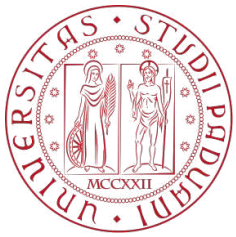


Università degli Studi di Padova

Dipartimento di Ingegneria Industriale
Corso di Laurea in Ingegneria Aerospaziale

Tesi di Laurea

Numerical study of the breakup of a porous medium due to fluid dynamic forces

Candidato
Roberto Dal Monte
Matricola: **2054474**

Relatore
Federico Dalla Barba

Anno accademico 2022-2023
20/10/2023

TESI

Roberto Dal Monte

July 2023

Abstract

The present manuscript treats the breakup of porous media due to the action of fluid dynamic forces. The problem of the interaction between solid structures and fluid flows, named Fluid-Structure Interaction (FSI), is a problem that interests a wide range of engineering applications and scientific fields, from aerospace, civil and biomedical engineering to geotechnics and planetary sciences. This phenomenon is often associated with hydraulic fracture due to fluid-dynamic forces, which act on immersed solids. To reproduce the phenomenon correctly fluid dynamics, solid mechanics and fracture mechanics have to be simultaneously considered. The major difficulty of the classical theories of solid mechanics is to predict and simulate the formation of cracks due to the arising of singularities in the derivatives of partial differential equations where discontinuities form.

The present manuscript intends to employ a novel numerical method to address FSI with solid fracturing of a porous medium, where solid and fracture mechanics are simulated with peridynamic, a well-established reformulation of continuum theory which, replacing partial differential equations with integral ones, intrinsically accounts for crack formation and branching. Instead, the dynamic of the fluid phase is reproduced by the three-dimensional incompressible formulation of the Navier-Stokes equations by using Direct Numerical Simulations (DNS). Then, the Immersed Boundary Method (IBM) is used to impose wall boundary conditions on the fluid-solid interfaces.

The simulations are run by using a massively parallel solver, written in Fortran 90 extended with a Message-Passing Interface (MPI) standard, which has been previously developed and validated. In the present manuscript, this new numerical tool has been employed to simulate the deformation and fragmentation of a linear-elastic porous medium. Three different simulations have been performed: the first reproduces the deformation of solid without fracture, the second with partial fracture and the third with total fracture. Then the comparisons between stress and strain distributions, before and after the fracturing process, are reported and discussed for all the three cases listed above. Furthermore, for these distributions, a failure criterion, which tries to predict when the fracture occurs, is presented. In addition, for all cases, the trend over time of the pressure drop, change in porosity and permeability are reported and discussed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With Fluid-Structure-interaction (FSI) term is considered a strongly non-linear problem that involves the intricate coupling between the governing equations of the fluid dynamics and solid mechanics where also the phenomenon of hydraulic fracture is taken into account. In FSI, solid and fluid phase interact with each other via the exchange of momentum through their interfaces which evolve in time, such that the result is a strong coupling of the dynamic of the fluid with that of the solids involved in the process.
FSI is a problem that interests a wide range of engineering applications and scientific fields, from aerospace, civil and biomedical engineering to geotechnics and planetary sciences. In the field of aerospace engineering for example, the problem is important for the modeling of aircraft wings and turbine blades [10, 31]. A typical example consists of the aeroelastic flutter [24], a dynamic and self-sustained instability of an elastic structure in a fluid flow, caused by positive feedback between the body's deflection and the forces exerted by the fluid flow. This condition can arise for the aircraft wings during normal flight conditions and the stress field in the wings may grow to such an extent that local damages, or even global failure, occur. Another example is represented by the rupture of ablative materials, which are usually porous, used for cooling the tanks of liquid propulsion space launch vehicles. It is clear how this problem is encountered in several critical areas and it must be accurately predicted during the designed process.
To achieve a full comprehension of the physics of FSI problems with flow-induced fracturing, fluid dynamics, solid mechanics and fracture mechanics must be taken into account at the same time. Since the aim of the present work is to reproduce the deformation and breakup of a porous medium due to fluid-dynamic forces, in the first chapter of the thesis a brief description of the physical properties of porous media are presented. Then the limits of local continuum theory are highlighted, since they are not reliable in the prediction of crack initiation and propagation in solid medium, due to the arising of singularities of derivatives in partial differential equations of solid mechanics. The most popular numerical techniques used to tackle numerically the problem are also presented

and it is shown how most of them lack of generality and are limited to very restricted cases, since they are also characterized by a poor description of the local fluid dynamics. This part is used as a preamble for the description of the novel numerical method, used in this thesis, where solid and fracture mechanics are addressed by employing peridynamics [7], a well-established reformulation of continuum theory that intrinsically accounts for crack initiation and branching, whereas the physics and dynamics of fluid phase is fully resolved by employing the three-dimensional Direct Numerical Simulation (DNS) of the incompressible Navier-Stokes equations. The *multi-direct forcing Immersed Boundary Method (IBM)* [14, 2] is used to prescribe wall boundary conditions on the fluid-solid interfaces, whereas *normal probe method* [20] is employed to compute the stress on the wet surfaces of the immersed solids.
In peridynamic theory the arising of singularities is avoided since the expression of governing laws of solid mechanics with partial differential equations is replaced by employing integral equations which allow to solve discontinuities avoiding the onset of singularities. This approach permits to describe crack formation and branching without using any empiric or macroscopic model. The work presented in this thesis is based on the so-called *bond-based peridynamic model* [7, 21, 17], which uses constitutive relations for *Prototype Micro-elastic Brittle Material*. The IBM [22, 12, 2, 14] is used to couple fluid and solid dynamics. The basic idea of the method is to prescribe no-slip and no-penetration boundary conditions on fluid-solid interfaces by applying to the discrete governing equations of the fluid flow, which is discretized using a fixed and structured computational grid that does not conform to the solid-fluid interfaces, a proper force distribution that set to zero the local relative velocity between the two phases in the vicinity of their interface, so that the effect of the boundaries is reproduced.
The source code is written in Fortran 90 extended with the Message-Passing Interface (MPI), a standardized and portable message-passing standard designed to function on parallel computing architectures. The code is composed of three different modules: a fluid solver, a peridynamic solver and a third module that manages the synchronization and the coupling of the Navier-Stokes and peridynamic equations via the IBM. The descriptions of their numerical implementations are reported. Finally the numerical tool has been employed for the simulation of a viscous three-dimensional incompressible channel-flow, with Poiseuille flow as inlet condition, that invests a solid porous medium located inside the channel. Three different simulations have been performed, which differ for the value of the *critical fracture energy release rate* of the material: the first reproduces the deformation of the solid without fracture, the second with partial fracture and the third with total fracture. For all cases listed above have been performed analyses of stress and strain distributions, before and after the fracturing process, trying to find a failure criterion which allows to predict when, basing on these distributions, the crack formation starts. Furthermore the pressure drop, the changing in porosity and permeability of the porous medium over time have been investigated and discussed.

Chapter 2

Overview of the subject

The interaction of a deformable body submerged in a fluid flow is encountered in a wide variety of engineering applications and scientific fields. These may include the response of an aerodynamic surface invested by a fluid flow or the vibrations of the blades inside a compressor or a turbine, rather than an ablative material that breaks down due to the effect of a fluid-dynamic forces induced by a fluid flow. All these different cases are included in the fluid-structure interaction (FSI) problem that so involves separate disciplines. For this reason is necessary to address the problem with a general point of view as well as possible. Typically, simulations where fluid-structure interaction is involved, the solid material usually presents a linear elastic behaviour (Hooke's law) and, for large deformations, a more complex non-linear behaviour of the material can be taken into account. Instead, the fluid is usually considered as a laminar flow with low Reynolds number and constant fluid density (incompressible flow hypothesis).
Since the target of the thesis is to solve numerically the deformation and the rupture of a solid porous medium immersed in a fluid flow, to get a clearer idea of the subject matter, a general discussion of the physical properties of the porous media is reported below.

2.1 Porous medium

A porous medium is a solid material which features pores in its volume. The skeleton portion of the material is often called "matrix" or "frame" , whereas the pores form a void space through which a fluid or gas can flow. This void space is identified as empty in the overall dimensions of the body and is determined and continuous if every pore structure is interconnected, or isolated if volume of the material appears to be confined from the outer environment [23]. Being filled with fluid, porous media are typical FSI problems and critical components for a broad range of disciplines and studies for a geological processes, (e.g. erosion, seawater filtration and underground hydrology, including water

contamination [36, 44]), as well as the numerical modeling of the degradation porous filters and inserts employed in many industrial applications, for examples electrodes for fuel cells and flow batteries.
For a better understanding of the characteristics of a porous medium a summary description of its main aspects is presented below.

2.1.1 Physical properties

Porous media are usually characterized by a random distribution of pores with different size and shape resulting in a void space of the medium extremely chaotic. In this way, the paths that the solid medium provides to the fluid when it is immersed in the latter are not straight but full of turns and crosses with other passages, so the fluid which flows through these passages is forced to travel way longer than its actual effective displacement considered as the length of the porous medium dimension along the macroscopic flow direction. For these reasons, to simplify the characterization of porous media, these kinds of solids are handled without completely taking into account their complex internal morphology [25]. Hence, the main parameters characterizing each porous medium, and eventually the conservation laws of the fluid flow inside its cavities, are averaged over a scale long enough to consider the segment homogeneous.
The most important parameter is *porosity* (ϕ) or void fraction. It's an intrinsic property of every porous medium and it is defined as the ratio between the volume of the pores and the total volume occupied by the medium [30]:

$$\phi = \frac{V_p}{V_m} \,, \tag{2.1}$$

where V_p is the volume of the void space, which is computed as the sum of the volumes of all the cavities presented inside the medium, whereas V_m is the total volume of the medium. In this case the void volume takes into account all the void space generated by interconnected pores; however, if the case of a porous medium immersed in a fluid flow is considered, most of the times not all the pore network is accessible for the fluid phase and so it is necessary to define the effective porosity, which considers only the void space that contributes to the fluid flow inside the solid. Porosity ϕ is a dimensionless parameter, which is included in $0 < \phi < 1$, where 0 means a solid medium without cavities and 1 indicates that there is not present a solid, so it refers to the condition of undisturbed flow. The higher is the volume filled by solid matrix, hence the lower is the porosity value, the flow struggles to pass through. In fact in this case, the flow will slow down and dissipates more of its energy, since a large contact surface is offered, increasing the overall drag and pressure drop.
Furthermore, different types of porosity exist: *convex porosity* and *non-convex porosity*. The first one is obtained by the formation of cavities inside the solid frame, forming sponge-like structures. These cavities usually have regular shape (for example spherical) and solid media which present this type of porosity usually have a less value of void space volume. The second one is typical of material assembled by progressive stacking of grains and particles of various

form and size, such as sand or concrete. The figure below displays a comparison between portions of porous media with *convex porosity* and *non-convex porosity*, respectively.

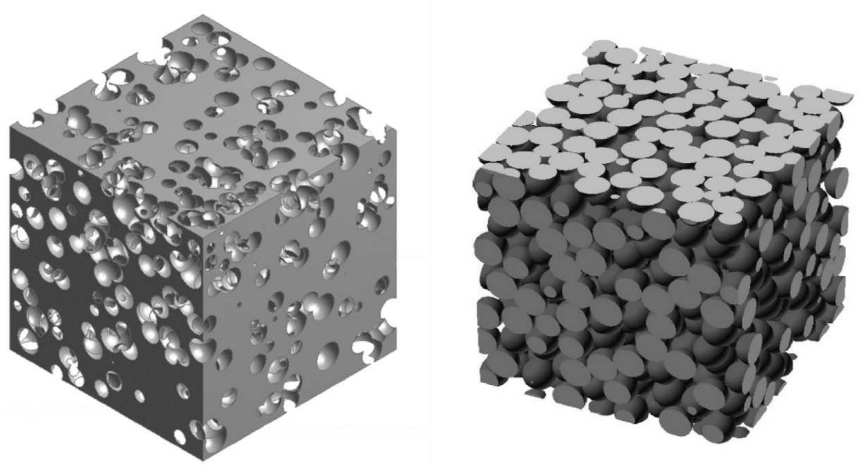


Figure 2.1: Representation of a portion of a solid with convex porosity (left side) and a portion of a solid with non-convex porosity (right side). Figure taken from [51].

As it is possible to see in the figure just presented, for the case of *non-convex porosity* cavities have not regular shape. For this reason, when dealing with *non-convex porosity*, is helpful to include an additional parameter, which defines the irregularity of granular particle: *sphericity*. The sphericity, ψ, of a particle is expressed as the ratio of the surface area of a sphere, with the same volume of the given particle, to the surface area of the particle [36]:

$$\psi = \frac{A_s}{A_p} = \frac{\pi^{1/3}(6V_p)^{2/3}}{A_p}\,, \tag{2.2}$$

where V_p and A_p are the volume and the surface of the considered particle, respectively; whereas A_s is the surface area of the sphere with the same volume of the given particle. Sphericity influences the value of porosity. It was proven by Kerimov et al. [36] that the lower is the average sphericity of the grains in a non-convex material, the lower is porosity.
Another important parameter for the description of the physical properties of a generic porous medium is the *permeability* (k) which is a measurement of the resistance encountered by the fluid when it crosses the porous medium. In the present work it is computed from petrophysical formulation of Darcy's law which is expressed as:

$$Q = \frac{kS}{L}\frac{\Delta p}{\mu}\,, \tag{2.3}$$

where Q is the flow rate of the fluid, S is the section through the fluid flows, k is the permeability, μ is the dynamic viscosity of the fluid and Δp is the pressure difference between two sections separated by a distance L along the channel. Therefore the *permeability* is defined as:

$$k = \frac{QL}{S}\frac{\mu}{\Delta p}\,. \tag{2.4}$$

It's clear from equation (2.4) that high values of *permeability* are related to a less resistance encountered by the fluid when it crosses the porous medium since $k \propto 1/\Delta p$.
Similarity to porosity, it was proven how irregularities in the grains' shape contribute to lower value of k, increasing the overall pressure drop.
In addition, always referring to the case of a porous medium immersed in a fluid, when the fluid flows through the cavities of the medium, the solid phase deflects the fluid extending its route inside the medium. To correct the effective flow path, and so to further predict the permeability of porous media, especially one presenting convex cavities, a new parameter, called *tortuosity*, is employed. Referred to this, there are different types of tortuosity depending on the phenomenon described. Since our field of interest involves fluid-structure interaction and crack propagation due to the fluid pressure, only the hydraulic definition of tortuosity is described.
Hydraulic tortuosity (τ_h) was introduced by Carman (1937) and is defined as the ratio of the effective hydraulic flow path L_h to the straight line distance in the direction of movement L [44]:

$$\tau_h = \frac{L_h}{L}\,, \tag{2.5}$$

hence, from its definition, it can be interpreted as a parameter which describes the average stretching of the flow path in a porous material respect to a non-disrupted flow. The length L_h is usually measured from the streamlines in a condition of a steady-state where the flow behaviour is unchanging with time. Still, identifying a streamline inside a porous frame is a very difficult process, so during the years some computation methods have been proposed to solve the problem. For instance, Zhang and Knackstedt (1995), for approximating the travel of a fluid particle inside a porous medium, evaluated a weight-average of all stream-lines traversing the medium, using as weight the corresponding time requested for a single flow particle to move along its entire flow path:

$$\tau_h = \frac{1}{L}\left(\frac{\sum_i w_i l_i}{\sum_i w_i}\right) = \frac{1}{L}\left(\frac{\sum_i l_i t_i^{-1}}{\sum_i t_i^{-1}}\right), \tag{2.6}$$

where l_i, w_i, and t_i are respectively the single streamline length, the related weight and the travel time for a particle. The equation 2.6, as it was later proven by Matyka and Koza (2012), for the case of laminar flow with low Reynolds number, can be written using the instantaneous particle velocity v_i obtaining:

$$\tau_h = \frac{\int_B v \mathrm{d}V}{\int_B v_x \mathrm{d}V} = \frac{\bar{v}}{\bar{v_x}}\,, \tag{2.7}$$

where B is the porous domain saturated by the fluid, v the particle velocity and v_x the particle velocity along the macroscopic flow direction. This equation is applicable under the assumption of incompressible flow.

All these parameters discussed above have been employed in mathematical models (e.g. Darcy's law, Kozeny-Carman equation, poroelasticity theory proposed by Biot, capillary model) which have been used over the years for trying to describe the behaviour of the fluid that flows through the cavities of a porous medium and consequently the interaction between the fluid phase and the solid one. On the other hand, in the present work the incompressible Navier-Stokes equations are numerically solved by employing a three dimensional Direct Numerical Simulation (DNS) technique which, thanks to a very dense numerical grid, allows to solve completely the dynamic of the fluid, also for the one inside the interstitial regions of the medium. Furthermore, by employing the *multi-direct forcing Immersed Boundary Method* (IBM) for prescribing no-slip and no-penetration boundary conditions and *normal-probe method* for evaluating fluid-dynamic forces acting on the solid surfaces, the need of other mathematical models, for reproducing the dynamic of the fluid inside the cavities of the medium and the interaction between the fluid phase and solid one, is removed. For these reasons, once the porous medium is discretized, the employment of the parameters discussed above is useless for reproducing the fluid dynamic through the porous medium, but it will be investigated how their values change after the breakup of the medium.

Chapter 3

Methodology

In this chapter are firstly discussed the limits of the most popular numerical methods used in the numerical modeling of solid and fracture mechanics. This part is a preamble of the main part of the chapter that reports the potential of peridynamics as the most suitable technique to numerically address a wide range of simulations of generic three-dimensional FSI problems with arbitrarily complex geometries, including rupture and crack propagation analysis. Furthermore, the descriptions of Direct Numerical Simulation (DNS) technique for reproducing the dynamic of fluid phase and the Immersed Boundary Method (IBM) to prescribe the wall boundary conditions on fluid-solid interfaces are presented.

3.1 Local and non-local theories

The fundamental theories which have been developed in the last years to describe the mechanics of solid media with complex mechanical behavior can be firstly classified in two main categories: local and non-local theories.
The basic assumption of local theories is that a solid medium can be treated as a continuum at any arbitrary small length-scales. The consequence of this assumption is that any material point, which composes solid medium, can be considered independent and it can interact, through zero-distance contact forces, only with points located in its immediate proximity [26].
In this frame the mathematical description of solid medium relies on partial different equation, but the internal structure of real material presents specific features (i.e. porosity, grains, crystalline structures) that cannot be correctly described by considering solid medium as a continuum. Furthermore, if the presence of discontinuities is taken into account, the local theories doesn't work well since the discontinuities lead to the emergence of singularities on partial differential equations.
On the contrary, the basic assumption of non-local theories relies on the hypothesis that the infinitesimal material volumes can mutually interact across a finite

distance, that represents the characteristic length-scale of the material and it can be related to some meaningful physical lengths, such as porosity, grain sizes or size of micro-cracks [21].
In this frame the state of a point can be influenced by the points located within a finite region centered around the considered one. Thanks to this approach it is possible to capture the effects of microstructure of a material, but although non-local theories introduce additional details to represent solid materials, they still suffer from the problem of singularities that arises when discontinuities (i.e. micro-cracks) are presents.
In the frame of non-local theories, peridynamics is a recent numerical reformulation of solid mechanics that attempts to completely solve the issue of singularities of derivatives in partial differential equations due to the presence of the discontinuities, by replacing them with integral equations, such that the derivatives are completely removed from the formulation avoiding the arising of singularities [7] [17].
In peridynamics each point can interact with the others through short-range forces that act over a finite threshold distance, named *horizon*. The governing laws of peridynamics consists of a integral balance of linear momentum, computed over the short-range distance (the *horizon*) and accounting for the mutual internal forces exchanged between material points.
As a result of this innovative formulation, crack initiation is described only by the deactivation of pair interactions where the crack occurs. Then the deactivated internal forces are removed from the momentum balance. Thanks to this approach damage is intrinsically taken into account in peridynamic theory, such that fractures occur as a natural outgrowth of the equation of motion and constitutive models [13].

3.1.1 Limits of most popular numerical methods for discontinuous solid media.

In the last years different numerical models have been developed to solve solid mechanics in presence of discontinuities. The most popular are *Enriched* (or *Extended*) *Finite Element Method* (*XFEM*) [5] and the *Distinct Element Method (DEM)* [1].
XFEM was initially developed to solve problems of classical *Finite Element Method* which require a priori-knowledge of the shape of the discontinuous features, such as cracks, void or inhomogeneities, for re-meshing at every time step in order to have the mesh conforms to them.
The basic idea behind *XFEM* relies on adding to the standard polynomial shape functions some appropriate discontinuous basis functions for nodes that belong to those elements that are intersected by some localized discontinuous features, such as a crack [8]. By using this method it isn't needed a priori-knowledge of the shapes of the features involved and the need for re-meshing to track the path of discontinuities is removed.
XFEM is quite accurate in the modelling of discontinuous features but it isn't

able to intrinsically take into account the presence of discontinuities, as peridynamics, but it requires the use of semi-empirical or simplified models and, for this reason, its resolution is not great.
DEM is another popular method employed to address numerical simulation of solid media in presence of discontinuities [1]. The idea is to split the solid in a set of finite size of polyhedral or spherical solid blocks for which is considered complex mechanical behaviors, such as the possibility of warping, to couple the method with *FEM* [19] [33].
This approach uses contact-force models in solid-solid contact regions [15] and its coupling with Navier-Stokes model is a frequent technique to address the simulation of multiphase flows involving discontinuous solid aggregates [37] [27]. However, a priori-knowledge of the block shapes is still necessary and the formation of cracks and discontinuities are possible only between distinct elements, not only through a single block, via the use of cohesive force models.
The limitations which have been listed above make the applications of *XFEM* and *DEM* difficult for general cases where it is necessary to simulate the mechanical behaviour of discontinuous solid media.

3.2 FSI problem: Arbitrary Lagrangian-Eulerian formulation vs. Immersed Boundary Method

During the last years different approaches have been developed to represent numerically the solid-fluid interactions. The most popular techniques are *Arbitrary Lagrangian-Eulerian (ALE)* [3] method and the *Immersed Boundary Method (IBM)* [2].
The basic idea of *ALE* method is to discretize the domain by using unstructured computational meshes that conform to the fluid-solid interfaces at every time step. The solution of governing equations of the fluid phase, which are temporal integrated by using explicit or implicit time-marching schemes, is updated over a discrete time step. Then, after surface forces exerted by the fluid phase on the solid are been calculated, the position of the nodes located on the surfaces of the immersed solids are updated and new mesh is created consequentially [3]. Therefore, no-slip and no-penetration boundary conditions are prescribed by directly imposing specific values of velocity and its derivatives on the mesh nodes located on the surfaces of the immersed solids.
Although this approach permits an accurate treatment of the interfaces, the needs to re-mesh and to compute the convective fluxes at every time step make the computational cost of this kind of simulation quite large.
Instead, with *IBM* approach, the need to re-mesh is completely removed [2]. This method in fact uses a fixed Eulerian-grid which is not conform with fluid-solid interfaces and it is time-independent. The only requirement is that it can provide an adequate local resolution to numerically solve the smallest turbulent-scale [12].

Wall boundary conditions are indirectly prescribed to the flow in correspondence of the solid surfaces by forcing functions which are added in the conservation of linear momentum equation of the fluid phase to reproduce the effect of the boundaries.
Specifically, in this manuscript, *IBM* is implemented with a discrete forcing approach, namely the forcing function, which accounts for the presence of the boundary, is introduced at the discrete level, thus after the discretization process of the governing equations [14].
In *Discrete Forcing IBM* frame the governing equations of the fluid phase are solved on a uniform Cartesian grid, refereed to as Eulerian grid, instead the solid surface is represented as a set of Lagrangian points which are equally distributed over the fluid-solid interface. The idea to impose no-slip and no-penetration boundary conditions by making sure that the velocity of the fluid phase in correspondence of the fluid-solid interfaces is equal to the velocity of the Lagrangian points distributed on solid surfaces. This is equivalent to impose that the relative velocity between fluid and solid phases is equal to zero in correspondence of their interfaces. These impositions are fulfilled by the forcing function term which is based on interpolation and spreading operations with a smooth transfer of the quantities between the Eulerian and the Lagrangian configurations by means of a regularized discrete delta function [6] [2].
To improve numerical accuracy and stability of the method it is used to employ schemes based on iterative procedure and different interpolation procedures are possible. This class of methods, called *multi direct forcing IBM* [22], has in the computation of the back-reaction exerted by the fluid on the solid its main issue. In the present manuscript is used a *multi direct forcing IBM* to impose boundary conditions on the fluid-solid interfaces coupled with a *normal probe* approach to compute hydrodynamic forced on deformable solid surfaces. [32] [39].

3.3 Peridynamics

Peridynamics is a recent continuum theory which is based on non-local approach and it is formulated with integral equations so that it is particularly suitable for studying crack initiation and branching. In fact, as it was explained above, the most popular methods use to numerically solve solid mechanics are based on partial differential equations which require a specific techniques when the presence of discontinuities is taken into account, and many of these require a priori-knowledge of the shape of the peculiar features involved, for example inhomogeneity or direction of crack propagation.
On the contrary peridynamics, thank to its integral formulation, can overcome the limits of these numerical methods and it doesn't need a priori-knowledge of the direction of crack propagation. The basic idea of this innovative theory is that a solid body is composed by a finite number of material points and everyone interacts only with the points located within a short range distance,

named *horizon*.

3.3.1 Definition and basic terminology

Let's considered a closed and bounded solid body, named $\boldsymbol{B}$, in its reference configuration. Then it is discretized in a finite number of cubic material particles, each having a fixed volume and mass according to Silling and Lehoucq [18]. Each of these is identified by a material point, which has zero mass, localized in the center of the material particle.
The Lagrangian coordinate of an arbitrary material point of the body in its reference configuration, which is at time $t = 0$, is referred to as $\boldsymbol{X}_0 \in \boldsymbol{B}$ and, in general, the notation $\boldsymbol{X}_0$ is used to indicate a generic material point. Instead, the coordinate of the point $\boldsymbol{X}_0$ in a deformed configuration of the body, at time $t \geq 0$, is expressed by the Lagrangian variable $\boldsymbol{X}(\boldsymbol{X}_0, t)$. Now that the reference frame is known, the fundamental units of peridynamics are presented.
Let's considered a material point, $\boldsymbol{X}_0$, it is called *neighborhood* of $\boldsymbol{X_0}$ the set of material points included in the spherical region of space centered in $\boldsymbol{X}_0$,

$$H_{\boldsymbol{X}_0} = \{\boldsymbol{X'}_0 \in B \wedge \boldsymbol{X'}_0 \neq \boldsymbol{X}_0, \|\boldsymbol{X'}_0 - \boldsymbol{X}_0\| < \delta\}, \tag{3.1}$$

where δ is the *horizon* that, as previously reported, is the finite maximum distance along which two material points can interact.
Then the *family* of $\boldsymbol{X}_0$ is defined as [21, 7]:

$$H = \{\boldsymbol{\xi} \in (\Re^3 \setminus \mathbf{0}), (\boldsymbol{\xi} + \boldsymbol{X_0}) \in (H_{\boldsymbol{X}_0} \cap B)\} \tag{3.2}$$

where a vector $\boldsymbol{\xi} \in H$ is called a *bond* connected to X_0, so a *family* of material point, for example $\boldsymbol{X}_0$, is the set of *bonds* that connect the reference material point with those that are included in its *neighborhood*.
Then, by definition, a *bond* ξ can be expressed as:

$$\boldsymbol{\xi} = \boldsymbol{X'}_0 - \boldsymbol{X}_0, \quad \forall \boldsymbol{X}_0 \in B, \quad \boldsymbol{X'}_0 \in H_{\boldsymbol{X}_0}, \tag{3.3}$$

whereas, the definition of a bond in a deformed configuration of the body is:

$$\boldsymbol{\zeta} = \boldsymbol{X'}(\boldsymbol{X'_0}, t) - \boldsymbol{X}(\boldsymbol{X_0}, t), \quad \forall \boldsymbol{X}_0 \in B, \quad \boldsymbol{X'}_0 \in H_{\boldsymbol{X}_0}, t \geq 0, \tag{3.4}$$

with,

$$\boldsymbol{\eta} = \boldsymbol{\zeta} - \boldsymbol{\xi}, \tag{3.5}$$

the *bond displacement vector*.

3.3.2 The governing equations

The fundamental governing equation of peridynamics relies essentially on a Lagrangian formulation of the conservation of linear momentum in continuum media. As previously reported, the Lagrangian coordinate of an arbitrary material point of the body in its reference configuration, named $\mathbf{B}$, is indicated as

$\boldsymbol{X}_0$, whereas the Lagrangian variable $\boldsymbol{X}(\boldsymbol{X}_0, t)$ denotes the coordinate of the point $\boldsymbol{X}_0$ in a deformed configuration of the body, at time $t \geq 0$.
In this frame, the velocity of a generic material particle within the solid is described by the Lagrangian variable:

$$\boldsymbol{V}(\boldsymbol{X}_0, t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}(\boldsymbol{X}_0, t). \tag{3.6}$$

Named $P \in B$ an arbitrary subregion of the body, the Newton's second law applied to it is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_P \rho(\boldsymbol{X}_0)\boldsymbol{V}(\mathbf{X}_0, t)\,\mathrm{d}V = \int_P [\boldsymbol{L}(\boldsymbol{X}_0, t) + \boldsymbol{F}(\boldsymbol{X}_0, t)]\,\mathrm{d}V, \tag{3.7}$$

where $\rho(\boldsymbol{X}_0)$ is the density of the body and, since it can change within the solid, it is indicated as a function of the coordinate $\boldsymbol{X}_0$. Then the notation $\boldsymbol{F}(\boldsymbol{X}_0, t)$ indicates an external body force per unit volume and $\boldsymbol{L}(\boldsymbol{X}_0, t)$ the force per unit volume acting at time t on $\boldsymbol{X}_0$ due to the interactions of the latter with other material points [7, 21]. By localization of equation (3.7), the Lagrangian equation governing the motion of any material point $\boldsymbol{X}_0$ is:

$$\rho(\boldsymbol{X}_0)\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{V}(\boldsymbol{X}_0, t) = \boldsymbol{L}(\boldsymbol{X}_0, t) + \boldsymbol{F}(\boldsymbol{X}_0, t), \qquad \forall \boldsymbol{X}_0 \in B, t \geq 0. \tag{3.8}$$

$\boldsymbol{L}(\boldsymbol{X}_0, t)$ is an internal force density and therefore, for the action-reaction's law, is self-equilibrating. So, if the Newton's second law is applied to the whole solid body $\mathbf{B}$ the Lagrangian equation governing the motion of the body becomes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_B \rho(\boldsymbol{X}_0)\boldsymbol{V}(\boldsymbol{X}_0, t)\,\mathrm{d}V = \int_B \boldsymbol{F}(\boldsymbol{X}_0, t)\,\mathrm{d}V. \tag{3.9}$$

and it means that

$$\int_B \boldsymbol{L}(\boldsymbol{X}_0, t)\,\mathrm{d}V = 0, \tag{3.10}$$

To avoid the problem of the onset of singularities due to the presence of discontinuities, in peridynamics frame, any internal force density acting on a material point is expressed by an integral relation, provided below in equation (3.12) computed over finite region of space radius, δ. Indeed, for any given force field $\boldsymbol{L}(\boldsymbol{X}_0, t)$, it always exists a vector-valued function, $\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$, that is anti-symmetric,

$$\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t) = -\boldsymbol{f}(\boldsymbol{X}_0, \boldsymbol{X'}_0, t), \qquad \forall \boldsymbol{X'}_0, \boldsymbol{X}_0 \in B, \quad t \geq 0, \tag{3.11}$$

and such that $\boldsymbol{L}(\boldsymbol{X}_0, t)$ can be expressed as:

$$\boldsymbol{L}(\boldsymbol{X}_0, t) = \int_{H_{\boldsymbol{X}_0}} \boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)\,\mathrm{d}V', \qquad \forall \boldsymbol{X}_0 \in B, t \geq 0. \tag{3.12}$$

where dV' is the differential volume evaluated at the position $\boldsymbol{X'}_0$ which is a material point included in the *neighborhood* of $\boldsymbol{X}_0$. The function $\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$ is called *pairwise force density* and it can be interpreted as the force per unit volume square acted by the point $\boldsymbol{X'}_0$ on $\boldsymbol{X}_0$ as is sketched in figure 3.1.

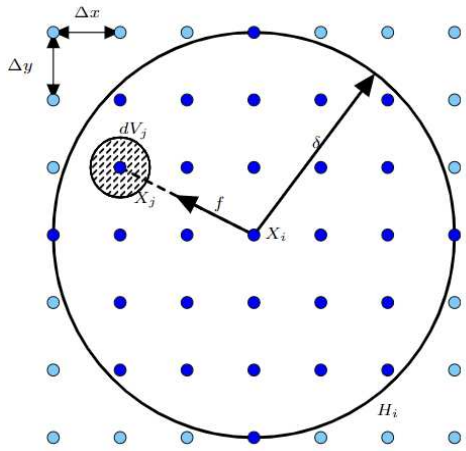


Figure 3.1: Two-dimensional sketch of the *horizon* of $\boldsymbol{X}_i$ and representation of force f acting between $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$. The material points included in the *neighborhood* are highlighted in dark blue. Figure taken from [34].

The *pairwise force density* can be expressed in the following way:

$$\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t) = \boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t) - \boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t), \qquad \forall \boldsymbol{X'}_0, \boldsymbol{X}_0 \in B, t \geq 0. \tag{3.13}$$

where $\boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$ is called *bond force density* and represents the basic quantity produced by a constitutive model in peridynamics theory [17]. For a better comprehension, in the figure below is represented the sketch of the *pairwise force density* and related *bond force density* for an arbitrary couple of material points of a peridynamic solid.

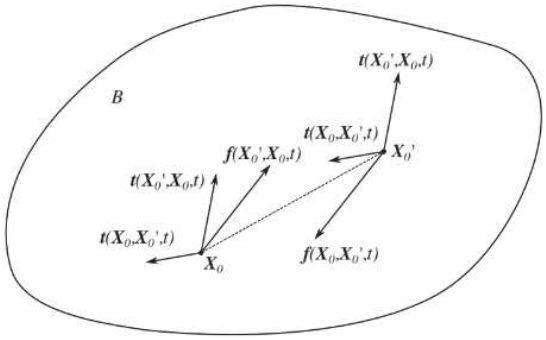


Figure 3.2: Representation of the couple of *pairwise force density* that the point $\boldsymbol{X}_0$ and $\boldsymbol{X'}_0$ exercise each other which are obtained by the contribution of the *bond force density* $\boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$ and $\boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$.

Finally, by considering the internal force acting on arbitrary point material as sum of all the *pairwise force density* that act on the reference material point, as it is provided in equation (3.12) and by considering the definition of Lagrangian velocity provided in equation (3.6), and the conservation of linear momentum of a single material point (3.8), the governing equations of peridynamics read:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}(\boldsymbol{X}_0,t) = \boldsymbol{V}(\boldsymbol{X}_0,t), \tag{3.14}$$

$$\rho(X_0)\frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{X}(\boldsymbol{X}_0,t) = \boldsymbol{L}(\boldsymbol{X}_0,t) + \boldsymbol{F}(\boldsymbol{X}_0,t), \tag{3.15}$$

$$\boldsymbol{L}(\boldsymbol{X}_0,t) = \int_{H_{\boldsymbol{X}_0}} [\boldsymbol{t}(\boldsymbol{X'}_0,\boldsymbol{X}_0,t) - \boldsymbol{t}(\boldsymbol{X'}_0,\boldsymbol{X}_0,t)]\,\mathrm{d}V \tag{3.16}$$

which are valid $\forall \boldsymbol{X}_0 \in B, t \geq 0$. To solve them it is necessary to provide a constitutive relation to compute the *bond force density* as a function of the macroscopic mechanical properties of the material and deformation state of the body.

Different constitutive relations can be derived for a peridynamic material and based on the constitutive model employed, it is possible to distinguish between three different peridynamic models: *bond-base peridynamics, ordinary state-based peridynamics, non-ordinary state-based peridynamics* [17]. In the former, which relies in the first peridynamic formulation developed by Silling [7], the constitutive model results in pairs of *bonde force density* vectors, $\boldsymbol{t}(\boldsymbol{X'}_0,\boldsymbol{X}_0,t)$ and $\boldsymbol{t}(\boldsymbol{X}_0,\boldsymbol{X'}_0,t)$, that are parallel to the relative position vectors in the deformed configuration of the body, $\boldsymbol{\zeta}$, and equal in magnitude. Whereas, In the ordinary state-based model, the *bond force density* vectors are still parallel to $\boldsymbol{\zeta}$, but they can have different magnitude. Finally, in non-ordinary state-based peridynamics, the *bond force density* vectors can be non-parallel to the related relative position vectors and different in magnitude.

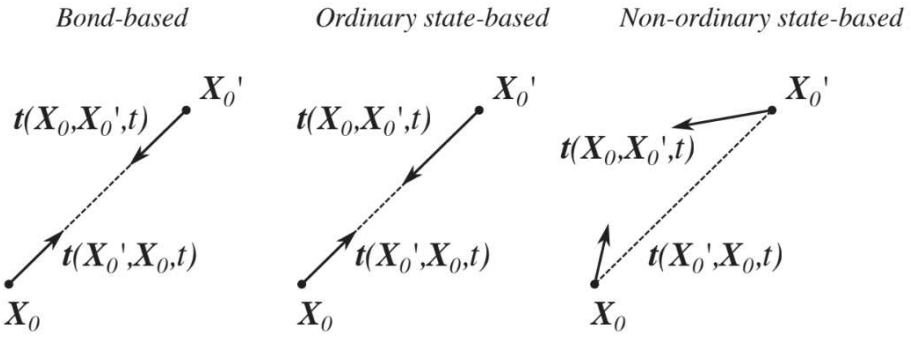


Figure 3.3: Differences of *bond force density* vectors definition between bond-based peridynamics, ordinary state-based peridynamics, non-ordinary state-based peridynamics

This manuscript focuses only on *bond-based peridynamics*, which can be seen as a special case of *ordinary state-based peridynamics*, because it is enough to reproduce the mechanical properties of the material of the porous solid medium that will be simulated. This model is the simplest of the peridynamic models and it can be entirely assessed without employing the formalism of peridynamic states (for additional information of peridynamic states and related algebra, the reader refers to the references [7, 17, 21]). It is important to remark that the real potential of peridynamics theory consists in the introductions of a non-locality by using of an integral relation to impose the internal force balance, avoids the onset of singularities that arise in local theories from the evaluation of partial derivatives in the presence of discontinuities, such as inhomogeneity and cracks, in the material.

3.3.3 Bond-based peridynamics

As previously reported, *bond-based peridynamics* relies on a special case of peridynamics although it can be entirely assessed without employing the formalism of peridynamic states. The basic assumption of bond-based theory is that each bond has its own constitutive relation, which is independent from the others. Since, as previously reported in last part of the section 3.3.2, the constitutive model of bond-based results in pairs of *bond force density* vectors that are parallel to the relative position vectors in the deformed configuration of the body, $\boldsymbol{\zeta}$, and equal in magnitude, the constitutive relation for *bond force density* vectors are derived:

$$\boldsymbol{t}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t) = \frac{1}{2}\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t), \tag{3.17}$$

$$\boldsymbol{t}(\boldsymbol{X}_0, \boldsymbol{X'}_0, t) = \frac{1}{2}\boldsymbol{f}(\boldsymbol{X}_0, \boldsymbol{X'}_0, t) = -\frac{1}{2}\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t). \tag{3.18}$$

As specified above, since each bond has its own constitutive relation, which is independent from the others, the value of *pairwise force density* $\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t)$, depends exclusively on the single bond, $\boldsymbol{\xi}$, connecting the material point $\boldsymbol{X}_0$ to $\boldsymbol{X'}_0$. This kind of material model, described by *bond-based peridynamics*, is referred to as *Prototype Microelastic Brittle (PMB)* which its properties are:

1. the *bond stretch* $s(\boldsymbol{\xi}, \boldsymbol{\eta})$ defined as:

$$s(\boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{\|\boldsymbol{\xi} + \boldsymbol{\eta}\| - \|\boldsymbol{\xi}\|}{\|\boldsymbol{\xi}\|} \tag{3.19}$$

   is calculated as the ratio of the *pairwise force density* $\boldsymbol{f}$ and the bond stiffness c_0, called *bond micro-modulus*;

2. the rupture of a bond occurs when the stretch of a bond overcomes a threshold value, s_0, called the *limit bond stretch*. The breakage of a bond is permanent and irreversible, therefore a bond is defined as time-dependent;

3. bond cannot break under compression.

Under these assumption, the *pairwise force density* can be expressed [7, 52]:

$$\boldsymbol{f}(\boldsymbol{X'}_0, \boldsymbol{X}_0, t) = \boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{\eta}) = c_0 s(\boldsymbol{\xi}, \boldsymbol{\eta}) \frac{\boldsymbol{\xi} + \boldsymbol{\eta}}{\|\boldsymbol{\xi} + \boldsymbol{\eta}\|} \lambda(\boldsymbol{\xi}, t) \tag{3.20}$$

where $\lambda(\boldsymbol{\xi}, t)$ is a history-dependent scalar-valued function that accounts crack formation and it will be discussed in the next section. The figure below represents the stretch of a bond in deformed configuration of the solid body due to the relative displacements between material points which are less distant of the limit cut-off distance δ in the reference configuration of the body.

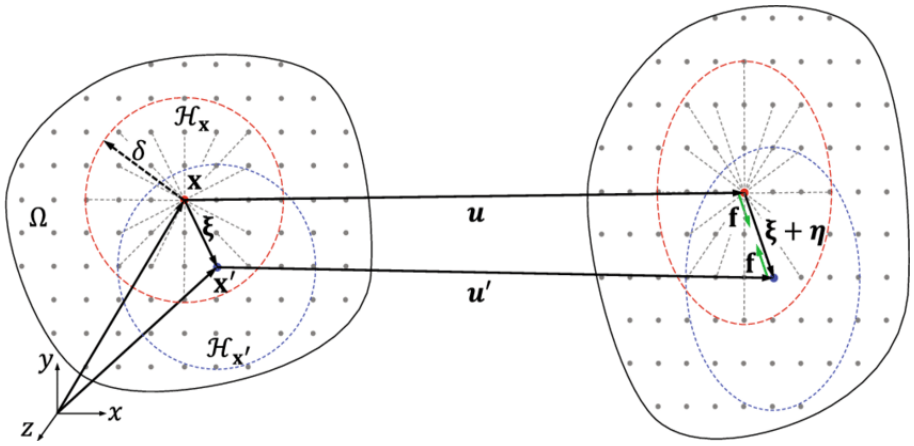


Figure 3.4: Deformation of a bond in bond-based peridynamics. To the left is represented the reference configuration of the body, referred as Ω, whereas to the right is depicted its deformed configuration. $\boldsymbol{u}$ and $\boldsymbol{u'}$ indicate respectively the displacement vectors of the Lagrangian variables $\boldsymbol{X}$ and $\boldsymbol{X'}$, whereas $\boldsymbol{\xi}$ indicates the undeformed bond and $\boldsymbol{\eta}$ the *bond displacement vector*. Figure taken from [38].

The *bond micro-modulus* c_0 can be put in relation with the macroscopic mechanical properties of the materials. The constitutive relation for c_0 depends on the geometry and loading conditions of the specific problem. In particular, the *bond micro-modulus* is a function of Young's modulus, E, and the peridynamic horizon δ:

$$c_0 = \begin{cases} \dfrac{9E}{\pi t \delta^3}, & \text{in 2D plane stress,} \\ \dfrac{48E}{5\pi t \delta^3}, & \text{in 2D plane strain,} \\ \dfrac{12E}{\pi \delta^4}, & \text{in 3D} \end{cases} \tag{3.21}$$

where t refers to the depth of the body along the out-of plane direction for the two dimensional cases. The Poisson's ratio ν_s is not included in equation (3.21) and, from derivation of equation 3.21, it can be proven that Poisson's ratio is fixed to the value of 1/4 for the three-dimensional and plain strain cases and to 1/3 for the plane stress case. The fact that Poisson's ratio is forced to have constant value for specific geometry and loading condition of the problem represents the main limitation of *bond-based peridynamics*. In the present manuscript, for semplicity, the *bond micro-modulus* is assumed to be uniform and constant.

3.3.4 Bond damage and fracture

As explained before, the definition of the internal force acting on an arbitrary material point as an integral balance computed over a finite region of space radius, δ, as described by the relation provided in equation (3.12) permits to intrinsically take into account of the damage at the bond level, such that fractures occur as a natural outgrowth of the equation of motion and constitutive models. The basic assumption of peridynamics is that when the stretch s, defined according to the equation (3.19), overcomes a threshold value s_0, called the *limit bond stretch*, the *pairwise force density* vectors between the involved material points are deactivated and their contribution to the internal force balance is permanently neglected. The deactivation of a pairwise interaction represents a rupture of the related bond and therefore the formation of a crack [13].
The figure below reports a schematic representation of a crack in peridynamic frame.

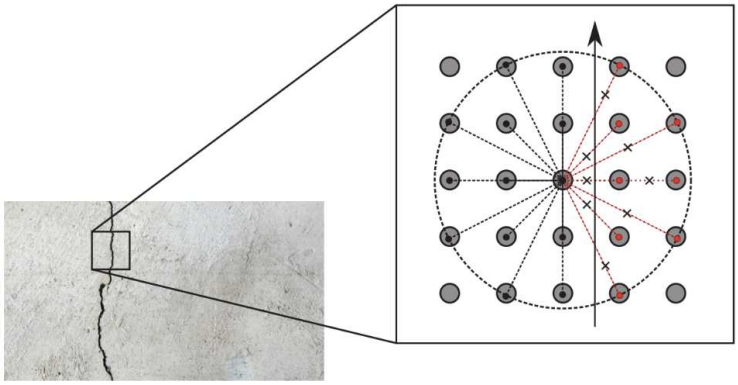


Figure 3.5: Schematic representation of a crack in peridynamic frame. The broken bonds are highlighted by the cross symbols and they don't contribute to the overall internal force balance on the central material point. The direction of crack propagation is represented by the black arrow.

To take into account the deactivation of the *pairwise force density* vectors when the bond stretch overcomes the *limit bond stretch*, and so the crack formation, there is the history-dependent scalar-valued function cointained in the *pairwise force density* expression in equation (3.20), that is defined in the following way:

$$\lambda(\boldsymbol{\xi}, t) = \lambda(s) = \begin{cases} 1, & s \leq s_0 \quad \forall t \geq 0, \\ \\ 0, & \text{otherwise.} \end{cases} \tag{3.22}$$

This parameter permits to neglect immediately the *pairwise force density* of the broken bonds from the internal force balance.
As the *bond micro-modulus* also the *limit bond stretch* s_0 can be expressed as a function of mechanical properties of the material through constitutive relations that depend on geometry and loading conditions of the specific problem. In particular the *limit bond stretch* can be expressed as a function of the *critical fracture energy release rate* of the material, G_0, the Young's modulus, E, and the peridynamic horizon, δ. It is worth remarking that the *critical fracture energy release rate* G_0 is expressed as the decrease in total potential energy per increase in fracture surface area, and is thus expressed in terms of energy per unit area. In the following a derivation of an expression for the *limit bond stretch* is provided for three-dimensional case in the frame of bond-based theory, considering a brittle micro-elastic material defined in section 3.3.3, by equaling the *critical fracture energy release rate* defined in the frame of classical fracture mechanics theory, with the work required to break all the bonds per unit fracture area computed in the peridynamic frame.
Let's consider a fracture in a brittle micro-elastic material and two arbitrary material points, $\boldsymbol{X}_0$ and $\boldsymbol{X'}_0$, located at the opposite sides of the fracture

surface. The figure below shows a schematic representation of the considered situation:

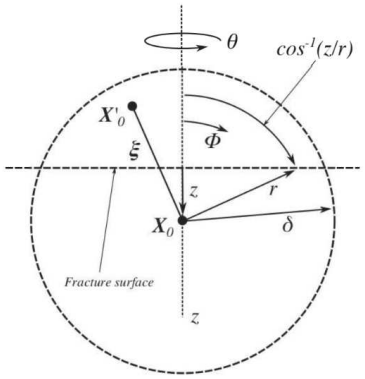


Figure 3.6: Representation of two arbitrary material points, $\boldsymbol{X}_0$ and $\boldsymbol{X'}_0$, located at the opposite sides of the fracture surface. The local z-axis measures the distance of the point $\boldsymbol{X}_0$ from the surface of the fracture. It is considered a spherical coordinate system centered at $\boldsymbol{X}_0$. Figure taken from [21].

The work required to break a single bond $\boldsymbol{\xi}$, connecting $\boldsymbol{X}_0$ to $\boldsymbol{X'}_0$, is:

$$w_0(\boldsymbol{\xi}) = \int_0^{s_0} c_0 s r \, \mathrm{d}s = \frac{1}{2} c_0 s_0^2 r, \tag{3.23}$$

where $r = \|\boldsymbol{\xi}\|$. Considering all the material points $\boldsymbol{X}_0$ along the *z-axes*, such that the z coordinates of these points are included in the interval $0 < z < \delta$, the work per unit fracture are W_0 required to break all the bonds connecting $\boldsymbol{X}_0$ to the points $\boldsymbol{X'}_0$ in the spherical cap to the opposite part of the fracture surface, can be computed as:

$$W_0 = \int_0^{\delta} \int_0^{2\pi} \int_z^{\delta} \int_0^{\cos^{-1}(z/r)} \left(\frac{1}{2} c_0 s_0^2 r \right) r^2 \sin(\Phi) \quad \mathrm{d}\Phi \, \mathrm{d}r \, \mathrm{d}\theta \, \mathrm{d}z. \tag{3.24}$$

By evaluating the integral above, the energy per unit fracture area for complete separation of the two halves of the considered body is:

$$W_0 = \frac{\pi c_0 s_0^2 \delta^5}{10} = G_0 \tag{3.25}$$

Finally, solving equation for s_0 and considering equation (3.21), which expresses c_0 as a function of E and δ, for the three-dimensional case, leads to:

$$s_0 = \sqrt{\frac{5G_0}{6E\delta}} \tag{3.26}$$

and, if a similar strategy is adopted for two-dimensional cases, considering plane stress and plane strain loading condition, the following expression are obtained:

$$s_0 = \begin{cases} \sqrt{\dfrac{4\pi G_0}{9E\delta}}, & \text{in 2D plane stress,} \\ \sqrt{\dfrac{5\pi G_0}{12E\delta}}, & \text{in 2D plane strain,} \\ \sqrt{\dfrac{5G_0}{6E\delta}}, & \text{in 3D.} \end{cases} \tag{3.27}$$

when a bond breaks a formation of a micro-crack occurs and it propagates locally in the direction normal to the broken bonds. As previously reported, since a bond breakage involves the removal of *pairwise force density*, related to the broken bond, from the internal force density balance, it leads a material weakening near crack tips that acts as a softer response of the material.
The advantage to introduce failure at bond level is that damage can be clearly identified at any arbitrary material point in the body and it can be quantified by the *damage level* computed as:

$$\Phi(\boldsymbol{X}_0, t) = 1 - \frac{\displaystyle\int_{H_{\boldsymbol{X}_0}} \lambda(\boldsymbol{\xi}, t)\, \mathrm{d}V'}{\displaystyle\int_{H_{\boldsymbol{X}_0}} \mathrm{d}V'} \tag{3.28}$$

The *damage level* is a scalar-valued function ranging from 0 to 1, with 0 corresponding to undamaged material and 1 corresponding to the disconnection of the reference material point from all the points which were originally part of its *neighborhood*.

3.3.5 Numerical discretization of peridynamics

In a peridynamic frame a solid body is defined as a set of finite-size material particles that mutually interact via the short-range force densities as specified above. Therefore, to numerically solve the governing laws of *bond-based peridynamics* it is necessary to adopt a discretization strategy which can represent a peridynamic solid body as well as possible. In the present thesis a "*mesh-free*" method, proposed by Silling [13], is used and so the solid domain is discretized with a uniform Lagrangian grid in a set of N_p finite-size, cubic and equally-spaced material particles, with fixed volume and fixed mass, each identified by a computational node located on their geometrical centroid.
With Δs is indicated the spacing between nodes in the reference configuration of the body and it is equal to the particle edge size. The ratio between the *horizon* δ and particle size is denoted as $m = \delta/\Delta s$ which is usually equal to 2 o 3. An arbitrary material particle, in the reference configuration of the body,

is identified by the Lagrangian variable $\boldsymbol{X}_{0,h}$ with $1 \leq h \leq N_p$, which refers to the coordinate of its geometrical centroid, whereas the Lagrangian variables $\boldsymbol{X}_h(\boldsymbol{X}_{0,h}, t)$ and $\boldsymbol{V}_h(\boldsymbol{X}_{0,h}, t)$ are used to indicate the position and velocity of the particle evaluated at time $t \geq 0$. These Lagrangian variables, as for the continuum case, are used to refer to both the particle centroid (and so to the corresponding node) as well as the discrete material particle itself.
The bond vector $\boldsymbol{\xi}_{h,l}$ in the reference configuration ($t = 0$), its image $\boldsymbol{\zeta}_{h,l}$ in a deformed configuration at $t \geq 0$ and the corresponding displacement vector $\boldsymbol{\eta}_{h,l}$ in the discrete case are respectively:

$$\boldsymbol{\xi}_{h,l} = \boldsymbol{X}_{0,l} - \boldsymbol{X}_{0,h}, \tag{3.29}$$

$$\boldsymbol{\zeta}_{h,l} = \boldsymbol{X}_l - \boldsymbol{X}_h, \tag{3.30}$$

$$\boldsymbol{\eta}_{h,l} = (\boldsymbol{X}_l - \boldsymbol{X}_h) - (\boldsymbol{X}_{0,l} - \boldsymbol{X}_{0,h}). \tag{3.31}$$

As previously reported by equation (3.1), the *neighborhood* of an arbitrary material point $\boldsymbol{X}_0$ is the set of material points enclosed into the space region of radius δ centered at $\mathbf{X}_0$. In the discrete frame of peridynamics, as it is possible to see in the figure 3.7, when a space region of radius δ centered at an arbitrary node $\boldsymbol{X}_{0,h}$ (which corresponds to the geometrical centroid of the associated discrete particle, as previously reported) is taken into consideration, there are some discrete particles belonging to its *neighborhood* which may present an intersection with the neighborhood boundary.

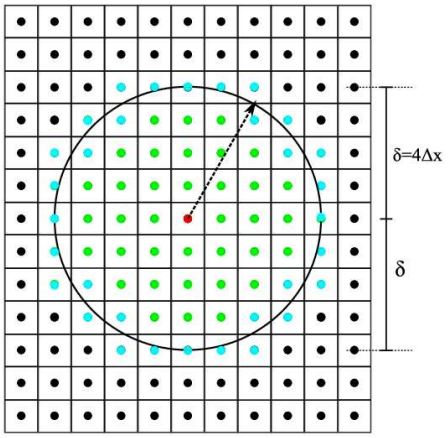


Figure 3.7: Schematic of the discretization of a peridynamic solid in the two-dimensional case and representation of the neighborhood boundaries of an arbitrary computational node highlighted in red. With Δx is indicated the grid spacing, δ is the *horizon* that in represented case is set as $\delta = 4\Delta x$. The nodes associated with the discrete particles which are completely included in the represented *neighborhood* are highlighted in light green, whereas the nodes associated with discrete particles which are partially enclosed in the space region covered by the considered *neighborhood* are highlighted in light blue.

To be sure that all the nodes that identify the discrete particles intersected by the neighborhood boundary of an arbitrary node $\boldsymbol{X}_{0,h}$ are included in its *neighborhood*, the definition of the *neighborhood* for the continuum case, provided by equation (3.1), is modified to be practicable in discrete case, as in the following:

$$H_{\boldsymbol{X}_{0,h}} = \left\{ \boldsymbol{X}_{0,l} \in \boldsymbol{B},\, \|\boldsymbol{X}_{0,l} - \boldsymbol{X}_{0,h}\| < \delta + \frac{1}{2}\Delta s \right\}. \tag{3.32}$$

The figure below reports a sketch of the *neighborhood* of a discrete material particle defined according to the equation (3.32):

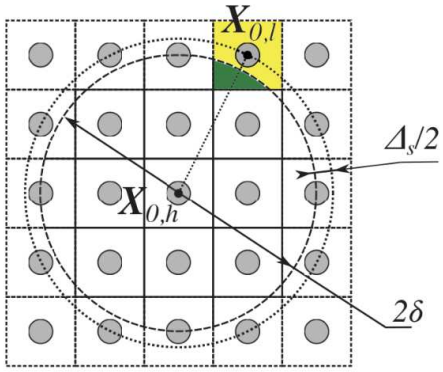


Figure 3.8: The *neighborhood* of a discrete material particle, $\boldsymbol{X}_{0,h}$ defined according to equation (3.32). The region highlighted in green corresponds to the portion of the cubic particle $\boldsymbol{X}_{0,l}$ enclosed into sphere of radius δ centered at $\boldsymbol{X}_{0,h}$. The remaining portion of the material particle is highlighted in yellow. Figure taken from [48].

As it was reported in section 3.3.2, the calculation of the internal force acting on arbitrary material point, provided by equation (3.12), is expressed as the integral of *pairwise force density* computed over the solid volume enclosed by the space region of the *neighborhood* of the considered material point. This is a fundamental quantity which is present in the linear momentum balance equation of the generic material point and it is necessary for the updating of the Lagrangian position and velocity of the considered material point. So, for reaching a reasonable accuracy of peridynamic numerical model, it is necessary to approximate the solid volume enclosed by neighborhood boundaries as well as possible. This can be done by employing the *volume reduction factor*, referred as $\Gamma_{h,l}$, which permits to take into account, for the discretize particles intersected by a neighborhood boundary, only their volume fraction that are enclosed in the space region covered by the *neighborhood*. In the following is reported its formulation for two-dimensional cases, by using a linear variation between a factor 1/2 and 1 depending on the neighborhood member's location

with respect to the *horizon* [26, 13], δ:

$$\Gamma_{h,l} = \begin{cases} \dfrac{\delta - \|\boldsymbol{\xi}_{h,l}\|}{\Delta s} + \dfrac{1}{2}, & \left(\delta - \dfrac{\Delta s}{2}\right) \leq \|\boldsymbol{\xi}_{h,l}\| \leq \delta, \\ 1, & 0 \leq \|\boldsymbol{\xi}_{h,l}\| \leq \left(\delta - \dfrac{\Delta s}{2}\right) \end{cases} \tag{3.33}$$

This parameter $\Gamma_{h,l}$ is defined as the ratio between the fraction of the volume of $\boldsymbol{X}_{0,l}$ comprised within a distance δ from $\boldsymbol{X}_{0,h}$ and the overall volume of $\mathbf{X}_{0,l}$, that in the present case is $\Delta V_l = \Delta s^3$. It is worth nothing that $\Gamma_{h,l}$ is computed in the reference configuration of the body and it does not depend on time.
For three-dimensional cases, instead, an analytical expression for *volume reduction factor* does not exist and so $\Gamma_{h,l}$ must be computed numerically. The most common approach consists of partitioning the cubic particle $\boldsymbol{X}_{0,l}$ into a set of N^3 cubic sub-region of size $\Delta s/N$ and the *volume reduction factor* is computed as $\Gamma_{h,l} = N_i/N^3$ where N_i is the number of cubes for which $r \leq \delta$, where r is the distance of the cubic sub-region centroids from $\boldsymbol{X}_{0,h}$ and they can be easily computed. The figure below reports a sketch of the partitioning of a discrete particle for two-dimensional cases:

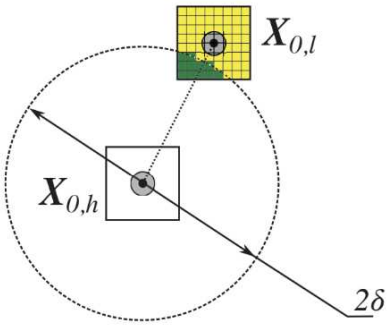


Figure 3.9: Two-dimensional schematic representation of the algorithm for the computation of *volume reduction factor*, $\Gamma_{h,l}$. The regions highlighted in green correspond to the portion of the cubic particle $\boldsymbol{X}_{0,l}$ enclosed into sphere of radius δ centered at $\boldsymbol{X}_{0,h}$. The remaining portion of the material particle is highlighted in yellow. Figure taken from [48].

The others fundamental peridynamic quantities expressed in discreet form are reported in the following:

- history-dependent scalar-valued function in discrete case:

$$\lambda_{h,l} = \begin{cases} 1, & s_{h,l} \leq s_0 \quad \forall t \geq 0, \\ \\ 0, & \text{otherwise.} \end{cases} \tag{3.34}$$

  where s_0 is the *limit bond stretch* defined in equation (3.27);

- the *bond stretch* in discrete case:

$$s_{h,l} = \frac{\|\boldsymbol{\xi}_{h,l} + \boldsymbol{\eta}_{h,l}\| - \|\boldsymbol{\xi}_{h,l}\|}{\|\boldsymbol{\xi}_{h,l}\|} \tag{3.35}$$

3.3.6 Discrete bond-based peridynamics

For *bond-based peridynamics*, as well as for the continuum case, the discrete *pairwise force density* $\boldsymbol{f}_{h,l}$ is expressed as a function of the *bond stretch* and the macroscopic mechanical properties of the material, as it is reported by equation (3.20):

$$\boldsymbol{f}_{h,l} = c_0 \, \lambda_{h,l} \, s_{h,l} \, \frac{\boldsymbol{\xi}_{h,l} + \boldsymbol{\eta}_{h,l}}{\|\boldsymbol{\xi}_{h,l} + \boldsymbol{\eta}_{h,l}\|} \tag{3.36}$$

where $s_{h,l}$ is the *bond stretch* defined by equation (3.35), whereas $\boldsymbol{\xi}_{h,l}$ and $\boldsymbol{\eta}_{h,l}$ are provided by equation (3.30) and(3.31), respectively. The bond micro-modulus c_0 is assumed to be constant and is defined in equation (3.21).
Therefore, the discrete equation of *bond-based peridynamics*, obtained by considering the corresponding equations from continuum case (3.14, 3.15, 3.16), are:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}_h = \boldsymbol{V}_h, \tag{3.37}$$

$$\rho_s \frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{X}_h = \boldsymbol{L}_h + \boldsymbol{F}_h + \boldsymbol{C}_h + \boldsymbol{D}_h \tag{3.38}$$

$$\boldsymbol{L}_h = \sum_{l=1}^{N_h} \boldsymbol{f}_{h,l} \, \Gamma_{h,l} \, \Delta V_l \tag{3.39}$$

where N_h is the finite number of particles which are part of the *neighborhood* of $\boldsymbol{X}_{0,h}$, ρ_s is solid density that is assumed to be constant in the present manuscript. The external force per unit volume acting on $\boldsymbol{X}_{0,h}$ is split into two different terms: the generic external fluid dynamic force field per unit volume, $\boldsymbol{F}_h$, and the force per unit volume arising from the contact of $\boldsymbol{X}_{0,h}$ with other material particles, which are not part of its *neighborhood* but are from other solid body or from another part of the same solid domain, $\boldsymbol{C}_h$. These types of forces, named *solid-solid contact forces*, are represented by pairwise short-range forces acting between couples of material particles which are located within a

cut-off distance, r_c. The expression that provides the short-range force acting on arbitrary discrete particle $\boldsymbol{X}_{0,h}$, due to the contact with another particle $\boldsymbol{X}_{0,l}$, which is not part to the *neighborhood* of $\boldsymbol{X}_{0,h}$, is:

$$\boldsymbol{C}_{h,l} = max\left\{k_c\left[\left(\frac{r_c}{\|\xi_{h,l}\|}\right)^{n_c} - 1\right],\, 0\right\}\frac{\boldsymbol{X}_l - \boldsymbol{X}_h}{\|\boldsymbol{X}_l - \boldsymbol{X}_h\|}\,, \tag{3.40}$$

where n_c and k_c are indicated as the short-range force exponent and the short-range force constant, respectively. Different choices of the values of n_c, r_c, k_c are possible and in the present thesis the values suggested by Macek and Silling [50] are employed, which are $n_c = 1$, $k_c = 15c_0$ and $r_c = \Delta s$. For reducing the computational cost of the simulation a *nearest-neighborhood search (NNS)* algorithm is used, which detects the pairs of particles located within the cut-off distance r_c, only for the particles located on the interfaces, without considering particles of the same *neighborhood*.
The term $\boldsymbol{D}_h$, instead, is an additional term which is not present in continuum case and is a function of the relative velocity between each pair of discrete particles:

$$\boldsymbol{D}_h = -k_d\sum_{l=1}^{N_h}\left[\left(\boldsymbol{V}_l - \boldsymbol{V}_h\right)\cdot\frac{\boldsymbol{\xi}_{h,l} + \boldsymbol{\eta}_{h,l}}{\|\boldsymbol{\xi}_{h,l} + \boldsymbol{\eta}_{h,l}\|}\right]\Gamma_{h,l}\Delta V_l\,, \tag{3.41}$$

width k_d the *internal dumping coefficient* in $[N\,s/m^7]$. This additional term is a self-equilibrating damping force which acts as a stabilizing numerical factor by suppressing the high-frequency modes of the sub-horizon motion of the discrete particles, which are related to acoustics and wave propagation that represents a disturbing phenomenon that afflicts the numerical stability of the method. Further explaining of the computation of this damping force is provided in the references [53, 28].
It can be demonstrated that the two fundamental conditions so that the discrete peridynamic model converge to the classical elasticity solution are *delta-convergence* ($\delta \to 0$) and *m-convergence* ($m \to 0$) [18].

3.3.7 Definition of the algorithm for interface detection and tracking

As it was reported before, a discretized peridyanamic solid is represented as a set of finite-size, cubic and equally-spaced material particles, each identified by a computational node located on its geometrical centroid. For reproducing the deformation and motion of the solid body is extremely important to detect the material particles located on the solid surfaces where the external forces, divided in fluid-dynamic and solid-solid contact forces, act.
For defining and tracking the interfaces that separate a solid object from the sorrounding space, it is used an algorithm, developed by Dalla Barba, which employs a set of Lagrangian markers, that form a specific subset of material

particles, distributed on the solid surfaces that move solidly with the latter. Thanks to this algorithm the interfaces are automatically tracked by evolving the Lagrangian governing equations of peridynamics (3.37)-(3.38)-(3.39) and it is simple to detect new interfaces by extending the material particles used as interface markers. The basic idea of the interface detection and tracking criteria consists of assuming that a material particle is located on interface if exist a region of space with a certain dimension where particles that interact with considered one are not present. In practice for each discrete particle, $\boldsymbol{X}_{0,h}$, is defined a circular region of space, divided in $N_s = 8$ sectors with azimutal extensions $\Delta\sigma = 2\pi/8$, with radius $R_c = \delta$, centered on the actual position of the particle centroid, $\boldsymbol{X}_h$.
Since the region of space considered is a sphere of radius δ centered at $\boldsymbol{X}_{0,h}$, all the discrete particles enclosed in the region considered, are part of the *neighborhood* of $\boldsymbol{X}_{0,h}$ and each one is associated to one and only one of the sectors previously defined.
For each sector, i, it is defined a mean relative distance, $r_{avg,i}$, for which the bond $\boldsymbol{\xi}_{h,l}$ that connects the central node $\boldsymbol{X}_{0,h}$ with the others enclosed in considered sphere region $\boldsymbol{X}_{0,l}$, has been deactivated ($\lambda_{h,l} = 0$) and it is computed as:

$$\bar{r}_{h,i} = \frac{\sum_{l=1}^{N_i} \|\boldsymbol{X}_l - \boldsymbol{X}_h\|(1 - \lambda_{h,l})}{\sum_{l=1}^{N_i}(1 - \lambda_{h,l})} \tag{3.42}$$

with $\boldsymbol{X}_h$ and $\boldsymbol{X}_l$ the Lagrangian coordinates of the particle centroids in deformed configuration at time $t \geq 0$, N_i the number of discrete particles which are part of the considered sector. The node $\boldsymbol{X}_{0,h}$ is considered as part of the interface if there are not discrete particles belong to at least $N_v = 2$ of its arbitrary consecutive sectors and, for these, all the bond $\boldsymbol{\xi}_{h,l}$ must be broken and their main relative distance $\bar{r}_{h,i}$ must be greater than a prescribed threshold value $\bar{r}_{thr}$, which in present work is equal to δ. Therefore, a new interface is detected only if the distance between solid parts is greater than the *horizon*, so δ represents both the characteristic length-scale of the material and the size of the smallest detectable interface. The figure below represents the identification of a material particle for two-dimensional case by using the algorithm just presented:

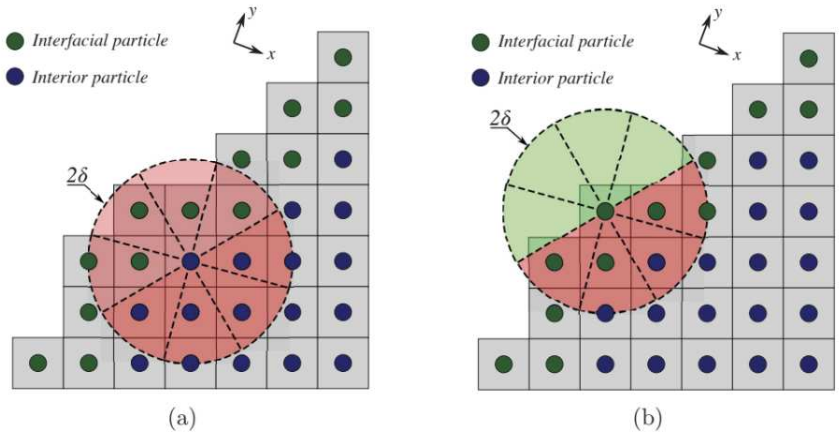


Figure 3.10: Two-dimensional schematic representation of the interface detection algorithm with $N_s = 8$, $N_v = 2$ and $r_{avg,t} = 2\Delta s$. (a) Representation of a material particle located into the solid interior. (b) Material particle located on the interface according to the prescribed criterion. Figure taken from [40].

3.3.8 Correction of peridynamic surface effect: volume-method

In a *bond-based peridynamic* model, the basic parameter that defines the value of *pairwise force density*, which represents the strenght of the interaction between two arbitrary material particles, is the *bond micro-modulus* c_0, but its expression, given by equation (3.21) is derived by assuming that each material point is located into the bulk of a solid and so it has a complete spherical *neighborhood*.
Nonetheless, as it is specified in the section 3.3.7, a material point located on the interface of the solid body doesn't have full spherical *neighborhood*. The missing *family* does not contribute to the overall deformation energy of the body and so, when this is loaded and stretched, the deformation energy density near the solid boundaries is lower than the bulk. The main consequence related to this phenomenon is a softening of the material response near the free surfaces as the local elastic modulus decreases. This effect is known as *peridynamic surface effect* or *boundary effect*.

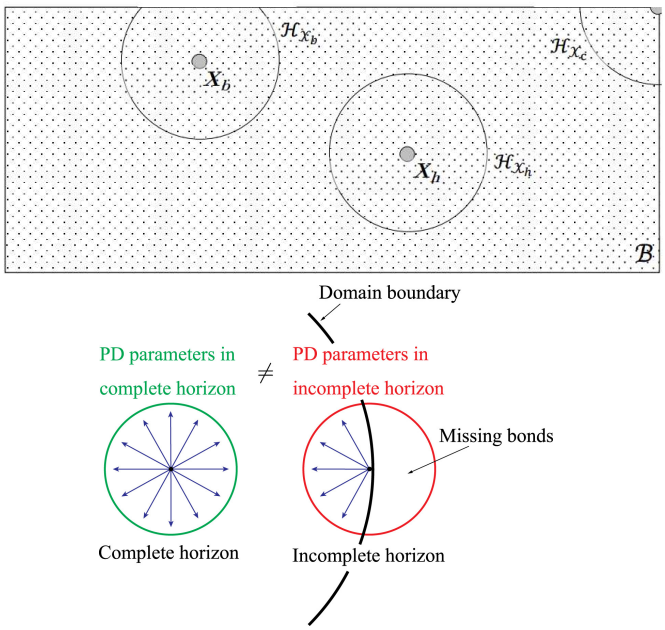


Figure 3.11: At the top of the figure is sketched a two-dimensional rectangular solid domain, $\boldsymbol{B}$, where the *neighborhoods* of three different material points, $\boldsymbol{X}_h, \boldsymbol{X}_b$ and $\boldsymbol{X}_c$ are represented. As it is possible to see $\boldsymbol{X}_h$ is an internal point because has a full spherical neighborhood, whereas $\boldsymbol{X}_c$ is a interface material point with a few material particles enclosed in its neighborhood, finally $\boldsymbol{X}_b$ is an internal point but it is close to the solid boundary and so has a truncated neighborhood. The bottom part show that material particles with truncated neighborhood have fewer bonds included in their families than the particles with full spherical neighborhood, and so they have different peridynamic parameters from the nominal case, in particular they have a lower value of the *bond micro-modulus* c_0. Figure taken from [42]

For mitigating the surface effect [49], in the present manuscript, the *volume method* is employed and its description in a discrete form is provided in the following.
The basic idea is that of counteracting the softening of the material near the solid boundaries, due to the incomplete families, by employing a multiplication factor $\alpha_{h,l}$ in the evaluation of *pairwise force density* acting between each pair of discrete particles.
Let's consider a generic material particle, $\boldsymbol{X}_{0,h}$, this multiplication factor stiffens each bond $\boldsymbol{\xi}_{h,l}$ of the latter and it is computed as:

$$\alpha_{h,l} = \frac{2V_0}{V_{h,l} + V_{H,h}} \tag{3.43}$$

where $V_0 = \frac{4}{3}\pi\delta^3$ is a volume of full spherical neighborhood, whereas $V_{H,h}$ and

$V_{H,l}$ are the actual volume of the *neighborhood* of $\boldsymbol{X}_{0,h}$ and $\boldsymbol{X}_{0,l}$ respectively which are computed as:

$$V_{H,l} = \sum_{i=1}^{N_h} \Gamma_{h,l} \Delta V_l \tag{3.44}$$

and the expression of $V_{h,l}$ is obtained by changing the subscripts h with l.
The result is the increasing of the micro-modulus of the bond around material points near surfaces, so that, under homogeneous deformation, each point has the same *strain energy density* as that of inner points.

3.3.9 Computation of tangent, normal and bi-normal interface vectors

In section 3.3.7, it is described the algorithm that is used in the present manuscript for detection and tracking the solid boundaries. Once interface has been detected, it is possible to compute an estimation of the tangent, normal and bi-normal vectors. Their knowledge is important for the computation of fluid-dynamic forces on the solid-fluid interface as it will be reported in the section 3.5.3. The unity normal vector, $\hat{\boldsymbol{n}}_h$, evaluated at Lagrangian position $\boldsymbol{X}_h$, can be easily computed considering the volume average of the relative positions, $\boldsymbol{X}_l - \boldsymbol{X}_h$, of the centroids of the material particles, $\boldsymbol{X}_{0,l}$, included in the *neighborhood* of $\boldsymbol{X}_{0,h}$, for which the bond $\boldsymbol{\xi}_{h,l}$ are active, as it is represented by the following expression:

$$\boldsymbol{n}_h = -\frac{\sum_{h=1}^{N_h} (\boldsymbol{X}_l - \boldsymbol{X}_h) \lambda_{h,l} \Gamma_{h,l} \Delta V_l}{\sum_{h=1}^{N_h} \lambda_{h,l} \Gamma_{h,l} \Delta V_l} \tag{3.45}$$

$$\hat{\boldsymbol{n}}_h = \frac{\boldsymbol{n}_h}{\|\boldsymbol{n}_h\|}. \tag{3.46}$$

Once the normal vector is know, the unity tangent vector is calculated accordingly from the relation $\hat{\boldsymbol{n}}_h \cdot \hat{\boldsymbol{t}}_{\boldsymbol{h}} = 0$. Finally, the bi-normal unity vector is obtained as $\hat{\boldsymbol{b}}_h = \hat{\boldsymbol{t}}_h \times \hat{\boldsymbol{n}}_h$. As the peridynamic discretization approaches, the computed approximation of the normal, tangent, bi-normal vectors converge to their exact counterparts as the *m-convergence* and *δ-convergence* conditions [18].

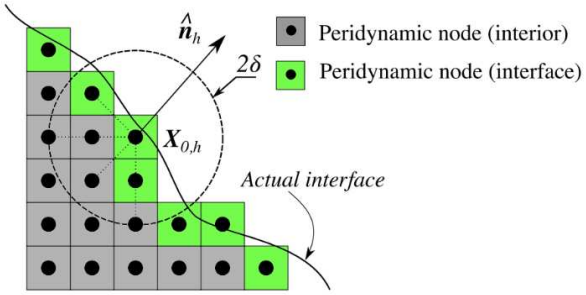


Figure 3.12: Representation of the schematic procedure employed to compute the unity normal vector to the solid boundaries at Lagrangian position $\boldsymbol{X}_h$. Figure taken from [43]

3.3.10 Temporal integration

For the temporal integration of the discrete governing equations of peridynamics full explicit low-storage third order Runge-Kutta time marching scheme is employed. It is summarized in the following:

do for r = *1,3*;
do for $h = N_p$;

$$\boldsymbol{X}_h^r = \boldsymbol{X}_h^{r-1} + \Delta t_s \left(\alpha_r \boldsymbol{V}_h^{r-1} + \beta_r \boldsymbol{V}_h^{r-2} - \gamma_r \boldsymbol{V}_h^{r-3} \right), \tag{3.47}$$

$$\boldsymbol{V}_h^r = \boldsymbol{V}_h^{r-1} + \frac{\Delta t_s}{\rho_s} \left(\alpha_r \boldsymbol{PRHS}_h^{r-1} + \beta_r \boldsymbol{PRHS}_h^{r-2} - \gamma_r \boldsymbol{PRHS}_h^{r-3} \right), \tag{3.48}$$

end do,
end do.

where r is the Runge-Kutta integration step, with $r = 0$ corresponds to the time level t_n, whereas $r = 3$ to t_{n+1}. Δt_s is the peridynamic solver time-step $t_{n+1} - t_n$; $\alpha_r, \beta_r, \gamma_r$ are the Runge-Kutta coefficients which are reported in the table below:

Runge-Kutta coeffiecients

r-step	$r = 1$	$r = 2$	$r = 3$
α_r	8/15	5/12	3/4
β_r	0	-17/60	-5/12
γ_r	8/15	2/15	1/3

PRHS is the right-hand side of the equation (3.38) which at step r is evaluated as:

$$\boldsymbol{PRHS}_h^r = \sum_{l=1}^{N_h} \left(\boldsymbol{t}_{h,l}^r - \boldsymbol{t}_{l,h}^r\right)\Gamma_{h,l}\Delta V_l \; + \; \boldsymbol{F}_h^r \; + \; \boldsymbol{C}_h^r \; + \; \boldsymbol{D}_h^r \,, \tag{3.49}$$

where the difference $\boldsymbol{t}_{h,l}^r - \boldsymbol{t}_{l,h}^r = \boldsymbol{f}_{h,l}^r$ is computed using the constitutive relation for *bond-based peridynamics* provided in equation (3.36).
As previously reported, the Runge-Kutta time marching scheme is an explicit method and so it propagates the solutions from the instant t_n to instant t_{n+1}, without any equilibrium being verified. This type of methods allow to increase the computational velocity but they need that the time-step employed is lower than a threshold value, so that the stability of the method is maintained. The maximum possible time-step, named CFL_{max} (from Courant-Friedrichs-Lewy conditions) is provided below:

$$\Delta t_{max} = \frac{\left(\|\boldsymbol{\xi}\|_{min}\right)}{\left(c_k\right)_{max}} \tag{3.50}$$

where $\left(\|\boldsymbol{\xi}\|_{min}\right)$ is the minimum bond length, that in the present case is equal to Δs, $\left(c_k\right)_{max}$ is the maximum speed at which information propagates in the material and it corresponds to the sound speed in the material which is computed as:

$$\left(c_k\right)_{max} = \sqrt{\frac{E}{\rho}} \tag{3.51}$$

To increase the certainty that the solution is stable and similar to the reality as much as possible, it is suggested to use a time-step $\Delta t = 0.5\Delta t_{max}$.

3.3.11 Boundary condition of solid domain

In general boundary conditions are not necessary for the solution of the integro-differential equations of peridynamics. However, boundary conditions can be imposed by prescribing specific values to the displacement and velocity, at each time level of a temporal integration scheme, on a ghost layer of material particles distributed along the boundary of a discrete peridynamic solid body. To be sure that the imposed conditions are sufficiently reflected on the actual material region the extent of the ghost layer is set to the *horizon* δ.

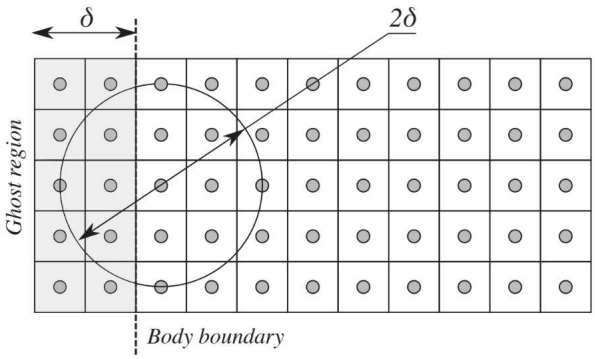


Figure 3.13: Schematics of the ghost-layer of material particles employed to impose boundary conditions on the boundaries of the solid domain. Figure taken from [40]

3.4 The fluid phase: Navier-Stokes equations

Fluid motion is described by a set of differential equations that are called Navier-Stokes equations. These relations mathematically express the conservation of mass and momentum for a viscous fluid that evolves in time and space; usually, they can be completed by an equation of state (energy balance equation) which links pressure, temperature and density together. They are expressed as:

$$\frac{\partial \rho}{\partial t} = -\frac{\partial \rho u_j}{\partial x_j} \tag{3.52}$$

$$\frac{\partial \rho u_i}{\partial t} = -\frac{\partial}{\partial x_j}\left(\rho u_i u_j + p\delta_{ij}\right) + \frac{\partial \sigma_{ij}}{\partial x_j} \tag{3.53}$$

$$\frac{\partial \rho E}{\partial t} = -\frac{\partial}{\partial x_j}\big((\rho E + p)u_j\big) + \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j}\right) + \frac{\partial(\sigma_{ij} u_{ij})}{\partial x_j} \tag{3.54}$$

where ρ indicates the fluid density, p the hydrodynamic pressure, T the temperature, $u_{i=1}^{3}$ the three velocity components, $E = e + e_k$ the total energy, e the internal energy and $e_k = 1/2 u_k u_k$ the kinetic energy. μ is the dynamic viscosity of the fluid, λ the diffusivity and finally σ_{ij} the viscous stress tensor computed as:

$$\sigma_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_s}{\partial x_s}\right) \tag{3.55}$$

The set of (3.52, 3.53, 3.54) consist of 5 equations in total and 7 variables; for this reason, they must be coupled with a constitutive model for both the fluid ($\rho = \rho(p, T)$) and the internal energy ($e = e(p, T)$). The equations just presented are valid for compressible fluids, but in the present work the fluis is

considered as incompressible. For this type of fluid ρ, μ and λ can be considered constant and the Navier-Stokes equations are simplified in the following way:

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{3.56}$$

$$\frac{\partial u_i}{\partial t} = -u_j \frac{\partial u_i}{\partial x_j} - \frac{\partial p \delta_{ij}}{\partial x_j} + \frac{\mu}{\rho} \frac{\partial^2 u_i}{\partial x_i xj} \tag{3.57}$$

$$\frac{\partial T}{\partial t} + u_i \frac{\partial T}{\partial x_j} = \lambda \frac{\partial^2 T}{\partial x_i x_j} \tag{3.58}$$

As it is possible to see from the equations (3.56, 3.57, 3.58) the mass e momentum equations for a incompressible fluid form a self-consistent system which does not require the energy balance equation to be solved. These last equations are usually presented in their non-dimensional form which are obtained from Buckingham theorem by implementing the *Reynolds number*:

$$\frac{\partial u'_j}{\partial x'_j} = 0 \tag{3.59}$$

$$\frac{\partial u'_j}{\partial t} = -u'_j \frac{\partial u'_i}{\partial x'_j} - \frac{\partial p' \delta_{ij}}{\partial x'_j} + \frac{1}{Re} \frac{\partial^2 u'_i}{\partial x'_i x'_j} \tag{3.60}$$

where the superscript (‘) indicates non-dimensional quantities. From equation (3.60) it can be easily seen that the convective-transport term is non-linear; for this reason, any convective flow is characterized by the non-linearity effect contained in the momentum equation.

As it was reported at the beginning, in the present thesis a novel numerical method is employed to address three-dimensional FSI problems with flow-induced fracturing of solids, where the physics of fluid flows is fully resolved. The incompressible formulation of the Navier-Stokes equations is used to reproduce the dynamics of the fluid phase in the frame of the three-dimensional *Direct Numerical Simulation (DNS)* of the incompressible Navier-Stokes equations. The latter relies on a well-established numerical framework that allows for accurate and high-resolution simulations, where all the spatial and temporal scales of the flow are directly resolved from first principles on a computational grid, e.g. without the need of a turbulence model.

3.4.1 Direct numerical simulation (DNS)

The most intuitive approach for solving Navier-Stokes equations consists of discretizing the fluid domain and directly solve equations (**??**) without introducing any turbulence approximation to simplify the computing process. This technique is called *Direct Numerical Simulation (DNS)*, as fluid equations are solved

directly [9], unlike the turbulence-models based techniques, and the results obtained represent an accurate and detailed reproduction of the flow under examination such that the simulation has the same value of a practical experiment. It is, for this reason, the most fundamental and precise simulation strategy, but since the considered equations enclose all the physics of the turbulent flows, it requires a truly dense computational grid both from spatial and temporal point of views, therefore the computational cost is very high.
Considering a generic turbulent flow, the introduction of kinetic energy takes place on space-scale comparable to the dimension of the body that generates the turbulence, than the transfer of this kinetic energy occurs from the large-dimension eddies to the smallest eddies where the energy is dissipated into heat. The dimension of dissipative scale, usually identified as the Kolmogorov's scale, represents the fundamental dimension of eddy formation and so the cell spacing of the mesh employed must be comparable to this dimension to correctly resolve the fluid motion and represent turbulence structures [46].
From Kolmogorov's theory (1941), we can evaluate the kinetic energy dissipation rate ϵ as:

$$\epsilon \propto \frac{U^3}{L} \tag{3.61}$$

where U is the velocity scale of the flow and L is the scale length for larger turbulent structures. From ϵ we can denote Kolmogorov's scale η as:

$$\eta \propto \frac{\nu^3}{\epsilon^{\frac{1}{4}}} \tag{3.62}$$

with ν the kinematic viscosity of the fluid.
The number of points for each direction is so $N_{x_i} = L_{x_i}/\Delta x_i \propto L/\eta$ with $i = 1, 2, 3$. An estimate of the total number of nodes to properly solve the fluid domain, remembering that the Reynolds number is $Re = UL/\nu$ is:

$$N \propto N_{x_i}^3 \propto \frac{L}{\eta} = \frac{L}{\left(\nu^3/\epsilon\right)^{\frac{1}{4}}} \propto \left(\frac{L^4U^3}{L\nu^3}\right)^{\frac{1}{4}} = Re^{\frac{9}{4}} \tag{3.63}$$

From this relation it is clear that the number of grid nodes necessary for correctly simulating a fluid flow dynamic grows fast with the Reynolds number.
Additionally, if we introduce also the Kolmogorov's velocity scale as $u_\eta \propto (\nu\epsilon)^{1/4}$ and the corresponding time scale $\tau_\eta \propto \eta/u_\eta \propto (\nu/\epsilon)^{1/2}$ the total computational time for a generic fluid domain is obtained:

$$T_{tot} \propto N \cdot \frac{U}{L} \propto Re^{11/4} \approx Re^3 \tag{3.64}$$

that remarks the negative effect of high Reynolds number on computational cost of *Direct Numerical Simulation*. For this reason, as in the present case, the *DNS* technique is used for low Reynolds numbers.

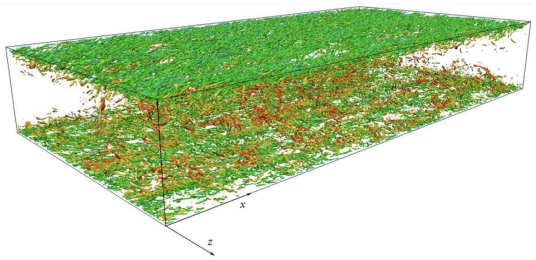


Figure 3.14: Vortical structure reproduced by DNS technique of a fluid channel flow with $Re_\tau = 600$ (vortical structures are colored with values of local streamwise velocity, reddish indicates high values, greenish indicates low values). Figure taken from [29]

3.4.2 Pressure-correction method

As it is possible to see from the incompressible formulation of the Navier-Stokes equations (**??**, the hydrodynamic pressure is a function of the space and time without an explicit link with the others terms that appear in the equations. Anyway, if for the compressible case the expression for the pressure is given from state equation ($p/\rho = RT$, for example) for incompressible case there is no explicit link between pressure and the other variables appearing in the equations. In other words, it is not trivial to express the pressure as a function of the velocity components and vice-versa. To make the link between pressure and velocity evident a manipulation of the incompressible Navier-Stokes momentum balance equation is necessary [47] from which the *Poisson equation* is obtained:

$$\frac{\partial^2 p}{\partial x_i x_j} = -\frac{\partial}{\partial x_i}\left(\frac{\partial u_i u_j}{\partial x_j}\right) \tag{3.65}$$

This equation expresses the equality between the laplacian of the pressure field to the divergence of the non-linear component of the convective term. Therefore the pressure is the real constraint that the mass equation exerts on the equations to ensure that at any time step the velocity field is solenoidal. Based on these considerations, the *pressure-correction method* is employed to solve the Navier-Stokes equations. A summary description is reported below.
As long as the pressure field influences instantaneously the velocity field due to the enforcement of mass conservation throughout the fluid domain, it is necessary for them to be solved together. This is impossible from a numerical point of view, as this coupling effect causes the algorithm to be implicit. *Pressure correction method* provides a solution to this issue, dividing the integrational step into two sub-steps following a predictor-corrector approach. The first sub-step assumes a first-attempt non-zero divergence velocity field u^* on the basis of the current velocity and pressure field; a predicted pressure field is then computed via the Poisson's equation (3.65). The second sub-step then proceeds to correct the velocity field via the predicted pressure field $\hat{p}$; the pressure p is

also updated consequently [4]. The method employed is summarized with the following steps:

1. Starting from the velocity and pressure field $\boldsymbol{u}^n$ and p^n, the predicted velocity field is computed as:

$$\boldsymbol{u}^* = \boldsymbol{u}^n + \Delta t(\boldsymbol{C}^n + \boldsymbol{D}^n - \nabla p^n) \tag{3.66}$$

   where $\boldsymbol{C}^n$ and $\boldsymbol{D}^n$ are the convective and diffusive term in the integrated equation.

2. Pressure field is then predicted in the following way:

$$\nabla^2 \hat{p} = \frac{\nabla u^*}{\Delta t} \tag{3.67}$$

3. finally, the velocity and pressure field is corrected $\boldsymbol{u}^{n+1}$ and p^{n+1} with:

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^* + \Delta t \cdot \nabla \hat{p}; \tag{3.68}$$

$$p^{n+1} = p^n + \hat{p}. \tag{3.69}$$

3.4.3 Spatial discretization and temporal integration

The computational domain for the fluid phase is a rectangular box region of dimensions $L_x \times Ly \times L_z$ along the x, y and z directions, respectively. For its discretization a fixed and structured computational *Eulerian* grid is used, with $N_x \times N_y \times N_z$ nodes along the same directions. Additionally, it is built as uniform and equally spaced along the three directions, with $\Delta_f = \Delta_x = \Delta_y = \Delta_z$ the grid size, that makes easier the prescription of boundary conditions from the *Immersed Boundary Method (IBM)* which is discussed in section 3.5.1. To prevent issues like odd-even decoupling between pressure and velocity the grid follows a staggered layout where the scalar quantities (i.e. *pressure*) being evaluated at the center of the grid cells, while vectorial ones on the cell faces (i.e. *velocity*) [16]. For the discretization of the Navier-Stokes equations (3.59)-(3.60) on the computational grid, second-order accurate finite difference schemes are employed and, to advance the solution in time, as the peridynamic temporal integration, a fully explicit Runge-Kutta time marching scheme is adopted [35]. To impose the boundary conditions on limit boundaries of the rectangular box domain, a layer of ghost nodes for both cell-centered and face-centered quantities is defined. The figure below represents the computational domain and a two-dimensional case reproduction of the Cartesian grid adopted.

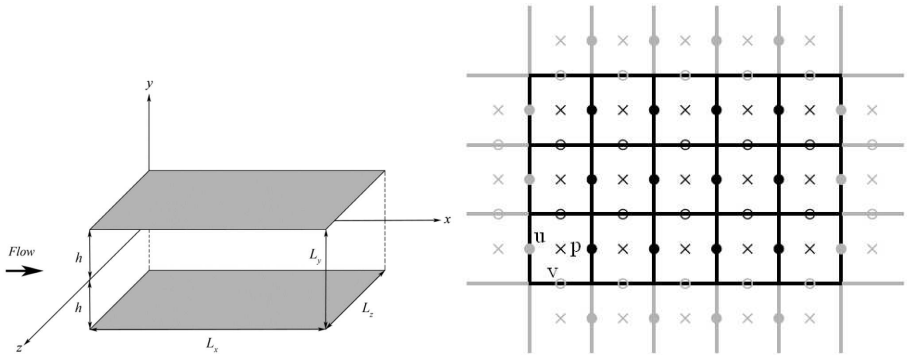


Figure 3.15: Left panel: sketch of the computational domain of the fluid phase. Right panel: two-dimensional representation of the staggered Cartesian grid with scalar quantities evaluated at the center of the grid cells, while vectorial ones on the cell faces. In light gray is represented the ghost layer adopted to impose boundary conditions on limit boundaries domain. Figure taken from [41]

The time marching scheme, employed to advance the solution in time, is summarized in the following:

do for $r = 1, 3$,

$$\boldsymbol{u}^* = \boldsymbol{u}^{r-1} + \frac{\Delta t}{\rho_f}\left(\alpha_r \boldsymbol{RHS_f}^{r-1} + \beta_r \boldsymbol{RHS_f}^{r-2} - \gamma_r \nabla p^{r-3/2}\right), \tag{3.70}$$

$$\nabla^2 \hat{p} = \frac{\rho_f}{\gamma_r \Delta t} \nabla \cdot \boldsymbol{u}^*, \tag{3.71}$$

$$\boldsymbol{u}^r = \boldsymbol{u}^* - \frac{\gamma_r \Delta t}{\rho_f} \nabla \hat{p}, \tag{3.72}$$

$$p^{r-1/2} = p^{r-3/2} + \hat{p}, \tag{3.73}$$

end do,

where the Δt is the time step, $\boldsymbol{u}^*$ is a non-solenoidal approximation of the velocity field, in according to the *pressure-correction method* summarized in section 3.4.2, denoted *first prediction velocity*, whereas the coefficients α_r, β_r and γ_r are the Runge-Kutta coefficients; the superscript r refers to the sub-steps of the time marching scheme, with $r = 0$ that corresponds to t_n and $r = 3$ to t_{n+1}. Finally, the right-hand side term, $\boldsymbol{RHS_f}$, evaluated at the sub-step r is computed as:

$$\boldsymbol{RHS_f}^r = -\rho_f \nabla \cdot \left(\boldsymbol{u}^r \boldsymbol{u}^r\right) + \mu_f \nabla^2 \boldsymbol{u}^r. \tag{3.74}$$

3.5 Immersed Boundary Method IBM

The *Immersed boundary Method (IBM)* is a popular technique which allows to simulate bi-phase fluids or, in general, the fluid motion around elements with complex-geometry. The method was originally proposed by Peskin (1972) to describe anatomical phenomena regarding cardiac mechanics related to blood flow. It was then developed and employed to solve the rising complexity of most industrial applications related to the presence of deformable solid bodies with complex geometry immersed in a fluid flow. In fact, as it is know from fluid-dynamic, when a solid object is immersed in a fluid domain, no-slip and no-penetration boundary conditions have to be prescribed on the solid-fluid interface and this, from a numerical point of view, it is not straightforward.
Originally, in situations were FSI problems were involved the so-called *Arbitrary Lagrangian-Eulerian (ALE)* [3] method was employed. This method directly imposes specific values for scalar and vector quantities on the grid nodes located on solid-fluid interface, by using a body fitted grid that is continually updated at every time step such that it conforms to the shape of the solid-surfaces at each discrete time level. However, the needs to re-grid and compute all the fluid-dynamic fields at every time step, especially in the case of DNS analysis, make the computational cost of this kind of simulations quite large such as to make ALE method impracticable for the most complex cases.
On the other hand, IBM [2, 11, 12] does not require the computational grid for the fluid phase to conform to the shape of the immersed solids, so it is used a non-body conformal Eulerian mesh which covers all the fluid domain, without considering the surface of the solid body. In this case, it still exists a Lagrangian surface grid, as some nodes are positioned on the interface of the body, so the solid boundaries cut through the grid elements, preventing traditional incorporation of the boundary conditions.

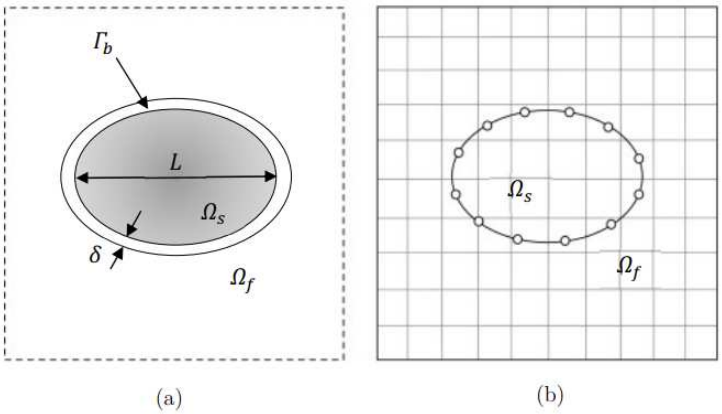


Figure 3.16: Generic solid body immersed in a fluid flow (a). The body is defined by its occupied volume Ω_s and surface Γ_b. Its characteristic length is L and δ is the thickness of the boundary layer. (b) Schematic of body immersed in a non-body conformal Eulerian grid employed in the framework of IBM where Ω_s denotes solid-domain, whereas Ω_f the fluid-domain . Figure taken from [12].

Due to this non-conformal nature of the mesh, no-slip and no-penetration wall conditions on the fluid-solid interfaces are prescribed in a indirect manner by forcing the flow to move with the same local-velocity of the solid (zero relative velocity) in its immediate proximity. This is possible by applying an additional force per unit mass q to the right-hand side of the momentum equation of the fluid phase, obtaining the expression below:

$$\rho_f \left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) = -\nabla p + \mu_f \nabla^2 \boldsymbol{u} + \rho_f \boldsymbol{q} \tag{3.75}$$

It is worth remarking that the use of non-conformal and stationary structural meshes features an important downside that is the impossibility to control the grid resolution in the proximity of the body and hence, so that the DNS analysis can be performed, the grid employed must be very dense throughout all the fluid domain. Nevertheless, the computational cost is not too high, since the need of re-mesh at each discrete time level is removed and the fluid equations are not solved for the nodes inside the solid body.
In the present case, the *Immersed Boundary Method (IBM)* [22] is employed to impose no-slip and no-penetration boundary conditions on the solid-fluid interfaces, that are summarized in *zero-relative velocity condition*. Hence the time marching scheme described above is modified by adding the contribution of the force $\boldsymbol{q}$ to the prediction velocity. This additional term is computed, through interpolation and spreading operations based on regularized Dirac delta function δ_Δ, between fluid nodes of Cartesian-Eulerian grid and peridynamic nodes of Lagrangian interface grid, according to an iterative scheme, named *multidirect forcing*. Then it is added at *first prediction velocity*, computed by equation

(3.66), obtaining the *second prediction velocity* $\hat{\boldsymbol{u}}$ which is used in the Poisson equation to calculate $\hat{p}$. The Dirac delta function and *multidirect forcing* method are discussed in the next sections.

3.5.1 Regularized Dirac Delta Function

A quantity which is expressed in the Eulerian reference system can be expressed in the Lagrangian one through an interpolation operation, whereas the inverse procedure is given by a spreading operation. In the present case; since there is the necessity to interpolate the fluid velocity in the Lagrangian interface grid for the computation of the forcing term used to impose boundary condition, and then it must be spreaded to the Eulerian grid so it can be added in momentum equation of the fluid; these interpolation and spreading operations are based on a regularized Dirac delta function δ_Δ, defining according to Roma et al. [6]:

$$\delta_\Delta(\boldsymbol{x}-\boldsymbol{x}_0) = \delta'_\Delta(x-x_0)\delta'_\Delta(y-y_0)\delta'_\Delta(z-z_0), \tag{3.76}$$

$$\delta'_\Delta(s-s_0) = \frac{1}{\Delta_f}\boldsymbol{\Phi}\left(\frac{s-s_0}{\Delta_f}\right), \tag{3.77}$$

$$\boldsymbol{\Phi}(t) = \begin{cases} \frac{1}{3}\left(1+\sqrt{1-3t^2}\right), & t \le 0.5, \\ \frac{1}{6}\left(5-3|t|-\sqrt{1-3(1-|t|)^2}\right), & 0.5 < |t| < 1.5, \\ 0, & |t| > 1.5 \end{cases} \tag{3.78}$$

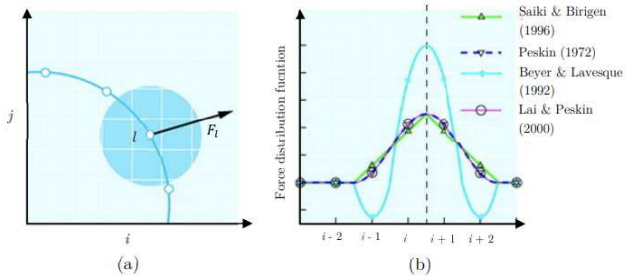


Figure 3.17: (a) Spreading of forcing term, indicated as $\boldsymbol{F_l}$ from Lagrangian boundary point $\boldsymbol{X_l}$ to surrounding Eulerian fluid nodes (shaded region signifies the extent of the force distribution). (b) Interpolation/distribution functions employed in various studies. Figure taken from [12].

The function Φ is defined like this because it has to fulfil some properties. In particular:

1. $\Phi(t)$ is a continuous function for all the real numbers t;
2. $\Phi(t) = 0$ for $|t| > 1.5$;
3. $\sum_i \phi(t-i) = 1 \quad \forall t$;
4. $\sum_i (t-i)\phi(t-i) = 1 \quad \forall t$;
5. $\sum_i \{\phi(t-i)\}^2 = \frac{1}{2} \quad \forall t$

with $-\infty < t < +\infty$.
Through the property (1) it is guaranteed that there are not discontinuities in interpolation and spreading operations; the property (2) guarantees that the influenced region of the function δ_Δ is limited (in this case is set to three Eulerian cells); the properties (3) and (4) guarantee together the conservation of angular momentum; the property (5) arises from the consideration that the forces acting on a interface material point affect the motion of that point and so, the property impose that the influence is always the same, independently from the point position relative to the mesh.
However, the use of a discrete kernel for interpolation and spreading operations involves the replacement of the sharp solid-fluid interfaces in the continuum case by a thin porous shell of "solidified fluid" in their discrete representation. The latter is centered on the nominal fluid-solid interfaces and has a spatial extension of the order of the support of the specific kernel employed, which in the present case is equal to $3\Delta_f$. Consequently the dimensions of the immersed solid objects result bigger because they are projected outwards of a length of $3\Delta_f/2$ relatively to their nominal dimension. This effect influences the overall-fluid dynamic forces acting on the immersed bodies and so it must be mitigate. To resolve this problem, as it is suggested by Breugem [22], it is employed a retraction distance $d_r = 0.5\Delta_f$ that involves an inward retraction of the solid surfaces so the estimation of the overall fluid-forces acting on the immersed body are more accurate.

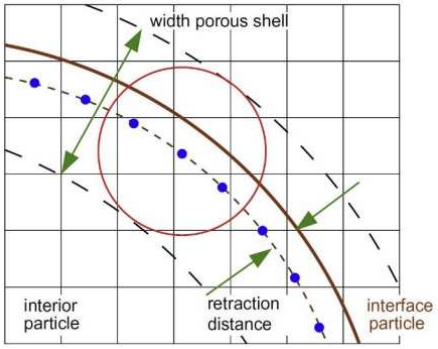


Figure 3.18: Sketch of the actual solid-fluid interface and spurious porous shell of "solidified" fluid in the proximity of the interface due to the use of discrete kernel for interpolation and spreading operations from regularized Dirac delta function. Figure taken from [22].

3.5.2 Multidirect Forcing method

Since the target of *Immersed boundary method (IBM)* is to impose no-slip and no-penetration wall conditions on solid-fluid interfaces, the idea is to impose an additional forcing term $\boldsymbol{q}$ to the discrete momentum equation of the fluid so that velocity of the fluid in correspondence to the solid boundaries, obtained by interpolation procedures, is equal, within a certain tolerance, to the Lagrangian velocity of material particles located on the interface. Hence, the use of a regularized Dirac delta function for spreading operations involves the distribution of the forcing term around the interface of the solid body. In figure below the circles represent the action radius of the delta function for the two Lagrangian interface points denoted by triangles.

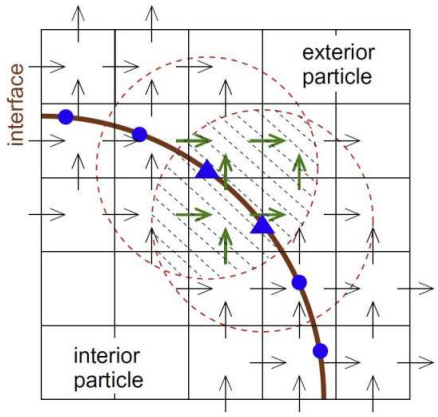


Figure 3.19: Illustration of the diffuse distribution of the IBM force around the interface of a particle. Lagrangian grid points are indicated with dots; the arrows indicate the force distribution over the Eulerian nodes. The circles identify the Dirac delta's range applied to the Lagrangian points designated by triangles. With dotted line is represented the overlapping zone where more Eulerian points force, at the same time, different Lagrangian points. Figure taken from [22].

Note that the circles overlap in some zones, indicating that Eulerian points in the overlapping zone force both Lagrangian points at the same time. Consequently, the fluid-velocity that has to be imposed in correspondence of the considered Lagrangian point may not be well imposed by the additional forcing term. To make the calculation of the forcing term as accurate as possible, an iterative procedure, named *multidirect forcing* [22] is employed. Therefore, considering the configuration of a solid object immersed in the computational domain evaluated at a generic time level t_n, the force $\boldsymbol{q}$ and so the *second prediction velocity* $\hat{\boldsymbol{u}}$ are computed after the solution of equation (3.70) and before the solution of equation (3.71) according to the *multidirect forcing* scheme which is reported below:
do for $s = 1, N_s$,
do for $h = 1, N_p$,

$$\hat{\boldsymbol{U}}_h^{s-1} = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} \hat{\boldsymbol{u}}_{i,j,k}^{s-1}\delta_\Delta\big(\boldsymbol{x}_{i,j,k} - \boldsymbol{X}_h^n\big)\Delta_f^3, \tag{3.79}$$

$$\boldsymbol{Q}_h^{r-1/2,s} = \boldsymbol{Q}_h^{r-1/2,s-1} + \frac{\boldsymbol{V}_h^n - \hat{\boldsymbol{U}}_h^{s-1}}{\Delta t}, \tag{3.80}$$

end do,

do for $[i,j,k]=[1,1,1],[N_x,N_y,N_z]$,

$$\boldsymbol{q}_{i,j,k}^{r-1/2,s}=\sum_{h=1}^{N_p}\boldsymbol{Q}_h^{q-1/2,s}\delta_\Delta\big(\boldsymbol{x}_{i,j,k}-\boldsymbol{X}_h^n\big)\Delta V_h, \tag{3.81}$$

$$\hat{\boldsymbol{u}}_{i,j,k}^{s}=\boldsymbol{u}_{i,j,k}^{*}+\Delta t\boldsymbol{q}_{i,j,k}^{r-1/2,s}, \tag{3.82}$$

end do,
end do.

where s is the index of iterative scheme, with initial value of $\hat{\boldsymbol{u}}$ is $\hat{\boldsymbol{u}}^0=\boldsymbol{u}^*$; $\hat{\boldsymbol{U}}_h^s$ is the Lagrangian variable obtained by the interpolation of the *second prediction velocity* of the fluid, $\hat{\boldsymbol{u}}^s$, at each Lagrangian coordinate $\boldsymbol{X}_h^n$ located on solid-fluid interface and evaluated at time level t_n; $\boldsymbol{Q}_h^{r-1/2,s}$ is the Lagrangian force computed at position $\boldsymbol{X}_h^n$, obtained from the ratio of the relative velocity $\boldsymbol{V}_h^n-\hat{\boldsymbol{U}}_h^{s-1}$ with the time step of the fluid time marching scheme, where $\boldsymbol{V}_h^n$ is the Lagrangian velocity at time level t_n of the material particles located on the interface. The Eulerian expression of the forcing term $\mathbf{q}^{r-1/2,s}$ is obtained by the spreading on the Eulerian grid of the Lagrangian quantity $\boldsymbol{Q}_h^{r-1/2,s}$ and finally the *second prediction velocity* is re-computed on each node of the Eulerian grid by updating the prediction velocity, $\boldsymbol{u}^*$. N_s is the total number of force iteration, that it was demonstrated by Breugem et al. [22] that the maximum number of iterations can be arbitrarily chosen, but a low value is recommended to keep the calculation time down. Specifically, it has been proven that setting $N_s=2$ it is possible to obtain good results, namely a second order of accuracy about the velocity of the fluid in proximity of the solid boundaries, with a reasonable computational cost of the procedure [22].

3.5.3 Computation of hydrodynamic forces: normal probe method

In the frame of immersed deformable solid bodies in a fluid-domain it is necessary to compute the stress distribution on the solid surfaces, due to the interaction with the fluid-phase, for predicting the dynamics of deformable solids. In the present manuscript, this is realized by using the so-called *normal probe method*, where the back-reaction of the fluid on the solid is estimated via the evaluation of the hydrodynamic pressure and viscous stresses in the proximity of the fluid-solid interfaces, considering a linear variation for the pressure gradient and velocity field in the normal direction [32]. This hypothesis is satisfied if the grid spacing is small enough to fully resolve the inner boundary layer of the interfaces, that is already a requirement of direct numerical simulations addressed in the present work. To simplify the problem, it has been considered the case of laminar boundary layer, which covers a wide range of applications. In such that case the flow can be approximated by the boundary layer equations in a local orthogonal curvilinear coordinate system $(\xi-\eta-\zeta)$ according to the directions

defined by interface unity vectors tangent, normal and bi-normal, respectively (section 3.3.7 and 3.3.9). To better understand the hydrodynamic stress model employed according to this last assumption the reader refers to [32]. Once the local orthogonal frame of reference is defined for each interfacial Lagrangian position, $\boldsymbol{X}_h$, a probe of length $\Delta l = 2\Delta_f$ is sent along the normal vector [39] with **e** and **m** denote the probe tip and root, respectively. As it is shown for the two-dimensional case in the figure below, a $2 \times 2 \times 2$ stencil with spacing Δ_f centered on **e** is defined.

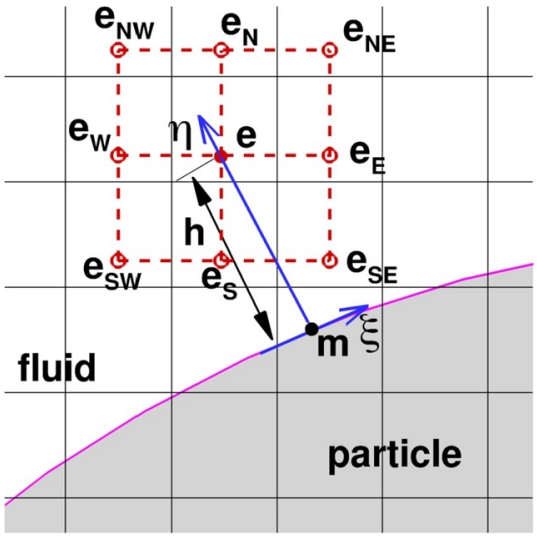


Figure 3.20: Two-dimensional scheme of the local curvilinear coordinate system and the stencil employed for the computation of the pressure and velocity gradient at the probe tip, **e**.

The use of the stencil permits to better approximate the value of the pressure and velocity field at the probe tip **e** and consequence on the probe root **m** and then in the solid-fluid interface. The pressure and velocity are computed on each point of the stencil via the regularized Dirac delta function defined by equation (3.78):

$$p|_{e_{a,b}} = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} p_{i,j,k}\delta_\Delta\left(\mathbf{x}_{i,j,k} - \mathbf{X}_{e_{a,b}}\right)\Delta_f^3 \,, \tag{3.83}$$

$$\mathbf{u}|_{e_{a,b}} = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} \boldsymbol{u}_{i,j,k}\delta_\Delta\left(\boldsymbol{x}_{i,j,k} - \boldsymbol{X}_{e_{a,b}}\right)\Delta_f^3 \,, \tag{3.84}$$

where $\boldsymbol{e}_{a,b}$ is a generic point of the stencil. Then, the pressure and velocity gradient calculation $\nabla p|_e$and$\nabla \boldsymbol{u}|_e$ is performed by central finite difference schemes using the point of the considered stencil. At this step, considering the linear

variation of the pressure gradient in the normal direction, the pressure is computed at the probe root **m** according to Wang et al. [39]:

$$p|_m = p|_e + \frac{1}{2}\left[\left.\frac{\partial p}{\partial \eta}\right|_m + \left.\frac{\partial p}{\partial \eta}\right|_e\right]\Delta l\,, \tag{3.85}$$

$$p|_e = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} p_{i,j,k}\delta_\Delta\big(\boldsymbol{x}_{i,j,k} - \boldsymbol{X}_{e_{a,b}}\big)\Delta_f^3\,, \tag{3.86}$$

$$\left.\frac{\partial p}{\partial \eta}\right|_m \approx \left.\frac{D\boldsymbol{u}}{Dt}\right|_m \cdot \hat{\boldsymbol{n}}_h \simeq \frac{d\boldsymbol{U}_h}{dt}\cdot\hat{\boldsymbol{n}}_h \tag{3.87}$$

$$\left.\frac{\partial p}{\partial \eta}\right|_e = \nabla p|_e \cdot \hat{\boldsymbol{n}}_h\,. \tag{3.88}$$

whereas, always from the laminar boundary layer assumption which allows to consider a linear variation of velocity field along the normal direction near the solid-fluid interface, the viscous shear stress is evaluated at the probe root, **m**, based on the velocity gradients computed at the probe tip, **e**:

$$\left.\frac{\partial u_\xi}{\partial \eta}\right|_m = \nabla \boldsymbol{u}|_e \cdot \hat{\boldsymbol{t}}\cdot\hat{\boldsymbol{n}}\,, \tag{3.89}$$

$$\left.\frac{\partial u_\eta}{\partial \eta}\right|_m = \nabla \mathbf{u}|_e \cdot \hat{\mathbf{n}}\cdot\hat{\mathbf{n}}\,, \tag{3.90}$$

$$\left.\frac{\partial u_\zeta}{\partial \eta}\right|_m = \nabla \boldsymbol{u}|_e \cdot \hat{\boldsymbol{b}}\cdot\hat{\boldsymbol{n}}\,, \tag{3.91}$$

which are employed to compute the shear and normal stresses at **m**:

$$\tau_\xi = \mu\left.\frac{\partial u_\xi}{\partial \eta}\right|_m\,, \tag{3.92}$$

$$\tau_\eta = \mu\left.\frac{\partial u_\eta}{\partial \eta}\right|_m - p|_m\,, \tag{3.93}$$

$$\tau_\zeta = \mu\left.\frac{\partial u_\zeta}{\partial \eta}\right|_m\,. \tag{3.94}$$

Then, with an appropriate coordinate transformation, the stress components are expressed in the global frame of reference (x-y-z):

$$\begin{bmatrix}\tau_x\\ \tau_y\\ \tau_z\end{bmatrix} = \begin{bmatrix}\hat{t}_x & \hat{n}_x & \hat{b}_x\\ \hat{t}_y & \hat{n}_y & \hat{b}_y\\ \hat{t}_z & \hat{n}_z & \hat{b}_z\end{bmatrix}\begin{bmatrix}\tau_\xi\\ \tau_\eta\\ \tau_\zeta\end{bmatrix}\,. \tag{3.95}$$

Once the stress distribution on solid-fluid interface are known, it is finally possible to compute the force per unit volume acting on each interface material particle $\boldsymbol{X}_h$ as:

$$\boldsymbol{F}_h = -\boldsymbol{\tau}_h \frac{A_h}{\Delta_s^3} \,, \tag{3.96}$$

with A_h the face area of the discrete interfacial material particles $\boldsymbol{X}_h$ and it is computed at the beginning of the simulation as the ratio of the immersed solid surfaces to the total number of the interfacial material particles. When a formation of crack occurs, new interfacial material particles are generated and for these the area A_h is set to $A_h = \Delta_s^{2/3}$, with Δ_s the grid spacing. As it was reported in section 3.3.7 the interface detection algorithm considers the generation of new interface only if the gap between solid surfaces is larger than the *horizon* δ, by setting $\bar{r}_{thr} = \delta$, so if the gap is closer than δ, to avoid overloading the code, the hydrodynamic forces are not computed and the particles are considered as interior one.

Chapter 4

Numerical Implementation

In this chapter it is presented the implementation of the methodologies discussed in the previous chapter. The whole code is written in Fortran 90, compiled in double precision, and it is composed of three different modules: a fluid solver, a peridynamic solver and a module that manages the synchronization and coupling of the equations and the momentum exchange across the interface. The fluid solver has been developed on the open source solver CaNS by Costa [45], whereas the peridynamic solver and coupling module have been implemented by Prof. Dalla Barba. All the solvers have been validated by Prof. Dalla Barba. All the code is compiled in parallel through the portable Message Passing Interface (MPI) library.

4.1 Fluid Solver

The temporal integration is provided from a fully-explicit third-order Runge Kutta time-marching scheme, which advances the Navier-Stokes equations from the discrete time level t_n to t_{n+1} with Δt_f the size of time step, where as the spatial integration is provided by a second-order accurate finite difference scheme, as described in section 3.4.3.
Arbitrary periodic, Neumann or Dirichlet boundary conditions can be applied to the external boundary of the computational domain by means of a ghost nodes, as reported in section 3.4.3. In this way, for the staggered grid that is used in the problem, the Dirichlet boundary conditions are easily applied as in the following:

for the velocity:

$$u_{x(i,j,k)} = \frac{u_{x(i+1/2,j,k)} + u_{x(i-1/2,j,k)}}{2} = 0 \implies u_{x(i+1/2,j,k)} = -u_{x(i-1/2,j,k)} \,, \tag{4.1}$$

for the pressure:

$$p_{(i,j,k)} = 0 \qquad \text{imposed that on the interested nodes,} \tag{4.2}$$

whereas Neumann boundary conditions:

for the velocity:

$$\frac{\partial u_x}{\partial x} = \frac{u_{x(i+1/2,j,k)} - u_{x(i-1/2,j,k)}}{\Delta x} = 0 \implies u_{x(i+1/2,j,k)} = u_{x(i-1/2,j,k)} \,, \quad (4.3)$$

for the pressure:

$$\frac{\partial p}{\partial x} = \frac{p_{(i,j,k)} - p_{(i-1,j,k)}}{\Delta x} = 0 \implies p_{(i,j,k)} = p_{(i-1,j,k)} \,. \quad (4.4)$$

The initial conditions are imposed on the velocity field by putting directly the velocity values to the interested nodes, for example $u_{(i,j,k)} = c$.
No-slip and no-penetration boundary conditions on the immersed fluid-solid interfaces are been imposed by the direct-forcing immersed boundary scheme, described in section 3.5.2, that is called directly inside the Navier-Stokes solver. Then the forces per unit volume acting on the immersed solids are computed according to the equations (3.96). The figure 4.1 reports the flowchart of the principal steps of the fluid solver, whereas the figure 4.2 reports the implementation of the *multi-direct forcing boundary method* (IBM) into the fluid solver:

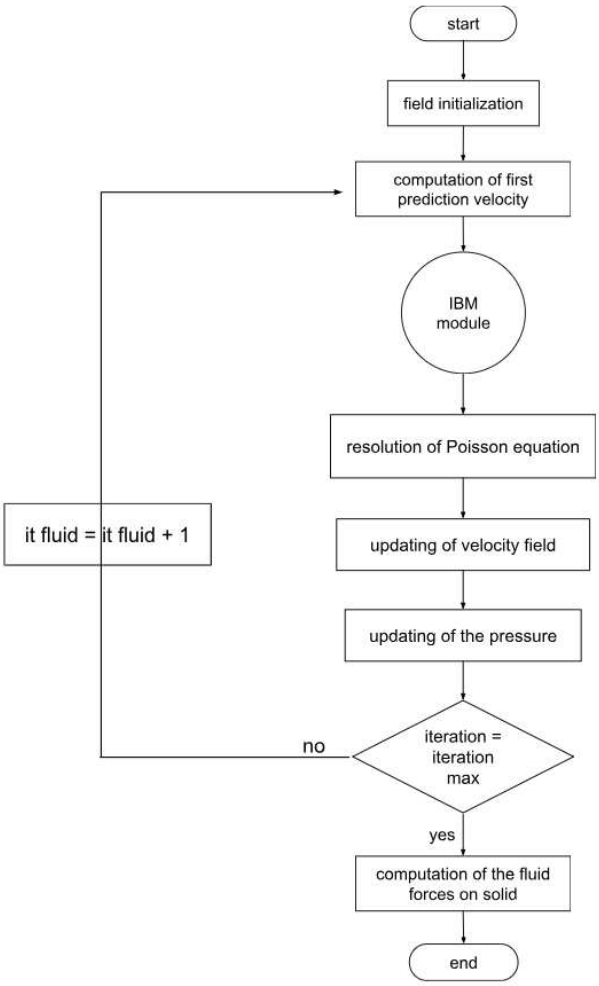


Figure 4.1: flowchart of the fluid solver

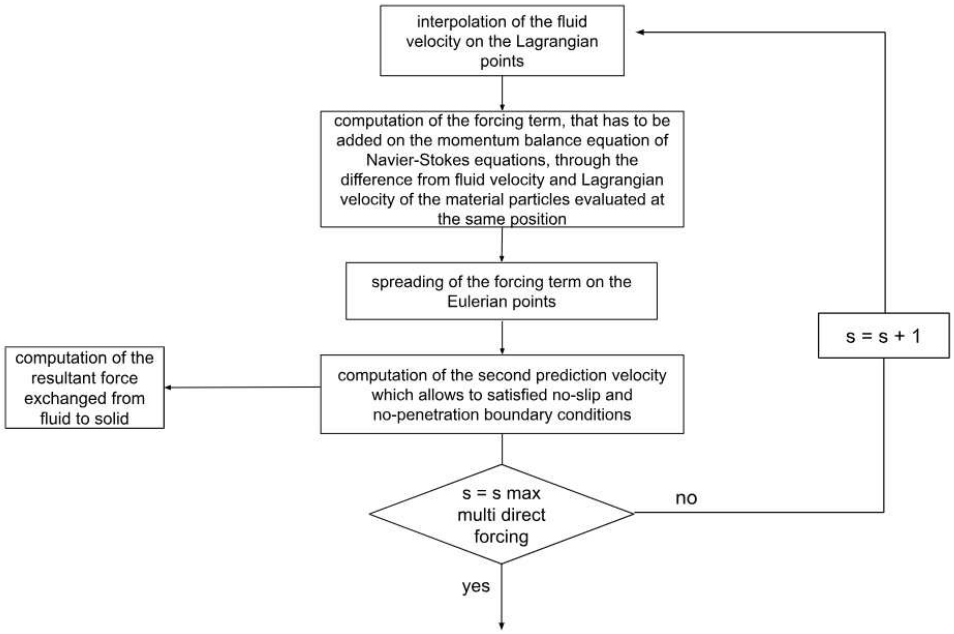


Figure 4.2: flowchart of the IBM module

4.2 Peridynamic solver

The solid bodies are discretized in their reference configurations (t = 0) in a set of cubic material particles with side equal to the Eulerian grid spacing, $\Delta s = \Delta f$. The temporal integration, like for the Navier-Stokes solver, is provided from a fully-explicit third-order Runge Kutta time marching scheme which advances the governing equations of peridynamics from time level t_n to t_{n+1}, with Δt_f the time step. All the discrete peridynamic quantities are computed as specified above. The flowchart of the peridynamic solver is reported in figure 4.3:

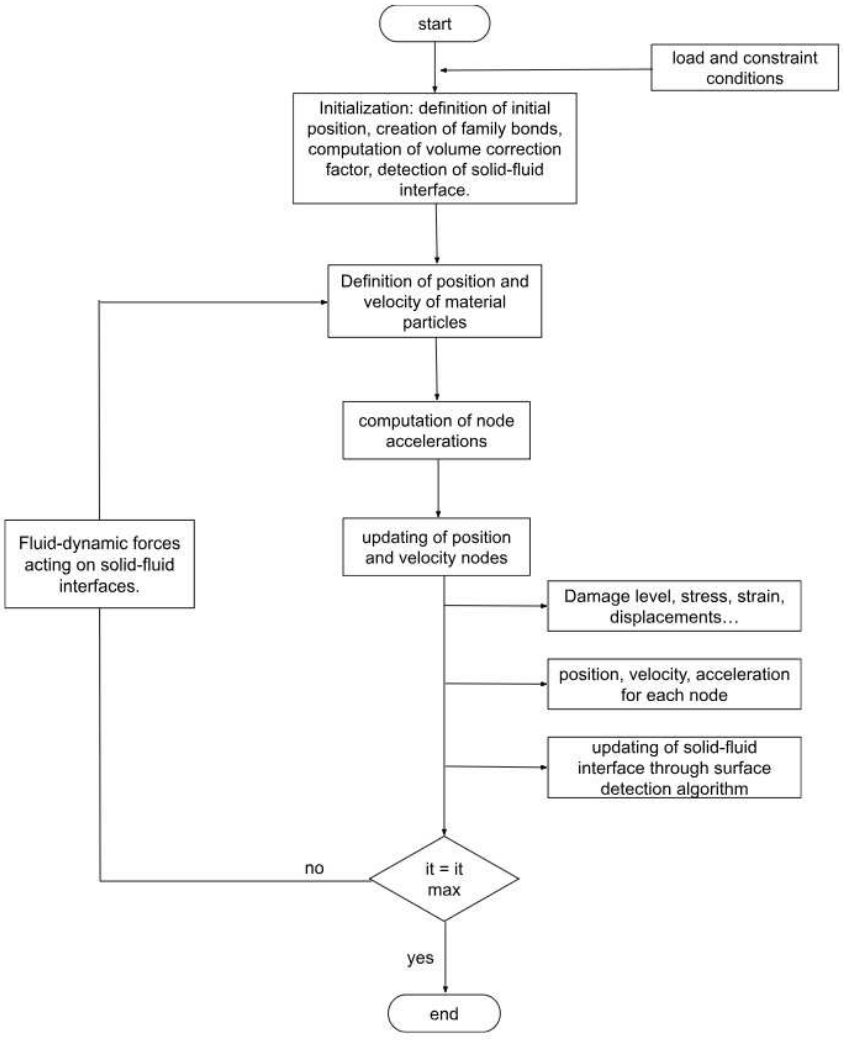


Figure 4.3: flowchart of peridynamic solver

4.3 Coupling of fluid solver and peridynamic solver

The code performs separately the two blocks relating to peridynamics and Navier-Stokes equations. These blocks are fully independent and exchange which each other the info of the applied loads on fluid-solid interfaces and the instant positions of the Lagrangian points of the solid bodies. The figure 4.4 reports the flowchart of the entire code. Thanks to this diagram it is possible to better understand how the two blocks dialogue with each other.
The scheme shows how the fluid-dynamic forces computed by fluid solver, which

are function of the fluid velocity field and of the instant position and velocity of the Lagrangian points, are given as input for the peridynamic solver. At the same way, it is highlighted how the Lagrangian point positions are computed just before solving the fluid-dynamic module. It is worth remarking that the maximum stable time step for the fluid is of the order of $\Delta t_f \propto \Delta f/U_f$, whereas that of the solid is $\Delta t_s \propto \Delta s/U_s$, with U_f the bulk velocity of the fluid and $U_s = \sqrt{E/\rho_s}$ the characteristic velocity-scale of the solid. In general, being $\Delta f = \Delta s$ and $U_s >> U_f$, the stability condition for the explicit peridynamic time-marching scheme is more severe than that for the stability of the fluid one, so the time step used by peridynamic solver is $\Delta t_s = R\Delta t_f$ with R a real number, $R \leq 1$. The factor $R = \Delta t_s/\Delta t_f \propto U_f/U_s$ is used to ensure the numerical stability. Being the characteristic time scale of the solid much lower than that of the fluid, the number of iterations required by the peridynamic time-marching scheme to achieve synchronization with the fluid one can be quite large, and increase with the decrease of R. Furthermore, the fully explicit coupling effects the temporal accuracy as well as the numerical stability of the scheme. In particular, it poses a lower limit for the solid-fluid density ratio, $\rho_s/\rho_f > 1$, due to an added-mass effect, such that stability issues do not arise.

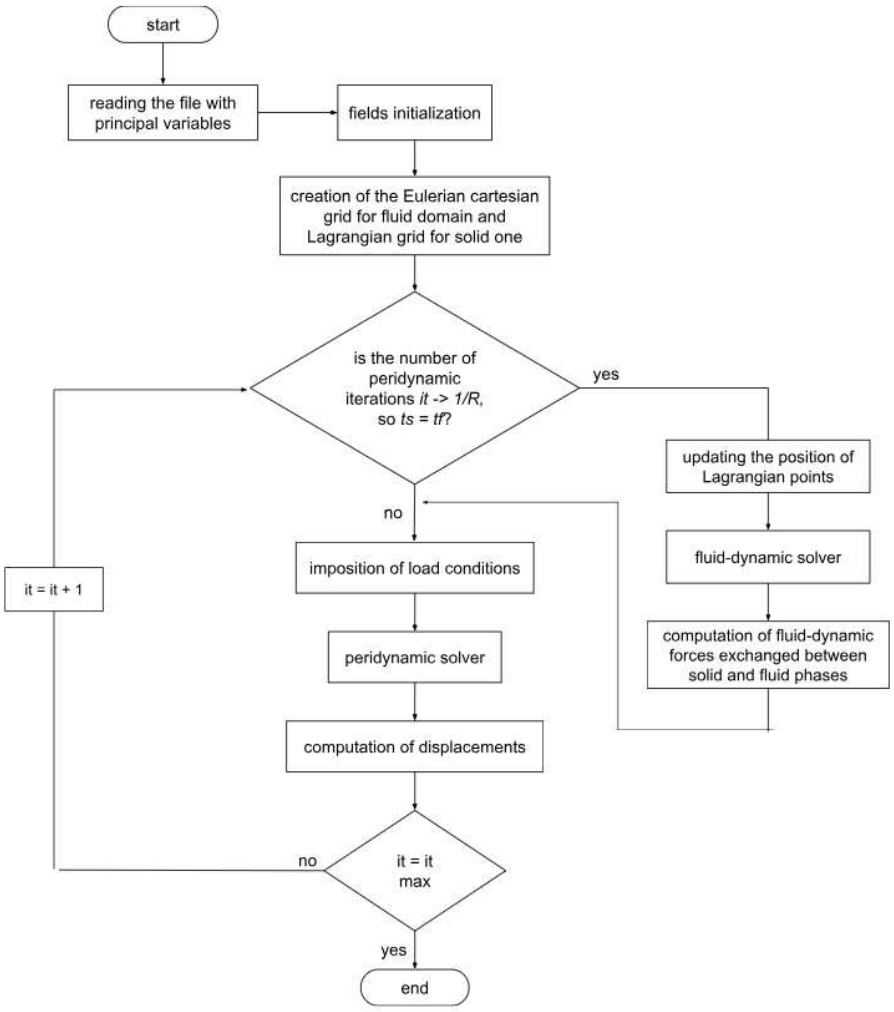


Figure 4.4: flowchart of the entire code.

Clarified how the code works, it is reported in the following a detailed representation of its the main steps:

do for $n = 1, N_t$

$t_{n+1} = t_n + \Delta t_f$

do for $r = 1, 3$

$$\boldsymbol{u}^* = \boldsymbol{u}^{r-1} + \frac{\Delta t_f}{\rho_f}\Big(\alpha_r \boldsymbol{FRHS}_f^{r-1} + \beta_r \boldsymbol{FRHS}_f^{r-2} - \gamma_r \nabla p^{r-3/2}\Big)$$

$$\hat{\boldsymbol{u}}^0 = \boldsymbol{u}^*$$

do for $s = 1, N_s$

do for $h = 1, N_p$

$$\hat{\boldsymbol{U}}_h^{s-1} = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} \hat{\boldsymbol{u}}_{i,j,k}^{s-1}\delta_\Delta\big(\boldsymbol{x}_{i,j,k} - \boldsymbol{X}_h^n\big)\Delta_f^3$$

$$\boldsymbol{Q}_h^{r-1/2,s} = \boldsymbol{Q}_h^{r-1/2,s-1} + \frac{\boldsymbol{V}_h^n - \hat{\boldsymbol{U}}_h^{s-1}}{\Delta t}$$

end do

do for $[i, j, k] = [1, 1, 1], [N_x, N_y, N_z]$,

$$\boldsymbol{q}_{i,j,k}^{r-1/2,s} = \sum_{h=1}^{N_p} \boldsymbol{Q}_h^{q-1/2,s}\delta_\Delta\big(\boldsymbol{x}_{i,j,k} - \boldsymbol{X}_h^n\big)\Delta V_h$$

$$\hat{\boldsymbol{u}}_{i,j,k}^s = \boldsymbol{u}_{i,j,k}^* + \Delta t \boldsymbol{q}_{i,j,k}^{r-1/2,s}$$

end do

end do

$$\nabla^2 \hat{p} = \frac{\rho_f}{\gamma_r \Delta t_f}\nabla \cdot \hat{\boldsymbol{u}}^{N_s}$$

Solving for $\hat{p}$

$$\boldsymbol{u}^r = \hat{\boldsymbol{u}}^{N_s} - \frac{\gamma_r \Delta t_f}{\rho_f}\nabla \hat{p}$$

$$p^{r-1/2} = p^{r-3/2} + \hat{p}$$

end do

Computing fluid forces on solid: $\boldsymbol{F}_h^n$

Updating solid positions and velocities via algorithm by peridynamic

solver. The cycle proceeds until synchronization is achieved, $t_s = t_{n+1}$:

do while $t_s \leq t_{n+1}$

$t_s = t_s + \Delta t_s$

do for $r = 1, 3$

do for $h = 1, N_p$

$$\boldsymbol{X}_h^r = \boldsymbol{X}_h^{r-1} + \Delta t_s \left(\alpha_r \boldsymbol{V}_h^{r-1} + \beta_r \boldsymbol{V}_h^{r-2} - \gamma_r \boldsymbol{V}_h^{r-3}\right)$$
$$\boldsymbol{V}_h^r = \boldsymbol{V}_h^{r-1} + \frac{\Delta t_s}{\rho_s} \left(\alpha_r \boldsymbol{PRHS}_h^{r-1} + \beta_r \boldsymbol{PRHS}_h^{r-2} - \gamma_r \boldsymbol{PRHS}_h^{r-3}\right)$$

Checking status of bonds

end do

end do

end do

Updating fluid-solid interface to time level t_f^{n+1} as described in section 3.3.7

end do

4.4 Organization of the code parallel architecture

As previously reported, the fluid domain has been discretized by a uniform and equally spaced Eulerian grid with nodes fixed in space, whereas the solid one has been discretized in a finite number of cubic material particles, each identified by the peridynamic node corresponding to their centroid, which its spatial position is reported in a Lagrangian frame and changes in time. For this reason, the definition of the parallel architecture of the code is not straightforward, because it is necessary to use different strategies for the Eulerian phase and Lagrangian one to have a uniform distribution of the computational operations as much as possible among the different MPI processes. Hence, it has been decided to adopt a *space-based decomposition* strategy for the Eulerian grid and an *index-based decomposition* for the Lagrangian one.
Specifically, the Eulerian domain is divided into $N_{MPi,x} \times N_{MPI,y}$ computational subdomains along x and y directions respectively and all the computation related to the space region covered by each subdomain are performed by a single MPI task, which stores all the related data in its own private memory area.

The so-called *halo cells* [35] is used in each subdomain to store a copy of the data pertaining to the boundary of the adjacent ones, since the use of the second-order, finite difference schemes for the spatial discretization requires the mutual communication of the variable stored in the cell-center and face-center of the grid-cells located on the boundaries of each computational subdomain. The figure below reports the partitioning of the Eulerian domain:

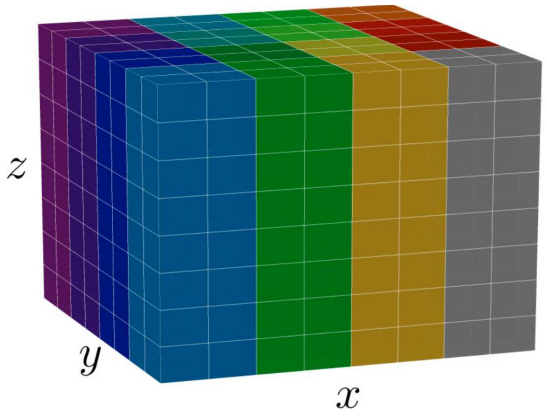


Figure 4.5: Partitioning of the Eulerian domain into $N_{MPI,x} \times N_{MPI,y}$ computational subdomains along the x and y directions, respectively, by using a two-dimensional, pencil like decomposition. Different colours represent different subdomains pertaining to different MPI processes.

On the other hand, the partitioning of the solid phase is obtained by the enumeration from 1 to N_p of the discrete solid particles, such that each is identified by an univocal integer *ID*. The discrete particles are then subdivided, based on their *ID*, into computational groups containing the same number of elements and each of these is assigned to a unique MPI process. Also in this case, in the memory area of each group is stored the Lagrangian positions and velocities of the particles located in its neighborhood but stored in a different group. The figure below reports a *index-based decomposition* for a generic peridynamic solid:

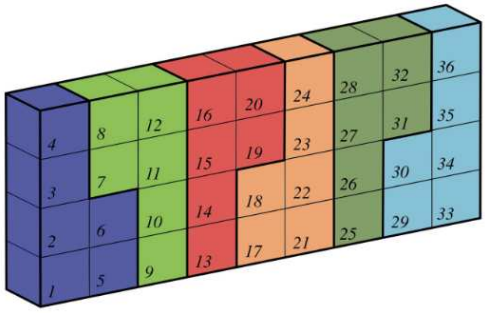


Figure 4.6: Example of the partitioning of the discrete particles into groups, identified by different colours, each associated with a different MPI processes. To minimize the amount of data communication, particles close in space are identified by close integer *ID*s.

It is worth remarking that the number of current MPI processes used for the *space-based domain decomposition* is equal to the number of these used for the *index-based* during a simulation. Since a significant amount of communications between different MPI processes of Eulerian phase and Lagrangian one are involved for interpolation and spreading operations to prescribe no-slip and no-penetration boundary conditions on the fluid-solid interfaces (as reported in the description of *multi-direct forcing* IBM method in section 3.5.2) and for calculating the fluid-dynamic forces acting on the solid boundaries (according to the equation 3.96), it is necessary to adopt an appropriate *index-based decomposition* strategy to minimize the number of required data communications. It has been proved that by numbering with close *ID*s particles which are closed in space allows a trade-off between a uniform distribution of the computational cost on concurrent processes and the number of data communications between them. Therefore, in the present work, the partitioning of the Lagrangian nodes is organized so that particles close in space are identified by close integer *ID*s so every solid subdomain, for prescribing boundary conditions and for computing fluid-dynamic forces, is forced to communicate with a less number of Eulerian computational subdomain and consequently the number of communications is reduced. It is worth remarking that these communications are restricted to the interface particles, only.

Chapter 5

Problem configuration

This document presents the simulation of the interaction between a porous medium that is invested by a incompressible fluid. This represents a typical fluid-structure interaction problem and is of interest for several research areas that involve porous materials and complex geometries. For example, in the field of aerospace engineering, there are several ablative materials, which have porous nature, that are use as external coverings for space vectors and satellites, which disintegrate under the action of fluid dynamic forces, contribute to maintain the structural element temperatures under prescribed threshold values during critical situations, for example during re-entry into the atmosphere.

In the present work a three-dimensional DNS of a laminar channel flow with immersed solid porous medium of a linear-elastic and brittle material is presented. The aim of the study is to reproduce the interaction between fluid phase and solid one where initiation of cracks, crack-branching and fractures are reproduced. The simulations involves both the solid and fluid solvers, as well as the multi-direct forcing algorithm and the normal probe method used to compute fluid-dynamic forces acting on the interfaces. Three different simulations which differs for the *critical fracture energy release rate* are presented. The target is to simulate three different cases: the first reproduces the deformation of the solid without fracture, the second with partial fracture and the third with total fracture. Then, in the next chapter, a comparison between the cases listed above is reported.

5.1 Configuration of computational domain

The Eulerian computational domain for the channel flow is a rectangular box extending for $L_x \times L_y \times L_z = L \times h \times h$ in the stream-wise, wall-normal, span-wise direction, respectively, with $L = 2 \cdot L_{ref}$ and $h = 1 \cdot L_{ref}$, where $L_{ref} = 0.1\ m$ is the reference length-scale. The discretization of the Eulerian domain consists of a Cartesian and uniform grid of $N_x \times N_y \times N_z = 128 \times 64 \times 64$ nodes along the

same directions. It results in a grid size $\Delta f = 1.5635 \cdot 10^{-3}\ m$. The solid porous medium is represented using a total of 54662 material particles. The ratio of the peridynamic horizon and the particle spacing is $m = \delta/\Delta_s = 3$, with $\delta = 4.6875 \cdot 10^{-3}\ m$ and $\Delta_s = \Delta_f$ the grid-spacing of Lagrangian frame. The peridynamic model adopted for the computation is the three-dimensional bond-based model. The porous medium is fixed to the lower wall of the channel. The constraint is applied to a ghost layer of solid material by disabling the dispacement along the three directions on three layer of material particles extending outside the Eulerian computational domain along the negative y direction. The thickness of ghost-layer is $\Delta_g = \delta$. The solid medium has the following dimensions $0.375L \times h \times h$ along the flow, span-wise, wall-normal directions respectively, with the axial position (i.e. flow direction x) of its bottom, leading corner set to $0.125 \cdot L$ from the inlet. The mechanical properties of the porous medium are set to $E = 10^8\ Pa$ $\nu_s = 1/4$, $\rho_s = 3000 kg/m^3$ with E the Young's modulus, ν_s the Poisson ratio, ρ_s the solid density. The value of the *critical fracture energy release rate* of the material G_0 is set to 0.05 J/m^2 in the first simulation, 0.03 J/m^2 in the second and 0.02 J/m^2 in the third. The bulk Reynolds number of the fluid is $Re_b = U_b h/\nu_f = 10.0$, with $U_b = 0.1\ m/s$ the bulk, axial velocity of the flow and $\nu_f = 1 \cdot 10^{-3}\ m^2/s$ is the kinematic viscosity of the fluid. No-slip and no-penetration boundary conditions are prescribed to the upper and lower sides of computational domain. A fully developed Poisseuille inflow is prescribed at the inlet, whereas a convective outflow condition is imposed at the outlet. Gravitational acceleration and buoyancy have been neglected. The figures below report the initial configuration of the porous medium and the configuration of the computational domain just described.

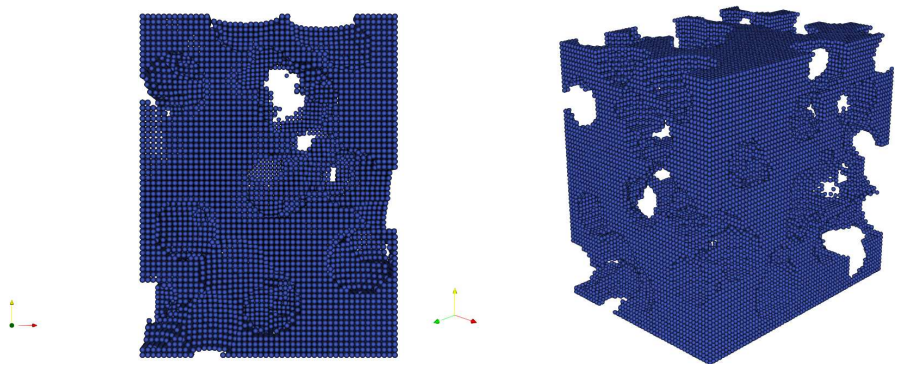


Figure 5.1: Representation of the initial configuration of the porous medium. The red axis corresponds to x axis, the yellow one to y and the green one to z.

As it is possible to see from the figure just presented, the simulation reproduces a solid medium with convex porosity whith pores of regular shapes and cavity dimensions that are quite larger than the grid-spacing, hence, since the DNS of the fluid is performed, it isn't necessary to use any mathematical model

to reproduce the dynamic of the flow inside these cavities.

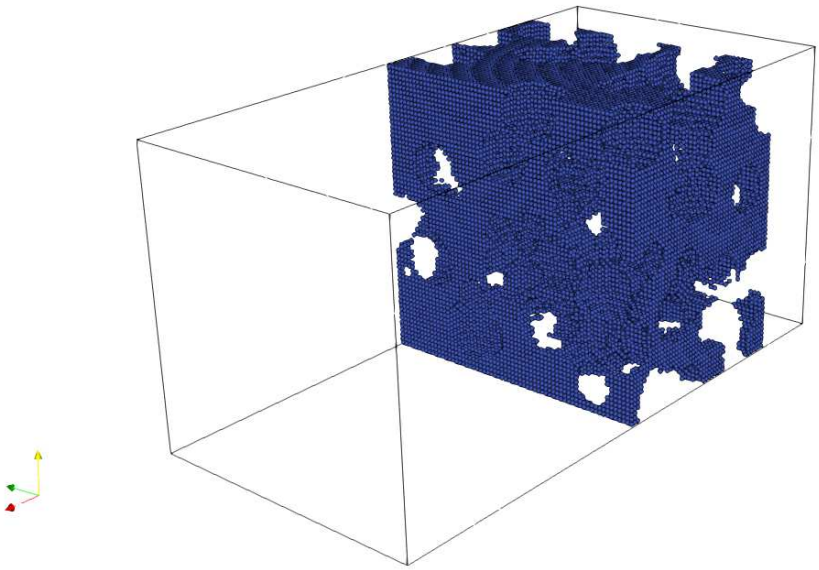


Figure 5.2: 3D render of the adopted computational domain with the final geometrical configuration. The fluid computational domain is indicated by the outer black outline. The red axis corresponds to x axis, the yellow one to y and the green one to z.

Each simulation has been performed in the following way:

1. For the first part of the simulation the fluid velocity is initialized as (0.1 m,0,0) and the fluid solver is run, keeping the solid solver disabled until a steady condition is established for the fluid flow. This part is ran for 5000 iterations employing as time step $\Delta t_f = 3.3569 \cdot 10^{-4}\ s$ and so is performed for a time $t \in [0;\ 1.6798]$ s. Due to the low Reynolds number that has been imposed the result of the initialization phase is a laminar fluid flow inside the domain.

2. In the second part, the solid solver is enabled, but the bond breakup is disabled and the coupled simulation is ran for other additional 5000 iterations and damping force is used until a statistically-stationary equilibrium configuration is achieved with the fluid stresses. The stability condition for the explicit peridynamic time-marching scheme is more severe than that for the stability of the fluid one, so a constant time step $\Delta t_s = 3.3569 \cdot 10^{-5}\ s$ is used. Hence this second part is performed for a time $t \in [1.6785\ ;\ 1.8463]$ s.

3. Finally the bond breakup is enabled and the simulation is ran for another

5000 iterations with the same time step of the second part, so for a time $t \in [1.8463\ ;\ 2.0142]$ s.

The present numerical study focuses on the effect of fluid stresses on the porous medium with linear-elastic and brittle material, in particular it aims to represent the initiation of cracks, crack-branching and fractures which occur when the bond stretches overtake the limit bond stretch s_0 of the material. For each simulation the probability density function (PDF) of the strains and stresses will be presented and then they will be compared and commented. The probability density function describes the probability density of a random variable at each point of the sample space. Hence, the probability of a random variable X, which has a probability density $\rho_x(X)$, that is included in a subset A of its sample space is given by:

$$P(X \in A) = \int_A \rho_x(X)\,\mathrm{d}x\,. \tag{5.1}$$

In addition it will be investigated the pressure drop of the fluid through the medium, the change in porosity and permeability over time for each case simulated.

Chapter 6

Results

This chapter presents the results obtained from the performed simulations and, in particular, it will be discussed the effectiveness of the model in describing the evolution of a porous material subjected to hydraulic fracturing. A total of three DNS simulations have been conducted, with the intents of studying the effect of solid fracturing and crack propagation by varying the *critical fracture energy release rate* of the material G_0 and to find a failure criterion, which allows to predict when the fracture occurs, basing on strain and stress distributions into the porous medium and their related threshold values. The physical parameters, expressed in non-dimensional units, employed in each simulation are:

- *bulk Reynolds number* $Re_b = U_b h/\nu_f = 10$;
- *solid to fluid density ratio* $\rho_s/\rho_f = 3$;
- the *non-dimensional Young's modulus* $E/(\rho_f U_b^2) = 10^7$;
- the *Poisson's ratio* of the solid material, $\nu_s = 0.25$

Whereas the values of *critical fracture energy release rate* G_0 employed for each simulation are:

$[J/m^2]$	S1	S2	S3
G_0	0.05	0.03	0.02

Table 6.1: Value of *critical fracture energy release rate* G_0, where S1, S2, S3 identify the first, second, third simulations respectively.

It is worth remarking that, as previously reported in section 3.3.4, the basic assumption of peridynamic theory is that cracks form, due to the rupture of bonds, that occurs when the stretch (equation 3.27) of a bond overcome a threshold value, called *limit bond stretch* s_0, that, for the 3D-cases, is defined as:

$$s_0 = \sqrt{\frac{5G_0}{6E\delta}}\,, \tag{6.1}$$

Still, bonds can break only under pure traction condition, in according with the *Prototype Microelastic Brittle (PMB)* material described by the employed *bond-based peridynamic* model.
As it is possible to see by equation 6.1, the value of the *limit bond stretch* s_0 is proportional to $\sqrt{G_0}$, which means that to lower G_0 values correspond lower s_0 values. Furthermore, material with a low value of s_0 are weaker and they tend to break sooner than materials which have a higher value of s_0. Therefore it is clear how changing the value of G_0 it is possible to make the material more or less brittle and consequently impose different deformation and stress distributions to it, as it was done in the present work.

Now a description of the three parts in which the simulations is divided are provided below. It is worth remarking that the first two parts (flow initialization and solid initialization) are the same for every simulation, as the value of G_0 influences only the last part where bond breakup is enabled.

6.1 Fluid-initialization

The fluid phase is evolved through the porous medium for 5000 iterations, by using a time step $\Delta t_f = 3.3569 \cdot 10^{-4}\ s$ and so for a time $t \in [0;\ 1.6798]\ s$. In this phase the peridynamic solver is disabled and so the the solid medium is interpreted as a rigid body that represents an obstacle for the fluid flow. Since the Reynolds number of the simulation is very low and a Poisseuille inflow is prescribed at the inlet, from a macroscopic point of view, the fluid flow inside the channel is laminar. However, due to the presence of the rigid solid medium, some isolated eddies are generated inside the channel. The figure below represents the contours of velocity at the end of the fluid-initialization phase.

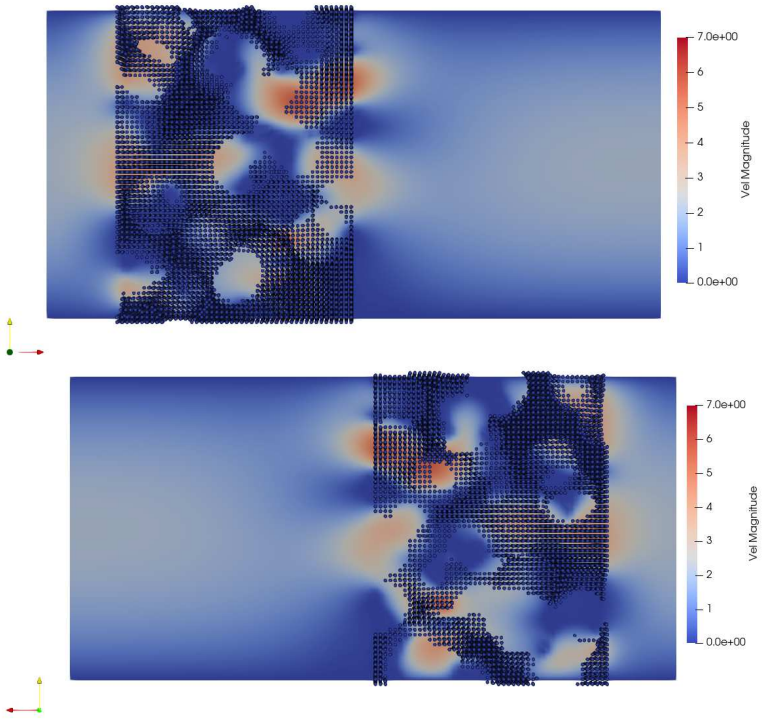


Figure 6.1: Contours of velocity at the end of the fluid-initialization part at $t = 1.6798\ s$. The red axis correspond to x axis, the yellow one to y and the green one to z. The $x - y - z$ directions refer to flow, wall-normal, span-wise directions, respectively.

It is also interesting to observe the hydrodynamic pressure distribution along the stream-wise direction at the end of this first initialization part. In fact, as it easy to understand, the presence of the porous medium inside the channel, that is modeled as a non-deformable rigid body since in this part the peridynamic solver is disabled, represents an obstacle to the movement of the fluid along the x direction. In particular the pressure value is the highest in front of the porous medium due to the effect of its frontal section, then the local pressure gradually decreases since part of the fluid travels through the internal porous cavities and finally, when the fluid crosses completely the porous medium, it can move undisturbed and so the pressure assumes its lowest value. The figures below represent the hydrodynamic pressure distribution at the end of the initialization phase where the fluid has reached a steady-state.

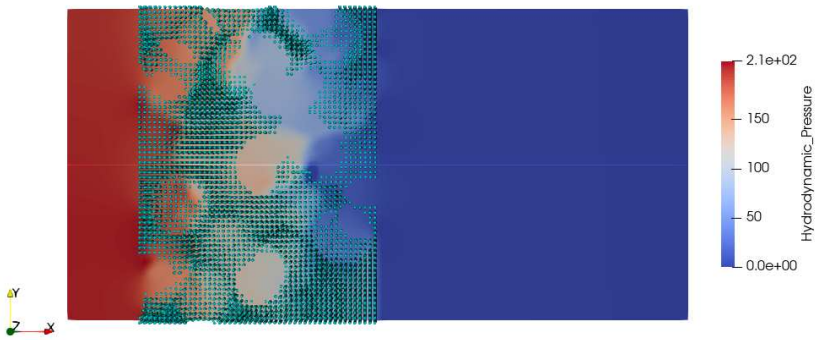


Figure 6.2: Contours of hydrodynamic pressure distribution at the end of the fluid-initialization part.

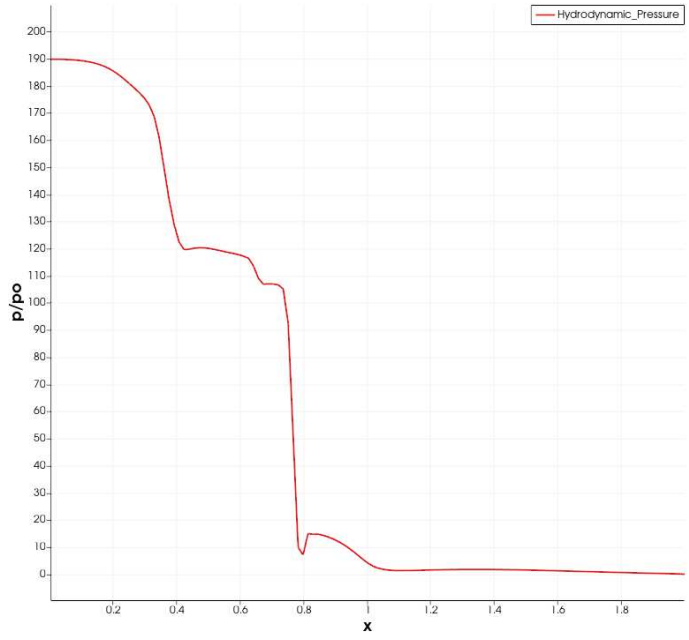


Figure 6.3: Distribution of the non-dimensional hydrodynamic pressure along the stream-wise direction at the end of the fluid-initialization part. P_0 is the reference value of pressure defined as $p_0 = \rho_f U_b^2$.

6.2 Solid-initialization

At the same way it has be done for the fluid, before to turning on the bond breakup, the solid solver is enabled and the coupled simulation is ran for addi-

tional 5000 iterations but, since the stability condition for the explicit peridynamic time-marching scheme is more severe than that for the stability of the fluid one, the time step employed is reduced to $\Delta t_s = 3.3569 \cdot 10^{-5}\ s$ and so this part involves the time $t \in [1.6785\ ;\ 1.8463]\ s$. In this part the peridynamic particles that compose the porous medium are forced to remain connected preventing that the possible immediate release of some solid particles could cause numerical errors capable of leading into a computational divergence. For this reason, to be sure that this effect cannot be realize, an additional dumping term related to the absolute particle velocities, is implemented.

The figure below represents the surface contours of the displacement field, associated to the peridynamic particles, for the front and back region of the porous material at the end of this part, namely at time $t = 1.8463\ s$.

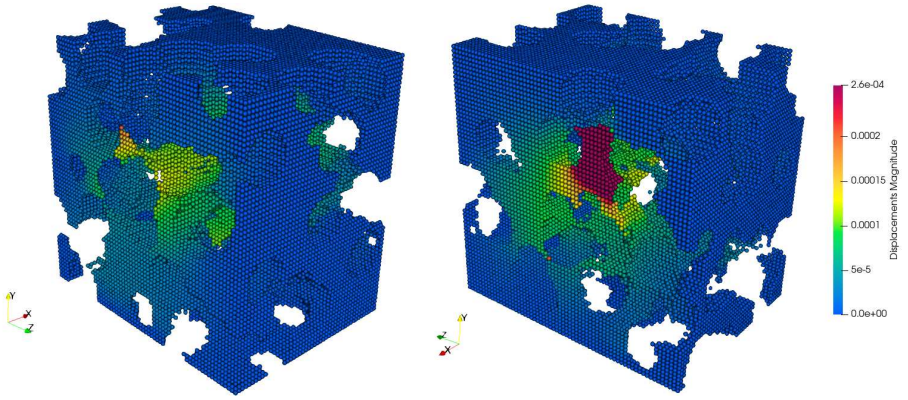


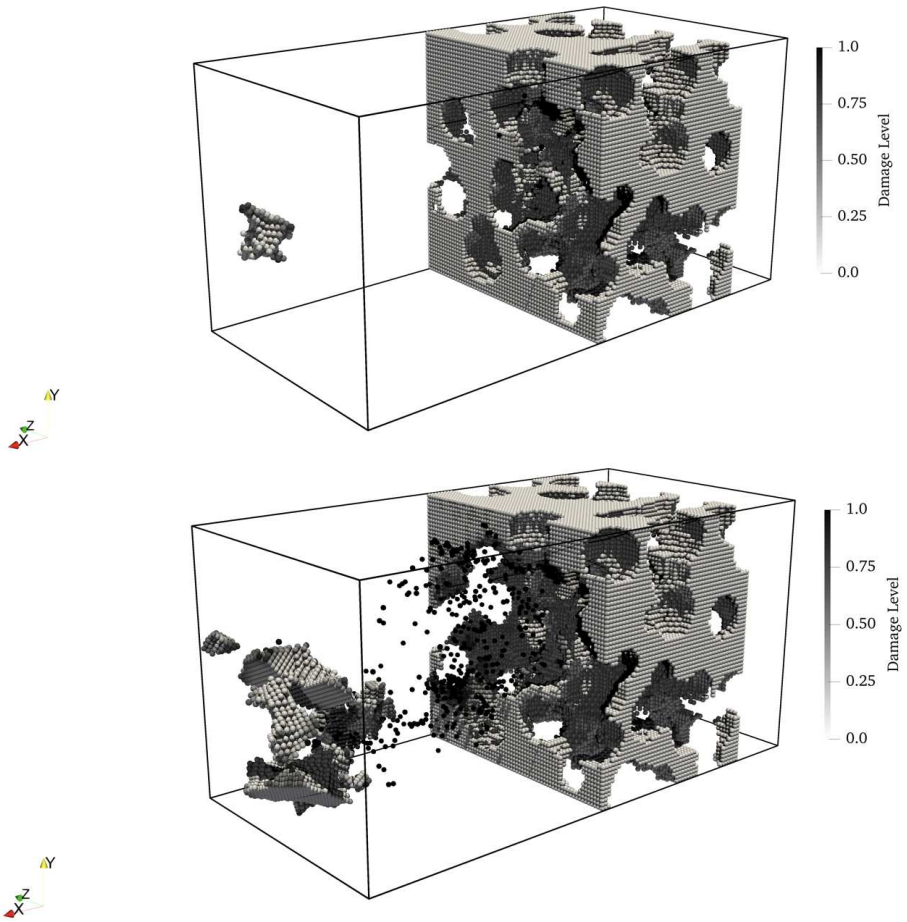
Figure 6.4: Surface contours of the non-dimensional displacement field in the front (left image) and back region (right image) of porous medium at time $t = 1.8463\ s$.

As it is possible to see in the figure above, the part of the porous medium that is subjected to greater deformation is the central one, whereas the part near the walls appears stiffer. This is a consequence of the imposed Poiseuille inflow condition, in which the fluid velocity field has a parabolic profile with a maximum value in the axis of symmetry of the channel, and it is zero in proximity of the walls. For this reason the action of the fluid is more severe for the core regions of the medium and less for the regions near the wall. In addition, it is clear from figure 6.4 that the largest displacements manifest in the back side of the solid and this result might seem in contrast with the hydrodynamic pressure distribution of the fluid (figure 6.3). This is due to the fact that the material particles located in the back region of the porous medium are less forced to maintain their positions because they don't suffer the bond forces of particles located behind them. It is therefore intuitive to think that the fracture process starts in the back side of the medium and propagates towards the core

region, whereas the part near the wall remains intact or it is the last to break.

6.3 Bond breakup enable

Once that the fluid and solid are initialized, the bond breakup is activated and the simulation continues to run for the interval $t \in [1.8463 \; ; \; 2.0141]$ s for further 5000 iterations with the same time step of the former part. Therefore the simulations are performed for a total of 15000 iterations. Let start observing the contour surfaces associated to the *damage factor* and the final geometrical configuration of the porous medium for each cases considered, which are reported by the figure below.



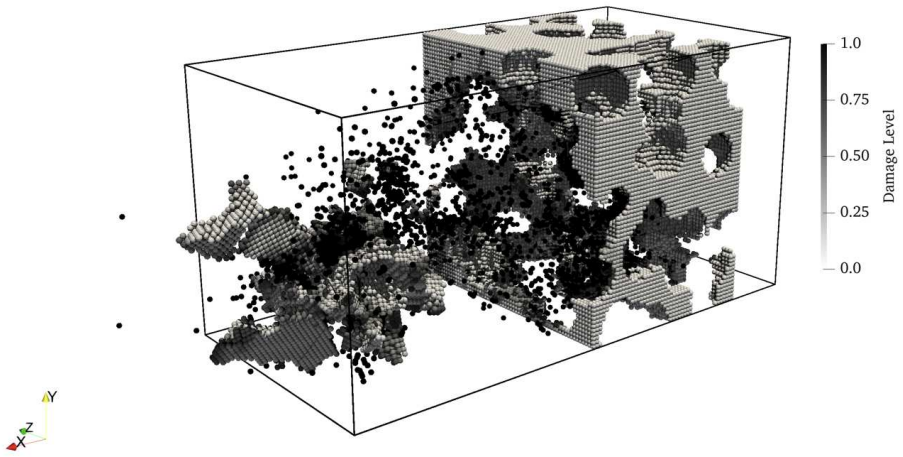


Figure 6.5: Representation of porous medium configurations and damage level associated for every peridynamic particles at the end of the fracturing process for each case considered. At the top is represented the case with the highest value of G_0, in the middle the one with the intermediate value of G_0 and at the bottom the one with the lowest value of G_0.

By observing the figure above it's evident that the three cases that we wanted to represent, which are the case with the medium intact, the case with the medium partially broken and the last with crushed solid, have been correctly reproduced. In particular, by observing the last frame of each simulation, it is possible to understand how the crack propagates inside the porous medium. In fact, referring to the cases with the lowest and the intermediate values of G_0, as expected from the displacement field distribution in figure 6.4, it is clear that the fracturing process starts from the back of the medium and then continues to propagate on posterior surface. Then, as it is possible to see in the case with the highest value of G_0, the breakage interests the innermost parts of the medium, whereas the parts near the walls remain intact, as expected.

6.4 Comparison of the cases

Once the way in which the simulations are performed has been explained, the comparison of the results obtained for each simulation can be reported and discussed. In particular the results about pressure drop through the medium, changing in permeability and porosity of the porous medium, stress and strain distribution are reported. These quantities have been computed separately at the end of each simulation, evaluating the fluid and solid data with a specific post-processing algorithm.

6.4.1 Comparison of pressure drops

In this section the change of the pressure difference between the front face and the back one of the porous medium is reported and discussed. As previously shown in figure (6.3) the initial difference of the non-dimensional pressure between the two faces $\Delta p/p_0 = 190$. This value is computed at the end of fluid-initialization part and remains approximately constant until the bond breakup is enabled. Then the fracture process changes the pressure distribution of the fluid and so the pressure difference through the porous medium.
The figure below reports the trend of the pressure differences for each case over time, considered as t/t_0 where t is the time measured from the activation of bond breakup and t_0 is the reference time scale computed as $t_0 = L_{ref}/U_b$:

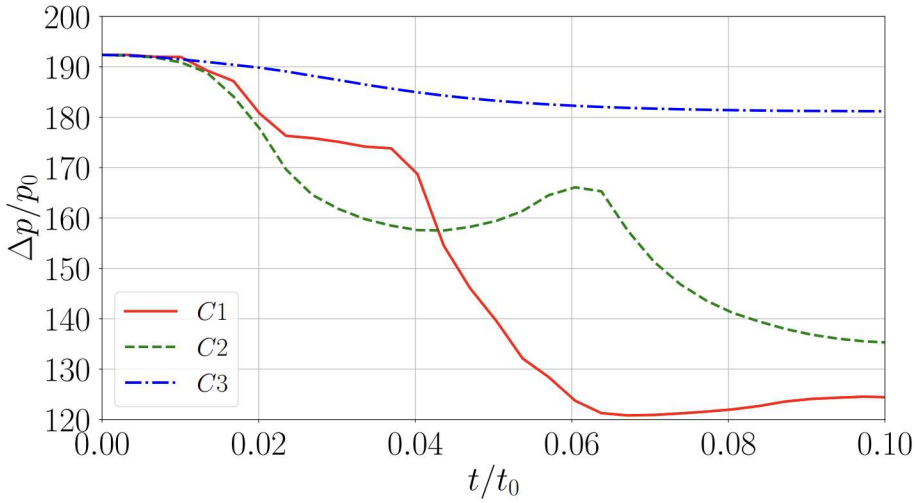


Figure 6.6: Representation of the pressure drop over time for the three different cases. In dark blue is represented the case without fracture, in green the case of partial fracture and in red the case with total fracture.

Observing the figure it is possible to notice that the pressure difference of the fluid between the front face and the back one changes little over time for the case without fracture (dark blue), because the porous medium remains almost intact and therefore its geometrical configuration remains similar to that before the rupture is activated. Instead, for the case with partial and total fracture, which are represented in green and red respectively, there is a significant pressure drop through the medium. This result is related to the fact that, as a consequence of the breakage and the detachment of solid fragments from the medium, the total frontal area of the solid, that hinders the fluid flow, decreases and therefore the fluid can flow easily through the medium. This brings to a reduction of the fluid pressure, specifically for the region before the porous medium that, as it is shown in figure 6.2, is the one with the highest value of pressure.

6.4.2 Comparison of the change in porosity and permeability of the porous medium over time

The interaction with the fluid phase and the consequent breakup of the porous medium cause a change of its physical properties. In particular in this section the changes of the *porosity* and *permeability* over time are presented for all the three cases simulated.
Let's start observing the trend of the *porosity* for each case over time, considered as t/t_0 where t is the time measured from the activation of bond breakup and t_0 is the reference time scale computed as $t_0 = L_{ref}/U_b$:

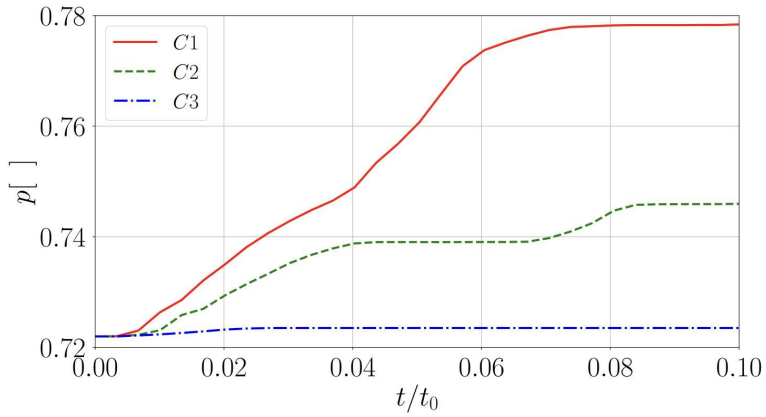


Figure 6.7: Representation of the trend of porosity over time for the three different cases. In dark blue is represented the case without fracture, in green the case of partial fracture and in red the case with total fracture.

As expected, when porous medium breaks and fragments leave the solid domain, the volume of the void space increases and consequently the porosity, since it is defined as the ratio between the volume of the void space and the total volume of the medium (equation 2.1). In fact, as it is possible to see from the figure above, for the case with partial fracture and total fracture of the medium, which are represented in green and red respectively, the porosity increases as the porous medium breaks down. In particular for the case of partial fracture we can observe that there are two different cases where one or more solid fragments detach from the main structure, since the porosity increases for the intervals t/t_0 = [0.01 - 0.04] and t/t_0 = [0.07 - 0.08], whereas it remains constant for t/t_0 = [0.04 - 0.06]. Instead, for the case of total fracture, the fracturing process continues without stopping until a steady state is reached.
On the contrary, for the case where hydraulic fracture phenomenon doesn't occur, which is represented in dark blue in the figure above, the porosity doesn't change since the porous medium deforms very little and remains almost intact.

Regarding the permeability, it was reported in section 2.1.1 that is a mea-

surement of the resistance encountered by the fluid when it crosses the porous medium and in the present work it is computed from Darcy's law (equation 2.3), so is expressed in the following way:

$$k = \frac{QL}{S}\frac{\mu}{\Delta p}\,, \tag{6.2}$$

where Q is the flow rate of the fluid, S is the section through the fluid flows, k is the permeability, μ is the dynamic viscosity of the fluid and Δp is the pressure difference between the front and back face of the medium. The main parameter which influences the value of permeability is the pressure difference Δp, since μ, L, S remain constant and Q is imposed. In the figure below, as the same way it has been done for *porosity*, the trend of permeability per unit area over time is reported:

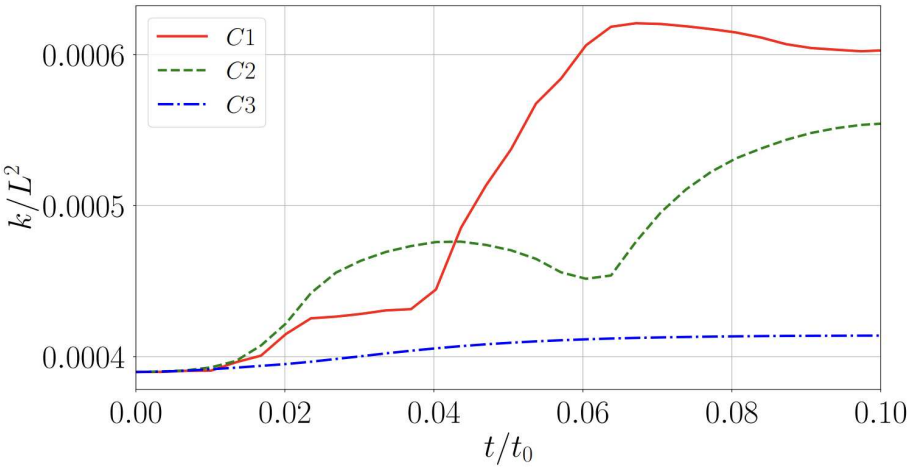


Figure 6.8: Representation of the trend of permeability over time for the three different cases. In dark blue is represented the case without fracture, in green the case of partial fracture and in red the case with total fracture.

From figure above it's possible to see that, for the case where the medium remains almost intact (dark blue), the permeability remains almost constant. This is due to the fact that the pressure drop for this case is very little (as previously reported in the former section) and therefore the permeability doesn't change.
Whereas, for both the case of partial and total fracture, the permeability changes significantly over time. This is due to the decrease of pressure difference through porous medium over time, in fact the graphs reported, for each case, follow the trend of the related pressure drop graphs reported in former section, but in the opposite way since $k \propto 1/\Delta p$.

6.4.3 Comparison of strain and stress distribution

Starting from notion of force per unit area in [7], the stress tensor is computed for each material particle and the strain tensor is also computed. Then they are rewritten in a principal reference system, and therefore diagonalized and, to associate at each coordinate only one value which identifies the local strain and stress, it was decided to take into consideration only the maximum value of the components for each diagonalized tensor.
Regarding to the choice of the threshold values, that identify when the fracture occurs, they are the *limit bond stretch* s_0 for the strain and the *limit stress* $\sigma_0 = E \cdot s_0$ for the stress. Referring to equation (6.1) the value of s_0 for each case are:

- $s_0 = 2.9814 \cdot 10^{-4}$ for the case with $G_0 = 0.05$ $[J/m^2]$;
- $s_0 = 2.3094 \cdot 10^{-4}$ for the case with $G_0 = 0.03$ $[J/m^2]$;
- $s_0 = 1.8856 \cdot 10^{-4}$ for the case with $G_0 = 0.02$ $[J/m^2]$.

whereas, the value of σ_0 for each case are:

- $\sigma_0 = 2981.4240\, Pa$ for the case with $G_0 = 0.05$ $[J/m^2]$;
- $\sigma_0 = 2309.4011\, Pa$ for the case with $G_0 = 0.03$ $[J/m^2]$;
- $\sigma_0 = 1885.6181\, Pa$ for the case with $G_0 = 0.02$ $[J/m^2]$.

Let's start evaluating the *PDF* of the strain represented by the figure below:

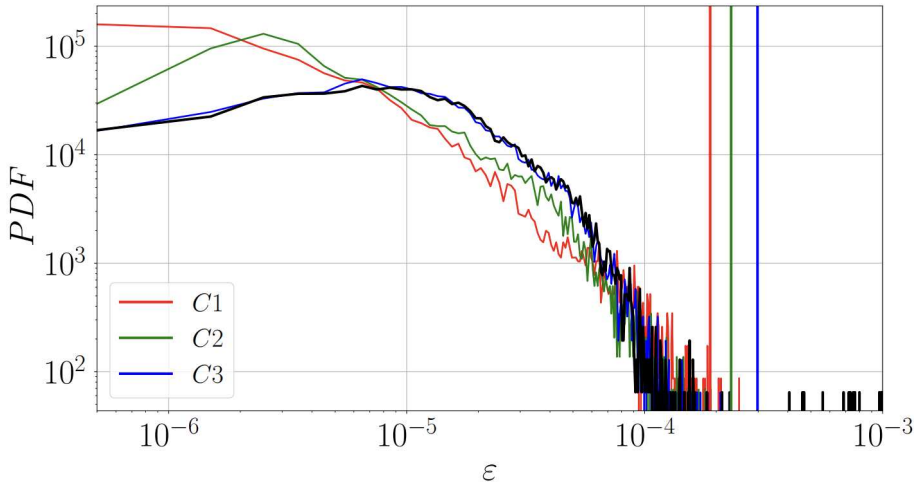


Figure 6.9: Probability density function (PDF) of the strains inside the medium. The case with the lowest value of G_0 is highlighted in dark blue, the one with the intermediate value in green and the case with the lowest value in red. All these graphs refer to the last iteration of each simulation. The black graphic, instead, represents the strain distribution at the end of solid-initialization phase, hence before that the bond breakup is enabled. The straight lines represent the threshold value for each case.

By observing the figure, in particular the strain distribution before the bond breakup is enabled (black graphics) and the threshold value for each case, it seems that the probability of breakage is about the same for each case simulated. This result is in contrast with what we have observed before, since it has been demonstrated that changing the value of G_0 the material is less or more brittle and therefore the probability of breakage must change accordingly.

This trend is also observed in stress distribution (figure 6.10) and it indicates that the criterion used to define the strain and stress distributions into the medium (maximum principal stress and strain) doesn't agree with the failure criterion employed to predict the breakage of the material (*limit bond stretch* s_0 for strain and *limit stress* $\sigma_0 = E \cdot s_0$ for stress).
The PDF of the stresses inside the medium is reported below:

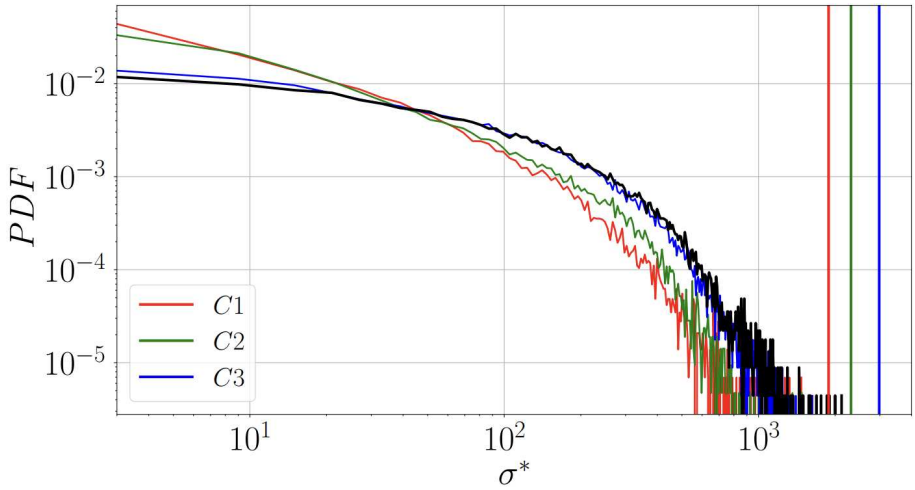


Figure 6.10: Probability density function (PDF) of the stresses inside the medium. The case with the lowest value of G_0 is highlighted in dark blue, the one with the intermediate value in green and the case with the lowest value in red. All these graphs refer to the last iteration of each simulation. The black graphic, instead, represents the strain distribution at the end of solid-initialization phase, hence before that the bond breakup is enabled. The straight lines represent the threshold value for each case.

As mentioned above, observing the stress distribution before the activation of bond breakup (black graphics) it seems that the medium remains intact for almost all cases considered, that is in contrast with the final configuration of the medium obtained for each simulation, and this error is due to the use of a incorrect failure criterion.
Furthermore, as it is possible to see in figures (6.9) and (6.10), referring to the case of partial fracture and completely fracture of the medium, which are highlighted in green and red respectively, both their strain and stress distributions are greater for the lower values and minor for higher value respect to the case where the solid remains intact, highlighted in dark blue. This trend is related to the fact that, when solid fragments detach from the main structure, as it happens for the considered cases, the fluid flows more easily through the medium and so the loads induced to the solid are lower.

Chapter 7

Conclusion

The aim of the present work is to solve numerically the interaction between laminar channel flow, with low Reynolds number, and a immersed porous medium. In particular, the main target is to reproduce the initiation of cracks, crack-branching and fracture of the solid body. This is a typical aspect of the fluid-structure-interaction (FSI) problem and is of interest for several science and engineering applications. For example, in the field of aerospace engineering, for protecting the space vehicle during atmospheric reentry operations, the external covering is composed by ablative materials that disintegrate under the action of fluid dynamic forces.
The numerical technique described in the present manuscript is capable to address FSI problems between incompressible flows and linear-elastic solids with brittle mechanical properties, where fracture mechanics is taken into consideration.
The dynamic of the fluid is solved by the incompressible formulation of the Navier-Stokes equations in the framework of the *Direct Numerical Simulations (DNS)*, whereas the solid mechanics and fracture mechanics are solved by employing peridynamics, a well-established reformulation of continuum theory that intrinsically accounts for crack initiation and branching. The fluid and solid dynamics are then coupled by the *immersed Boundary Method (IBM)*, which prescribes no-slip and no-penetration boundary conditions on the fluid-solid interfaces, that is detected by a specific algorithm. Then, the surface forces exerted by the fluid on the immersed solids are computed by using the *normal-probe method*, which allows to compute the stress distribution on wet surface, necessary for predicting the dynamics of deformable solids.
The source code is written in Fortran 90 extended with the *Message-Passing Interface (MPI)*, a standardized and portable message-passing standard designed to function on parallel architectures. The code is composed of three different modules: a fluid solver, a peridynamic solver and the third module that manages the synchronization of the coupling of the Navier-Stokes and peridynamic equations via the *IBM*.
A total of three different simulations have been performed, which differ for the

value of the *critical fracture energy release rate* of the material: the first reproduces the deformation of the solid without fracture, the second with partial fracture and the third with total fracture.
For all the cases have been performed analyses of stress and strain distributions, before and after fracturing process is occurred, the trend over time of fluid pressure drop through the solid, the change in porosity and permeability of the porous medium. Regarding the stress and strain distributions, for predicting when crack formation occurs, a failure criterion (*maximum principal stress and strain*) has been employed. However, it has been observed that, applying this criterion for both stresses and strains, it isn't possible to predict exactly the breakage of the porous medium. Anyway, others criteria can be implemented to represent stress and strain distributions (for example Von Mises criterion) or for prediction of the crack formation (for example the Griffith energy balance). Regarding the *DNS* of the breakup of porous medium in a laminar channel flow several improvements are possible in the near future developments of the method. For example, a lubrication model can be added to the solid-solid contact model to account for the effect of thin layer of fluid in interstitial regions created by cracks that are not resolved on computational grid. In fact, in the actual implementation of the model only the thin interstitial regions larger than the size of three Eulerian grid cells are detected. Another improvement concerns the addiction of turbulence model (e.g Smagorinsky) in the frame of *Large Eddy Simulations (LES)*, for reproducing the test case in frame where Reynolds number is higher, therefore where the usage of *Direct Numerical Simulations (DNS)* is not technically feasible, due to the high computational costs and the large time of simulation.

Bibliography

[1] Peter A Cundall. "A computer model for simulating progressive, large-scale movement in blocky rock system". In: *Proceedings of the international symposium on rock mechanics*. Vol. 8. 1971, pp. 129–136.

[2] Charles S Peskin. "Flow patterns around heart valves: a numerical method". In: *Journal of computational physics* 10.2 (1972), pp. 252–271.

[3] Cyrill W Hirt, Anthony A Amsden, and JL Cook. "An arbitrary Lagrangian-Eulerian computing method for all flow speeds". In: *Journal of computational physics* 14.3 (1974), pp. 227–253.

[4] Michael Thomadakis and Michael Leschziner. "A pressure-correction method for the solution of incompressible viscous flows on unstructured grids". In: *International journal for numerical methods in fluids* 22.7 (1996), pp. 581–601.

[5] Nicolas Moës, John Dolbow, and Ted Belytschko. "A finite element method for crack growth without remeshing". In: *International journal for numerical methods in engineering* 46.1 (1999), pp. 131–150.

[6] Alexandre M Roma, Charles S Peskin, and Marsha J Berger. "An adaptive version of the immersed boundary method". In: *Journal of computational physics* 153.2 (1999), pp. 509–534.

[7] Stewart A Silling. "Reformulation of elasticity theory for discontinuities and long-range forces". In: *Journal of the Mechanics and Physics of Solids* 48.1 (2000), pp. 175–209.

[8] Natarajan Sukumar et al. "Extended finite element method for three-dimensional crack modelling". In: *International journal for numerical methods in engineering* 48.11 (2000), pp. 1549–1570.

[9] John L Lumley and Akiva M Yaglom. "A century of turbulence". In: *Flow, turbulence and combustion* 66 (2001), pp. 241–286.

[10] Volker Carstens, Ralf Kemme, and Stefan Schmitt. "Coupled simulation of flow-structure interaction in turbomachinery". In: *Aerospace Science and Technology* 7.4 (2003), pp. 298–306.

[11] Gianluca Iaccarino and Roberto Verzicco. "Immersed boundary technique for turbulent flow simulations". In: *Appl. Mech. Rev.* 56.3 (2003), pp. 331–347.

[12] Rajat Mittal and Gianluca Iaccarino. “Immersed boundary methods”. In: *Annu. Rev. Fluid Mech.* 37 (2005), pp. 239–261.

[13] Stewart A Silling and Ebrahim Askari. “A meshfree method based on the peridynamic model of solid mechanics”. In: *Computers & structures* 83.17-18 (2005), pp. 1526–1535.

[14] Markus Uhlmann. “An immersed boundary method with direct forcing for the simulation of particulate flows”. In: *Journal of computational physics* 209.2 (2005), pp. 448–476.

[15] Harald Kruggel-Emden et al. “Review and extension of normal force models for the discrete element method”. In: *Powder Technology* 171.3 (2007), pp. 157–173.

[16] Pieter Rauwoens, Jan Vierendeels, and Bart Merci. “A solution for the odd–even decoupling problem in pressure-correction algorithms for variable density flows”. In: *Journal of Computational Physics* 227.1 (2007), pp. 79–99.

[17] Stewart A Silling et al. “Peridynamic states and constitutive modeling”. In: *Journal of elasticity* 88 (2007), pp. 151–184.

[18] Stewart A Silling and Richard B Lehoucq. “Convergence of peridynamics to classical elasticity theory”. In: *Journal of Elasticity* 93 (2008), pp. 13–37.

[19] A Bobet et al. “Numerical models in discontinuous media: review of advances for rock mechanics applications”. In: *Journal of geotechnical and geoenvironmental engineering* 135.11 (2009), pp. 1547–1561.

[20] Marcos Vanella and Elias Balaras. “Short note: A moving-least-squares reconstruction for embedded-boundary formulations”. In: *Journal of Computational Physics* 228.18 (2009), pp. 6617–6628.

[21] Stewart A Silling and Richard B Lehoucq. “Peridynamic theory of solid mechanics”. In: *Advances in applied mechanics* 44 (2010), pp. 73–168.

[22] Wim-Paul Breugem. “A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows”. In: *Journal of Computational Physics* 231.13 (2012), pp. 4469–4498.

[23] Bao-Lian Su, Clément Sanchez, and Xiao-Yu Yang. “Insights into hierarchically structured porous materials: from nanoscience to catalysis, separation, optics, energy, and life science”. In: *Hierarchically Structured Porous Materials* (2012), pp. 1–27.

[24] Raymond L Bisplinghoff, Holt Ashley, and Robert L Halfman. *Aeroelasticity*. Courier Corporation, 2013.

[25] Behzad Ghanbarian et al. “Tortuosity in porous media: a critical review”. In: *Soil science society of America journal* 77.5 (2013), pp. 1461–1477.

[26] Erdogan Madenci and Erkan Oterkus. “Peridynamic theory”. In: *Peridynamic theory and its applications*. Springer, 2013, pp. 19–43.

[27] Aman G Kidanemariam and Markus Uhlmann. "Direct numerical simulation of pattern formation in subaqueous sediment". In: *Journal of Fluid Mechanics* 750 (2014), R2.

[28] QV Le, WanKan Chan, and Justin Schwartz. "A two-dimensional ordinary, state-based peridynamic model for linearly elastic solids". In: *International Journal for Numerical Methods in Engineering* 98.8 (2014), pp. 547–561.

[29] Amin Rasam. "Anisotropy-resolving surgid-scale modelling using explicit algebraic closures for large-eddy simulation". PhD thesis. Mar. 2014.

[30] Mayken Espinoza et al. "Analysis of porosity and tortuosity in a 2D selected region of solid oxide fuel cell cathode using the lattice Boltzmann method". In: *ECS Transactions* 65.1 (2015), p. 59.

[31] Roham Rafiee, Mojtaba Tahani, and Mohsen Moradi. "Simulation of aeroelastic behavior in a composite wind turbine blade". In: *Journal of Wind Engineering and Industrial Aerodynamics* 151 (2016), pp. 60–69.

[32] Marco D de Tullio and Giuseppe Pascazio. "A moving-least-squares immersed boundary method for simulating the fluid–structure interaction of elastic bodies with arbitrary thickness". In: *Journal of Computational Physics* 325 (2016), pp. 201–225.

[33] Chengzeng Yan et al. "Combined finite-discrete element method for simulation of hydraulic fracturing". In: *Rock mechanics and rock engineering* 49 (2016), pp. 1389–1410.

[34] Piermario Campagnari. "Studio dell'interazione fluido-struttura con un metodo" immersed boundary" accoppiato alla peridinamica". In: (2017).

[35] Pedro Costa. "A FFT-based finite-difference solver for massively-parallel direct numerical simulations of turbulent flows". In: *Computers & Mathematics with Applications* 76.8 (2018), pp. 1853–1862.

[36] Abdulla Kerimov et al. "The influence of convex particles' irregular shape and varying size on porosity, permeability, and elastic bulk modulus of granular porous media: insights from numerical simulations". In: *Journal of Geophysical Research: Solid Earth* 123.12 (2018), pp. 10–563.

[37] Hong-Chao Zheng et al. "Coupled CFD-DEM model for the direct numerical simulation of sediment bed erosion by viscous shear flow". In: *Engineering geology* 245 (2018), pp. 309–321.

[38] Christer Stenström and Kjell Eriksson. "The J-contour integral in peridynamics via displacements". In: *International Journal of Fracture* 216 (2019), pp. 173–183.

[39] Shizhao Wang, Marcos Vanella, and Elias Balaras. "A hydrodynamic stress model for simulating turbulence/particle interactions with immersed boundary methods". In: *Journal of Computational Physics* 382 (2019), pp. 240–263.

[40] Federico Dalla Barba and Francesco Picano. “A Novel Approach for Direct Numerical Simulation of Hydraulic Fracture Problems”. In: *Flow, Turbulence and Combustion* 105 (Aug. 2020), pp. 1–23. DOI: 10.1007/s10494-020-00145-x.

[41] Yangyang Qiao and Steinar Evje. “A general cell-fluid Navier-Stokes model with inclusion of chemotaxis”. In: *Mathematical Models and Methods in Applied Sciences* 30 (Feb. 2020). DOI: 10.1142/S0218202520400096.

[42] Ming-Jyun Dai et al. “Ordinary state-based peridynamic shell model with arbitrary horizon domains for surface effect correction”. In: *Theoretical and Applied Fracture Mechanics* 115 (2021), p. 103068.

[43] Federico Dalla Barba and Francesco Picano. “Direct numerical simulation of the scouring of a brittle streambed in a turbulent channel flow”. In: *Acta Mechanica* 232 (Dec. 2021), pp. 1–24. DOI: 10.1007/s00707-021-03075-5.

[44] Jinlong Fu, Hywel R Thomas, and Chenfeng Li. “Tortuosity of porous media: Image analysis and physical simulation”. In: *Earth-Science Reviews* 212 (2021), p. 103439.

[45] *Descrizione del campo continuo della propagazione della cricca.*

[46] *Flussi turbolenti.*

[47] *Il metodo di correzione della pressione.*

[48] *Interazione fluido-struttura 3D con la frattura: un nuovo metodo con applicazioni.*

[49] Florin Le QV e Bobaru. *Correzioni superficiali per modelli peridinamici in elasticità e frattura.*

[50] *Peridinamica tramite analisi agli elementi finiti.*

[51] Luca Placco. “Direct numerical simulation of the hydraulic fracturing of porous media”. In: ().

[52] *Propagazione delle crepe con perfezionamento della griglia adattiva nella peridinamica 2D.*

[53] *Teoria peridinamica per la previsione del cedimento progressivo in materiali omogenei ed eterogenei.* L’Università dell’Arizona.