of $C = v_{max}\eta/\gamma_{sf}$ for the two-fluid simulation

γ_{sf}/η		10^1	10^2	10^3	10^4	10^5
C	[81]	9.1×10^{-3}	9.0×10^{-3}	8.8×10^{-3}	9.1×10^{-3}	9.4×10^{-3}
	Pino Muñoz et al. [70]	3.7×10^{-3}	3.7×10^{-3}	3.7×10^{-3}	3.7×10^{-3}	3.7×10^{-3}
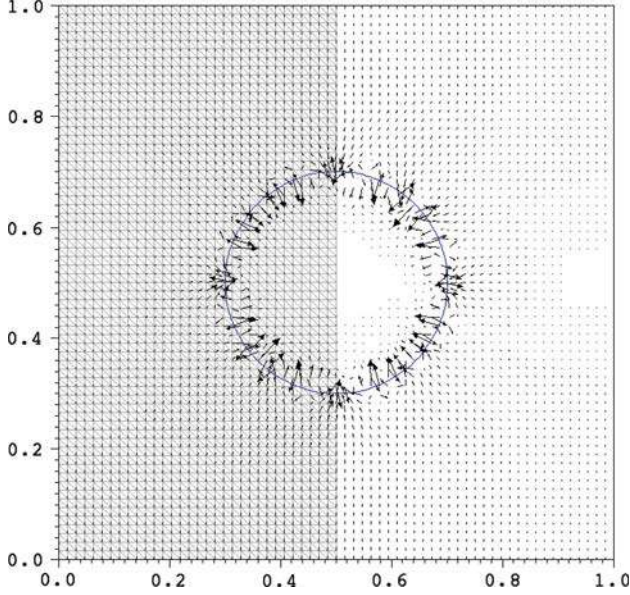
Fig. 32 Parasitic currents in a static bubble simulation over a square computational domain of side 1 and element size $h = 1/64$ (*left* with mesh, *right* velocity only)

Fig. 33 Relative error on the mean pressure computed inside the bubble

Table 5 Relative error on the mean pressure as a function of the stabilized finite element and curvature computation methods

Stabil.	Curvature κ Analytical	Direct	Tensorial
ASGS	1.72×10^{-2}	1.72×10^{-2}	1.73×10^{-2}
$P1+/P1$	6.20×10^{-3}	6.29×10^{-3}	8.65×10^{-3}

ber of nodes $N^{1/2}$. Four of these curves have been extracted from [7]: results named "CSF" are computed by Brackbill et al. using the *Continuum Surface Force* method [8] within a Eulerian VOF framework; Results named "HAREM" correspond to an augmented Lagrangian method combined with a VOF approach [87]; the "Front Tracking" results correspond to a: LS Finite Differences method [78]; and the "Monte Carlo" approach is based on energy potential including interface and volume energies [7].

The fifth curve in Fig. 33, referred to as "Pino et al." corresponds to the simulation carried out by using the direct computation of the curvature ($\kappa/\Delta_s\kappa$), and the SLR method for surface tension integral evaluation. For comparison, the sixth curve, referred to as "Pino et al. Dirac" is obtained in the same condition of curvature computation, but using an approximation of the Dirac delta function to compute the surface integral similarly to the "CSF" method. Both simulations show a similar trend, with a first order spatial convergence, and can be compared with relevancy to the recent results obtained by Bordère et al. [7], referred to as "Monte Carlo" in Fig. 33. The mesh adaptation strategy presented in Sect. 4.4.4 has also been used to perform the same simulation presented previously and the corresponding relative error on the mean pressure is also plotted in Fig. 33 (referred

to as "Pino et al. Adapt"). This mesh adaptation leads to a non uniform element size distribution which has an element size smaller close to the interface and larger away from $\Gamma_{s/f}$. Compared with an structured mesh with the same number of nodes, the adapted mesh has an element size that could be up to 20 times smaller that the structured one. This clearly yields an improved convergence of the numerical approach.

Finally a comparison between the two stabilized finite element methods used to solve the mechanical problem, that is $P1+/P1$ and ASGS, is reported in Table 5. For this comparison the same incompressible bubble inclusion simulation as described previously has been considered, using a structured mesh with an element size of 1/256. The curvature is computed in three different manners: in the first one ("Analytical") the curvature is replaced by its exact value $\kappa = 1/R$, the second method ("Direct") is the mixed solver $\kappa/\Delta_s\kappa$, and the third way ("Tensorial") corresponds to the tensorial expression given in Eq. (108).

For this test case the best results are found by using a $P1+/P1$ stabilization whatever the curvature assessment procedure. Furthermore, it is an interesting fact to verify, like in [29], that the result obtained by computing the cur-

Table 6 Material properties of the solid and the fluid considered

Properties	Value
Elastic solid	
μ	156 GPa
K	260 GPa
ν	0.25
γ_{sf}	0.9 J/m^2
Incompressible fluid	
η	1 Pa s

Fig. 34 L^2-relative error on the pressure computed inside the elastic inclusion

vature with the direct method allows us to get a result as accurate as in the case where the curvature is replaced by its exact analytical value. It is also important to highlight that the ASGS stabilization does not present any relevant dependence upon the curvature computation method. However, it has to be underlined that $P1 + /P1$ stabilization appears slightly better, for this case, than the ASGS method when the error is estimated with the mean pressure. If the error is now based on the maximum pressure value (i.e. L^∞-norm), the conclusions are different as shown in next section.

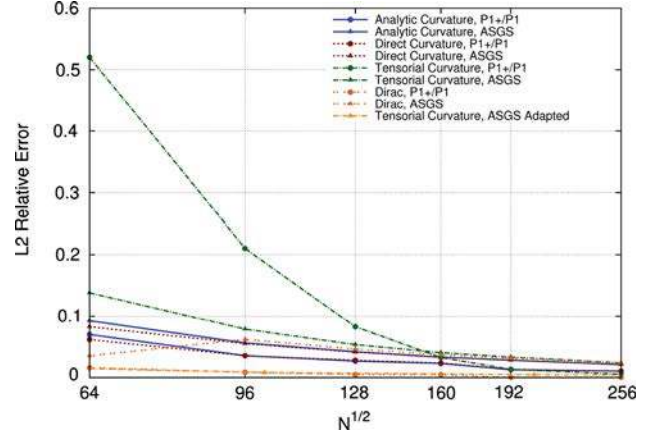### 6.3.2 Fluid–Elastic Solid Interaction

The mechanical monolithical formulation presented in Eqs. (93) and (94) will be used from now on to simulate the interaction between an incompressible fluid and a linear elastic solid with surface tension. This kind of interaction presents some new challenges such as coping with the high ratio between the mechanical properties of the two phases and the compressible nature of elastic solids.

Spherical elastic inclusion This section focuses on the case of an elastic cylindrical 2D inclusion embedded into a Newtonian fluid matrix. Like in the previous section, the stress vector is set to $-p_{out}\mathbf{n}$ over the boundary of the computational domain. The mechanical properties of both phases, solid and fluid, are presented in Table 6. This problem is solved under the plane strain hypothesis. The pressure inside the elastic inclusion depends on both the curvature κ and surface tension coefficient γ_{sf}, but is also a function of the mechanical properties of the solid. The analytical solution of this problem is:

$$p_{in} = p_{out} + \frac{3K}{3K + \mu}\gamma_{sf}\kappa = p_{out} + \frac{2}{3}\gamma_{sf}\kappa\,(1 + \nu) \quad (109)$$

where p_{in} is the pressure inside the solid (Ω_s), p_{out} is the pressure in the fluid (Ω_f), and ν is the solid Poisson's ratio. In the considered case p_{out} is set to zero, and therefore the pressure inside the solid should be $p_{in} = 3.75$ MPa considering the mechanical properties in Table 6.

Within the Eulerian approach considered in this work, the surface tension at $\Gamma_{s/f}$ introduces a discontinuity of the normal stress, and therefore, a discontinuity of pressure. However, the presented simulations have been carried out by approximating the pressure with a continuous piecewise linear function. With this continuous approximation, the pressure discontinuity and the material property discontinuity at the fluid–solid interface, generate some pressure oscillations that do not represent finely the physical nature of the problem. Nevertheless, continuous approximations remain popular and easy-to-implement methods to deal with two-phase problems [7,8,36,49,50,54,74,81], although other techniques as the extended finite element method, the discontinuous Galerkin method, or a recent modified continuous Galerkin method [4] exist and can be used.

Figure 34 shows the relative L^2-error on the pressure, as a function of the square root of the number of nodes. The surface integral has been evaluated by using the SLR method which was proved to yield the best results. The curvature can be computed into three different ways: the first one corresponds to its analytical value (solid lines) and the two other ones are the direct formulation ($\kappa/\Delta_s\kappa$) (dotted lines) and the tensorial method (dashed lines) expressed in Eq. (108). Both stabilization methods studied in this work are also plotted: the $P1 + /P1$ stabilization (filled circles) and the ASGS method (filled triangles). It is important to note that the six first curves are obtained with structured meshes, and the last one (referred to as "Tensorial Curvature, ASGS Adapted") is obtained from an adapted unstructured mesh.

It can clearly be seen that the surface tension term is not accurately computed by the tensorial method for structured meshes with less than 160^2 nodes (green curves). However, when the mesh size is small enough, the tensorial method leads to results with a relative error lower than 2 %, which is of the same order as the error obtained with the analytical value of the curvature, or obtained with the direct computation of the curvature. The stabilization method also has an effect on the relative error: with the structured meshes, the
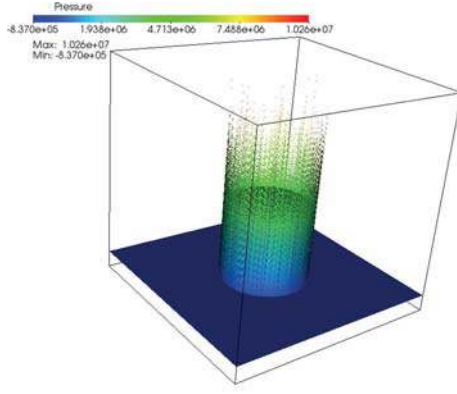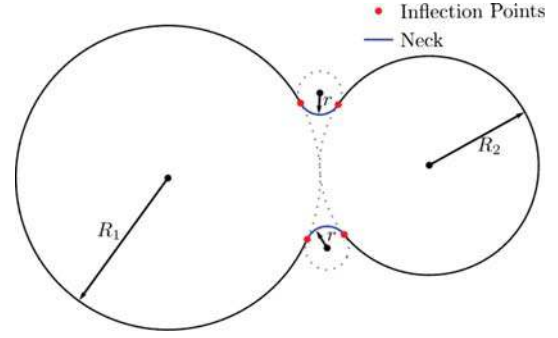
(a)
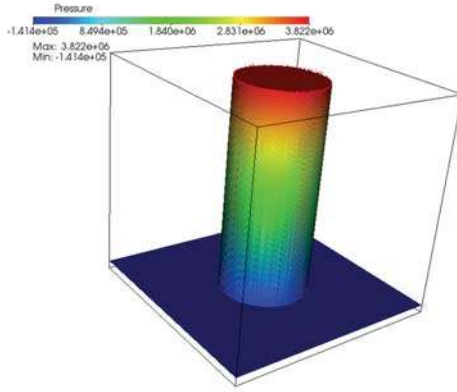
(b)

Fig. 35 Oscillations of the pressure field at the fluid–solid interface, using MINI-element or ASGS stabilization (with two different scales). **a** MINI-element. **b** ASGS method

Fig. 36 Two-particle problem—Radii and inflection points (2D)

$P1 + /P1$ stabilization seems to lead to slightly better result than the ASGS method. However, even if the relative L^2-error on the pressure is better when using the $P1 + /P1$ stabilization, the nonphysical oscillations of the pressure (L^∞-error) are significantly amplified when using the $P1 + /P1$ stabilization, as it can be seen in Fig. 35. Accordingly, the pressure analytical solution should be zero in the fluid, but as a result of the non physical oscillation of the pressure close to the interface, this pressure is lower than zero, and larger for the $P1 + /P1$ case ($-8.35E5$ Pa $\simeq 24 \% \ p_{in}$) than for the ASGS method ($-1.41E5$ Pa $\simeq 4 \% \ p_{in}$). Since we seek the smoothest pressure distribution, especially for physical phenomena driven by pressure gradients, the following simulations are performed by using the ASGS method. The impact of the curvature computation method is discussed further in the next section.

Two spherical particles with neck The case described in Fig. 36 is now considered: two spherical elastic particles of radii R_1 and R_2, are connected by a neck of radius r. The analytical value of the curvature is known all over the surface: over the neck it is equal to $1/r$, over the surface of each particle it is equal to $1/R_1$ and $1/R_2$, respectively, and at the inflection points (see Fig. 36) the curvature is zero. A 2D simulation is carried out (plane strains assumption), for which the computational domain is the unit square, and the particles radii are $R_1 = 0.2\mu$m and $R_2 = 0.1\mu$m and the neck radius is $r = 0.01\mu$m. The material properties of both fluid and solid phases are those given in Table 6.

The results obtained with the approach presented in this work are compared with those obtained by performing the same simulation using *Abaqus 6.10®*. Contrary to *CimLib*, the framework adopted in *Abaqus®* for the elastic analysis is a Lagrangian one. Consequently, the following comparison methodology has been used. First, the mesh of the computational domain is created using the *Gmsh* free mesher (see [34]), in such a way the fluid–solid interface is exactly represented by a set of interior element edges, as shown in Fig. 37. A Eulerian computation is then carried out with *CimLib*, using this mesh made up of ASGS-stabilized triangles, the SLR algorithm, and three methods for calculating the curvature: the analytical one, the "direct" one, and the "tensorial" one. A Lagrangian simulation is subsequently achieved with *Abaqus®*, using the particles mesh, extracted from the previous fluid–solid mesh. Furthermore, the curvature is computed in two different ways to study the impact of slight curvature variations onto the mechanical equilibrium, independently of the FE solver for the mechanical problem. The first method uses the analytical value of the curvature. The second one uses *CimLib* with the direct method to compute the curvature that is subsequently imported into *Abaqus®*. The Laplace's law (88) is then applied as a Neumann condition over the mesh boundary.

Figure 38 shows the variation of the computed pressure along the vertical line connecting both necks, as depicted in the sketch of the particles configuration. Note that the elastic analysis in *Abaqus®* is carried out with displacement based FE. The pressure field is then post-treated and equal to minus one-third the trace of the stress tensor. Looking at the *CimLib* and *Abaqus®* simulations using the analytical value of the curvature in Fig. 38 ("Analytical Curvature"),
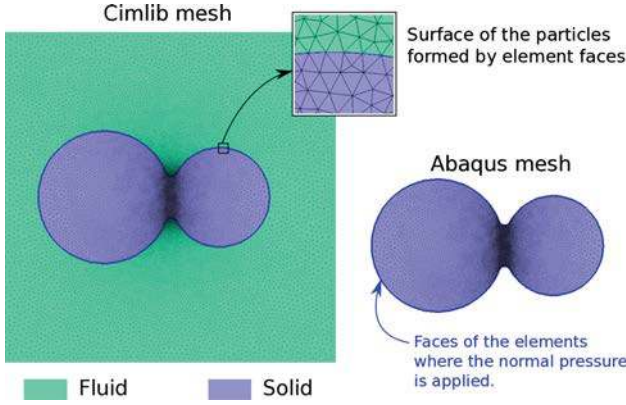
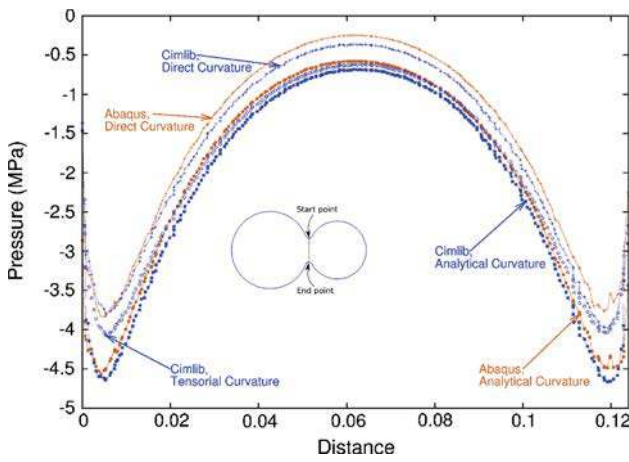**Fig. 37** Both meshes used for the *CimLib–Abaqus*® comparison

Fig. 38 Comparison of the pressure along the vertical line connecting both necks in 2-grain test case (sketch in *left* corner)

Fig. 39 Comparison between the pressure field obtained with *Abaqus* using analytical curvature (*upper*) and *CimLib* using Tensorial curvature (*lower*)

Fig. 40 Comparison between the magnitude of the displacements obtained with *Abaqus* (*upper*) and *CimLib* (*lower*). $u_x = u_y = 0$ at point "A" and $u_y = 0$ at point "B". The analytical curvature (*upper*) and Tensorial curvature (*lower*) methods were used

an excellent agreement is found. The same kind of agreement is found regarding the *CimLib* and *Abaqus*® simulations using the direct method for computing the curvature ("Direct Curvature"). But a slight variation of the curvature (Analytical or Direct) has an impact on the pressure, which means that the mechanical problem itself is intrinsically very sensitive to slight curvature variations. Obviously the mechanical approach developed here also presents this sensitivity. Pressure obtained with *CimLib* using the tensorial method to express the surface tension term is bounded by the pressure computed by using the Analytical and Direct method for the curvature, demonstrating the precision of this method for curvature computation even for a non-structured mesh.

The tensorial method still yields good results, as it can be seen in Fig. 39, where the pressure isovalues obtained with this approach under *CimLib* are compared with the *Abaqus*® results. Furthermore, two advantages have been found to the tensorial method compared with the direct curvature evaluation. First, in the limit case where the neck curvature tends to high negative values, the pressure field exhibits a better
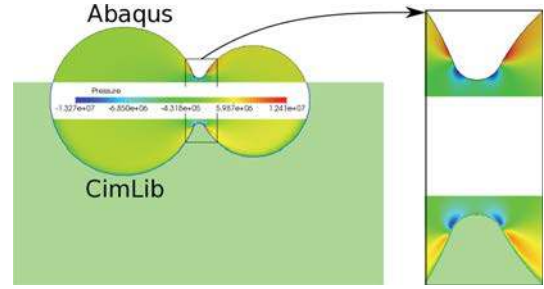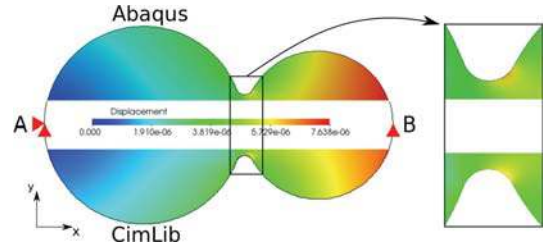
behavior when it is computed using the tensorial method. Second, when using an iterative method to solve the linear algebraic system obtained from the finite element discretization, the convergence is found to be better with this approach than when the curvature is computed. And finally, as the explicit curvature estimation is avoided, the computation time is reduced.

As a complement, even if the variational formulation of the problem has been developed as a function of the velocity, the displacement inside the solid can be computed during the post-processing by using the Euler explicit scheme (Eq. (92)). The Eulerian framework used can generate a rigid body displacement of the solid. In order to be able to establish a comparison between the results obtained from *Abaqus*® and *CimLib*, this rigid body displacement should be subtracted to match the Dirichlet boundary conditions applied to the *Abaqus*® simulation. Consequently, a translation is applied to the solid in *CimLib* in order to set the displacement of point "A" in Fig. 40 equal to zero. And finally, a rotation is applied to the solid to prescribe the displacement in the Y direction equal to zero at point "B" (Fig. 40). After the post-processing, a comparison of the displacement magnitude between the results obtained from *Abaqus*® (upper) and *CimLib* is presented in Fig. 40. A very good agreement between both simulations is found.

Finally, it was established in Sect. 2.2.3 that the fluid velocity associated with volume sintering is proportional to the
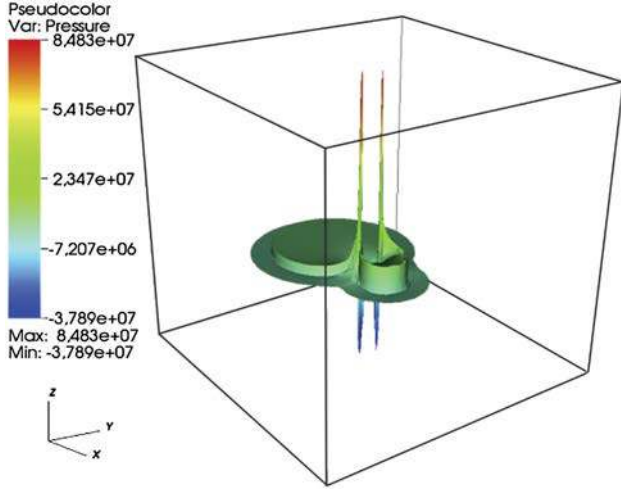
Fig. 41 3D representation of the pressure distribution (Pa) in 2D particles connected with a neck

Fig. 42 Pressure (Pa) field in a particle packing

gradient of the hydrostatic pressure (Eq. 25). This means that even very slight oscillations of the pressure can lead to high magnitude localized velocity, and consequently to a degeneration of the surface of the particles after a few time steps. It can be shown that the numerical approach presented allows to obtain a pressure field with very low unphysical oscillations, even in cases where the surface tension induces a very high gradient of pressure. For instance, Fig. 41 presents the pressure computed for the case of two 2D particles connected through a neck using the same mechanical characteristics (Table 6) for particles of radius $R_1 = 0.2$ μm and $R_2 = 0.05$ μm and the neck radius is $r = 0.01$ μm. The curvature close to the neck between the particles is very high, which leads to large pressure gradients but still no strong pressure oscillations can be observed (Fig. 41).

Multiple spherical particles in 3D. To conclude this section dedicated to the numerical tools developed for volume diffusion sintering, the simulation of a particle packing embedded into a Newtonian fluid is shown in Fig. 42. Packing is made up of 178 elastic spherical particles, which can initially slightly intersect each other, with a uniform radius distribution ranging from 0.05 to 0.1 μm. The material properties of both fluid and solid phases are those from Table 6. The mesh has been adapted in the vicinity of the particles surface using the strategy presented in Sect. 4.4.4, in order to improve the description of the microstructure while keeping a reasonable number of elements. In the simulation shown in Fig. 42 about 700,000 nodes and 4,000,000 elements have been used. The simulation has been performed using the SLR combined to the tensorial method, for computing the surface tension term, and the multiscale ASGS technique to stabilize the $P1/P1$ velocity–pressure formulation. Regarding Fig. 42, the developed methodology allows the pressure field to be accurately
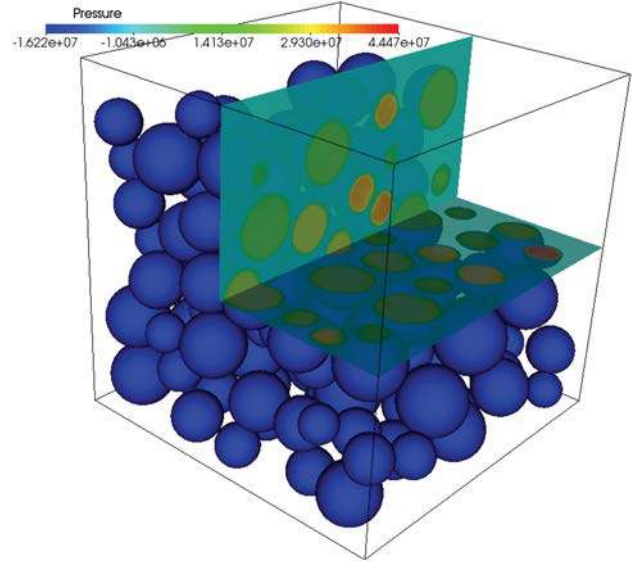
computed in this particles compact, with limited oscillations at the fluid–solid interface.

6.4 Volume Diffusion Velocity

Solving numerically for the matter velocity in volume diffusion requires two more difficulties to be sorted out. First, the pressure discontinuity at the solid-fluid interface does not allow to compute directly any pressure gradient although matter velocity depends on it, and second the mass balance must be fulfilled over time since no densification is activated during volume diffusion (see Table 1).

6.4.1 Pressure Discontinuity

Equation 25 shows that the lattice flux diffusion depends on the gradient of the pressure computed by solving previous mechanical problem. However, as demonstrated in Fig. 35b, since one single field is used to describe both fluid and solid pressure fields, this field exhibits a discontinuity at the interface, and consequently its gradient can not be computed in a straightforward way over this interface. This issue is overcome by computing the pressure gradient only inside the grains, up to a distance λ from the interface (see Fig. 43). The pressure normal gradient (or normal velocity) computed over $\{\alpha = -\lambda\}$ is subsequently projected onto the interface $\{\alpha = 0\}$. Providing the pressure p^n has been computed, these steps are performed by solving the following equation

$$\xi^n = \nabla p^n \cdot \mathbf{n}_\alpha \text{ if } \alpha < -\lambda \tag{110}$$

$$\nabla \xi^n \cdot \mathbf{n}_\alpha = 0 \text{ if } \alpha > -\lambda \tag{111}$$

where $\mathbf{n}_\alpha$ is given by Eq. (51) from the LS definition. Equation (110) is solved by a finite element method. At each time
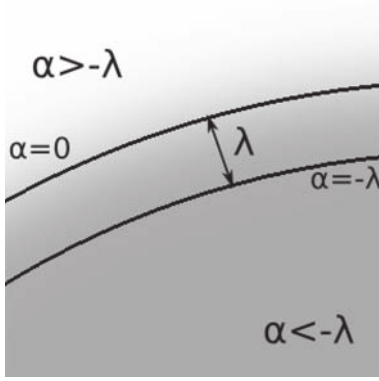
Fig. 43 ξ^n is directly computed from the gradient of the pressure in the region $\alpha < -\lambda$ (*darker grey region*) and convected in the outward normal direction in the region $\alpha > -\lambda$ (*lighter grey region*)

t_n, ξ^n is approximated by a continuous piecewise linear function defined over Ω. Since second expression in (110) is a stationary advection equation, a Galerkin approximation can not be applied, and the SUPG method (Eq. 72) is used. Once ξ^n is computed, the volume diffusion velocity is simply

$$\mathbf{v}_{v\alpha}^n = -C_v \xi^n \mathbf{N}_\alpha \tag{112}$$

with $C_v = (1-f)\Omega_0^a D_v/kT$, and $\mathbf{N}_\alpha$ is the piecewise linear vector obtained as the normalized nodal average of the $\mathbf{n}_\alpha$ vectors (which are constant per element).

6.4.2 Artificial Volume Conservation Velocity

The velocity computed to represent volume diffusion must also verify the mass balance in the system. Indeed, volume diffusion induced by matter transport from the surface to the neck does not induce any mass change, no densification in our case (see Table 1). If no particular attention is paid to this point, an artificial volume loss of some percents can be induced due to the numerical schemes.

This problem is also present when the grain boundary diffusion is considered. In literature different solutions are proposed, for example by Pan [20,21,52,63,65,66]. Some additional degrees of freedom are added to each particle corresponding to the translational and rotational velocities of a rigid body motion, in such a way that the volume of the particles is conserved. Another solution considering the sintering by surface and grain boundary diffusions of two particles of the same size [88], consists in the addition of a relative velocity between the two particles to represent the effect of the grain boundary diffusion and at the same time establish the coupling between the two diffusion mechanisms. Those solutions are more adapted for Lagrangian approaches where the nodes of the mesh are placed over the surface of the particles and the grain boundaries and mainly used in 2D prob-

lems. It would be complex to integrate them within the LS approach presented in this work as the velocity field should be computed over the vicinity of the interface and not only at the interface, and also their extension to 3D problems raises some additional challenges.

Another solution can be proposed. The idea consists in adding an artificial volume conservation velocity on the fluid-solid interface, intended to recover the volume loss/gain (ΔV) due to the diffusion phenomena [13,69]. This artificial volume conservation velocity $\mathbf{v^c}$ is defined in such a way that its magnitude is constant over the whole fluid-solid interface and it is oriented in the local normal direction of the particles surface:

$$\mathbf{v^c} = v^c \mathbf{n} \tag{113}$$

It can be easily shown that the volume change induced by any velocity field $v^c \mathbf{n}$ will be proportional to the fluid-solid surface S

$$\Delta \tilde{V} \approx v^c \Delta \tau S_{|t} \tag{114}$$

Then this correction, or artificial velocity, can be computed over the fluid-solid interface such that it induces a volume change to counterbalance the volume change ΔV due to matter diffusion $\mathbf{v}_{v\alpha}^n$ over a time step Δt

$$\Delta \tilde{V}_{|\Delta\tau} = -\Delta V_{|\Delta t}$$
$$= \int_{\Omega_s|_{t+\Delta t}} dV - V(t) \tag{115}$$

In this way, Eqs. (114) and (115) allow to find an expression for the artificial volume conservation velocity $\mathbf{v^c}$:

$$v^c = -\frac{\Delta V_{|\Delta t}}{\Delta \tau S_{|t}} \tag{116}$$

Considering this volume conservation velocity, some numerical tests are carried below.

6.5 Time-Stepping Strategy

Time-stepping strategy for the simulation of sintering by volume diffusion is summarized in Algorithm 3. It consists in computing the coupled fluid-solid mechanical equilibrium for pressure p^n, then deduce the pressure normal gradient ξ^n to compute the corresponding surface velocity $\mathbf{v}_{v\alpha}^n$, update the surface geometry, and then correct this velocity to enforce volume conservation.

6.6 Sintering Between Two Grains

In the following simulations, the computational domain is a cube or a square with a side length equal to 1μm. Table 7 summarizes the values of the different parameters used in the lattice diffusion simulations. The grain boundary is not

38

Algorithm 3 Time-stepping for volume diffusion

$n \leftarrow 0, \alpha \leftarrow \alpha_0$ (Eq. 49)
Initial mesh adaptation
while $t_n \leq \Theta$ **do**
    $p^n \leftarrow$ from fluid–solid coupled problem (93)–(94) with the "best" combination
    $\xi^n \leftarrow$ from Eq. (110)
    $\mathbf{v}^n_{\nu\alpha} \leftarrow$ from Eq. (112)
    $\alpha^{n+1} \leftarrow$ from Eq. (68) with $\mathbf{v} \leftarrow \mathbf{v}^n_{\nu\alpha}$
    $\mathbf{v^c} \leftarrow$ from Eq. (116)
    $\alpha^{n+1} \leftarrow$ from Eq. (68) with $\mathbf{v} \leftarrow \mathbf{v^c}$
    if necessary **then**
        mesh adaptation step
    end if
    $n \leftarrow n + 1$
end while

**Table 7** Parameters used in simulations of lattice diffusion

	Parameters
Mechanical Problem	$G = 156$ GPa, $K = 390$ GPa
System (87)	(Poisson's coefficient $\nu = 0.25$), $\eta = 1,000$ Pa.s
Laplace's law	$\gamma_{sf} = 0.9$, $\gamma_{gb} = 0$ N.m^{-1}
Diffusion Eq. (112)	$(1 - f)^2 \Omega_0^a D_v / kT = 0.013$ m^4.s^{-1}.N^{-1}
Time step	3.10^{-4} s $\leq \Delta t \leq 10^{-3}$ s

taken into account, we have therefore $\gamma_{gb} = 0$. Two remarks can be done. First, the value of the viscosity of the surrounding medium is very high, here $\eta = 1,000$ Pa.s. However, since the role of the surrounding fluid is only to transmit the stresses from the computational domain boundary to the grain surfaces, an accurate description of the dynamics of this medium is not of interest here. Consequently, the key parameter is not η but the ratio $\eta/(\Delta t G)$ which has to be small enough to guarantee that the surrounding medium does not perturbate the motion of the grain surfaces. The present ratio, lower than 2.10^{-5}, satisfies this condition. The second remark is that the parameter $(1 - f)\Omega_0^a D_v / kT$ (through the diffusion coefficient D_v) allows the time scale to be set. Here, the time is expressed in seconds. However, this choice is arbitrary.

The neck growth by lattice diffusion between two spherical grains of same radii is first analyzed. The geometrical models developed in the literature have been presented in the first section of this paper, they can be expressed under the form of power law (Eq. 33). For volume diffusion a common expression of the neck growth over time writes [73]

$$\left(\frac{x(t)}{R}\right)^n = \frac{B D_l \gamma \Omega_m}{RT \, R^4} t = B t^* \tag{117}$$

where B is a constant, $t^* = (D_v \gamma_{sf} \Omega_0^a)/(kT \, R^4)$ the dimensionless time, and the value of the parameter n stands between 4 et 5 according to [28] (Table 2). Figure 44 shows the growth
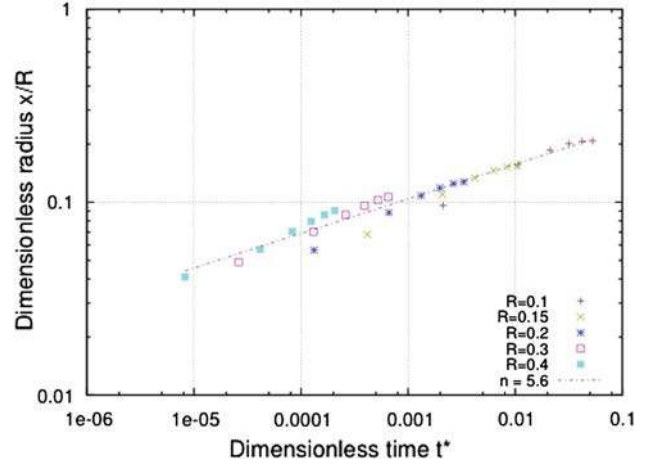

Fig. 44 Growth by lattice diffusion of the dimensionless neck radius x/R over dimensionless time $t^* =$ (logarithmic scale) for different values of R

of the dimensionless neck radius x/R versus t^*, computed by finite element for a grain radius ranging from 0.1 to 0.4. The best curve fitting these data, obtained by a least-square minimization, is $x/R = 0.36 t^{*1/5.6}$ and is referred to as "n = 5.6". Similarly to surface diffusion, this value which is larger than the upper bound predicted by the theory represents a sort of average value that takes into account the different stages of the sintering. However, when these simulations are examined individually, the coefficient n is shown to depend on the grain size and to vary slightly into each simulation. More precisely, the simulations provide a coefficient n that decreases when the grain size increases: n is equal to 4.85 when $R = 0.1$, to 4.23 when $R = 0.2$, to 4.14 when $R = 0.3$ and to 3.88 when $R = 0.4$. However, it has to be underlined that, even if it gives a good indication on the relevancy of the numerical approach, a strict comparison between simulations and expression (117) is not fully appropriate since this expression does not depend on material parameters. Indeed, expression (117) corresponds to the limit case where grains are considered as rigid bodies.

Figure 45 presents the change in time, by volume diffusion, of two spherical particles of different sizes. This simulation has been performed in 2D and the plane strain assumption is considered. The initial radii of the two particles are 0.25 and 0.1 μm, respectively. The mesh adaptation strategy is used to refine the mesh over a narrow band around the interface Γ_{sf}, as shown in Figs. 46a and 45c. The mesh is built up of about 55,000 elements.

The pressure field computed at the initial configuration is shown in Fig. 46a. After 250 time steps as described in Algorithm 3, the pressure field is shown in Figs. 46b and 45c. At that time ($t = 250 \Delta t = 0.1$ s), the neck between the particles is about 87 % of the radius of the smaller particle and as the curvature is lower then, as expected, the pressure
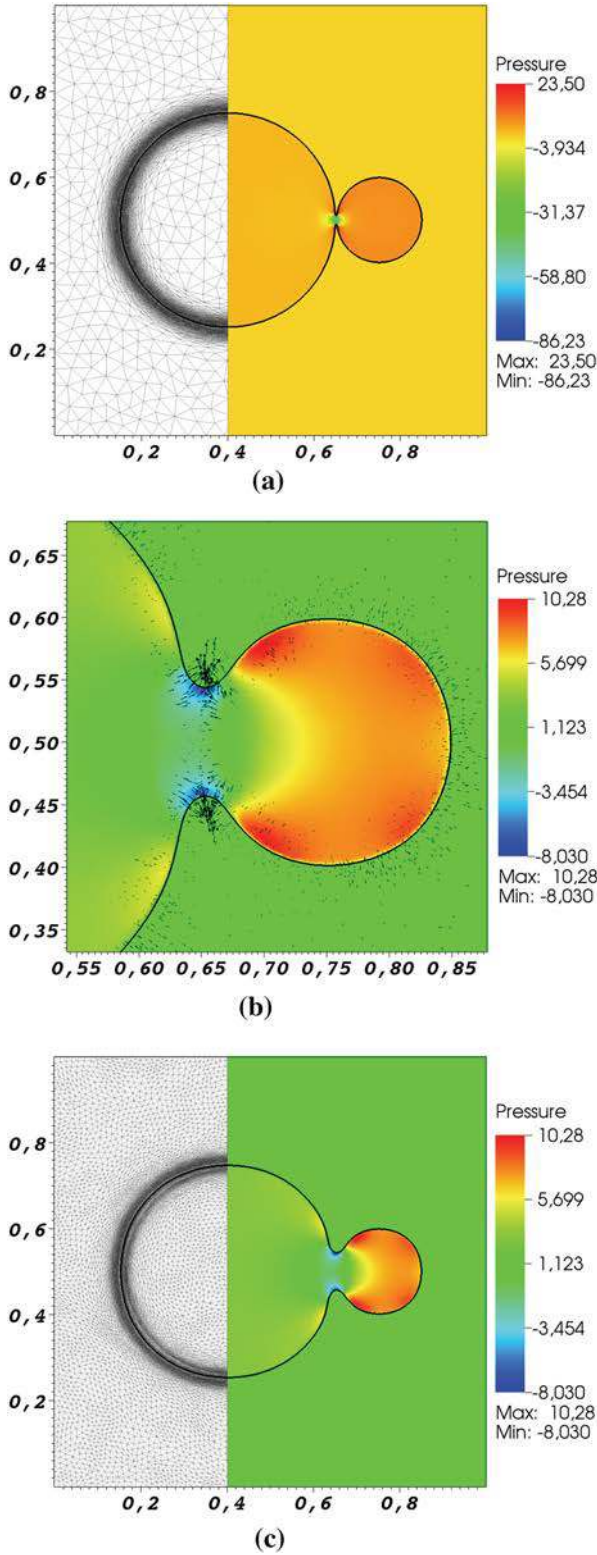
Fig. 45 2D simulation, under plane strains assumption, of volume diffusion for two cylindrical particles of different size: **a** and **c** mesh refined around the interface and pressure isovalues (MPa), respectively on the initial configuration and after 250 computation increments; **b** pressure field (MPa) and induced diffusion velocity after 250 computation increments

is not as strong as it was at the beginning. It is important to highlight that the change on the grain volume during the whole simulation is limited to 2.2×10^{-6} % which can be considered as zero.

## 7 Toward a Full Sintering Simulation

A numerical framework for the simulation of sintering by surface and volume diffusion has been established in previous sections. The third important diffusion route, the grain boundary diffusion, is discussed in Sect. 7.1. However, we shall see that the proposed numerical framework is not yet satisfactory to deal with this diffusion route. That is why Sect. 7.2 proposes to couple together only the surface and volume diffusion velocities, and Sect. 7.3 investigates the sintering of two grains under this mechanism. Finally, the simulation of sintering of a granular packing is shown in Sect. 7.4.

7.1 Grain Boundary Diffusion

According to Eqs. (26) and (30), the velocity associated with the grain boundary diffusion path is function of the second derivative of the normal stress along the grain boundary. The strategy presented in this paper is a first step toward the sintering simulation by grain boundary diffusion since the surface tension at the grain boundary can be taken into account in the mechanical problem (87). Hence, Fig. 46 shows the pressure field computed with non-zero surface tension both at the free surface and at the grain boundary.

However, some major difficulties are yet existing, and make that the numerical strategy allowing to compute the velocity due to grain boundary diffusion is yet to be developed. Among these difficulties, we can note that the description of two grains separated by a grain boundary, as shown in Fig. 46, requires two LS functions. More generally, one LS function has to be associated with each grain if one wants to describe grain boundaries (even if some optimization can be done using a graph coloration algorithm as explained in [41]). Beyond this number of LS functions, dealing with two level sets in contact still remains a difficult task: the triple junction is not geometrically well-defined when using a mesh, holes can be artificially created at the grain boundary and in the vicinity of the triple junction, grains can overlap. Some solutions are provided in references [56,80] but in a two-dimensional finite difference framework. Another difficulty is that the normal stress computed with our strategy is not smooth enough to obtain a good approximation of its Laplacian over the grain boundary.

For these reasons, simulation of sintering by grain boundary diffusion remains a challenging task for the future.

(a) Pressure field in the domain

(b) Pressure distribution along a horizontal cut
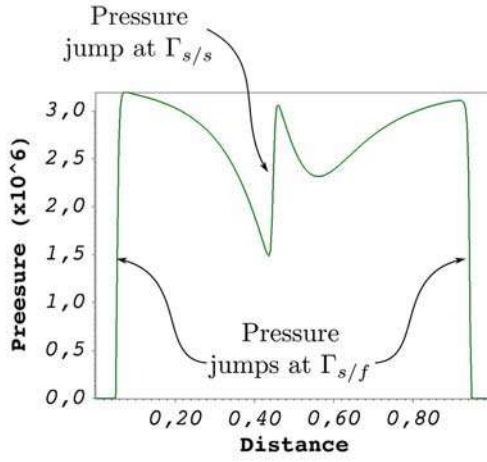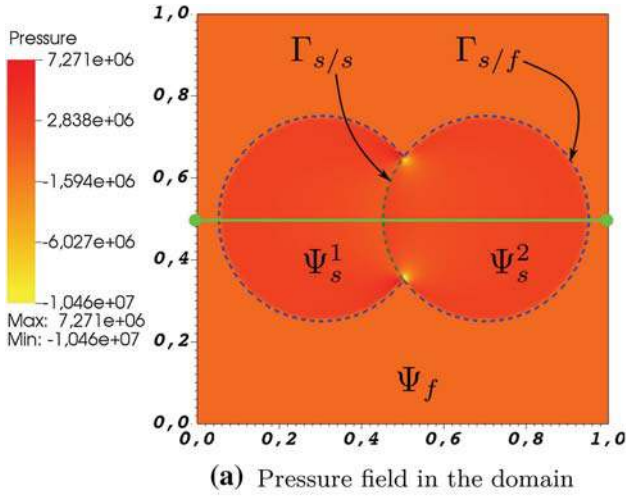
Fig. 46 Pressure field computed by taking into account the surface tension over Γ_{sf} and Γ_{gb} (grain boundary). **a** pressure field on the entire domain, and **b** pressure distribution along a straight line (*green line* in **a**)—pressure jumps across interfaces Γ_{sf} and Γ_{gb}. (Color figure online)

7.2 Coupling Surface and Volume Diffusion Routes

In order to couple together surface and volume diffusion routes, some hypotheses have to be made. The surface diffusion mass flux $\mathbf{j}_s$ and the volume diffusion mass flux $\mathbf{j}_v$ are assumed to be independent and therefore the corresponding velocities are also independent. Additionally, both diffusion mechanisms are supposed to occur simultaneously. In fact, those diffusion fluxes mainly depend on the geometry of the structure and more precisely on the curvature κ. The surface flux $\mathbf{j}_s$ is directly proportional to the surface gradient of the curvature and the volume flux is proportional to the pressure gradient which also depends on the curvature through the Laplace's law. A way to establish a coupling between the two diffusion paths consists in computing the coupled veloc-

Table 8 Material properties used

Property	Value	Units
$D_s \Omega_0^a \gamma_{sf} \delta_s / kT$	1×10^{-7}	m mol/s
$D_v \Omega_0^a (1 - f) / kT$	55.16	m mol/N s
K	260	GPa
G	156	GPa
η	1e-3	Pa/s
γ_{sf}	0.9	N/m
Ω_0^a	8.55×10^{-6}	m^3/mol

ity $\mathbf{v_{sv}}$ as the result of the vectorial addition of each individual diffusion velocity. This coupling velocity is obtained through a population balance when considering $\mathbf{j}_s$ and $\mathbf{j}_v$ as independent. Considering the expressions for surface diffusion (Eq. 29) and volume diffusion (Eq. 32) velocity, we have:

$$\mathbf{v_{sv}} = \mathbf{v_s} + \mathbf{v_v}$$
$$= (1 - f)\frac{\Omega_0^a}{kT}\left(D_s \delta_s \gamma_{sf} \Delta_s \kappa - (1 - f)D_v \nabla p \cdot \mathbf{n}\right) \mathbf{n}$$
$$(118)$$

The time-stepping strategy for the simulation of sintering by coupling surface and volume diffusions is given by Algorithm 4.

**Algorithm 4** Time-stepping for coupling surface diffusion and volume diffusion

---

$n \leftarrow 0, \alpha \leftarrow \alpha_0$ (Eq. 49), $A^0 \leftarrow \| \nabla \alpha_0 \|$
Initial mesh adaptation
while $t_n \leq \Theta$ **do**
$\quad \left(\kappa_\alpha^n, v_{s\alpha}^n\right) \leftarrow$ from Eq. (83)
$\quad \mathbf{v}_{s\alpha}^n \leftarrow$ from Eq. (84)
$\quad \xi^n \leftarrow$ from Eq. (110)
$\quad \mathbf{v}_{v\alpha}^n \leftarrow$ from Eq. (112)
$\quad \alpha^{n+1} \leftarrow$ from Eq. (68) with $\mathbf{v} \leftarrow \mathbf{v}_{v\alpha}^n + \mathbf{v}_{s\alpha}^n$
$\quad \mathbf{v^c} \leftarrow$ from Eq. (116)
$\quad \alpha^{n+1} \leftarrow$ from Eq. (68) with $\mathbf{v} \leftarrow \mathbf{v^c}$
$\quad$**if** necessary **then**
$\quad\quad$ mesh adaptation step
$\quad$**end if**
$\quad n \leftarrow n + 1$
end while

The material properties used in the following simulations correspond to the properties of Alumina (Al_2O_3) and are summarized in Table 8. It is important to recall that the diffusion related properties are significantly larger than the real properties of the material (because the time unit is not fixed), but this does not have any impact on the kinetics of the phenomena.
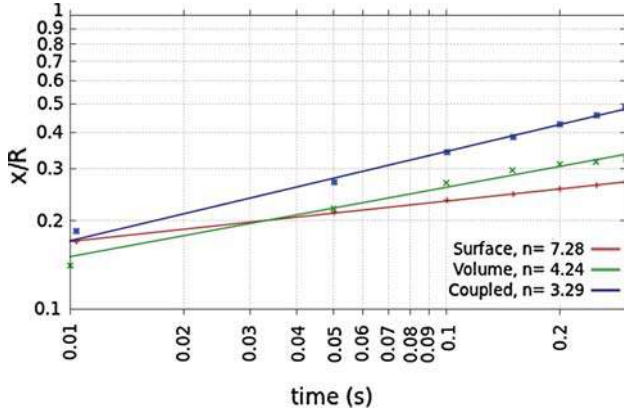
Fig. 47 Evolution of the dimensionless neck radius x/R over the time t (logarithmic scale)

7.3 Sintering Between Two Grains

Surface and volume diffusions are coupled to simulate the sintering of two particles by these two diffusion mechanisms. When the surface diffusion and volume diffusion mechanisms were presented, the case of two particles was used to validate the results obtained. As indicated several times throughout the presented approach, the analytical models for the neck growth between particles can be written under the form of power laws (Eq. 33) for any diffusion mechanism, with appropriate expressions of exponents n an dm (Table 2) and with B which stands for a physical parameters combination including radius particle R (see Eq. 85) for surface diffusion and Eq. (117) for volume diffusion).

Change in the dimensionless neck radius x/R for two particles of radius $R = 0.2 \ \mu$m is considered. First, the surface and the volume diffusion were considered alone. The change in the dimensionless neck radius x/R for these two diffusion mechanisms is plotted in Fig. 47, where the red and the green lines correspond to the surface and volume diffusions, respectively. As stated in previous Sects. 5.4 and **??**, the kinetics of these two diffusion mechanisms is well represented by the numerical approach developed. The evolution of the dimensionless neck radius x/R for the coupled diffusion is also plotted in Fig. 47 with a blue line. As expected the neck growth is significantly faster when the coupled diffusion is considered. By using a least square interpolation of the obtained data, the exponent corresponding the coupled diffusion is $n = 3.29$. Validation of this coupling is very complex since analytical models for the neck growth are not available for these two mechanisms working simultaneously. However, by considering the kinetics obtained, one may state that the results are qualitatively correct.

Figure 48 permits to compare the neck geometry, obtained at a same fixed time, provided by three different diffusion mechanisms: surface diffusion alone, volume diffusion alone, and the coupling between surface and volume diffu-
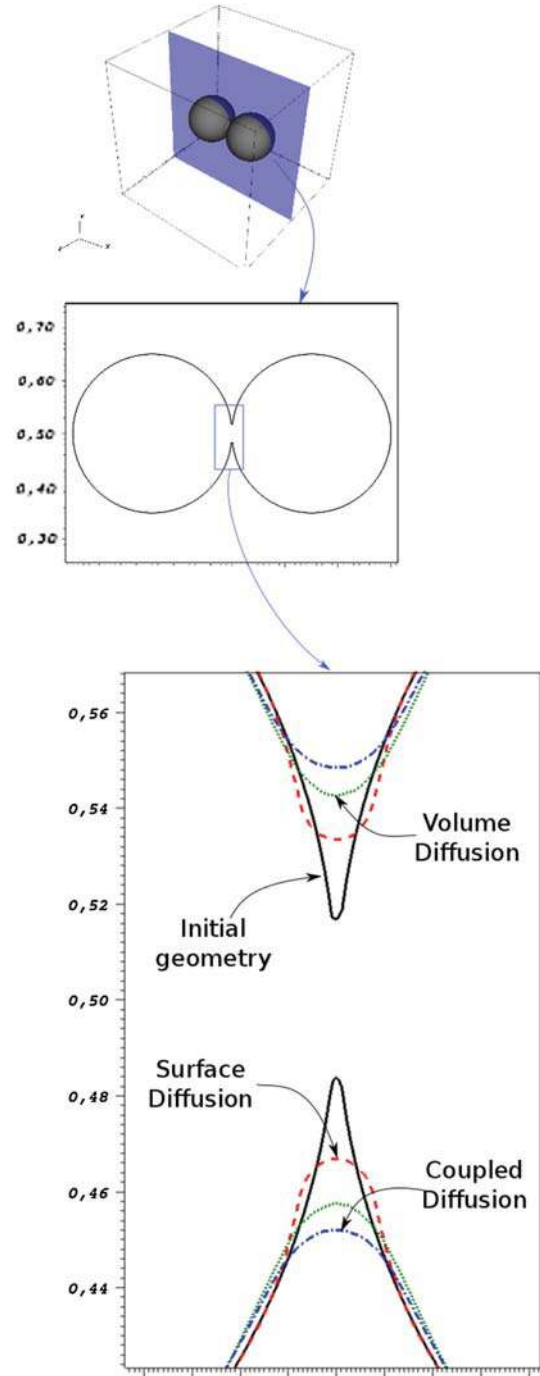
Fig. 48 Geometry of the neck between the particles after 65 time steps

sions. As expected, the neck grows significantly faster when surface and volume diffusion take place simultaneously.

7.4 Sintering of a Granular Packing

A sintering simulation by coupled surface and volume diffusions over a more realistic granular packing is presented. A

42

set of 154 particles with radii ranging from 0.0633 to 0.0797 μm is embedded into a computational domain given by a cube of side 1.2 μm. The material properties used are presented in Table 8. Figure 49a shows the initial packing as well as a cut of the refined mesh that is made up of about 2 millions nodes and about 11 millions tetrahedral elements.

The changes in structure under surface and volume diffusion is shown in Fig. 49a–d. In the initial geometry, grains are set to be quasi-tangent. As the volume diffusion takes place, the necks between the particles grow up to a point (Fig. 49d) where the particles can not be distinguished any more (recall that grain boundaries do not exist in this simulation).

One of the most important advantages of the numerical approach developed in this work is related to its capability to supply information about the local state of the structure at any time step. Furthermore, since the surface and volume diffusions are coupled together, the results obtained should get closer to the microstructures that can be obtained in real experiments, but still important differences are present. Specially, the grain boundary diffusion mechanism has a huge contribution among the diffusion paths and therefore it is not yet possible to make qualitative comparisons with the results available experimentally where all the diffusion mechanisms are activated. Nevertheless, important information can be extracted from this kind of simulation.

For example the grain packing presented in Fig. 49a was initially formed by a set of 154 particles with no closed porosity. As coupled diffusion takes place, the structure changes and closed porosity appears after about 250 time steps ($t \approx 0.2$ s). Figure 50a shows the first pore that appears inside the powder compact. In fact this pore evolves until a roughly spherical shape is reached (Fig. 50a–c). The structure is still changing, and after $520\Delta t$ it is possible to identify multiple pores of different sizes inside the powder compact, as can be seen in Fig. 50d.

As stated previously, neither the mass nor the density of the particles change during the sintering process, the volume of the grains must remain constant. Considering the simulation shown in Fig. 49, the change of total volume of the grains after 200 time steps is about 0.12 %, which is negligible (thanks to the correction velocity—Eq. 113). Because of this volume conservation, shrinkage effect results naturally from the microstructure changes which occur during the simulation. This simulation involves 550 time steps and has been performed in about 245h by using a parallel computing strategy on 24 cores.

8 Conclusion

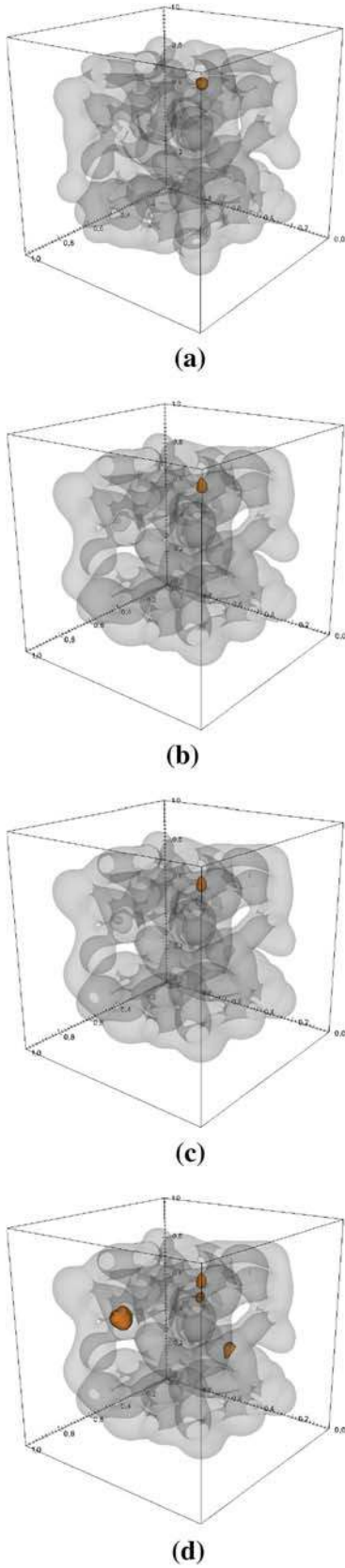Sintering is a very complex process involving several multiphysics phenomena. From a practical point of view there are many variables that have to be controlled in order to

(a)

(b)

(c)

(d)
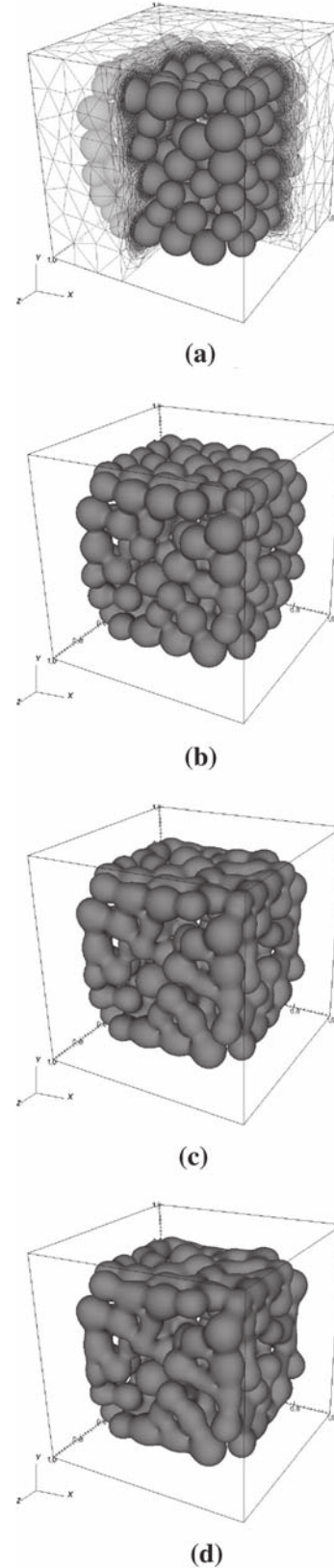
Fig. 49 Changes occurring in a granular packing under surface and volume diffusion. **a** $t = 0$ and adapted mesh; **b** $t = 65\Delta t$; **c** $t = 130\Delta t$; **d** $t = 170\Delta t$

43

Fig. 50 Closed porosity at different time steps. **a** $t = 270\Delta t$; **b** $t = 470\Delta t$; **c** $t = 520\Delta t$; **d** $t = 520\Delta t$ (all the pores developed)

obtain the desired properties of the final product. Because of these many variables and their interdependency, it is difficult to extract useful information from experimental data. Therefore numerical simulations represent a powerful tool that can provide meaningful information about this phenomena.

Considering the numerical tools available aiming at the simulation of the sintering process at the particle scale, a lack of a numerical approach able to handle the different diffusion mechanisms, complex geometries, and strong topological changes in 2D and even more drastically in 3D becomes obvious. In this work was developed a numerical approach able to integrate efficient simulations of sintering by multiple diffusion mechanisms at the grain scale, allowing to study the changes occurring into the representative elementary volume of a powder compact. The LS method which is an Eulerian-based approach was chosen because of its capability to handle strong topological changes in 2D and specially in 3D without any kind restriction concerning the geometry and the evolution of the system.

Within this LS framework, a numerical strategy, based on stabilized mixed finite elements, was developed for simulating sintering both by surface and volume diffusions. A characteristic of the proposed approach for volume diffusion, is to express the mass flux with respect to the gradient of pressure. Hence, contrary to the literature where grains are considered as rigid bodies, grains were assumed to behave here as elastic bodies. Consequently, a finite element analysis of the mechanical problem coupling elastic bodies with a surrounding fluid medium through Laplace's law was addressed.

Both surface and volume diffusion simulations were compared individually with success to usual geometrical models of two grains. However, the interest of the proposed approach is that it allows to cope with the severe topological changes and complex geometries that characterize the sintering process. Hence, combining surface and volume diffusion routes, the simulation of sintering of a 3D granular packing involving more than 150 grains was presented. During this simulation, the structure, which is initially made up of tangential grains, changes until developing closed porosity.

It is important to highlight that this kind of simulations are computationally very expensive, especially in 3D. In fact, the 3D simulations of the 150 particle packing sintering were performed by using a mesh built-up of about ten millions of elements and 24 processors were used, which required a computational time of about 200 hours. It has also to be outlined that mesh adaptation technique is another key point of these simulations. Without this mesh adaptation, such simulations would not be accessible.

Outlook can be drawn from this work. Since the framework for the simulation of the grain boundary diffusion path has already been fixed, the most straight outlook is the introduction of this diffusion mechanism into the numerical approach. However, this still represents a challenging

task since the transport of multiple LS functions has to be handled and the normal stress over the grain boundary has to be computed in a more accurate way. Coupling between those three main diffusion mechanisms (surface, volume and grain boundary) could lead to comparisons with experimental data and calibrated powder compact sintering. Moreover, the microstructural evolution of the powder compact could be embedded into macroscopic models.

Additionally, this numerical tool also allows to deal with the sintering of multi-materials or the study of the sintering of doped powders. In fact, all the diffusion mechanisms are numerically controlled by the value of the material properties—i.e. diffusion coefficients, mechanical properties, surface tension coefficients—therefore it would be possible to evaluate different material properties from some physical considerations to represent the multi-materials sintering or the sintering of doped powders. That corresponds to a work in progress.

References

1. Adalsteinsson D, Sethian JA (1995) A fast level set method for propagating interfaces. J Comput Phys 118(2):269–277
2. Ashby MF (1974) First report on sintering diagrams. Acta Metallurgica 22(3):275–289
3. Aulisa E, Manservisi S, Scardovelli R, Zaleski S (2007) Interface reconstruction with least-squares fit and split advection in three-dimensional cartesian geometry. J Comput Phys 225(2):2301–2319
4. Ausas RF, Sousa FS, Buscaglia GC (2010) An improved finite element space for discontinuous pressures. Comput Methods Appl Mech Eng 199(17–20):1019–1031
5. Badia S, Codina R (2009) Unified stabilized finite element formulations for the stokes and the darcy problems. SIAM J Numer Anal 47(3):1971–2000
6. Badia S, Codina R (2010) Stabilized continuous and discontinuous galerkin techniques for darcy flow. Comput Methods Appl Mech Eng 199(25–28):1654–1667
7. Bordère S, Vincent S, Caltagirone JP (2010) Stochastic energetic approach devoted to the modeling of static two-phase problems dominated by surface tension. Comput Fluids 39(3):392–402
8. Brackbill JU, Kothe DB, Zemach C (1992) A continuum method for modeling surface tension. J Comput Phys 100(2):335–354
9. Braginsky M, Tikare V, Olevsky E (2005) Numerical simulation of solid state sintering. Int J Solids Struct 42(2):621–636
10. Brezzi F, Fortin M (1991) Mixed and hybrid finite elements methods. Springer-Verlag, New York
11. Brooks AN, Hughes TJR (1982) Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. Comput Methods Appl Mech Eng 32(1):199–259
12. Bross P, Exner HE (1979) Computer simulation of sintering processes. Acta Metallurgica 27(6):1013–1020
13. Bruchon J, Drapier S, Valdivieso F (2011) 3d finite element simulation of the matter flow by surface diffusion using a level set method. Int J Numer Methods Eng 86(7):845–861
14. Bruchon J, Pino-Muñoz D, Valdivieso F, Drapier S (2012) Finite element simulation of mass transport during sintering of a granular packing. Part i: surface and lattice diffusion. J Am Ceram Soc 95(8):2398–2405
15. Bruchon J, Pino-Muñoz D, Valdivieso F, Drapier S, Pacquaut G et al (2010) 3d simulation of the matter transport by surface diffusion within a level-set context. Eur J Comput Mech 19:281–292
16. Burger Martin, Hauíer Frank, Stöcker Christina, Voigt Axel (2007) A level set approach to anisotropic flows with curvature regularization. J Comput Phys 225(1):183–205
17. Cervera M, Chiumenti M, Codina R (2010) Mixed stabilized finite element methods in nonlinear solid mechanics: Part i: formulation. Comput Methods Appl Mech Eng 199(37–40):2559–2570
18. Chen IW, Hassold GN, Srolovitz DJ (1990) Computer simulation of final-stage sintering. II. Influence of initial pore size. J Am Ceram Soc 73(10):2865–2872
19. Chen LQ (2002) Phase-field models for microstructure evolution. Ann Rev Mater Res 32(1):113–140
20. Ch'ng HN, Pan J (2004) Cubic spline elements for modelling microstructural evolution of materials controlled by solid-state diffusion and grain-boundary migration. J Comput Phys 196(2):724–750
21. Ch'ng HN, Pan J (2005) Modelling microstructural evolution of porous polycrystalline materials and a numerical study of anisotropic sintering. J Comput Phys 204(2):430–461
22. T. Coupez (2006) Réinitialisation convective et locale des fonctions level set pour le mouvement de surfaces et d'interfaces. Journées Activités Universitaires de Mécanique-La Rochelle, 31 août et 1er septembre 2006
23. Coyajee ERA (2007) A front-capturing method for the numerical simulation of dispersed two-phase flow. Ph.D. Thesis, Delft University of Technology
24. Daly BJ, Pracht WE (1968) Numerical study of density-current surges. Phys Fluids 11:15
25. Digonnet H, Silva L, Coupez T (2007) Cimlib: a fully parallel application for numerical simulations based on components assembly. In: Materials processing and design: modeling, simulation and applications, vol. 908, pp 269–274. American Institute of Physics, 2 Huntington Quadrangle, Suite 1 NO 1, Melville, NY, 11747–14502
26. Djohari H, Derby JJ (2009) Transport mechanisms and densification during sintering. II. Grain boundaries. Chem Eng Sci 64(17):3810–3816
27. Djohari H, Martínez-Herrera JI, Derby JJ (2009) Transport mechanisms and densification during sintering. I. Viscous flow versus vacancy diffusion. Chem Eng Sci 64(17):3799–3809
28. Exner HE, Arzt E, Cahn RW, Haasen P (1996) Sintering processes. Physical metallurgy, 4th edn. North-Holland, Oxford, pp 2627–2662
29. Fortin A, Béliveau A, Demay Y (1998) A two-dimensional numerical method for the deformation of drops with surface tension. Int J Numer Methods Fluids 10(3):225–240
30. Garikipati K, Bassman L, Deal M (2001) A lattice-based micromechanical continuum formulation for stress-driven mass transport in polycrystalline solids. J Mech Phys Solids 49(6):1209–1237
31. Gerlach D, Tomar G, Biswas G, Durst F (2006) Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows. Int J Heat Mass Transf 49(3–4):740–754
32. German RM, Lathrop JF (1978) Simulation of spherical powder sintering by surface diffusion. J Mater Sci 13(5):921–929
33. German RM (1996) Sintering theory and practice. Wiley, New York
34. Geuzaine C, Remacle J-F (2009) Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. Int J Numer Methods Eng 79(11):1309–1331
35. Gibbs JW (1928) The collected works of J. Willard Gibbs, vol 1. Longmans, Green, New York
36. Groß S, Reusken A (2007) An extended pressure finite element space for two-phase incompressible flows with surface tension. J Comput Phys 224(1):40–58

37. Harlow FH, Welch JE et al (1965) Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Phys Fluids 8(12):2182

38. Hassold GN, Chen IW, Srolovitz DJ (1990) Computer simulation of final-stage sintering: I, model kinetics, and microstructure. J Am Ceram Soc 73(10):2857–2864

39. Herring C (1951) Surface tension as a motivation for sintering. Phys Powder Metall 27(2):143–179

40. Hirt CW, Nichols BD (1981) Volume of fluid (VOF) method for the dynamics of free boundaries. J Comput Phys 39(1):201–225

41. Hitti K (2011) Direct numerical simulation of complex representative volume elements (RVEs): generation, resolution and homogenization. Ph.D. thesis, Ecole Nationale Supérieure des Mines de Paris

42. Hoge CE, Pask JA (1977) Thermodynamic and geometric considerations of solid state sintering. Ceramurgia Int 3(3):95–99

43. Holm EA, Glazier JA, Srolovitz DJ, Grest GS (1991) Effects of lattice anisotropy and temperature on domain growth in the two-dimensional potts model. Phys Rev A 43(6):2662

44. Howard RE, Lidiard AB (1964) Matter transport in solids. Rep Prog Phys 27:161

45. Hughes TJR (1987) Recent progress in the development and understanding of supg methods with special reference to the compressible euler and navier-stokes equations. Int J Numer Methods Fluids 7(11):1261–1275

46. Hughes TJR, Sangalli G (2007) Variational multiscale analysis: the fine-scale Green's function, projection, optimization, localization, and stabilized methods. SIAM J Numer Anal 45(2):539–557

47. Thomas JR (1995) Hughes. Multiscale phenomena: Green's functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. Comput Methods Appl Mech Eng 127(1–4):387–401

48. Hughes TJR, Feijóo GR, Mazzei L, Quincy J-B (1998) The variational multiscale method-a paradigm for computational mechanics. Comput Methods Appl Mech Eng 166(1–2):3–24

49. Hysing S (2006) A new implicit surface tension implementation for interfacial flows. Int J Numer Methods Fluids 51(6):659–672

50. Hysing S (2012) Mixed element FEM level set method for numerical simulation of immiscible fluids. J Comput Phys 231(6):2449–2465. doi:10.1016/j.jcp.2011.11.035

51. Job G, Herrmann F (2006) Chemical potential: a quantity in search of recognition. Eur J Phys 27:353

52. Kucherenko S, Pan J, Yeomans JA (2000) A combined finite element and finite difference scheme for computer simulation of microstructure evolution and its application to pore-boundary separation during sintering. Comput Mater Sci 18(1):76–92

53. Kuczynski GC (1949) Self-diffusion in sintering of metallic particles. Trans AIME 185:169–178

54. Lafaurie B, Nardone C, Scardovelli R, Zaleski S, Zanetti G (1994) Modelling merging and fragmentation in multiphase flows with SURFER. J Comput Phys 113(1):134–147

55. Martin CL, Bouvard D, Shima S (2003) Study of particle rearrangement during powder compaction by discrete element. J Mech Phys Solids 51(4):667–693

56. Merriman B, Bence JK, Osher S (1994) Motion of multiple junctions: a level set approach. J Comput Phys 112(2):334–363

57. Mesri Y, Zerguine W, Digonnet H, Silva L, Coupez T (2008) Dynamic parallel adaption for three dimensional unstructured meshes: Application to interface tracking. In: Proceedings of the 17th International Meshing Roundtable, Springer, pp 195–212

58. Nichols FA, Mullins WW (1965) Morphological changes of a surface of revolution due to capillarity-induced surface diffusion. J Appl Phys 36:1826–1835

59. Nichols FA, Mullins WW (1965) Morphological changes of a surface of revolution due to capillarity induced surface diffusion. J Appl Phys 36(6):1826–1835

60. Noh W, Woodward P (1976) Slic (simple line interface calculation). In: Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28–July 2, 1976 Twente University, Enschede, Springer, pp 330–340

61. Osher S, Fedkiw RP (2001) Level set methods: an overview and some recent results. J Comput Phys 169(2):463–502

62. Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations. J Comput Phys 79(1):12–49

63. Pan J, Ch'ng HN, Cocks ACF (2005) Sintering kinetics of large pores. Mech Mater 37(6):705–721

64. Pan J, Cocks ACF (1995) A numerical technique for the analysis of coupled surface and grain-boundary diffusion. Acta Metallurgica et Materialia 43(4):1395–1406

65. Pan J, Cocks ACF, Kucherenko S (1997) Finite element formulation of coupled grain-boundary and surface diffusion with grain-boundary migration. In: Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 453(1965):2161–2184

66. Pan J, Le H, Kucherenko S, Yeomans JA (1998) A model for the sintering of spherical particles of different sizes by solid state diffusion. Acta Materialia 46(13):4671–4690

67. Papadakis G (2008) A novel pressure-velocity formulation and solution method for fluid–structure interaction problems. J Comput Phys 227(6):3383–3404

68. Peng D, Merriman B, Osher S, Zhao H, Kang M (1999) A pde-based fast local level set method. J Comput Phys 155(2):410–438

69. Pino-Muñoz D (2012) High-performance computing of sintering process at particle scale. Ph.D. Thesis, École Nationale Supérieure des Mines de Saint-Étienne

70. Pino Muñoz D, Bruchon J, Drapier S, Valdivieso F (2013) A finite element-based level set method for fluid-elastic solid interaction with surface tension. Int J Numer Methods Eng 93(9):919–941

71. Popinet S, Zaleski S (1999) A front-tracking algorithm for accurate representation of surface tension. Int J Numer Methods Fluids 30(6):775–793

72. Qiu F, Egerton TA, Cooper IL (2008) Monte carlo simulation of nano-particle sintering. Powder Technol 182(1):42–50

73. Rahama MN (1995) Ceramic processing and sintering. Marcel Dekker, New York

74. Renardy Y, Renardy M (2002) PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method. J Comput Phys 183(2):400–421

75. Rider WJ, Kothe DB (1998) Reconstructing volume tracking. J Comput Phys 141(2):112–152

76. Riedel H, Zipse H, Svoboda J (1994) Equilibrium pore surfaces, sintering stresses and constitutive equations for the intermediate and late stages of sintering-ii. diffusional densification and creep. Acta Metallurgica et Materialia 42(2):445–452

77. Rudman M (1997) Volume-tracking methods for interfacial flow calculations. Int J Numer Methods Fluids 24(7):671–691

78. Shin S, Juric D (2002) Modeling three-dimensional multiphase flow using a level contour reconstruction method for front tracking without connectivity. J Comput Phys 180(2):427–470

79. Silva L, Puaux G, Vincent M, Laure P (2010) A monolithic finite element approach to compute permeabilityatc microscopic scales in LCM processes. Int J Mater Form 3(1):619–622

80. Smith K, Solis F, Chopp D (2002) A projection method for motion of triple junctions by level sets. Interfaces Free Boundaries 4(3):263–276

81. Smolianski A (2005) Finite-element/level-set/operator-splitting (FELSOS) approach for computing two-fluid unsteady flows with free moving interfaces. Int J Numer Methods Fluids 48(3):231–269

82. Sussman M, Smereka P, Osher S (1994) A level set approach for computing solutions to incompressible two-phase flow. J Comput Phys 114(1):146–159

83. Tikare V, Braginsky M, Bouvard D, Vagnon A (2010) Numerical simulation of microstructural evolution during sintering at the mesoscale in a 3D powder compact. Comput Mater Sci 48(2):317–325

84. Tikare V, Braginsky M, Olevsky EA (2003) Numerical simulation of solid-state sintering: I, sintering of three particles. J Am Ceram Soc 86(1):49–53

85. Uskoković DP, Exner HE (1977) Kinetics of contact formation during sintering by diffusion mechanisms. Sci Sinter 9(3):265–303

86. Ville L, Silva L, Coupez T (2011) Convected level set method for the numerical simulation of fluid buckling. Int J Numer Methods Fluids 66(3):324–344

87. Vincent S, Caltagirone JP, Lubin P, Randrianarivelo TN (2004) An adaptative augmented lagrangian method for three-dimensional multimaterial flows. Comput Fluids 33(10):1273–1289

88. Wakai F, Brakke KA (2011) Mechanics of sintering for coupled grain boundary and surface diffusion. Acta Materialia 59(14):5379–5387

89. Wang YU (2006) Computer modeling and simulation of solid-state sintering: a phase field approach. Acta Materialia 54(4):953–961

90. Weaire D, Kermode JP, Wejchert J (1986) On the distribution of cell areas in a Voronoi network. Philos Mag B 53(5):101–105

91. Williams MW, Kothe DB, Puckett EG (1998) Accuracy and convergence of continuum surface tension models: fluid dynamics at interfaces. Cambridge University Press, Cambridge

92. Wu FY (1983) Erratum: the potts model. Rev Mod Phys 55(1):315

93. Youngs DL (1982) Time-dependent multi-material flow with large fluid distortion. Numer Methods Fluid Dyn 24:273–285