UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: *Università degli Studi di Padova*

CISAS - Centro d'Ateneo di studi ed Attività Spaziali "G. Colombo"

SCUOLA DI DOTTORATO DI RICERCA IN: Scienze Tecnologie e Misure Spaziali (STMS)

INDIRIZZO: Scienze e Tecnologie per Applicazioni Satellitari e Aeronautiche (STASA)

CICLO XXVIII

Structural material damage: novel methods of analysis

Direttore della Scuola: *Ch.mo Prof. Giampiero Naletto*

Coordinatore d'indirizzo: *Ch.mo Prof. Giampiero Naletto*

Supervisore: *Ch.mo Prof. Ugo Galvanetto*

Co-supervisore: *Ch.mo Prof. Mirco Zaccariotto*

Dottoranda: *Giulia Sarego*

To *Enzo, Teresa, Lorella,*
Raffaella, Andrea
and all my family

Abstract

A new nonlocal theory of continuum, called Peridynamics, was introduced recently. While the classical theory of solid mechanics employs partial derivatives in the equation of motion and, consequently, requires the derivability of the displacement field, Peridynamics employs an integral formulation which leads to the possibility to analyze structures without specific techniques whenever discontinuities, like cracks, are involved. Peridynamics has proven to be very suitable to predict failure of materials and to handle several phenomena such as crack branching and multiple crack interaction in all its variants, bond-based and state-based. The former is a particular case of the latter, which can also be found in two versions, the ordinary, in which the interaction force between two nodes is aligned with their current relative position, and the non-ordinary, in which interaction forces can have different directions and classical models may be directly introduced in the formulation.

In this thesis, Peridynamics is adopted for numerical analyses. Both static and dynamic solvers are employed to reproduce fracture patterns for brittle homogeneous isotropic materials. In particular, the static solver adopts a direct stiffness matrix approach where the linear system of equations is solved by the biconjugate gradient stabilized method, while the dynamic solver employs an explicit velocity-Verlet time integration scheme.

Two types of convergence in the numerical implementation of Peridynamics are investigated: *δ-convergence*, which is related to the maximum distance of nonlocal interaction (called *horizon*, δ), and *m-convergence*, where the m-ratio is the ratio between the grid spacing and the horizon. Both the bond-based formulation of a brittle linear elastic material model (called "prototype microelastic brittle model")

and the corresponding ordinary state-based formulation (called "linear peridynamic solid model") are adopted. Differing failure criteria are discussed and implemented, and numerical results are compared to analytical data and experimental tests found in literature.

Summary

In the classical continuum theory of solid mechanics, the mathematical framework involves partial derivatives to represent the state of deformation of a solid body. A significant drawback due to derivatives is related to the unphysical results given near the discontinuities, because they are undefined wherever a continuous field of displacements is not verified, such as in the presence of dislocations, voids, cracks, interfaces between different phases within the same body and grain boundaries.

Various techniques were employed for overcoming this incapability of the classical theory in describing material behavior in such conditions; in fact, spontaneous formation and growth of discontinuities are of great importance in solid mechanics: they lead to fractures and failures of systems that must be avoided, especially in aerospace structures, primarily, for safety reasons and, secondly, for economic purposes.

One of these new approaches concerns employing nonlocal theories, based on integral formulations (more precisely integro-differential formulations), defined even when non-derivable displacement fields are involved. Peridynamics is one of these theories: it was suggested by Stewart Silling in 2000 [1] in order to adopt a consistent formulation describing material behavior not only when a continuous displacement field is provided, but also whenever discontinuities are present, avoiding partial differential equations or pre-setting of conditions which can influence the results. There are two versions of peridynamic models: *bond-based*, which was introduced first (see [1, 2]) and *state-based*. In the bond-based version, forces between two material points depend solely on their relative displacement, their relative initial position, and material properties. Due to its simplicity compared

to the state-based version, most of the peridynamic applications have employed bond-based Peridynamics. However, bond-based models result in several limitations (the same of other atomistic or molecular dynamics models [3], although this is a continuum theory, not a discrete one), the most important of these is the fixed value of Poisson's ratio: 1/4 in 3D or 2D plane strain, and 1/3 in 2D plane stress (see e.g. [1, 4]). This peculiarity implies other restrictions, such as the impossibility of reproducing plastic incompressibility in an accurate way. Nevertheless, for many purposes, bond-based Peridynamics fits the requirements and gives satisfying results. State-based peridynamic models remove these restrictions by allowing the interaction ("bond") between a pair of points to potentially depend on all other bonds connected to the two points.

Moreover, there are two types of state-based peridynamic formulations: *ordinary* and *non-ordinary* [2, 5, 6]. In the former, the forces between two material points act along the vector connecting the points in the deformed configuration. In the latter, such characteristic is not present. The ordinary state-based formulation requires specific derivation of constitutive models, see examples of viscoelasticity and plasticity models in [7, 8]. For non-ordinary state-based formulation, two approaches have been proposed: the development of an explicit model for the peridynamic force state [2] and the development of a map thanks to which classical mechanics constitutive relations are incorporated to indirectly establish the relationship between the interaction force and the deformation. The latter approach is called *correspondence* model [2].

The purpose of this thesis has been the investigation of possible advantages and drawbacks of this new and unexplored theory, so to identify some guidelines for choosing parameters fundamental for the analyses and the development of models for particular structural analyses.

In the first year of the PhD course, the state of the art of this theory was studied and the bond-based linear and nonlinear static solvers developed in Matlab® were analyzed, employed and improved.

During the second year of PhD course, the author of this thesis has focused her attention on the second version of the theory, based on concepts of advanced

mathematics. She has become familiar with it, thanks to the functional analysis course that she had attended in the first year.

One of the main original contributions of the present work to the existing literature is the development of the 2D linearization of the state-based "linear peridynamic solid" model in the state-based formulation. These models are useful whenever simplifying assumptions of plane stress and plane strain can be adopted for the simulation of a system, which, otherwise, would be described by a 3D model requiring high computational resources (time and memory). Particular attention is paid to this aspect, because, being a nonlocal model, implementing a peridynamic code is, in general, more computationally expensive than a code based on a local approach. The study of the state-based version started before going abroad and the development of the 2D models was completed during the six month stay at the University of Nebraska-Lincoln in USA. Both static and dynamic codes have been developed and the relevant parameters of these models have been analyzed. These linearized models are described in chapter 1.2.2.

The study of failure criteria in state-based Peridynamics and the improvement of the algorithms in Matlab® to accelerate the codes and to optimize memory resources have been the main issues of the third year research. Some failure criteria, presented in section 1.2.3, have been proposed for brittle homogeneous linear elastic materials. They are criteria based on the maximum admissible stretch: a given bond fails at a critical stretch obtained by the work required to break that bond and this work is related to the fracture energy of the material. The results are compared to experimental data both for static and for dynamic cases, in bond-based and in state-based formulations. The detailed description of the algorithms can be found in chapter 3, while the results are illustrated in chapters 4 and 5.

Sommario

Nella teoria classica della meccanica dei solidi, la formulazione matematica include derivate parziali, grazie alle quali si possono rappresentare stati di deformazione come funzioni degli spostamenti relativi dei nodi in cui è discretizzato il sistema continuo. Una carenza rilevante dovuto all'utilizzo delle derivate è legato ai risultati privi di significato fisico ottenuti in prossimità delle discontinuità perché le derivate non sono definite laddove manca un campo di spostamenti continuo, come può capitare in presenza di dislocazioni, vuoti, cricche, interfacce tra fasi differenti nello stesso corpo e bordi dei grani.

Dato che la formazione spontanea e la crescita di discontinuità sono di grande importanza in meccanica dei solidi, diverse tecniche sono state utilizzate per superare questa incapacità della teoria di descrivere il comportamento dei materiali in tali condizioni, perché situazioni in cui le strutture sono incapaci di continuare a svolgere la propria funzione devono essere evitate, specialmente per strutture aerospaziali, in primo luogo, per ragioni di sicurezza ed, in secondo luogo, per motivi economici.

Uno di questi nuovi approcci riguarda l'utilizzo di teorie non locali basate su formulazioni integrali (più precisamente formulazioni integro-differenziali), definite anche quando campi di spostamento non derivabili sono presenti. La teoria "Peridynamics" è una di queste teorie: è stata proposta da Stewart Silling nel 2000 [1] così da adottare una formulazione unica e coerente capace di descrivere i comportamenti dei materiali in corpi sia continui che discontinui, evitando l'uso di equazioni alle derivate parziali o la definizione *a priori* di alcune condizioni che possono influenzare (e in un certo senso favorire) dei risultati. Ci sono due versioni di modelli peridinamici: la *state-based*, e un suo caso particolare, la *bond-based*, che

è stata introdotta per prima (vedi [1, 2]). Nella versione bond-based, le forze tra due punti materiali dependono unicamente dal loro spostamento relativo e dalla loro posizione relativa iniziale, oltre che dalle proprietà del materiale. Vista la sua semplicità a confronto con la seconda versione, la maggior parte delle applicazioni e degli articoli sulla Peridynamica ha adottato la formulazione bond-based. Tuttavia, i modelli nella formulazione bond-based sono caratterizzati da alcune limitazioni (le stesse dei modelli di altre teorie atomistiche e dei modelli di dinamica molecolare [3], anche se la Peridinamica è una teoria del continuo, non discreta), la più notevole di queste è il modulo di Poisson fisso: 1/4 nelle simulazioni 3D oppure in caso di deformazione piana 2D, e 1/3 nelle simulazioni in stato di tensione piana 2D (si veda per esempio [1, 4]). Questa particolarità implica altre restrizioni, come l'impossibilità di riprodurre la condizione di incomprimibilità plastica in maniera accurata. Tuttavia, per la maggior parte degli scopi, la formulazione bond-based è sufficiente e fornisce risultati approssimati soddisfacenti.

I modelli della versione state-based rimuovono queste restrizioni, permettendo che le interazioni tra due punti possano dipendere da tutte le interazioni (i "bond") connessi ad almeno uno dei due punti, tramite delle mappe avanzate chiamate "states". Inoltre, ci sono due tipi di formulazioni state-based: la *ordinary* e la *non-ordinary* [2, 5, 6]. Nella formulazione ordinary, le forze tra due punti materiali agiscono lungo la congiungente i due punti nella configurazione deformata, mentre nella formulazione non-ordinary, questa caratteristica non è più vera. La formulazione ordinary della state-based necessita di modelli costitutivi appositamente derivati, come per esempio i modelli di viscoelasticità e platicità in [7, 8]. Per la formulazione non-ordinary della state-based, due approcci sono stati proposti: lo sviluppo di un modello esplicito per l'espressione dello state della forza peridinamica [2] e lo sviluppo di una mappa grazie alla quale le relazioni costitutive della meccanica classica sono incorporate per stabilire indirettamente la relazione tra la forza d'interazione e la deformazione. I modelli derivanti dal secondo approccio sono chiamati modelli *correspondence* [2].

L'argomento di questa tesi è lo sviluppo di modelli per particolari tipi di analisi e la ricerca di possibili vantaggi e inconvenienti di questa teoria nuova ed inesplorata,

così da identificare alcune linee guida per la scelta di parametri fondamentali per le analisi.

Durante il primo anno del corso di dottorato, lo stato dell'arte relativo a questa teoria è stato studiato e i solutori statici lineari e non lineari nella formulazione bond-based sviluppati precedentemente in ambiente Matlab® sono stati analizzati, usati e migliorati.

Durante il secondo anno, l'autrice di questa tesi si è concentrata sulla seconda versione, basata su concetti di matematica avanzata con cui ha preso dimestichezza grazie al corso di analisi funzionale seguito il primo anno. Uno dei principali contributi originali alla letteratura esistente presenti in questa tesi è lo sviluppo dei modelli linearizzati 2D del modello solido lineare nella formulazione state-based. Questi modelli sono particolarmente utili quando semplificazioni di stato piano di tensione o di deformazione possono essere assunte per la simulazione di un sistema tridimensionale, che altrimenti verrebbe descritto da un modello 3D che necessiterebbe di risorse computazionali più elevate (in termini di tempo e memoria). Una particolare attenzione è richiesta per quest'aspetto, perché, essendo un approccio non locale, implementare un codice basato sulla teoria peridinamica richiede in generale più risorse computazionali di un codice basato su un approccio locale. Lo studio della versione state-based è iniziato prima di andare all'estero e lo sviluppo dei modelli 2D si è poi completato durante il soggiorno di sei mesi alla University of Nebraska-Lincoln negli Stati Uniti. Sono stati sviluppati sia un codice dinamico che uno statico. I parametri principali di questi modelli sono stati analizzati e i modelli linearizzati si possono trovare descritti nel capitolo 1.2.2.

Lo studio dei criteri di frattura adottabili nella formulazione state-based e il miglioramento degli algoritmi in Matlab® per accelerare i codici e ottimizzare le risorse di memoria e gestione dei dati sono stati gli argomenti principali del terzo anno. Alcuni criteri di frattura, presentati nel capitolo 1.2.3, sono stati proposti per materiali lineari elastici omogenei e caratterizzati da frattura fragile. Sono criteri basati sul massimo allungamento: un'interazione non locale ("bond") viene meno quando un valore critico di allungamento è raggiunto; questo valore di allungamento critico è calcolato dal lavoro richiesto per rompere il bond e questo lavoro è a sua

volta legato all'energia di frattura. I risultati ottenuti sono stati confrontati con dati sperimentali per casi sia statici che dinamici, sia nella formulazione bond-based che in quella state-based. La descrizione dettagliata degli algoritmi si trova nel capitolo 3, mentre i risultati sono riportati nei capitoli 4 e 5.

Acknowledgements

During these three years many things happened and lots of people shared part of their lives with mine. I will try to acknowledge all of them. If someone is missing, it is due to my lack of memory.

First of all, I want to express my deep gratitude to Prof. Ugo Galvanetto, for giving me this opportunity and for his supervision, and to Dr. Mirco Zaccariotto, for his continuous help every time I needed during the last three years. I'd also like to thank Prof. Florin Bobaru, for his support during my period of stay in Lincoln, Nebraska.

I'm deeply indebted to my mother and my sister Raffaella for all the support and love they gave me, even though they have no idea of what I've been doing since long ago. If I have to say the names of people who really did whatever they could to let me reach this point, the names coming to my mind are definitely theirs. I also want to thank my father and my sister Lorella, who have taught me patience and perseverance, and my brother Andrea, who literally let me invade his private spaces (PC and bedroom) without complaining too much... or at least, I haven't listened to them, so they were mild.

I would like to thank all my friends for their support, their help and the funny times spent together. New friends I have encountered in this period and old time friends who shared with me this experience are really too many to count.

First of all, Elisa and Jelena, who are like sisters for me and have been with me since childhood.

I particularly want to thank Marco, who literally was involved (willingly or not) in all the aspects of the life of a PhD student, he practically had been bothered by (me) all the issues of a PhD course without being enrolled in one. And unbelievably,

without complaining...

In addition, I want to thank my best friends, Daniele, who has shared ideas and suggestions in many discussions, both in private matters as a friend and in work-related topics as a colleague, and Matteo, who is always absent-mindedly funny, but reliable when I need clever and serious opinions.

I want to mention all my colleagues, Siamak, Teo, Arman, Soheil, Fabio, for their constant help, funny stories and diverting times and I hope to enjoy more and more times with you in the future; Guanfeng and Yenan, for our time together when I was in Nebraska; Mattia, Guido and all the PhD students of the 4^{th} floor for our meetings and discussions, especially those in front of the vending machines; Lorenzo, Laura, Livia, and the guys of "AF" group, I enjoyed going to CISAS headquarters because I knew I would have found them there, and helping them in the symposium organization was a great experience.

I would like to thank also Elena, Eleonora, Anna, Anaïs, Mita, Luca and Nicola, because no matter how rarely we see each other, it's always like we have been together till the day before; Elena, Marianna and Silvia, because our meetings are always so relaxing and funny; Anna, Ivan, Elisa, Federico, Francesco, Alice, Mirko and Cristiano for all the fun and the evenings spent together.

Lastly, I'm grateful also to my newest friends Veronica, Renè, Isabel and Barbara, who have been incredibly supportive in the short period I've known them.

Publications

Articles

Sarego G., Le Q. V., Bobaru F., Zaccariotto M., Galvanetto U. (to be published) Linearized State-based Peridynamics for 2D problems, International Journal for Numerical Methods in Engineering.

Dipasquale D., Sarego G., Zaccariotto M., Galvanetto U. (under revision) Dependence of crack paths on the orientation of regular peridynamic grids, Engineering Fracture Mechanics.

Zaccariotto M., Luongo F., Sarego G., Galvanetto U. (2015) Examples of applications of the peridynamic theory to the solution of static equilibrium problems. The Aeronautical Journal, vol. 119, p. 677-700, ISSN: 0001-9240.

Proceedings

Sarego G., Zaccariotto M., Galvanetto U. (2015) Mixed-mode crack patterns in ordinary state-based Peridynamics. In: Advances in Fracture and Damage Mechanics XIV. Key Engineering Materials, vol. 665, p. 53-56, Trans Tech Publications, ISBN: 978-3-03835-541-0, ISSN: 1013-9826, doi: 10.4028/www.scientific.net/KEM.665.53, Budva, Montenegro, 21-23 September 2015.

Zaccariotto M., Sarego G., Dipasquale D., Shojaei A., Mudric T., Galvanetto U. (2015). Discontinuous mechanical problems studied with a peridynamic-based approach. In: Proceedings of the 23th AIDAA Conference, Turin,

Italy, 17-19 November 2015.

Zaccariotto M., Sarego G., Dipasquale D., Galvanetto U. (2015) Alternative thin plate formulation using a peridynamic approach . In: SPB 2015 International Conference on Shells, Plates and Beams. Proceedings: Structural and Computational Mechanics, p. 55-56, Società Editrice Esculapio, ISBN: 978-88-7488-886-3, ISSN: 2421-2822, doi: 10.15651/978-88-748-8886-3, Bologna, 9-11 September 2015.

Dipasquale D., Zaccariotto M., Sarego G., Duzzi M., Galvanetto U. (2014) Peridynamics computations with variable grid size. In: Proceeding of the 27th Nordic Seminar on Computational Mechanics (NSCM-27). Stockholm (Sweden), 22-24 September 2014.

Zaccariotto M., Luongo F., Sarego G., Dipasquale D., Galvanetto U. (2013) Fatigue Crack Propagation with Peridynamics: a sensitivity study of Paris law parameters. In: Proceedings of the 4th CEAS Conference, p. 849-854, Linkoping University Electronic Press, ISBN: 9789175195193, Linkoping, Sweden, 16-19 September 2013.

Abstracts

Galvanetto U., Zaccariotto M., Dipasquale D., Sarego G. (2015) Enhanced 2D lamina formulation for composite materials, simulation with a peridynamic approach. In: International Conference on Advances in Composite Materials and Structures, Book of Abstracts. p. 60-61, Istanbul, Turkey, 13-15 April 2015.

Dipasquale D., Sarego G., Shojaei A., Zaccariotto M., Galvanetto U. (2015) Addressing grid sensitivity in Peridynamics: an adaptive refinement approach. In: International Conference on Computational Modeling of Fracture and Failure. p. 93-94, Rennes: CCSd (Centre pour la Communication Scientifique Directe), Cachan, France, 3-5 June 2015.

Zaccariotto M., Sarego G., Dipasquale D., Galvanetto U. (2015) Remarks on constitutive laws and influence functions used in the Peridynamic theory. In: International Conference on Computational Modeling of Fracture and Failure. p. 248-249, Rennes:CCSd (Centre pour la Communication Scientifique Directe), Cachan, France, 3-5 June 2015.

Sarego G., Dipasquale D., Galvanetto U., Zaccariotto M. (2015) Study of fracture patterns generated by different failure criteria in ordinary state-based Peridynamics. In: International Conference on Computational Modeling of Fracture and Failure. p. 303-304, Rennes: CCSd (Centre pour la Communication Scientifique Directe), Cachan, France, 3-5 June 2015.

Zaccariotto M., Sarego G., Dipasquale D., Galvanetto U. (2015) Strategies for fatigue damage modeling with Peridynamics. In: Book of Abstracts of the 8th International Congress of Croatian Socirety of Mechanics. Croatian Society of Mechanics, ISBN: 978-953-7539-20-7, Opatija, Croatia, 29 September - 2 October 2015.

Dipasquale D., Sarego G., Zaccariotto M., Galvanetto U. (2014) Peridynamics with adaptive grid refinement. In: Joint 11th World Congress on Computational Mechanics (WCCM XI), the 5th European Conference on Computational Mechanics. International Center for Numerical Methods in Engineering, p. 4009-4020, ISBN: 9788494284472, Barcelona, Spain, 20-25 July 2014.

Zaccariotto M., Sarego G., Dipasquale D., Galvanetto U. (2014) Fatigue Crack Propagation and their Interaction Modelling with a Peridynamics Approach. Joint 11th World Congress on Computational Mechanics (WCCM XI), the 5th European Conference on Computational Mechanics. Barcelona, Spain, 20-25 July 2014.

Posters

Dipasquale D., Zaccariotto M., Duzzi M., Galvanetto U., Sarego G. (2014) Dynamic and Static Simulations with Peridynamic Approach Using Finite Element Analysis. In: Poster presented at International CAE Conference. Pacengo del Garda (Verona), 27-28 October 2014.

Duzzi M., Dipasquale D., Sarego G., Zaccariotto M., Galvanetto U. (2014) A concurrent multiscale model to predict crack propagation in nanocomposite materials with the peridynamic theory. In: Poster presented at NanotechITALY 2014. Venice, 26-28 November 2014.

Awards

Winner of the 2013 "Mela d'Oro" Prize of the Marisa Bellisario Foundation for the Most Distinguished Newly Graduated Female Aerospace Engineer.

Author winner of 2015 A.I.D.A.A. "Premio in memoria del Prof. Teodoro Merlini" prize for best paper in XXIII AIDAA Conference in Turin, Italy, 17-19 November 2015.

Contents

List of Tables

List of Figures

Abbreviations

BBP	Bond-Based Peridynamics
SBP	State-Based Peridynamics
PMB	Prototype Microelastic Brittle
LPS	Linear Peridynamic Solid
FEM	Finite Element Method
PMMA	PolyMethyl MethAcrylate)
SCB	Semi-Circular Bend
DLSP	Diagonally Loaded Square Plate
CFL	Courant-Friedrichs-Lewy approach
MTS	Maximum Tangential Stress

Nomenclature

α_1	rotation angle between global and local reference systems
β	crack propagaton angle in a mixed mode case
$\Delta x, \Delta y, \Delta z$	grid spacings respectively in x, y, z directions
Δt	time step size
Δt_{crit}	critical time step for stable solutions
δ	horizon, nonlocal length scale
ε_{ij}	deformation tensor component
$\boldsymbol{\eta}$	relative displacement
θ	peridynamic dilatation
$\mu\left(\boldsymbol{\xi}\right)$	time dependent variable of the damage of the bond
μ	shear modulus
ν	Poisson's ratio
$\boldsymbol{\nu}(\mathbf{x}, t)$	collapsed peridynamic stress tensor
$\boldsymbol{\xi}, \boldsymbol{\zeta}$	relative initial positionx, bonds
ρ	material density
σ_{ij}	stress tensor component
$\omega_0\left(\boldsymbol{\xi}\right)$	work required to break a bond
$\underline{\omega}$	influence function
Ω_+, Ω_-	halves of the horizon volume
a	pre-crack length
$\mathbf{b}$	body force density
$\mathcal{B}$	solid body
$\hat{\mathbf{b}}$	incremental body force density

c	bond stiffness
$C(\boldsymbol{\xi})$	micromodulus
c_w	wave speed
dV_i, dV_j	infinitesimal volumes associated to points $\mathbf{x}_i, \mathbf{x}_j$
dV_p, dV_q	infinitesimal volumes associated to points $\mathbf{p}, \mathbf{q}$
dV_-, dV_+	infinitesimal volumes associated to points in Ω_+ and Ω_-
E	Young's modulus
$\underline{e}$	scalar extension state
$\underline{e^d}$	deviatoric extension state
$\mathbf{f}$	pairwise force function
$\{F_{ext}\}$	external nodal force vector
f_V	integration weight associated to a node to identify the portion of the node volume inside the horizon sphere of the source node
$\mathbf{F}(\mathbf{x}, t)$	peridynamic deformation gradient tensor
$\mathbf{g}$	out of balance force
G_0	critical energy release rate
$\mathcal{H}_{x_i}$	the neighborhood of the point x_i containing all the points x_j interacting with x_i
$\mathcal{H}$	set of all bonds in SBP
$I_i^{n-1}, I_{n_x}^{n-1}$	integral of the internal forces at the $n-1$ time step
I_{bond}	matrix storing the initial bonds of the structure
K	bulk modulus
$[K]$	stiffness matrix
$[K]_T$	tangent stiffness matrix
k', α	constant related to material properties
$\mathbf{K}(\mathbf{x})$	shape tensor
$\underline{\mathbb{K}}$	double state, derivative of the force state
k_{spring}	stiffness of a spring
k	iteration number
$\mathbf{L_u}$	integral of internal forces
$\mathcal{L}_m$	set of all tensors of order m
m-ratio	ratio of the horizon to the grid spacing
$\underline{M}$	deformation direction state
M	maximum number of family nodes
n	superscript indicating the time step
N	total number of nodes
n_x, n_p, n_q	ID numbers of nodes in SBP numerical discretization
$\mathbf{P}$	force applied on a specimen

q, m	weighted integration volumes in 2D and 3D
R, R^T	rotation matrix
$\mathrm{R_s}$	radius of the SCB specimen
s	bond stretch
s_0	critical stretch
S	semi-distance between the two bottom supports in SCB test
t	time
t_h	thickness of a 2D system
$\underline{t}$	modulus state
$\underline{T}$	force state
$\ddot{\mathbf{u}}$	acceleration vector relatively to the initial position
$\mathbf{u}$	displacement vector
$\underline{U}$	displacement state
$\{U\}$	nodal displacement vector
$\mathcal{W}$	strain energy density
$\underline{x}$	reference scalar state
$\underline{X}$	reference state
$\mathbf{x}_i, \mathbf{x}_j$	material points in BBP
$\mathbf{x}, \mathbf{p}, \mathbf{q}$	material points in SBP
$\mathbf{u}_i, \mathbf{u}_j$	displacements of point $\mathbf{x}_i, \mathbf{x}_j$
$\mathbf{y}$	current position vector
$\ddot{\mathbf{y}}$	current absolute acceleration vector

Introduction

Classical continuum mechanics

Continuum mechanics is a part of mechanics whose main objective is to predict the response of a body under the action of external and internal forces. The main assumptions are

1. that a body has a continuous distribution of matter in the Newtonian space-time and no reference to its discrete structure at microscopic length scales is made [13],

2. that contact forces are the only possible type of internal forces (see [14]).

From the 15^{th} to the 18^{th} century, two main areas of analysis were established separately, one related to the deformation of solids and one concerning the flow of fluids [15], while solid/fluid interaction concepts were studied later. Classic continuum mechanics is a powerful tool which deals with the macroscopically relevant properties of the material through mathematical models easily treated either analytically or numerically. The 20^{th} century saw a prosperous development in modern continuum mechanics, thanks to the enrichment of mathematical tools (such as concepts in tensor calculus, partial differential equations, numerical analysis and so on) and improvements in information technology allowing to handle big amount of data for its numerical implementation.

Classical continuum mechanics can be applied whenever the characteristic length of the deformation is larger than the representative volume element (i.e. the smallest volume over which a measurement can be made that will yield a value representative of the whole) of the response of the material to a homogeneous deformation.

Therefore, the classical continuum theory lacks an internal length parameter that would allow modeling at different scales referring to the microstructure [16, 17].

Alternative Approaches

The assumptions on which the theory is based justify the use of partial derivatives to represent the strain state of a body. However, these partial derivatives are undefined along discontinuities, such as cracks, dislocations and interfaces, so any method employing them breaks down. As a matter of fact, whenever deformation localizes, the classical continuum theory starts to break down [18, 19, 20, 21, 22] and eventually fails when discontinuities are involved. More precisely, if the characteristic wavelength of the deformation field is longer than the resolution level of the material model (i.e. the representative volume), a conventional continuum description can be adequate [23].

In the last decades, new physical and mathematical models have been intensively developed to overcome the shortcomings of the classical theory, not only to be able to describe fracture behavior, but also to meet the needs of a growing demand for cutting-edge applications concerning new technological processes and new materials working either in extreme conditions or with miniaturized components where microstructures or long-range forces play important roles for the component behavior [24].

Several approaches were adopted to overcome this huge limitation, such as higher order gradient continuum models [25, 26, 27, 28] or other remedies involving cumbersome techniques that treat fracture as a special case instead of an inherent material behavior [29, 30, 31, 32]. New approaches deal with models taking into account the microstructure of the materials [33], trying to express macroscopic behavior of solids from the microscopic level, such as lattice dynamics [34, 35, 36], or by linking macroscopic properties to microscopic ones [37, 38].

If the displacement is still continuous, for example in a change of phase, then partial derivatives can, actually, be employed in the weak solution of the underlying partial differential equation [39], but this is not applicable with material cracks, where the only useful method seems to be the one redefining the body, with the

crack considered to be a boundary, where physical laws regarding merely the relevant macroscopic properties are applied [40]. However, this technique can't be efficiently employed to keep track of defects if crack patters are complex. In both cases, crack/discontinuity paths have to be known *a priori* and the techniques provide reliable results only for a specific geometry/load configuration. In such cases, analytical work get more difficult and numerical approximation methods may become computationally costly.

Nonlocal theories

Material responses that couldn't yet be described are the primary reason for the massive development of nonlocal theories, able to take into consideration the influence of microstructures in the response of elastic media (see [35, 41]). In the 1970s, although finite-element programs made it suddenly feasible to simulate the distributed cracking observed in failure tests of concrete structures (e.g. concrete vessels for nuclear reactors), they demonstrated that a local inelastic constitutive law with strain-softening damage inevitably leads to spurious localization of damage into a zone of limit-to-zero volume [42]. Therefore, new *nonlocal* models that were variant with respect to a rescaling of partial coordinates and had an internal length were proposed.

The nonlocal theory of continuous media establishes a connection between the classical continuum mechanics and molecular dynamics models [43]. Atomistic simulations are the most detailed and realistic ones among all the possible models to describe material behavior, particularly material fractures [44, 45]. They have been used mostly to understand the basic physical mechanisms of crack formation and growth, rather than for prediction, because of the limited computational resources available, which make them unsuitable for describing complete engineering structures. In addition, nonlocal models where the long-range interaction length is related to the statistical length of heterogeneities [46, 47, 48] have been considerably employed in statistical continuum mechanics, providing the link between the laws governing the elementary particles and those of gross matter [49]. Therefore, they are particularly suitable to model phenomena both at small length scale

related to the intimate structure of materials and at bigger ones.

Nonlocal elasticity [50, 51, 52] has been first developed to improve the agreement between continuum theories and phenomena taking place in crystals on a scale comparable to the range of interatomic forces (vacancies and dislocations). For example, nonlocal models do not present classical singularities in the expressions of the stress field at the crack tips [53, 54]. There were some attempts to extend nonlocal theories to damage [55] and to plasticity [56, 57]. Although nonlocal continuum theory leads to finite stress at the crack tips, it still preserves the derivatives in the formulation which are unsuitable along discontinuities [43, 58].

Other types of nonlocal theories use displacement fields instead of their derivatives to prevent this shortcoming [41, 50].

Peridynamic theory

Peridynamics [1] is a member of the larger class of nonlocal formulations [59] of solid mechanics [35, 41, 50, 60]. It employs an integro-differential formulation without using partial derivatives; differently from the first introduced nonlocal theories, it includes nonlinear material behavior and damage within its original formulation, while most nonlocal theories were proposed for limited material responses.

The word Peridynamics is a portmanteau word coming from two Greek words, roots of near (peri, $\pi\varepsilon\rho\acute{\iota}$) and force (dynami, $\delta\acute{\upsilon}\nu\alpha\mu\eta$). The peridynamic formulation assumes that the body can be considered as a set of material points associated with an infinitesimal volume of the body. The particles can interact with each other not through contact (local) forces, like in the classical theory, but through nonlocal forces. The behavior of each particle depends on these forces computed thanks to an integral function of the displacement field within a neighborhood of the points on which the forces are evaluated.

The obvious advantage is that it can be applied to discontinuities without particular treatment or *ad hoc* criteria. The motion equation is similar to that of traditional molecular dynamics [59], in which interaction between discrete particles (i.e. atoms or molecules) is determined by integration of Newton's second law and forces and potential energy depends on molecular mechanic force fields (co-

valent bonds, electrostatic forces, Van der Waals forces). However, a fundamental difference between this two theories is that, while in molecular dynamics the material body is seen as a collection of discrete particles, in Peridynamics the material is considered continuous.

Peridynamics takes into consideration all the interaction forces of a point with points within a finite distance, called *horizon*, and the stresses at a certain point depend on the strain of all the points in the subregion corresponding to the horizon neighborhood, while in classic continuum models the state of a material point is influenced only by the points in its immediate vicinity, through contact forces. As the horizon increases, peridynamic models become the continuum form of molecular dynamic models. Peridynamics and molecular dynamics are so similar that peridynamic integration was recently done with LAMMPS Molecular Dynamics Simulator [61, 62]. Peridynamic theory turns out to be the link between molecular dynamic discrete theory and local classical continuum theory, having the possibility of describing material behavior on different length scales, from macroscale to nanoscale, since an internal characteristic length (the horizon) can be controlled. Peridynamic theory also contemplates nonlinear material response and includes damage as a feature of the original theory. As a result, this theory can capture spontaneous processes of crack formation and propagation, or any process related to discontinuities and singularities [63], wherever it is energetically favorable without resorting to special crack growth criteria. EMU computer code [64], the first peridynamic code, and *Peridigm* code [65], in which Silling is directly involved, are continuously developing, since many aspects have yet to be investigated in this theory.

State of the art

There are two main formulations for peridynamic models (see Figure 1): state-based (SBP) and bond-based (BBP), which was introduced first [1, 2].

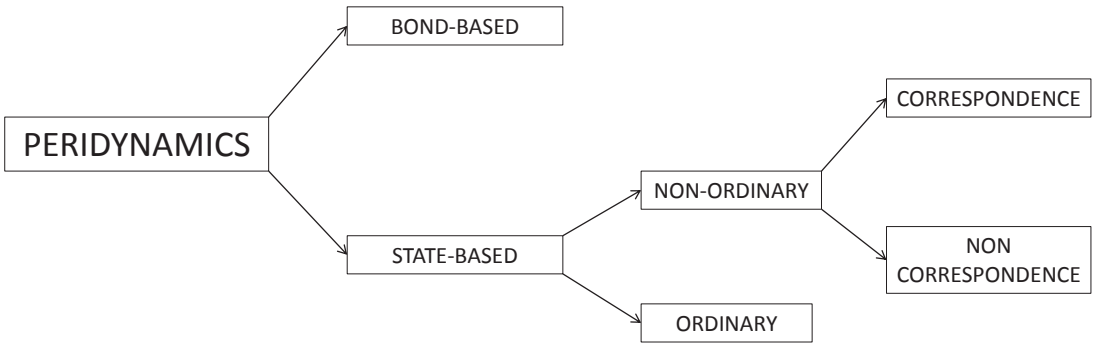


Figure 1: Different types of formulation in Peridynamics.

In the bond-based version, forces between two material points depend solely on their relative displacement. Bond-based Peridynamics is characterized by some limitations, such as a fixed Poisson's ratio: 1/4 in 3D or 2D plane strain, and 1/3 in 2D plane stress [1, 4]. Its value is exactly the same found by the Cauchy relation for a solid made of a lattice of points that interact only through a central force potential [66]. Nevertheless, bond-based Peridynamics has been used to predict dynamic crack growth (see e.g. [67, 68, 69, 70, 71, 72, 73]) and fatigue crack growth (see e.g. [74, 75]). As a matter of fact, to date, most of the peridynamic applications use the bond-based version. The effectiveness of peridynamic models has already been demonstrated in several sophisticated applications, including the fracture and failure of composites [71, 76], crack instability [77, 67], the fracture of polycrystals [78, 79], and nanofiber networks [80, 73].

SBP models remove this restriction by allowing interactions between two points to also depend on elongations of all other bonds connected to these two points. There are two types of SBP formulations: ordinary and non-ordinary [2, 5, 6]. In ordinary peridynamic formulations, the forces between two material points acts along the vector connecting the points in the deformed configuration. In non-ordinary formulation, this is not true. Ordinary state-based formulations require explicit derivation of constitutive models [7, 8].

For non-ordinary state-based formulation, Silling proposed two approaches: the development of explicit models for the peridynamic force states [2] and the development of maps, called *correspondence* models [2], incorporating classic mechanics constitutive relations to relate interaction forces and deformations. Non-ordinary

SBP models have been applied to study material elasticity, plasticity, fatigue and dynamic fracture [81, 82, 83, 84, 85, 86]. Correspondence models suffer from the problem of zero-energy modes [81]. Zero-energy modes arise in the correspondence formulation because of the weak coupling between peridynamic points and their own families [87]. Non-ordinary Peridynamics is characterized by a formulation similar to that of other meshfree methods: in fact, for uniform grid, the discretizations for the Element free Galerkin Method [88], the Reproducing Kernel Particle method [89] and Smoothed-Particle Hydrodynamics [90, 91, 92] are equivalent to SBP discretization [93]. All these methods seem to suffer from the zero-energy mode problem and different techniques to control or minimize it have been proposed in [94, 95, 96] and in particular in [97, 98, 99] for the correspondence formulation. However, the problem is not completely solved [81, 87]. Ordinary state-based formulation is not affected by zero-energy modes [85, 86].

Meshfree methods are characterized by other problems, such as how to impose boundary conditions: in the discretization of the classical equation it has been analyzed in papers as [100, 101, 102], while for Peridynamics this has been addressed in [103, 104] and further investigated for coupling with FEM in [64, 105, 106].

In this thesis, a uniform grid discretization with either a bond-based formulation or an ordinary state-based one are employed for all the numerical simulations. The "peridynamic stiffness matrix" for a linearized ordinary SBP model was derived, similarly to what was derived for peridynamic bond-based in [107]. The peridynamic stiffness matrix was used to solve elastostatic problems to verify that the peridynamic model is able to reproduce the intended elastic material properties, in particular different Poisson's ratios. While the peridynamic model should converge, in the limit to zero volume of the nonlocal region (the peridynamic horizon neighborhood), to the classical elasticity model [5]), the practical size of the nonlocal region to obtain solutions of acceptable engineering accuracy is analyzed. In the thesis, the difference between the solution of peridynamic problems and the exact solutions of the classical local problems is called "error", even if it is not an error: the two problems are distinct and the peridynamic formulation is expected to converge to the classical one only in the limit to zero horizon.

The main original contributions presented in this thesis are the following:

- the plane stress and the plane strain formulations for linearized 2D State-Based Peridynamics were derived in section 1.2.2.
- A set of fracture criteria, for the State-Based Peridynamics, has been proposed and numerically evaluated in section 1.2.3.
- A strategy to drastically reduce the computational time required for the assembly of stiffness and tangent matrix was defined in section 3.3.
- Convergence of the peridynamic solution to the classical elasticity by means of static analyses are studied in chapter 4.
- A new set of influence functions has been proposed and their effects on peridynamic solutions has been studied in section 4.2.3.
- Numerous numerical analyses (both static and dynamic) have been performed to show the capabilities of Peridynamics to reproduce experimentally obtained crack paths (also in mixed mode fracture) in chapter 5.

In addition, chapter 1 shows in detail the two main formulations of Peridynamics, in particular, it presents the linearized versions and the adopted failure criteria. Chapter 2 shows the main aspects of Peridynamics in its discretized counterpart, so to highlight important parameters for the numerical simulation to converge. Chapter 3 shows the developed solvers. In chapters 4 and 5, the results of the peridynamic problems are compared to the analytical solutions, when existing, or with experimental data. Chapter 6 draws conclusions and highlights some comments and possible future activities, with this thesis as a starting point.

Chapter 1

Mathematical Formulation

As previously mentioned, Peridynamics has been first introduced in 2000 by Silling [1]. Its introduction is due to his author's dissatisfaction with the existing theories lacking a comprehensive description of some of the most important phenomena involved in material responses, in particular formation of discontinuities and growth of cracks. In fact, most of the existing theories have been employing *ad hoc* criteria lacking generality, since they are developed according to specific geometric and load conditions and give good approximate results merely for those cases.

Differently from the classical theory, Peridynamics uses an integral formulation to compute the forces on a material particle and such equation remains valid even if discontinuities are involved. A unique framework of mathematical equations can be used both when discontinuities are involved and when they are not, thanks to the integral formulation. In this way, not only crack propagation direction do not have to be known *a priori*, but also crack initiation points do not have to be located in advance, since both phenomena are inherently captured.

Peridynamics is included in the set of nonlocal theories, since two body particles can interact with each other even when separated by a finite distance, provided it is smaller than a limit distance called *horizon*, which is a length scale similar to the one in molecular dynamics; according to this approach, contact forces can be considered a subset of integral forces in the limit to zero of the horizon and they are included in the formulation as additional nodal force densities [108]. Nonlocal linear elasticity and microcontinua have been studied for several decades [52, 60, 109] in

an effort to account for the long-range effects; however, differential operators and their limitations were still present in the formulation.

The fundamental element of this theory is the connection between two points, called "*bond*", and two mathematical formulations have been proposed, based on how bonds influence each other: if each bond acts independently of all the others, the variant is called *Bond-Based* Peridynamics (BBP), otherwise it is called *State-Based* Peridynamics (SBP).

BBP version was the first proposed by Silling [1]. The peridynamic force exerted through the bond can be unequivocally described by the relative initial and current position vectors between the two points identifying the bond, provided that the material properties have been associated to these points (not just microscopic ones, such as density, but also macroscopic ones, such as Young's modulus and Poisson's ratio): no further information from their surrounding points or bonds is required. Even though the main concept is preserved, the second version, SBP, has been innovative in its formulation. It has been introduced in 2007 [2] and more extensively described in 2010 [110]. Bonds can influence each other thanks to mathematical functions called *states*. Besides, the definitions of (nonlocal) strain and stress are included in the formulation. In this way, limitations typical of molecular dynamics or atomic lattice models, such as a fixed Poisson's ratio [3, 66], are overcome, so that general behaviors and complex phenomena can be described.

The peridynamic equation of motion for the BBP version is the following [1]:

$$\rho(\mathbf{x}_i)\ddot{\mathbf{y}}(\mathbf{x}_i,t)dV_i = \left[\int_{\mathcal{H}_{x_i}} \mathbf{f}\left(\mathbf{u}_j\left(t\right)-\mathbf{u}_i\left(t\right),\mathbf{x}_j-\mathbf{x}_i\right)\,dV_j + \mathbf{b}(\mathbf{x}_i,t)\right] dV_i \tag{1.1}$$

usually found in the form where the dimension of the terms is $[Force]/[Length^3]$

$$\rho(\mathbf{x}_i)\ddot{\mathbf{y}}(\mathbf{x}_i,t) = \int_{\mathcal{H}_{x_i}} \mathbf{f}\left(\mathbf{u}_j\left(t\right)-\mathbf{u}_i\left(t\right),\mathbf{x}_j-\mathbf{x}_i\right)\,dV_j + \mathbf{b}(\mathbf{x}_i,t) \tag{1.2}$$

where $\rho(\mathbf{x}_i)$ is the density of the point at the located at $\mathbf{x}_i$, $\ddot{\mathbf{y}}(\mathbf{x}_i,t)$ its current acceleration vector (i.e. $\mathbf{y}(\mathbf{x}_i,t)$ is its current position vector, see Figure 1.1), $\mathcal{H}_{x_i}$ the neighborhood of the point $\mathbf{x}_i$ containing all the points $\mathbf{x}_j$ interacting with i, $\mathbf{f}$ the pairwise force function, $\mathbf{u}_i\left(t\right)$ is the displacement of point $\mathbf{x}_i$, dV_j is the infinitesimal volume associated to a $\mathbf{x}_j$ point and $\mathbf{b}(\mathbf{x}_i,t)$ is the given body force density.

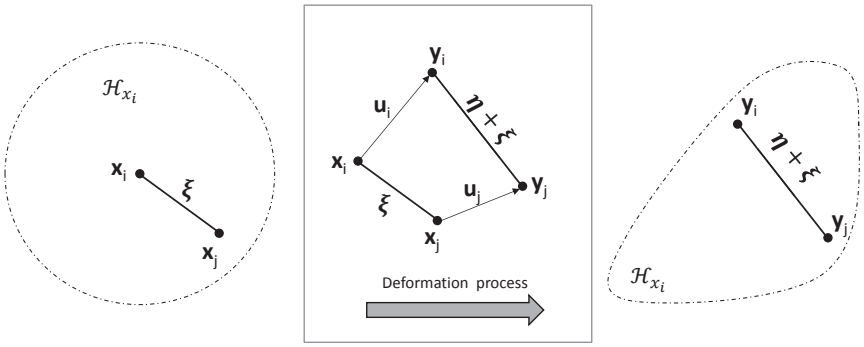


Figure 1.1: Description of the geometric quantities involved in Eq. (1.2) in a 2D case.

As for the SBP, the integral of the internal forces is substituted by the integral of the force state

$$\rho(\mathbf{x})\ddot{\mathbf{y}}(\mathbf{x},t) = \int_{\mathcal{H}} \left\{ \underline{T}\left[\mathbf{x},t\right]\langle \mathbf{p}-\mathbf{x}\rangle - \underline{T}\left[\mathbf{p},t\right]\langle \mathbf{x}-\mathbf{p}\rangle \right\} dV_p + \mathbf{b}(\mathbf{x},t) \tag{1.3}$$

where $\underline{T}\left[\mathbf{x},t\right]\langle \mathbf{p}-\mathbf{x}\rangle$ is the force state at time t, applied to the bond $\langle \mathbf{p}-\mathbf{x}\rangle$ and at point $\mathbf{x}$, dV_p is the infinitesimal volume associated to $\mathbf{p}$.

In both formulations, the horizon δ is the maximum distance at which two points are connected by a bond (i.e. the maximum initial length of a bond is equal to δ), but that does not mean that it is the maximum distance at which a peridynamic force acts between two points for both of them - this aspect is explained in the linearization process.

As for the boundary conditions, the enforcement of nonstandard nonlocal displacement loading conditions in Peridynamics has been studied in [103] and requires special treatments similar to those in the meshfree Galerkin methods [111]. Coupling schemes with the finite element method [112, 113], ghost particles [61] and alteration of particle volume integration near the boundary [114] have been developed to impose essential boundary conditions.

Let us consider a body subjected to mechanical loads and divide it into two domains (i.e. halves of the horizon neighborhood) [114, 115]: the material points located in domain Ω_+ interact with the points in domain Ω_-, as shown in Figure 1.2. The force densities acting on points in domain Ω_+ must be determined by integrating the response function over domain Ω_- as

$$\int_{\Omega_-} f \, dV_- \tag{1.4}$$

The resulting force can be computed by volume integration of these force densities over domain Ω_+

$$\mathbf{F} = \int_{\Omega_+} \int_{\Omega_-} f \, dV_- dV_+ \tag{1.5}$$

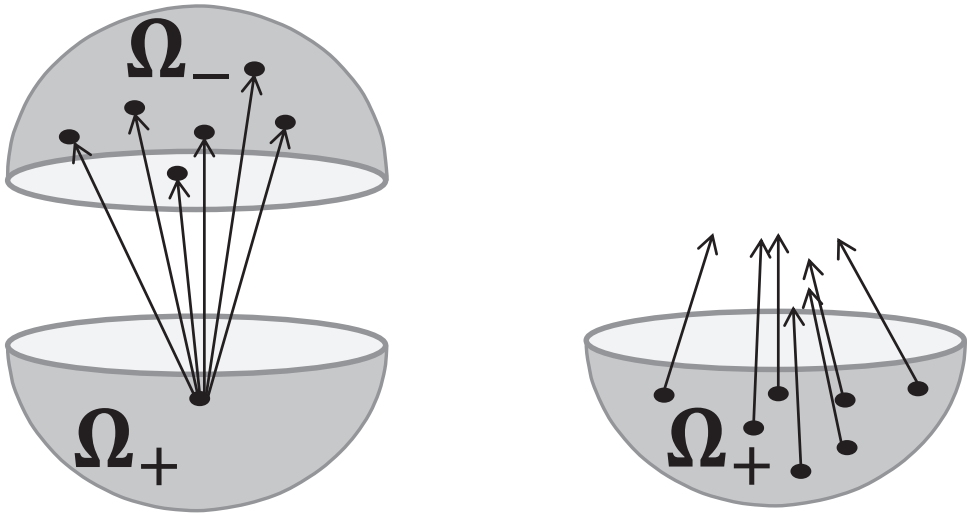


Figure 1.2: Boundary conditions: interaction of a point in domain Ω_+ with domain Ω_- and force densities acting on domain Ω_+ due to domain Ω_-.

Therefore, the tractions or point forces cannot be applied as boundary conditions, since their volume integrations result in a zero value: they are applied over the volumes as body forces, displacements, and velocities [114].

In general, being a nonlocal model, Peridynamics suffer from a "surface effect": the constant parameters linking the peridynamic model to the material properties are computed assuming the horizon neighborhood of the point fully embedded in the body, but this is not true whenever a point is near a surface. Due to the lack of integration points, the behavior of the material near the boundary is unexpected. Several studies have been done to avoid it or at least reduce it [64, 116], but in the simulations in this thesis no surface correction are adopted because the existing techniques are not general.

The difference between these two variants is explained in the following paragraphs through an image connected to simple concepts. Looking at Figure 1.3, imagine that the rope is a bond of the structure and that each person in the picture represents a particle in the analyzed body. In Figure 1.3, the two nodes are represented by the two individuals pulling with the same amount of force. They "behave" similarly towards that specific bond (i.e. the rope), that's the reason why they are portrayed as twins (i.e. considering a different rope connected to one of the two main protagonists of this figure, the individual will change aspect and will look like a twin of the guy at the other end on the new rope). No other particle (person in the figure) affects their pairwise interaction. The crowd is just there and see their game and does not take part in it.

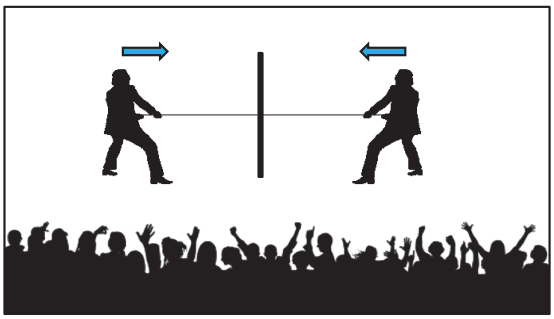


Figure 1.3: Representation of BBP theory through a play.

In order to compare the material behavior simulation of BBP and SBP, a more detailed picture (Figure 1.5) of what the twins are actually pulling is needed. They are not pulling one rope, but two at the same time through a rigid handle: one of the ropes is connected to a fixed pole (Figure 1.3 shows it as a black stick while Figure 1.5 as a black circle), the other is directly linking the twins. The force exclusively exerted by one twin can be seen in the tension of the rope connected to the fixed pole (see Figure 1.4: their forces are independent from each other, but they turn out to be equal in magnitude due to the BBP formulation).

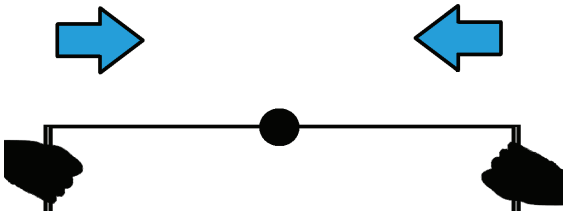


Figure 1.4: Representation of the actual forces exerted by the twins on the rope connected to the fixed pole (the black circle).

What can be computed through the BBP formulation is the tension of the rope linking the two twins. The total tension acting along the rope is twice the single twin force. Therefore, every node contributes for half of the total force acting along a bond [110]. Huilong et al. in [117] describe them as the algebraic sum of the active force exerted by the particle and the passive (or received) one exerted by the other particle on the first one. In BBP, they are always equal in magnitude[1] for a homogeneous material. However, the internal force computed in the BBP theory does not take into consideration these forces separately, its contribution is directly the algebraic sum (i.e. points contributing for exactly half of the total force has been formally pointed out only when SBP has been proposed, 10 years after the introduction of BBP). Figure 1.5 shows the forces actively exerted by the single twin on the upper part of the picture, and their algebraic sum (made by the two forces and as a unique force vector) in the lower part. BBP explicitly deals with the forces in the box, the half forces do not take part in the formulation.

[1]This is theoretically true, but when discretization is involved, issues related to type of grid and horizon size of the single points might affect the validity of this statement, for example when the horizon size is not constant throughout the structure, ghost forces arise at the interface and this statement is not valid anymore.

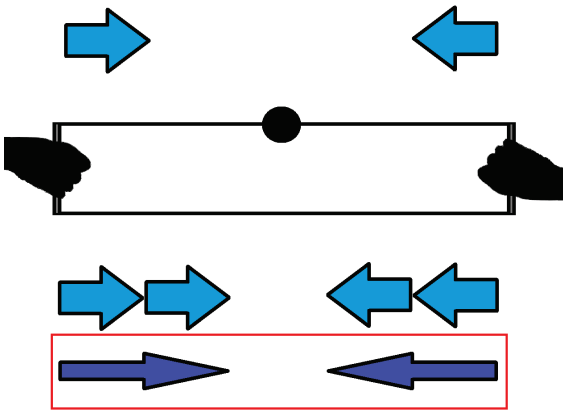


Figure 1.5: Representation of the actual ropes pulled by the twins. The forces explicitly indicated in the BBP formulation are those in the box.

Figure 1.6 shows what happens in SBP. The behaviors of particles connected by a bond differ from each other, because they depend on the influence of their surrounding particles. Therefore, the amount of force they "actively" exert on the other is different (see Figure 1.7).

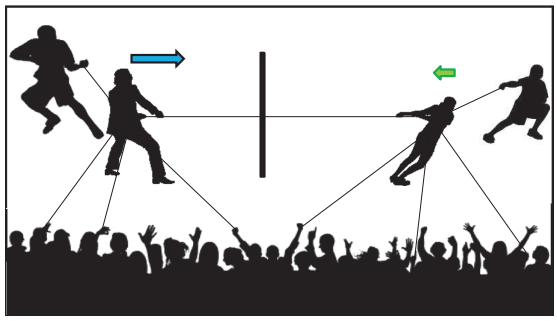


Figure 1.6: Representation of SBP theory.

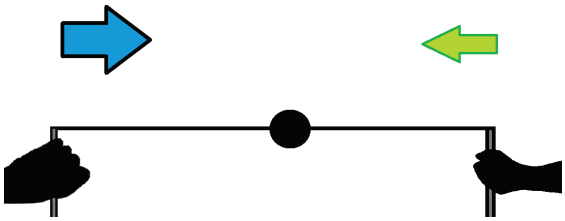


Figure 1.7: Representation of the different actual forces exerted by the two individuals on the rope connected to the fixed pole.

The active forces are those exerted on the rope connected to the fixed pole (Figure 1.7): in general, they are different from each other. In this thesis, only the

ordinary type of SBP has been investigated, but, as previously mentioned, SBP is more general and the exerted forces can be not aligned with the current relative position vector (in this case, this example would be not thorough).

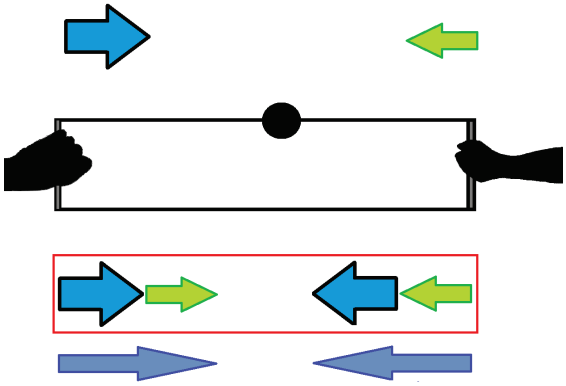


Figure 1.8: Representation of the actual ropes pulled by the two people. The forces explicitly indicated in the SBP formulation are those in the red box.

Figure 1.8 shows in the upper part the exerted forces by the guys distinctively, while it shows their algebraic sum in the lower part. In SBP, the formulation deals directly with the separate contributions of the two guys and their algebraic sum is explicitly present in the equation of motion (Eq. (1.3)), while the BBP deals with the sum (Eq. (1.2)) in a model too simplified to allow for a varying Poisson's ratio.

In sections 1.1 and 1.2 the two formulations are described in a detailed way.

1.1 BBP Version

Suppose a body $\mathcal{B}$ occupies a region in a reference system $Oxyz$. This theory assumes that the body is composed by material points located at generic positions $\mathbf{x}_i$ to which infinitesimal volumes dV_i are associated. The body, differently from molecular dynamics, is considered to be continuous (at least initially within its boundary), even if not necessarily homogeneous (i.e. density $\rho(\mathbf{x}_i)$ can change from material point to material point) or isotropic. There are, indeed, no assumptions on the material required for the validity or derivation of the theory, as well as no assumptions that the internal forces are zero in the reference configuration.

Consider two points identified by their initial position $\mathbf{x}_i$ and $\mathbf{x}_j$ (see Figure 1.9), define the relative position of these two material points in the reference configuration (i.e. at time $t = 0$)

$$\boldsymbol{\xi} = \mathbf{x}_j - \mathbf{x}_i \tag{1.6}$$

and their relative displacement at time $t \geq 0$ indicated by $\boldsymbol{\eta}$

$$\boldsymbol{\eta} = \mathbf{u}\left(\mathbf{x}_j, t\right) - \mathbf{u}\left(\mathbf{x}_i, t\right) = \mathbf{u}_j\left(t\right) - \mathbf{u}_i\left(t\right) \tag{1.7}$$

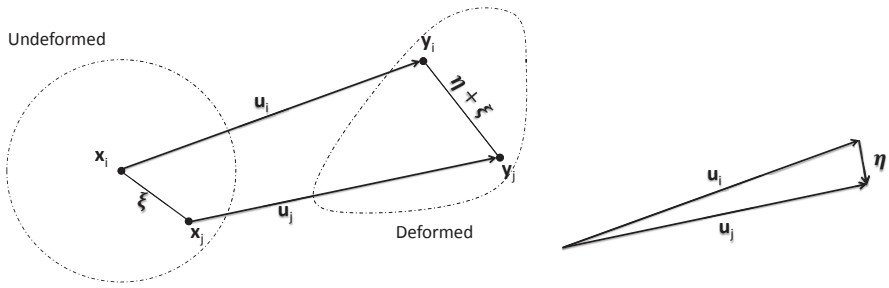


Figure 1.9: Relative positions (initial and current), displacements and relative displacement vector.

Their interaction, called *bond*, is a pairwise non local force, also called *response function force*, which is a vector-valued function of their relative initial position and of their relative displacement, as well as of the material properties at the points. Its dimension is force per unit volume squared. In the equation of motion the acceleration of each material point is influenced by this force through the following integral.

$$\mathbf{L}_{\mathbf{u}}\left(\mathbf{x}_i, t\right) = \int_{\mathcal{B}} \mathbf{f}\left(\mathbf{u}\left(\mathbf{x}_j, t\right) - \mathbf{u}\left(\mathbf{x}_i, t\right), \mathbf{x}_j - \mathbf{x}_i\right) dV_j \qquad \forall \mathbf{x}_j \in \mathcal{B} \quad t \geq 0 \tag{1.8}$$

where $\mathbf{f}$ is, in fact, the pairwise peridynamic force vector function that the material point $\mathbf{x}_j$ exerts on point $\mathbf{x}_i$ and dV_j is the infinitesimal volume associated to point $\mathbf{x}_j$. Thus, the equation of motion proposed in the original version of peridynamic theory is

$$\rho(\mathbf{x}_i)\ddot{\mathbf{y}}\left(\mathbf{x}_i, t\right) = \rho(\mathbf{x}_i)\ddot{\mathbf{u}}\left(\mathbf{x}_i, t\right) = \mathbf{L}_{\mathbf{u}}\left(\mathbf{x}_i, t\right) + \mathbf{b}\left(\mathbf{x}_i, t\right) \tag{1.9}$$

where $\rho(\mathbf{x}_i)$ is mass density, $\ddot{\mathbf{y}}(\mathbf{x}_i, t)$ is the acceleration of the point[2] and $\mathbf{b}(\mathbf{x}_i, t)$ is a given body force density, which represents the external force per unit reference volume.

In this first developed theory, each bond is independent of all other local condition, as well as of the deformation steps which might lead to the deformed configuration, since it is referred to the initial configuration. Therefore, any kind of damage model related to the irreversible rupture of bonds is not yet included.

It is defined a positive number δ called *horizon* so that the peridynamic force exists even if two particles are not in contact, provided that their distance is less than the horizon length

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = 0 \qquad \forall |\boldsymbol{\xi}| > \delta, \ \forall \boldsymbol{\eta} \tag{1.10}$$

It is reasonable (in comparison to molecular dynamics, where such a parameter is not involved) to assume that no significant interaction occurs between points which are farther than the horizon. As a consequence of this assumption, the equation of motion is limited to the integration of the forces in the domain $\mathcal{H}_{x_i}$, which is a spherical neighborhood (in three-dimensional problems, while it is a circular neighborhood in two-dimensional problems and a linear neighborhood in mono-dimensional ones) of $\mathbf{x}_i$ in $\mathcal{B}$ within the range identified by δ:

$$\rho(\mathbf{x}_i)\ddot{\mathbf{u}}(\mathbf{x}_i, t) = \int_{\mathcal{B}} \mathbf{f}(\mathbf{u}(\mathbf{x}_j, t) - \mathbf{u}(\mathbf{x}_i, t), \mathbf{x}_j - \mathbf{x}_i) \, dV_j + \mathbf{b}(\mathbf{x}_i, t) \tag{1.11}$$
$$\forall \mathbf{x}_i \in \mathcal{B}, \ \forall \mathbf{x}_j \in \mathcal{H}_{x_i}, \ t \geq 0$$

in fact, in a more rigorous way, the mathematical domain in the BBP version is

$$\mathcal{H}_{x_i} = \{\mathbf{x}_j \in \mathbb{R}^3 \mid |\mathbf{x}_j - \mathbf{x}_i| < \delta\} \tag{1.12}$$

The pairwise peridynamic force has to satisfy Newton's Third Law, for conservation of linear momentum, so that it has to be

$$\mathbf{f}(-\boldsymbol{\eta}, -\boldsymbol{\xi}) = -\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) \tag{1.13}$$

[2]Note that the acceleration of a node may be identified either by $\ddot{\mathbf{y}}(\mathbf{x}_i, t)$ or $\ddot{\mathbf{u}}(\mathbf{x}_i, t)$, because they are equal, even though $\mathbf{y}(\mathbf{x}_i, t) \neq \mathbf{u}(\mathbf{x}_i, t)$.

Conservation of angular momentum has to be satisfied as well, verifying the condition

$$(\boldsymbol{\eta} + \boldsymbol{\xi}) \times \mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = 0 \tag{1.14}$$

which means that the force vector has to be parallel to the current relative position vector, so $\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi})$ can be also expressed as

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = F(\boldsymbol{\eta}, \boldsymbol{\xi})(\boldsymbol{\eta} + \boldsymbol{\xi}) \qquad \forall\, \boldsymbol{\eta}, \boldsymbol{\xi} \tag{1.15}$$

where $F(\boldsymbol{\eta}, \boldsymbol{\xi})$ is an appropriate scalar value even function.

In BBP, elasticity can be introduced through the definition of *microelasticity*. A material is called microelastic if it fulfills the following condition [1]:

$$\oint_{\Gamma} \mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) \cdot d\boldsymbol{\eta} = 0 \qquad \forall\, \textit{closed curve}\, \Gamma,\ \forall\, \boldsymbol{\xi} \neq \mathbf{0} \tag{1.16}$$

where $d\boldsymbol{\eta}$ is an elementary vector path length along the closed curve Γ. Such condition means that the line integral is path independent so that the net work done by the response force along any closed curve is zero, similarly to elasticity in the classical theory. In the same way, as Stokes' Theorem states, if $\mathbf{f} = (f_1, f_2, f_3)$ is continuously differentiable with respect to $\boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3)$, a necessary and sufficient condition for respecting the previous property is for the vector curl operator evaluated with respect to the coordinates of $\boldsymbol{\eta}$ to be zero:

$$\begin{aligned} \nabla_{\boldsymbol{\eta}} \times \mathbf{f} = & \\ = \left(\frac{\partial f_3}{\partial \eta_2} - \frac{\partial f_2}{\partial \eta_3}\right)\mathbf{i} + \left(\frac{\partial f_1}{\partial \eta_3} - \frac{\partial f_3}{\partial \eta_1}\right)\mathbf{j} + \left(\frac{\partial f_2}{\partial \eta_1} - \frac{\partial f_1}{\partial \eta_2}\right)\mathbf{k} = \mathbf{0} & \\ \forall\, \boldsymbol{\xi} \neq \mathbf{0} & \end{aligned} \tag{1.17}$$

where, if the orthonormal base of the reference configuration system $Oxyz$ is $(\mathbf{i}, \mathbf{j}, \mathbf{k})$.

As a consequence of Stokes' Theorem, since the vector force field is conservative and irrotational, the pairwise force function can be derived from a micropotential function ω, which is a differentiable and scalar-valued function, such that

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = \frac{\partial \omega}{\partial \boldsymbol{\eta}}(\boldsymbol{\eta}, \boldsymbol{\xi}) \tag{1.18}$$

The micropotential depends only on the relative displacement vector through the scalar distance between the deformed points. Therefore, a general peridynamic force function for a microelastic material can be written as a scalar value depending on the relative distance and is aligned with the relative position vector

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi})(\boldsymbol{\eta} + \boldsymbol{\xi}) \qquad \forall\, \boldsymbol{\eta}, \boldsymbol{\xi} \tag{1.19}$$

where $H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi})$ is a scalar-valued even function obtained by derivation of the micropotential (it is $F(\boldsymbol{\eta}, \boldsymbol{\xi})$ in Eq. (1.15), but here the dependence on the relative distance is highlighted).

In BBP theory, the simplest developed model is the prototype microelastic brittle (PMB) constitutive model [68]. The bonds in such material are similar to springs in classic solid mechanics. The peridynamic force is a linear function of the bond stiffness, also called *spring stiffness*, c, though, generally, it can be a varying function of the relative position $\boldsymbol{\xi}$. The ratio between the vector force $\mathbf{f}$ and this spring constant is the bond stretch s; the bond breaks and, consequently, fails when its stretch reaches a limit value s_0, called *critical stretch* [68]. The details on how to compute it are in section 1.1.2. After the rupture, the bond is not taken into consideration anymore, because the process is irreversible. For this type of material, the scalar-valued even function $H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi})$ is a linear function of the spring stiffness and of the bond stretch:

$$H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})\,s}{|\boldsymbol{\eta} + \boldsymbol{\xi}|} \tag{1.20}$$

where $\mu(\boldsymbol{\xi})$ is the history dependent scalar-valued function which takes into consideration bond failure state and it is expressed as

$$\mu(\boldsymbol{\xi}) = \begin{cases} 1 & s < s_0 \\ 0 & s \geq s_0 \end{cases} \tag{1.21}$$

and s is the current bond stretch, which is defined as

$$s = \frac{|\boldsymbol{\eta} + \boldsymbol{\xi}| - |\boldsymbol{\xi}|}{|\boldsymbol{\xi}|} \tag{1.22}$$

A clarification has to be pointed out for Eq. (1.21): it would be correct to define the condition on the maximum time-history stretch $(max(s(t)) \gtreqless s_0,\ for\ t >$

0) because $\mu(\boldsymbol{\xi}) = 0$ even though the current stretch is lower than the critical one, assuming that the stretch has overcome the limit s_0 during the load history. Nevertheless, this expression is commonly found in literature, since as soon as the stretch overcomes the limit, the bond is irreversibly broken and it is not considered anymore in the computation (no stretch or other variables are computed for the next steps). This means that, if the current stretch is less than the limit (i.e. it is still computed), all the stretch values of the previous time steps satisfy this condition too.

Hence, the pairwise peridynamic BBP force vector for a prototype microelastic brittle material is

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})\,s\frac{\boldsymbol{\eta} + \boldsymbol{\xi}}{|\boldsymbol{\eta} + \boldsymbol{\xi}|} \tag{1.23}$$

The spring constant c for a prototype microelastic brittle material in three dimensional models results [68]

$$c = \frac{18K}{\pi\delta^4} = \frac{6E}{\pi\delta^4(1-2\nu)} \tag{1.24}$$

where K is the bulk modulus of the material, δ the horizon length, E Young's modulus and ν Poisson's ratio. The peridynamic spring constant c for a two-dimensional structure can be determined by surface-integration of an infinitely large plate[3] in which a uniform expansion loading and a pure shear are applied separately, so that the elastic energy can be computed both for plane stress and plane strain conditions [4], from which the values of the bond stiffness is

$$c = \frac{12E}{\pi t_h \delta^3(1+\nu)} \qquad \begin{cases} plane\ stress & \nu = \frac{1}{3} \\ plane\ strain & \nu = \frac{1}{4} \end{cases} \tag{1.25}$$

where t_h is the plate thickness.

1.1.1 Linearized prototype microelastic brittle model

In this thesis, the studied cases taken into consideration are mainly focused on elastic response of materials, even though general peridynamic theory does not restrict its validity to such limitations and it can be applied to large deformation

[3] This equation is not valid near the boundary, thus the surface effect arises.

cases. Dealing with small displacements and deformations, a linearization of the vector force function can be introduced, by solving the equation in an incremental approach.

Assume $\boldsymbol{\eta} \ll 1$, the vector force function in a neighbor of the point $(\mathbf{0}, \boldsymbol{\xi})$ can be represented by a Taylor's series expansion of the first order. Keeping $\boldsymbol{\xi}$ constant, the vector force function becomes

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = C(\boldsymbol{\xi})\,\boldsymbol{\eta} + \mathbf{f}(\mathbf{0}, \boldsymbol{\xi}) \tag{1.26}$$

where $C(\boldsymbol{\xi})$ is a second-order tensor, called *micromodulus*, of the vector force which in general can be computed as

$$C(\boldsymbol{\xi}) = \frac{\partial \mathbf{f}}{\partial \boldsymbol{\eta}}(\boldsymbol{\eta}, \boldsymbol{\xi}) = \begin{bmatrix} \frac{\partial f_x}{\partial \eta_x}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_x}{\partial \eta_y}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_x}{\partial \eta_z}(\boldsymbol{\eta}, \boldsymbol{\xi}) \\ \frac{\partial f_y}{\partial \eta_x}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_y}{\partial \eta_y}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_y}{\partial \eta_z}(\boldsymbol{\eta}, \boldsymbol{\xi}) \\ \frac{\partial f_z}{\partial \eta_x}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_z}{\partial \eta_y}(\boldsymbol{\eta}, \boldsymbol{\xi}) & \frac{\partial f_z}{\partial \eta_z}(\boldsymbol{\eta}, \boldsymbol{\xi}) \end{bmatrix} \tag{1.27}$$

The tensor from Eq. (1.27) can be written as

$$C(\boldsymbol{\xi}) = \boldsymbol{\xi} \otimes \frac{\partial F}{\partial \boldsymbol{\eta}}(\mathbf{0}, \boldsymbol{\xi}) + F(\mathbf{0}, \boldsymbol{\xi})\, I \tag{1.28}$$

where $\otimes$ is the dyadic product or tensor product between two vectors which gives as a result a tensor of second order and F is a scalar-valued even function which relates the force vector to the current relative position vector. $C(\boldsymbol{\xi})$ is symmetric, due to the Stokes' theorem validity for a microelastic material. This means that a number $\lambda(\boldsymbol{\xi})$ exists so that

$$\boldsymbol{\xi} \otimes \frac{\partial F}{\partial \boldsymbol{\eta}}(\mathbf{0}, \boldsymbol{\xi}) = \lambda(\boldsymbol{\xi})\, \boldsymbol{\xi} \otimes \boldsymbol{\xi} + F(\mathbf{0}, \boldsymbol{\xi})\, I \tag{1.29}$$

Hence, the linearized vector force function can be explicitly reformulated as

$$\mathbf{f}(\boldsymbol{\eta}, \boldsymbol{\xi}) = [\lambda(\boldsymbol{\xi})\, \boldsymbol{\xi} \otimes \boldsymbol{\xi} + F(\mathbf{0}, \boldsymbol{\xi})\, I]\, \boldsymbol{\eta} + \mathbf{f}(\mathbf{0}, \boldsymbol{\xi}) \tag{1.30}$$

For the prototype microelastic brittle model,

$$F(\boldsymbol{\eta}, \boldsymbol{\xi}) = H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi}) \tag{1.31}$$

where $H(|\boldsymbol{\eta} + \boldsymbol{\xi}|, \boldsymbol{\xi})$ is taken from Eq. (1.20). So

$$F(\boldsymbol{\eta}, \boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\, \mu(\boldsymbol{\xi})\, s}{|\boldsymbol{\eta} + \boldsymbol{\xi}|} \tag{1.32}$$

Therefore

$$\frac{\partial F}{\partial \boldsymbol{\eta}}(\boldsymbol{\eta},\boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\xi}|}\frac{\partial}{\partial \boldsymbol{\eta}}\left(1-\frac{|\boldsymbol{\xi}|}{|\boldsymbol{\eta}+\boldsymbol{\xi}|}\right) = -c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})\frac{\partial}{\partial \boldsymbol{\eta}}\frac{1}{|\boldsymbol{\eta}+\boldsymbol{\xi}|} \tag{1.33}$$

This derivative becomes [118]

$$\frac{\partial F}{\partial \boldsymbol{\eta}}(\boldsymbol{\eta},\boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\eta}+\boldsymbol{\xi}|^3}(\boldsymbol{\eta}+\boldsymbol{\xi}) \tag{1.34}$$

and its evaluation at $(\mathbf{0},\boldsymbol{\xi})$ is

$$\frac{\partial F}{\partial \boldsymbol{\eta}}(\mathbf{0},\boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\xi}|^3}\boldsymbol{\xi} \tag{1.35}$$

From Eq. (1.28)

$$C(\boldsymbol{\xi}) = \begin{bmatrix}\xi_x\\ \xi_y\\ \xi_z\end{bmatrix} \otimes \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\xi}|^3}\begin{bmatrix}\xi_x\\ \xi_y\\ \xi_z\end{bmatrix} = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\xi}|}\begin{bmatrix}\frac{\xi_x^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_x\xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_x\xi_z}{|\boldsymbol{\xi}|^2}\\ \frac{\xi_y\xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_y^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_y\xi_z}{|\boldsymbol{\xi}|^2}\\ \frac{\xi_z\xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_z\xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_z^2}{|\boldsymbol{\xi}|^2}\end{bmatrix} \tag{1.36}$$

where the matrix contains the directional cosines of the bond vector in the global reference system. The force vector at $(\mathbf{0},\boldsymbol{\xi})$ is

$$\mathbf{f}(\mathbf{0},\boldsymbol{\xi}) = \frac{c(\boldsymbol{\xi})\,\mu(\boldsymbol{\xi})}{|\boldsymbol{\xi}|}\begin{bmatrix}\frac{\xi_x^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_x\xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_x\xi_z}{|\boldsymbol{\xi}|^2}\\ \frac{\xi_y\xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_y^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_y\xi_z}{|\boldsymbol{\xi}|^2}\\ \frac{\xi_z\xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_z\xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_z^2}{|\boldsymbol{\xi}|^2}\end{bmatrix}\begin{bmatrix}\eta_x\\ \eta_y\\ \eta_z\end{bmatrix} \tag{1.37}$$

This procedure can be employed for any starting point, not only for $(\mathbf{0},\boldsymbol{\xi})$, as described in Eq. (1.26).

1.1.2 Failure Criterion

The adopted failure criterion for BBP is a maximum stretch criterion introduced in [68]. The bond stretch s has the same form of the engineering strain in the classic continuum theory, a linear proportionality between bond elongation and macroscopic elongation is expected. The critical bond stretch s_0 can be related to macroscopic known quantities such as the energy release rate when the structure is subjected to isotropic extension till its critical value G_0 or the dissipated energy per unit area of fracture surface during the growing of a crack. This relation can be established under the assumptions of complete separation of the fracture surfaces

and of the absence of other dissipative mechanisms [68]. The work required to break a single bond for the PMB model is equal to

$$\omega_0 = \int_0^{s_0} H(s)\, d\eta = \int_0^{s_0} H(s)\, \xi ds = \frac{c s_0^2 \xi}{2} \tag{1.38}$$

In order to create a new fracture surface, all the bonds crossing the mentioned surface (i.e. connecting pairs of points of different subregions separate by the surface itself) have to be broken and the energy per unit surface area required to break them all is equal to the critical energy release rate G_0 of classic continuum theory, as assumed in Griffith's theory. Griffith's criterion [119] is a nonlocal criterion, since it is based on the energy balance of the whole material surrounding the crack. The two energies can be related as

$$G_0 = \int_0^{2\pi} \int_0^{\delta} \int_z^{\delta} \int_0^{\arccos\left(\frac{z}{\xi}\right)} \left(\frac{c s_0^2 \xi}{2}\right) \xi^2 \sin\phi \, d\phi \, d\xi \, dz \, d\theta = \frac{\pi c s_0^2 \delta^5}{10} \tag{1.39}$$

where the variables can be seen in Figure 1.10.

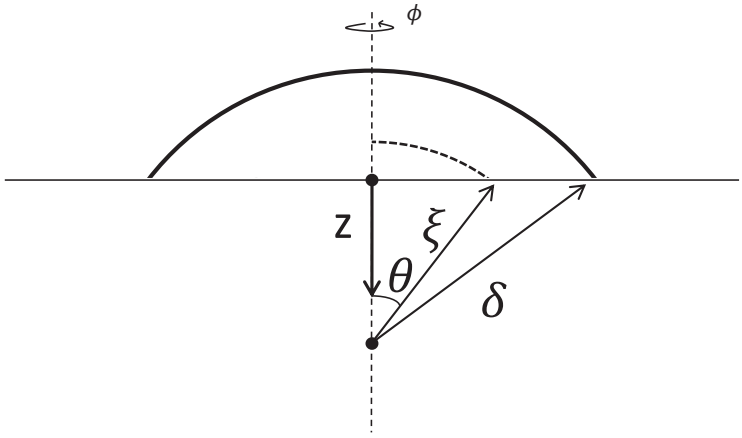


Figure 1.10: Variables involved in the computation of the critical stretch value.

s_0 can be evaluated as

$$s_0 = \sqrt{\frac{10 G_0}{\pi c \delta^5}} = \sqrt{\frac{5 G_0}{9 K \delta}} \tag{1.40}$$

for three-dimensional structures, while for two-dimensional it becomes

$$s_0 = \sqrt{\frac{4\pi G_0}{9 E \delta}} \tag{1.41}$$

For mode I opening, the energy release rate can be related to the stress intensity factor K_I or to the *fracture toughness* through the relation

$$G_0 = \frac{K_I^2}{E'} \tag{1.42}$$

where $E' = E$ for plane stress conditions while $E' = E/(1-\nu^2)$

The local damage index at the point located at $\mathbf{x}_i$ is denoted by the function $\phi(\mathbf{x}_i, t)$ defined as

$$\phi(\mathbf{x}_i, t) = 1 - \frac{\int_{\mathcal{B}} \mu(\mathbf{x}_i, t)\, dV_j}{\int_{\mathcal{B}} dV_j} \tag{1.43}$$

Therefore, it is the ratio of the sum of broken interactions and all the initial interactions of the points.

1.2 SBP Version

Peridynamic states were first introduced by Silling in [2] and then described in a more sistematic way in [110]. They can be considered as functions useful to map pair of points (or better bonds) into some quantity.

Consider a body $\mathcal{B}$ and a positive number representing the maximum interaction length, called *horizon*, usually identified by the Greek letter δ (see Figure 1.11). In this body, consider a material point, usually identified by an ID number n_x and its position $\mathbf{x}$ in an inertial reference system, and all its family nodes, n_p located at $\mathbf{p}$, within its horizon sphere.

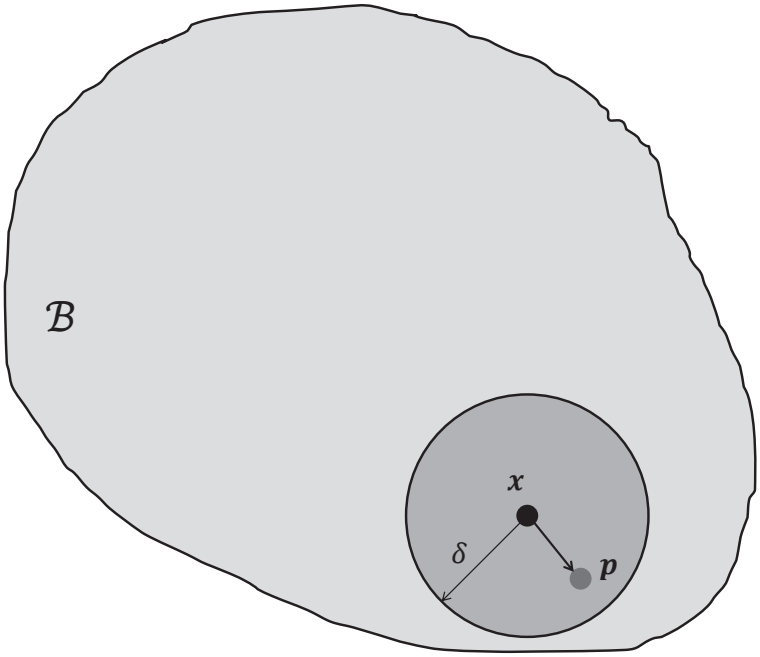


Figure 1.11: Horizon sphere of the material point $\mathbf{x}$ within the body $\mathcal{B}$ and one of its family point $\mathbf{p}$.

The general equation of motion is

$$\rho(\mathbf{x})\ddot{\mathbf{y}}(\mathbf{x},t) = \int_{\mathcal{H}} \{\underline{T}[\mathbf{x},t]\langle\mathbf{p}-\mathbf{x}\rangle - \underline{T}[\mathbf{p},t]\langle\mathbf{x}-\mathbf{p}\rangle\}\, dV_p + \mathbf{b}(\mathbf{x},t) \tag{1.44}$$

where $\rho(\mathbf{x})$ is the density of the point $\mathbf{x}$, $\ddot{\mathbf{y}}(\mathbf{x},t)$ its acceleration at time t, $\underline{T}[\mathbf{x},t]\langle\mathbf{p}-\mathbf{x}\rangle$ the force exerted by $\mathbf{x}$ on $\mathbf{p}$, $\underline{T}[\mathbf{p},t]\langle\mathbf{x}-\mathbf{p}\rangle$ the force exerted by $\mathbf{p}$ on $\mathbf{x}$, dV_p the infinitesimal volume associate to point $\mathbf{p}$, and $\mathbf{b}(\mathbf{x},t)$ the force density.

Differently from the domain in BBP which is stated in Eq. (1.12), in SBP a new set of elements, called *bond domain*, is employed to define the "states". The set is

$$\mathcal{H} = \{\boldsymbol{\xi} \in (\mathbb{R}^3 \setminus \mathbf{0}) \mid (\boldsymbol{\xi} + \mathbf{x}) \in \mathcal{H}_x \cap \mathcal{B}\} \tag{1.45}$$

$\mathcal{H}$ contains vectors, called *bonds* and it's centered at $\mathbf{0}$.

A peridynamic state is a *function* defined in the bond domain: it is applied to one or more bonds, so that every state maps them into a quantity. A state is called *scalar* if the output quantity is scalar (i.e. the state belongs to the set of all tensor of order 0, called $\mathcal{L}_0$[4]), *vector* if this quantity is a vector (first order) or

[4] $\mathcal{L}_m$ denotes the set of all tensors of order m, thus $\mathcal{L}_0 = \mathbb{R}$ [2]

double state if it maps pairs of bonds $(\boldsymbol{\xi},\ \boldsymbol{\zeta}) \in \mathcal{H}$ into second order tensors[5].

A vector state is equivalent to a second order tensor in the classical theory of solid mechanics, since it maps vectors into vectors, but it is not required to be either a linear transformation of $\boldsymbol{\xi}$ or a continuous one. As a matter of fact, any state may be a *noncontinuous, nonlinear function* of bonds, differently from the usual tensors. Besides, while the set of vector states (identified by $\mathcal{V}$) is an infinite-dimensional real Euclidean space, the set $\mathcal{L}_2$ of second order tensors in classical mechanics has dimension 9. Although the sets are mathematically different, both of them are real Euclidean spaces, and thus denoted by the same symbols, bearing in mind the broader meaning when used in the peridynamic formulation.

States usually are identified by an underscore, angle brackets $\langle\cdot\rangle$ identify the bond on which they are applied, parentheses $(\cdot)$ identify the state on which the state depends (if any) and square brackets $[\cdot]$ identify other quantities on which the state may depend, such as the source point and the time step.

Several mathematical definitions are provided in [2, 110, 120]. The *sum* and the *difference* are defined for two states of the same order as

$$\underline{A}\langle\cdot\rangle : \mathcal{H} \mapsto \mathcal{L}_m,\ \underline{B}\langle\cdot\rangle : \mathcal{H} \mapsto \mathcal{L}_m \quad \mapsto \quad \begin{aligned} (\underline{A} + \underline{B})\langle\boldsymbol{\xi}\rangle &= \underline{A}\langle\boldsymbol{\xi}\rangle + \underline{B}\langle\boldsymbol{\xi}\rangle \\ (\underline{A} - \underline{B})\langle\boldsymbol{\xi}\rangle &= \underline{A}\langle\boldsymbol{\xi}\rangle - \underline{B}\langle\boldsymbol{\xi}\rangle \end{aligned} \tag{1.46}$$

The composition of states $\underline{A} : \mathcal{H} \mapsto \mathcal{L}_m$ and $\underline{V}\langle\cdot\rangle : \mathcal{H} \mapsto \mathcal{L}_1$[6] is

$$(\underline{A} \circ \underline{V})\langle\boldsymbol{\xi}\rangle = \underline{A}\langle\underline{V}\langle\boldsymbol{\xi}\rangle\rangle \tag{1.47}$$

In a Cartesian coordinate system, an order m state has m components that are written as $\underline{A}_{i_1,i_2,\cdots,i_m}$. The point product of two states $\underline{A} : \mathcal{H} \mapsto \mathcal{L}_{m+p}$ and $\underline{B} : \mathcal{H} \mapsto \mathcal{L}_p$ is a state in $\mathcal{L}_m$ defined by

$$(\underline{AB})_{i_1,i_2,\cdots,i_m}\langle\boldsymbol{\xi}\rangle = \underline{A}_{i_1,i_2,\cdots,i_m,j_1,j_2,\cdots,j_p}\langle\boldsymbol{\xi}\rangle\, \underline{B}_{j_1,j_2,\cdots,j_p}\langle\boldsymbol{\xi}\rangle \tag{1.48}$$

Interesting states for a first implementation are the scalar state, the dot product (between states of the same order and between states of different orders) and the

[5] There are higher order of set of states, but they are not involved in the formulation.

[6] $\mathcal{L}_1$ is also denoted as $\mathcal{V}$, the set of all vector states; the composition is possible only with a vector state

Fréchet derivative: these are briefly illustrated in the following paragraphs since they are useful for the numerical analyses.

The scalar state of $\underline{A}\langle\cdot\rangle : \mathcal{H} \mapsto \mathcal{L}_m$ is defined as

$$|\underline{A}|\langle\boldsymbol{\xi}\rangle = \sqrt{(\underline{A}\underline{A})\langle\boldsymbol{\xi}\rangle} \qquad \forall\, \boldsymbol{\xi} \in \mathcal{H} \tag{1.49}$$

If another vector state $\underline{B}\langle\cdot\rangle : \mathcal{H} \mapsto \mathcal{L}_m$ is taken into account, the dot product is

$$\underline{A} \bullet \underline{B} = \int_{\mathcal{H}} \underline{A}\langle\boldsymbol{\xi}\rangle \cdot \underline{B}\langle\boldsymbol{\xi}\rangle \, dV_{\boldsymbol{\xi}} \tag{1.50}$$

Note that here the notation introduced by Silling in [2] for state-based Peridynamics is adopted: although the differential volume $dV_{\boldsymbol{\xi}}$ is equivalent to the differential volume dV_j (i.e. is the differential volume associated to the node to which the source node is connected through bond $\boldsymbol{\xi}$), the domain of states is formally composed by bonds, so the author changed the notation consistently.

Dealing with a double state $\underline{\mathbb{K}}$, the right product of $\mathbb{K}$ and $\underline{A}$ is defined by

$$(\underline{\mathbb{K}} \bullet \underline{A})_i\langle\boldsymbol{\xi}\rangle = \int_{\mathcal{H}} \mathbb{K}_{ij}\langle\boldsymbol{\xi}, \boldsymbol{\zeta}\rangle \underline{A}_j\langle\boldsymbol{\zeta}\rangle \, dV_{\boldsymbol{\zeta}} \qquad \forall \boldsymbol{\xi} \in \mathcal{H} \tag{1.51}$$

in which Einstein's notation is adopted.

As for the Fréchet derivative, if we consider a state (function) of a state $\underline{T}(\cdot) : \mathcal{L}_n \mapsto \mathcal{L}_m$ and if a state-valued function $\nabla\underline{T} \in \mathcal{L}_{n+m}$ exists such as that for any state $\underline{A} \in \mathcal{L}_n$ and any small increment of it $\Delta\underline{A}$

$$\underline{T}(\underline{A} + \Delta\underline{A}) = \underline{T}(\underline{A}) + \nabla\underline{T}(\underline{A}) \bullet \Delta\underline{A} + o(\|\Delta\underline{A}\|) \tag{1.52}$$

then $\underline{T}$ is defferentiable and $\nabla\underline{T}$ is its Fréchet derivative.

The two most important states in the setting of Peridynamics are the force state $\underline{T}$, which is supplied by the *constitutive model* containing all the information about the material response, and the deformation state $\underline{Y}$, which connects bonds to their deformed image.

Silling et al. in 2007 [2] have derived the linear peridynamic solid model (LPS) in a general framework, then Quang et al. in 2014 [121] have developed the 2D plane stress and 2D plane strain models. In section 1.2.2 the linearized isotropic elastic models are illustrated.

1.2.1 Linear isotropic elastic models

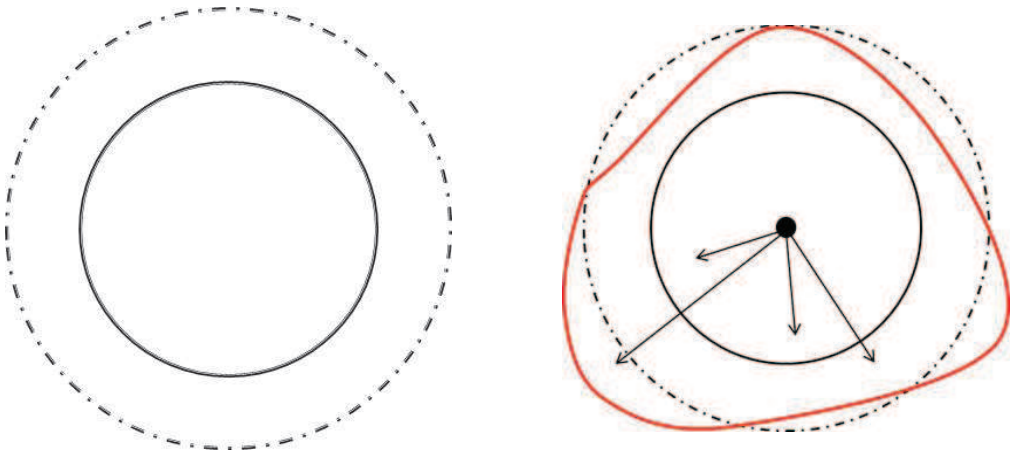
In [2] the force state $\underline{T}$ is described as the product of its modulus state $\underline{t}$ and its deformed direction state $\underline{M}(Y)$

$$\underline{T}(Y) = \underline{t}(Y) \cdot \underline{M}(Y) \tag{1.53}$$

where the modulus state is expressed as

$$\underline{t} = \frac{3K\theta}{m}\underline{\omega x} + \alpha\underline{\omega e^d} \tag{1.54}$$

in which K and α are positive constants related to material properties (the former is the bulk modulus and the latter is proportional to the shear modulus), the other quantities are states and are defined in the following paragraphs, together with basic states involved in the computation, even though not explicitly mentioned in Eq. (1.54). The force state, as well as the deformation state, is decomposed in the force determining a pure change of volume of the horizon sphere (Figure 1.12b) and in the force causing a change of shape of the horizon sphere (Figure 1.12b). These two terms are called the *co-isotropic* and the *co-deviatoric* parts of the force, rispectively corresponding to the first and the second terms of $\underline{t}$ in Eq. (1.54).



(a) (b)

Figure 1.12: Decomposition of deformation state: (a) is related to a pure change of volume of the horizon sphere, while (b) refers to a pure change of shape.

The author of [7] explains the mathematical definition of several states and shows how the Fréchet derivative is calculated for the linearized model. Here,

in the following paragraphs, some of these states are described, because they are employed extensively in the following sections. These states are:

- the *reference state* is the state which associates each pair of points to their initial relative position vector (see Figure 1.13):

$$\underline{X}\langle\boldsymbol{\xi}\rangle = \mathbf{p} - \mathbf{x} = \boldsymbol{\xi} \tag{1.55}$$

- the *scalar reference state* is the state which associates each pair of points to their bond length and it practically refers to the relative initial distance:

$$\underline{x} = |\underline{X}\langle\boldsymbol{\xi}\rangle| = |\mathbf{p} - \mathbf{x}| = |\boldsymbol{\xi}| \tag{1.56}$$

- the *deformation state* is the state which associates each pair of points to their relative current position (see Figure 1.13):

$$\underline{Y}[\mathbf{x}, t]\langle\boldsymbol{\xi}\rangle = y(\mathbf{p}, t) - y(\mathbf{x}, t) = \boldsymbol{\eta} + \boldsymbol{\xi} \tag{1.57}$$

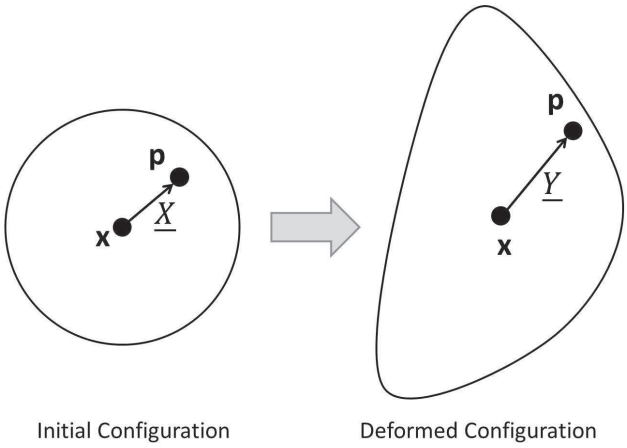


Figure 1.13: Reference state $\underline{\mathrm{X}}$ at the initial time and defomation state $\underline{\mathrm{Y}}$ at a generic time t.

- the *deformation direction state* is the state which associates each pair of points to their relative position unit vector and it can be seen as their relative current unit position vector (note that the quantities on which $\underline{Y}$ depends are omitted in the first part of the equation for simplicity):

$$\underline{M}(\underline{Y}) = \frac{\underline{Y}[\mathbf{x}, t]\langle\boldsymbol{\xi}\rangle}{|\underline{Y}[\mathbf{x}, t]\langle\boldsymbol{\xi}\rangle|} = \frac{\boldsymbol{\eta} + \boldsymbol{\xi}}{|\boldsymbol{\eta} + \boldsymbol{\xi}|} \tag{1.58}$$

- the *displacement state* is the state which associates to each pair of points their relative displacement:

$$\underline{U}[\mathbf{x},t]\langle\boldsymbol{\xi}\rangle = u(\mathbf{p},t) - u(\mathbf{x},t) = \boldsymbol{\eta} \tag{1.59}$$

- the *scalar extension state* is the state which associates each pair of points to the elongation of the bond (i.e. in the BBP version, this is equivalent to the numerator of the stretch):

$$\underline{e}(\underline{Y}) = |\underline{Y}[\mathbf{x},t]\langle\boldsymbol{\xi}\rangle| - |\underline{X}\langle\boldsymbol{\xi}\rangle| = |\boldsymbol{\eta}+\boldsymbol{\xi}| - |\boldsymbol{\xi}| \tag{1.60}$$

- the *influence function* state $\underline{\omega}$ is introduced in [2] and it is a scalar state to be used to select which bonds within a deformation state are to participate in determining the force state. It is used also to determine the different weights of bond contributions to the global behavior of the material. Its only restriction is the non-negative condition in the entire bond domain, and if it depends only on the scalar reference state, it is said to be *spherical*. For the linear ordinary peridynamic model, it is spherical, so the described material is isotropic.

- a weighted volume, identified by the letter m, takes into consideration how many bonds are in the horizon sphere (Figure 1.14), it shows if the source point is near a surface or if its horizon sphere is completely embedded within the body[7]:

$$m = (\underline{\omega x}) \bullet \underline{x} = \int_{\mathcal{H}} \omega(|\boldsymbol{\xi}|)|\boldsymbol{\xi}|^2 dV_p \tag{1.61}$$

[7]In fact, it is analytically computed assuming the horizon sphere fully embedded, but this is not true for points near the boundaries and thus it affects the behavior of material in the region, thus it is called surface effect

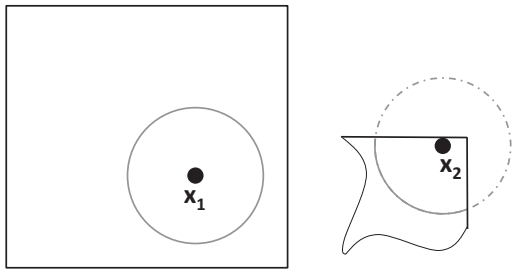


Figure 1.14: m weight for different points within the body.

- the *dilatation* θ is a scalar state indicating the deformation of the horizon neighborhood of a point; it depends on the point, on the deformation state $\underline{Y}$ of all its bonds and on the m weight. It takes into consideration how the radius of the horizon sphere changes during the deformation (Figure 1.12a):

$$\theta(\underline{Y}) = \frac{3}{m}(\underline{\omega x}) \bullet \underline{e} = \frac{3}{m}\int_{\mathcal{H}} \omega(|\boldsymbol{\xi}|)|\boldsymbol{\xi}||\boldsymbol{\xi}+\boldsymbol{\eta}|dV_p - 3 \tag{1.62}$$

- the *deviatoric extension state* is the state which associates each pair of points to the portion of elongation of the bond which is related to a change of shape of the horizon sphere:

$$\underline{e}^d(\underline{Y}) = \underline{e}(\underline{Y}) - \frac{\theta(\underline{Y})|\underline{X}\,\langle\boldsymbol{\xi}\rangle\,|}{3} = |\boldsymbol{\eta}+\boldsymbol{\xi}| - \frac{[\theta(\underline{Y})+3]\,|\boldsymbol{\xi}|}{3} \tag{1.63}$$

In [122], the BBP version is shown as a particular case of the SBP version, assuming for the $3D$ model

$$\nu = \frac{1}{4}, \qquad \underline{\omega}\langle\boldsymbol{\xi}\rangle = \frac{1}{|\boldsymbol{\xi}|} \tag{1.64}$$

In [121], Le et al. have developed the two linearized models for the plane stress and plane strain cases: the strain tensor and the stress tensor of classical mechanics are taken into consideration, so the authors can derive the peridynamic equivalent of the strain energy.

For the ***plane stress*** case, while all the stress components not in the $x-y$ plane are zero, the strain component orthogonal to the plane ε_{zz} is not null. By rearranging its expression in terms of volume dilatation, the new expression for the

energy density in classical mechanics becomes

$$\mathcal{W} = \left[\frac{K}{2} + \mu\left(\frac{\nu+1}{3(2\nu-1)}\right)^2\right]\left(\frac{dV}{V}\right)^2 + \mu\sum_{i,j=x,y}\varepsilon_{ij}^d\varepsilon_{ij}^d \tag{1.65}$$

$\left(\frac{dV}{V}\right)$ is equivalent, for small homogeneous deformation, to the peridynamic scalar-valued dilatation function θ which is computed for this case as

$$\theta = \frac{2(2\nu-1)}{\nu-1}\frac{\underline{\omega x}\bullet\underline{e}}{q} \tag{1.66}$$

in which the states are those previously defined, ν is Poisson's ratio and q is the weighted volume in two dimensions (equivalent to m in 3D cases, although its integration volume is a disk).

After some mathematical manipulation, in the two-dimensional plane stress model, the force modulus state is expressed as

$$\underline{t} = \frac{2(2\nu-1)}{\nu-1}\left(k'\theta - \frac{\alpha}{3}(\underline{\omega e}^d)\bullet\underline{x}\right)\frac{\underline{\omega x}}{q} + \alpha\underline{\omega e}^d \tag{1.67}$$

where k' and α are positive constants depending on the bulk modulus K and the shear modulus μ as following

$$k' = K + \frac{\mu}{9}\frac{(\nu+1)^2}{(2\nu-1)^2} \qquad\qquad \alpha = \frac{8\mu}{q} \tag{1.68}$$

In the ***plane strain*** case, while all the strain components not in the $x{-}y$ plane are zero, the stress component σ_{zz} orthogonal to the plane is not null. This means that

$$\varepsilon_{zz}^d = \varepsilon_{zz} - \frac{1}{3}\frac{dV}{V} = -\frac{1}{3}\frac{dV}{V} \tag{1.69}$$

Thus, the new expression for the energy density becomes

$$\mathcal{W} = \left[\frac{K}{2} + \frac{\mu}{9}\right]\left(\frac{dV}{V}\right)^2 + \mu\sum_{i,j=x,y}\varepsilon_{ij}^d\varepsilon_{ij}^d \tag{1.70}$$

where $\left(\frac{dV}{V}\right)$ is equal to θ, which in this case is computed as

$$\theta = 2\frac{\underline{\omega x}\bullet\underline{e}}{q} \tag{1.71}$$

while other parameters are previously defined. After rearrangement, in the two-dimensional plane strain model, the force modulus state is expressed as

$$\underline{t} = 2\left(k'\theta - \frac{\alpha}{3}(\underline{\omega e}^d)\bullet\underline{x}\right)\frac{\underline{\omega x}}{q} + \alpha\underline{\omega e}^d \tag{1.72}$$

where k' and α are positive constants depending on the bulk modulus K and the shear modulus μ as following

$$k' = K + \frac{\mu}{9} \qquad \alpha = \frac{8\mu}{q} \tag{1.73}$$

1.2.2 Linearized models

In classical mechanics, one of the common strategies to work with is the concept that any configurations can be obtained as a superposition of small incremental displacements. This can be carried out with a model linear with respect to displacements. The main idea is that a small displacement field can be superposed on a reference configuration in which displacements can be large with respect to the initial configuration, provided that it is a complete known and equilibrated configuration and assumed the material response to be *linear* in a small neighborhood of the configuration itself. The Fréchet derivative of the force state is needed for the model linearization, since the peridynamic models are, in general, not linear with respect to the displacements.

The linearized model is theoretically developed in [120], while the mathematical expression is derived for 3D models in [7]. In this section, the method is explained and the linearized model for 2D cases is developed.

Consider a reference configuration, with a 0 superscript, and a small displacement field $d\underline{Y} \simeq d\underline{U}$ superposed to it. At the following step the deformation is $\underline{Y}^0 + d\underline{U}$, while the strain energy is

$$\mathcal{W}(\underline{Y}^0 + d\underline{U}) = \mathcal{W}(\underline{Y}^0) + \underline{T}(\underline{Y}^0) \bullet d\underline{U} + \frac{1}{2} d\underline{U} \bullet d\underline{T}(\underline{Y}^0) \tag{1.74}$$

where $d\underline{T}(\underline{Y}^0)$ can be written as

$$d\underline{T}(\underline{Y}^0) = \underline{\mathbb{K}}(\underline{Y}^0) \bullet d\underline{U} \tag{1.75}$$

in which $\underline{\mathbb{K}}(\underline{Y}^0)$ is the double state computed as the second Fréchet derivative of the strain energy, as well as the first Fréchet derivative of force state:

$$\underline{\mathbb{K}}(\underline{Y}^0) = \underline{\nabla T}(\underline{Y}^0) = \frac{\partial}{\partial \epsilon} \underline{T}(\underline{Y}^0 + \epsilon \underline{U})|_{\epsilon=0} \tag{1.76}$$

Assuming that the body $\mathcal{B}$ in Eq. (1.2) is subjected to a body force density $\mathbf{b}^0$ resulted in an equilibrated deformation $\mathbf{y}^0$, if an additional body force density $\hat{\mathbf{b}}$ is

applied, the motion equation can be written as

$$\rho(\mathbf{x})\left(\ddot{\mathbf{y}}_{\mathbf{0}}+\ddot{\mathbf{u}}\right)(\mathbf{x},t)=\int_{\mathcal{B}}\{(\underline{T}^0[\mathbf{x}]+\underline{\mathbb{K}}[\mathbf{x}]\bullet\underline{U}[\mathbf{x}])\langle\mathbf{p}-\mathbf{x}\rangle \tag{1.77}$$
$$-(\underline{T}^0[\mathbf{p}]+\underline{\mathbb{K}}[\mathbf{p}]\bullet\underline{U}[\mathbf{p}])\langle\mathbf{x}-\mathbf{p}\rangle\}\,dV_p+\left(\mathbf{b}^{\mathbf{0}}+\hat{\mathbf{b}}\right)(\mathbf{x},t)$$

Since we assume that the previous time step t^0 is equilibrated, then

$$\int_{\mathcal{B}}\{\underline{T}^0[\mathbf{x}]\langle\mathbf{p}-\mathbf{x}\rangle-\underline{T}^0[\mathbf{p}]\langle\mathbf{x}-\mathbf{p}\rangle\}\,dV_p+\mathbf{b}^0(\mathbf{x})=\ddot{\mathbf{y}}^0=\mathbf{0} \tag{1.78}$$

and Eq. (1.44) becomes

$$\rho(\mathbf{x})\ddot{\mathbf{u}}(\mathbf{x},t)=\int_{\mathcal{B}}\{(\underline{\mathbb{K}}[\mathbf{x}]\bullet\underline{U}[\mathbf{x}])\langle\mathbf{p}-\mathbf{x}\rangle-(\underline{\mathbb{K}}[\mathbf{p}]\bullet\underline{U}[\mathbf{p}])\langle\mathbf{x}-\mathbf{p}\rangle\}\,dV_p+\hat{\mathbf{b}}(\mathbf{x},t) \tag{1.79}$$

Writing the dot products explicitly, it becomes

$$\rho(\mathbf{x})\ddot{\mathbf{u}}(\mathbf{x},t)=\int_{\mathcal{B}}\int_{\mathcal{B}}\{\underline{\mathbb{K}}[\mathbf{x}]\langle\mathbf{p}-\mathbf{x},\mathbf{q}-\mathbf{x}\rangle(\mathbf{u}(\mathbf{q},t)-\mathbf{u}(\mathbf{x},t)) \tag{1.80}$$
$$-\underline{\mathbb{K}}[\mathbf{p}]\langle\mathbf{x}-\mathbf{p},\mathbf{q}-\mathbf{p}\rangle(\mathbf{u}(\mathbf{q},t)-\mathbf{u}(\mathbf{p},t))\}\,dV_q dV_p+\hat{\mathbf{b}}(\mathbf{x},t)$$

where $\mathbf{q}\in(\mathcal{H}_x\cup\mathcal{H}_p)$. $\mathbf{q}$ and $\mathbf{x}$ can interact with each other indirectly (so even if $\delta<|\mathbf{q}-\mathbf{x}|<2\delta$, see Figure 1.15) because of the intermediate point $\mathbf{p}$ which has both of them as family nodes (it can be seen from the computation of $\underline{\mathbb{K}}[\mathbf{p}]$).

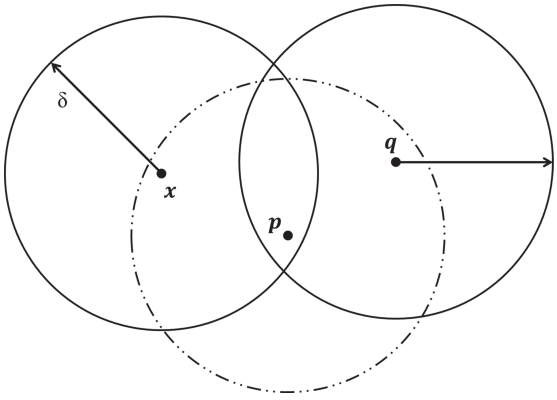


Figure 1.15: Points located at $\mathbf{x}$ and $\mathbf{q}$ are both in the family of point $\mathbf{p}$, which is why they "interact" with each other (see Eq. (1.80)).

In [7] the double state $\underline{\mathbb{K}}$ for the linear elastic model is equal to

$$\underline{\mathbb{K}}[\mathbf{x}]\langle\boldsymbol{\zeta},\boldsymbol{\xi}\rangle=\left(\frac{9K-\alpha m_{\mathbf{x}}}{m_{\mathbf{x}}^2}\right)\underline{\omega}(|\boldsymbol{\zeta}|)\,\underline{\omega}(|\boldsymbol{\xi}|)\,|\boldsymbol{\zeta}|\,|\boldsymbol{\xi}|\,\underline{M}(\underline{Y}^0)\langle\boldsymbol{\zeta}\rangle\otimes\underline{M}(\underline{Y}^0)\langle\boldsymbol{\xi}\rangle$$
$$+\alpha\underline{\omega}(|\boldsymbol{\zeta}|)\,\underline{M}(\underline{Y}^0)\langle\boldsymbol{\zeta}\rangle\otimes\underline{M}(\underline{Y}^0)\langle\boldsymbol{\xi}\rangle\,\Delta(\boldsymbol{\zeta}-\boldsymbol{\xi}) \tag{1.81}$$
$$+\underline{t}(\underline{Y}^0)\langle\boldsymbol{\zeta}\rangle\left(\frac{I-\underline{M}(\underline{Y}^0)\langle\boldsymbol{\xi}\rangle\otimes\underline{M}(\underline{Y}^0)\langle\boldsymbol{\xi}\rangle}{|\underline{Y}^0|\langle\boldsymbol{\xi}\rangle}\right)\Delta(\boldsymbol{\zeta}-\boldsymbol{\xi})$$

where $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$ are bonds of the source point $\mathbf{x}$ on which the double state depends (see Figure 1.16[8]) and $\Delta(\cdot)$ is the Dirac delta function in $\mathbb{R}^3$, K is the bulk modulus, α a constant related to material properties, the states are those previously defined, I identifies the identity 3×3 matrix and $m_{\mathbf{x}}$ identify the weight of Eq. (1.61), where the subscript $\mathbf{x}$ is added to underline the dependence on the point.

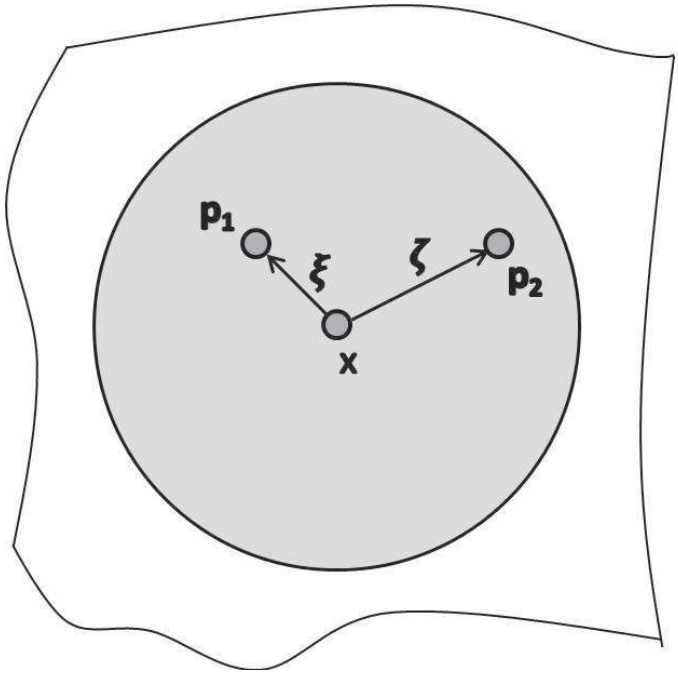


Figure 1.16: Pairs of bonds involved in the computation of the double state for the discretized system.

Note that, bearing in mind that Eq. (1.81) shows the expression of $\underline{\mathbb{K}}$, from a computational point of view, the subcases are several: for instance if $\mathbf{q} \in (\mathcal{H}_x \setminus \mathcal{H}_p)$, $\underline{\mathbb{K}}[\mathbf{p}]$ is null, if $\mathbf{q} \in (\mathcal{H}_p \setminus \mathcal{H}_x)$, $\underline{\mathbb{K}}[\mathbf{x}]$ is null; besides, there are cases in which $\mathbf{q} = \mathbf{p}$ or $\mathbf{q} = \mathbf{x}$ and the Dirac Δ-functions are not null.

In [123], the authors have developed the linearized formulation for the 2D plane stress and plane strain models.

The linearized model of a force state can be decomposed in two components (Figure 1.17): one aligned to the force state at the previous time step and one perpendicular to it, similarly to the derivation of a generic vector.

[8]Points in the neighborhood of the point located as $\mathbf{x}$ are located at a $\mathbf{p_i}$ position here. Using $\mathbf{q}$ may have been misleading, because the point located at $\mathbf{q}$ may be outside of the neighborhood of the point located at $\mathbf{x}$.

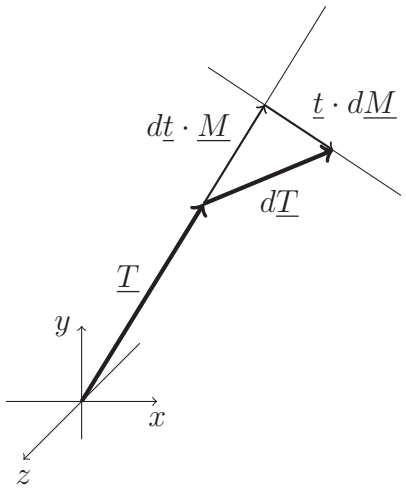


Figure 1.17: Decomposition of the linearized force state with respect to the displacement.

In fact, a force state can be considered as the product between the modulus force state $\underline{t}$ and its direction state $\underline{M}$ and its Fréchet derivative is computed by the chain rule. Consider a small infinitesimal variation in the force state due to an infinitesimal change in the deformation state $d\underline{Y}$, it is computed as

$$d\underline{T}(d\underline{Y}) = d(\underline{t}\underline{M}) = d\underline{t} \cdot \underline{M} + \underline{t} \cdot d\underline{M} \tag{1.82}$$

As shown by Silling in [120], an infinitesimal change in the deformation state $d\underline{Y} \simeq d\underline{U}$ produces an infinitesimal change of the scalar extension state such as

$$d\underline{e}\,\langle\boldsymbol{\xi}\rangle = \frac{\underline{Y}\,\langle\boldsymbol{\xi}\rangle \cdot d\underline{Y}\,\langle\boldsymbol{\xi}\rangle}{|\underline{Y}\,\langle\boldsymbol{\xi}\rangle\,|} \simeq \frac{\underline{Y}\,\langle\boldsymbol{\xi}\rangle \cdot d\underline{U}\,\langle\boldsymbol{\xi}\rangle}{|\underline{Y}\,\langle\boldsymbol{\xi}\rangle\,|} \tag{1.83}$$

since the scalar extension state is defined as

$$\underline{e}\langle\boldsymbol{\xi}\rangle = |\underline{Y}\langle\boldsymbol{\xi}\rangle| - |\underline{X}\langle\boldsymbol{\xi}\rangle| \tag{1.84}$$

In the same way, since

$$\underline{M}\langle\boldsymbol{\xi}\rangle = \frac{\underline{Y}\langle\boldsymbol{\xi}\rangle}{|\underline{Y}\langle\boldsymbol{\xi}\rangle|} \tag{1.85}$$

a small change in the deformation state induces

$$d\underline{M}\langle\boldsymbol{\xi}\rangle = \frac{I - \underline{M}\langle\boldsymbol{\xi}\rangle \otimes \underline{M}\langle\boldsymbol{\xi}\rangle}{|\underline{Y}\langle\boldsymbol{\xi}\rangle|} \tag{1.86}$$

where $\otimes$ is the dyadic product.

Another useful property is that a state $\underline{A}$ depending on the single bond can be seen as an integral of that state times the Dirac delta function in the bond domain, for example

$$\underline{A}\langle\boldsymbol{\xi}\rangle = \int_{\mathcal{H}} \underline{A}\langle\boldsymbol{\zeta}\rangle \Delta\left(\boldsymbol{\zeta}-\boldsymbol{\xi}\right) dV_{\boldsymbol{\zeta}} \tag{1.87}$$

Thus, the Fréchet derivative of the force state $\underline{T}$ can be computed as shown in the following formulas.

Consider the force state for the ***plane stress*** model: its modulus is expressed in Eq. (1.67) and applying the chain rule shown in Eq. (1.82), the infinitesimal increments are needed. For simplicity, the modulus state is divided in three contributions in the following derivations:

$$\underline{t}\langle\boldsymbol{\xi}\rangle = \underbrace{\frac{2(2\nu-1)}{\nu-1}k' d\theta \frac{\underline{\omega}\langle\boldsymbol{\xi}\rangle \underline{x}\langle\boldsymbol{\xi}\rangle}{q}}_{d\underline{t}_1\langle\boldsymbol{\xi}\rangle} - \underbrace{\frac{2(2\nu-1)}{3(\nu-1)}\frac{\alpha}{q}\left(\underline{\omega} d\underline{e}^d \bullet \underline{x}\right)\underline{\omega}\langle\boldsymbol{\xi}\rangle \underline{x}\langle\boldsymbol{\xi}\rangle}_{d\underline{t}_2\langle\boldsymbol{\xi}\rangle} + \underbrace{\alpha \underline{\omega} d\underline{e}^d}_{d\underline{t}_3\langle\boldsymbol{\xi}\rangle} \tag{1.88}$$

$$\begin{aligned}
d\underline{T}_1\langle\boldsymbol{\xi}\rangle &= d\underline{t}_1\langle\boldsymbol{\xi}\rangle \cdot \underline{M}\langle\boldsymbol{\xi} = \frac{2(2\nu-1)}{\nu-1}k' d\theta \frac{\underline{\omega}\langle\boldsymbol{\xi}\rangle \underline{x}\langle\boldsymbol{\xi}\rangle}{q}\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \frac{2(2\nu-1)}{\nu-1}k' \frac{2(2\nu-1)}{\nu-1}\int_{\mathcal{H}} \frac{\underline{\omega}\langle\boldsymbol{\zeta}\rangle \underline{x}\langle\boldsymbol{\zeta}\rangle \underline{M}\langle\boldsymbol{\zeta}\rangle d\underline{U}\langle\boldsymbol{\zeta}\rangle}{q} dV_p \frac{\underline{\omega}\langle\boldsymbol{\xi}\rangle \underline{x}\langle\boldsymbol{\xi}\rangle}{q}\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}} \left\{ \left[\frac{2(2\nu-1)}{q(\nu-1)}\right]^2 k' \underline{\omega}\langle\boldsymbol{\xi}\rangle \underline{\omega}\langle\boldsymbol{\zeta}\rangle |\boldsymbol{\xi}||\boldsymbol{\zeta}| \underline{M}\langle\boldsymbol{\xi}\rangle \otimes \underline{M}\langle\boldsymbol{\zeta}\rangle \right\} \cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}} \underline{\mathbb{K}}_1\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle \cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \underline{\mathbb{K}}_1\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle \bullet d\underline{U}\langle\boldsymbol{\zeta}\rangle
\end{aligned} \tag{1.89}$$

$$
\begin{aligned}
d\underline{T}_2\langle\boldsymbol{\xi}\rangle &= d\underline{t}_2\langle\boldsymbol{\xi}\rangle \cdot \underline{M}\langle\boldsymbol{\xi} = -\frac{2(2\nu-1)}{3(\nu-1)}\frac{\alpha}{q}\left(\underline{\omega}d\underline{e}^d \bullet \underline{x}\right)\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= -\frac{2(2\nu-1)}{3(\nu-1)}\frac{\alpha}{q}\int_{\mathcal{H}}\underline{\omega}\langle\boldsymbol{\zeta}\rangle\underline{x}\langle\boldsymbol{\zeta}\rangle\left(\underline{M}\langle\boldsymbol{\zeta}\rangle d\underline{U}\langle\boldsymbol{\zeta}\rangle - \frac{d\theta\underline{x}\langle\boldsymbol{\zeta}\rangle}{3}\right)dV_p\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\left\{-\frac{2(2\nu-1)}{3(\nu-1)}\frac{\alpha}{q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \frac{2(2\nu-1)}{9(\nu-1)}\frac{\alpha}{q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\int_{\mathcal{H}}\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\zeta}|^2 dV_p\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p + \frac{2(2\nu-1)}{9(\nu-1)}\frac{\alpha}{q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\, q\,\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{\frac{4(2\nu-1)^2}{9(\nu-1)^2}\frac{\alpha}{q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\left\{\left[-\frac{2(2\nu-1)}{3(\nu-1)}+\frac{4(2\nu-1)^2}{9(\nu-1)^2}\right]\frac{\alpha}{q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \underline{\mathbb{K}}_{2}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\bullet d\underline{U}\langle\boldsymbol{\zeta}\rangle
\end{aligned}
\tag{1.90}
$$

$$
\begin{aligned}
d\underline{T}_3\langle\boldsymbol{\xi}\rangle &= d\underline{t}_3\langle\boldsymbol{\xi}\rangle\cdot\underline{M}\langle\boldsymbol{\xi} = \alpha\underline{\omega}d\underline{e}^d\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \alpha\underline{\omega}\langle\boldsymbol{\xi}\rangle\left(\underline{M}\langle\boldsymbol{\xi}\rangle d\underline{U}\langle\boldsymbol{\xi}\rangle - \frac{d\theta\underline{x}\langle\boldsymbol{\xi}\rangle}{3}\right)\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\left\{\alpha\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\xi}\rangle\Delta(\boldsymbol{\xi}-\boldsymbol{\zeta})\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p - \frac{\alpha}{3}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\,\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{-\frac{\alpha}{q}\frac{2(2\nu-1)}{3(\nu-1)}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle\underline{x}\langle\boldsymbol{\zeta}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{-\frac{\alpha}{q}\frac{2(2\nu-1)}{3(\nu-1)}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \underline{\mathbb{K}}_{3}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\bullet d\underline{U}\langle\boldsymbol{\zeta}\rangle
\end{aligned}
\tag{1.91}
$$

The last contribution comes from the $d\underline{T}_4 = \underline{t} \cdot d\underline{M}$ derivation, which is the product of the modulus state in Eq. (1.67) and the derivation of the direction state computed in Eq. (1.86).

Rearranging all the terms, the Fréchet derivative of the force state is a 2×2 tensor[9] given by the following expression

$$\begin{aligned}\underline{\mathbb{K}}\langle \boldsymbol{\xi}, \boldsymbol{\zeta} \rangle = & \frac{\gamma}{q^2} \underline{\omega}(|\boldsymbol{\xi}|)\underline{\omega}(|\boldsymbol{\zeta}|)|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle \boldsymbol{\xi} \rangle \otimes \underline{M}\langle \boldsymbol{\zeta} \rangle \\ & + \frac{4E}{q\,(1+\nu)} \underline{\omega}(|\boldsymbol{\xi}|)\underline{M}\langle \boldsymbol{\xi} \rangle \otimes \underline{M}\langle \boldsymbol{\xi} \rangle \Delta\,(\boldsymbol{\xi} - \boldsymbol{\zeta}) \\ & + \underline{t}(\underline{Y})\langle \boldsymbol{\xi} \rangle \left(\frac{I - \underline{M}(\underline{Y})\langle \boldsymbol{\xi} \rangle \otimes \underline{M}(\underline{Y})\langle \boldsymbol{\xi} \rangle}{|\underline{Y}|\langle \boldsymbol{\xi} \rangle} \right) \Delta(\boldsymbol{\zeta} - \boldsymbol{\xi})\end{aligned} \tag{1.92}$$

where

$$\gamma = \frac{2(3\nu - 1)E}{1 - \nu^2} \tag{1.93}$$

in which E is Young's modulus and ν Poisson's ratio.

The same approach can employed fot the ***plane strain*** derivations, where the mathematical contributions are taken from Eq. (1.72)

$$\begin{aligned} d\underline{T}_1\langle \boldsymbol{\xi} \rangle &= d\underline{t}_1\langle \boldsymbol{\xi} \rangle \cdot \underline{M}\langle \boldsymbol{\xi} = 2k' d\theta \frac{\underline{\omega}\langle \boldsymbol{\xi} \rangle \underline{x}\langle \boldsymbol{\xi} \rangle}{q} \underline{M}\langle \boldsymbol{\xi} \rangle \\ &= 4k' \int_{\mathcal{H}} \frac{\underline{\omega}\langle \boldsymbol{\zeta} \rangle \underline{x}\langle \boldsymbol{\zeta} \rangle \underline{M}\langle \boldsymbol{\zeta} \rangle d\underline{U}\langle \boldsymbol{\zeta} \rangle}{q} dV_p \frac{\underline{\omega}\langle \boldsymbol{\xi} \rangle \underline{x}\langle \boldsymbol{\xi} \rangle}{q} \underline{M}\langle \boldsymbol{\xi} \rangle \\ &= \int_{\mathcal{H}} \left\{ \frac{4}{q^2} k' \underline{\omega}\langle \boldsymbol{\xi} \rangle \underline{\omega}\langle \boldsymbol{\zeta} \rangle |\boldsymbol{\xi}||\boldsymbol{\zeta}| \underline{M}\langle \boldsymbol{\xi} \rangle \otimes \underline{M}\langle \boldsymbol{\zeta} \rangle \right\} \cdot d\underline{U}\langle \boldsymbol{\zeta} \rangle dV_p \\ &= \int_{\mathcal{H}} \underline{\mathbb{K}}_1\langle \boldsymbol{\xi}, \boldsymbol{\zeta} \rangle \cdot d\underline{U}\langle \boldsymbol{\zeta} \rangle dV_p \\ &= \underline{\mathbb{K}}_1\langle \boldsymbol{\xi}, \boldsymbol{\zeta} \rangle \bullet d\underline{U}\langle \boldsymbol{\zeta} \rangle \end{aligned} \tag{1.94}$$

[9] The linearization is relative to 2D models, thus the force state is a 2×2 tensor.

$$
\begin{aligned}
d\underline{T}_2\langle\boldsymbol{\xi}\rangle &= d\underline{t}_2\langle\boldsymbol{\xi}\rangle \cdot \underline{M}\langle\boldsymbol{\xi} = -\frac{2\alpha}{3q}\left(\underline{\omega}d\underline{e}^d \bullet \underline{x}\right)\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= -\frac{2\alpha}{3q}\int_{\mathcal{H}}\underline{\omega}\langle\boldsymbol{\zeta}\rangle\underline{x}\langle\boldsymbol{\zeta}\rangle\left(\underline{M}\langle\boldsymbol{\zeta}\rangle d\underline{U}\langle\boldsymbol{\zeta}\rangle - \frac{d\theta\underline{x}\langle\boldsymbol{\zeta}\rangle}{3}\right)dV_p\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle \quad (1.95)\\
&= \int_{\mathcal{H}}\left\{-\frac{2\alpha}{3q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \frac{2\alpha}{9q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\int_{\mathcal{H}}\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\zeta}|^2 dV_p\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p + \frac{2\alpha}{9q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\, q\, \underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{\frac{4\alpha}{9q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\left\{-\frac{2\alpha}{9q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{2}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \underline{\mathbb{K}}_{2}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\bullet d\underline{U}\langle\boldsymbol{\zeta}\rangle
\end{aligned}
$$

$$
\begin{aligned}
d\underline{T}_3\langle\boldsymbol{\xi}\rangle &= d\underline{t}_3\langle\boldsymbol{\xi}\rangle\cdot\underline{M}\langle\boldsymbol{\xi} = \alpha\underline{\omega}d\underline{e}^d\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \alpha\underline{\omega}\langle\boldsymbol{\xi}\rangle\left(\underline{M}\langle\boldsymbol{\xi}\rangle d\underline{U}\langle\boldsymbol{\xi}\rangle - \frac{d\theta\underline{x}\langle\boldsymbol{\xi}\rangle}{3}\right)\underline{M}\langle\boldsymbol{\xi}\rangle \quad (1.96)\\
&= \int_{\mathcal{H}}\{\alpha\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\xi}\rangle\Delta(\boldsymbol{\xi}-\boldsymbol{\zeta})\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p - \frac{\alpha}{3}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\, d\theta\,\underline{M}\langle\boldsymbol{\xi}\rangle \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{-\frac{2\alpha}{3q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{x}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle\underline{x}\langle\boldsymbol{\zeta}\rangle\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3a}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&+ \int_{\mathcal{H}}\left\{-\frac{2\alpha}{3q}\underline{\omega}\langle\boldsymbol{\xi}\rangle\underline{\omega}\langle\boldsymbol{\zeta}\rangle|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle\otimes\underline{M}\langle\boldsymbol{\zeta}\rangle\right\}\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \int_{\mathcal{H}}\underline{\mathbb{K}}_{3}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\cdot d\underline{U}\langle\boldsymbol{\zeta}\rangle dV_p \\
&= \underline{\mathbb{K}}_{3}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle\bullet d\underline{U}\langle\boldsymbol{\zeta}\rangle
\end{aligned}
$$

The last contribution is due to the $d\underline{T}_4 = \underline{t}\cdot d\underline{M}$ so the first Fréchet derivative of the force state is

$$\underline{\underline{\mathbb{K}}}\langle\boldsymbol{\xi},\boldsymbol{\zeta}\rangle = \frac{\gamma}{q^2}\underline{\omega}(|\boldsymbol{\xi}|)\underline{\omega}(|\boldsymbol{\zeta}|)|\boldsymbol{\xi}||\boldsymbol{\zeta}|\underline{M}\langle\boldsymbol{\xi}\rangle \otimes \underline{M}\langle\boldsymbol{\zeta}\rangle \tag{1.97}$$

$$+ \frac{4E}{q\,(1+\nu)}\underline{\omega}(|\boldsymbol{\xi}|)\underline{M}\langle\boldsymbol{\xi}\rangle \otimes \underline{M}\langle\boldsymbol{\xi}\rangle\Delta\,(\boldsymbol{\xi}-\boldsymbol{\zeta}) \tag{1.98}$$

$$+ \underline{t}(\underline{Y})\langle\boldsymbol{\xi}\rangle\left(\frac{I - \underline{M}(\underline{Y})\langle\boldsymbol{\xi}\rangle \otimes \underline{M}(\underline{Y})\langle\boldsymbol{\xi}\rangle}{|\underline{Y}|\langle\boldsymbol{\xi}\rangle}\right)\Delta(\boldsymbol{\zeta}-\boldsymbol{\xi})$$

where

$$\gamma = \frac{2\,(4\nu-1)\,E}{(1-2\nu)\,(1+\nu)} \tag{1.99}$$

in which E is Young's modulus and ν Poisson's ratio. It is interesting to notice, since it is expected, that the γ constant is equal to zero for a Poisson's ratio equal to $\frac{1}{3}$ for the plane stress model and to $\frac{1}{4}$ for the plane strain one. To summerize, the modulus states have the same mathematical components, with different constants, related to different properties in the two cases (Table 1.1).

Term	**Plane Stress**	**Plane Strain**
$d\underline{T}_1\langle\boldsymbol{\xi}\rangle$	$\left[\frac{2(2\nu-1)}{q(\nu-1)}\right]^2 k'$	$\frac{4}{q^2}k'$
$d\underline{T}_2\langle\boldsymbol{\xi}\rangle$	$\left[\frac{4(2\nu-1)^2}{9(\nu-1)^2} - \frac{2(2\nu-1)}{3(\nu-1)}\right]\frac{\alpha}{q}$	$-\frac{2}{9}\frac{\alpha}{q}$
$d\underline{T}_3\langle\boldsymbol{\xi}\rangle$	$\alpha;\ -\frac{2(2\nu-1)}{3(\nu-1)}\frac{\alpha}{q}$	$\alpha;\ -\frac{2}{3}\frac{\alpha}{q}$
$d\underline{T}_4\langle\boldsymbol{\xi}\rangle$	1	1

Table 1.1: Material constants of the different terms of the force modulus state that are linearized separately.

The double tensor is employed in the computation of the contributions to the components of the stiffness matrix, as shown in section 3.2.

1.2.3 Failure Criteria in SBP

In SBP, the force contributions are related not only to the isotropic part of the deformation (i.e. the stretch of the single bond), but also to the the deviatoric

part of the deformation (i.e. the change in the relative position between a bond and those contained in the same set) [2]. For this reason, several failure criteria have been employed in literature, such as equivalent strain criterion [6] or implementing the standard Johnson-Cook damage model where the failure occurs when the damage parameter reaches the unit value [98]. In this thesis, criteria based on the maximum stretch or the maximum energy are implemented.

The energy in the single bond is stored both in terms of elongation and in terms of change of relative position with respect to the surrounding bonds, but employing a maximum stretch criterion implies that the energy is considered stored merely during the elongation process. This approximation is due to the dependence of the bond total stored energy on the stepwise bond positions.

So the obvious question is: which critical value of which quantity should be employed? If a failure stretch is known *a priori*, then that can be used for the simulation, but this case is not so common. In this thesis three different critical values are computed:

- the critical value s_0 is the limit stretch computed in the BBP variant. To take into account this value means, as previously mentioned, to neglect the contribution of the deviatoric strain tensor to the total stored elastic energy in the sample. In this case, the maximum storable energy is linearly proportional to the initial bond length, in fact $s_0 \propto \dfrac{\omega_0}{\xi}$, in particular s_0 is expressed by Eqs. (1.40) and (1.41).

- a limit energy ω_0 storable in the bond, independently of other properties of the bond (i.e. ω_0 is constant), so it can be computed for a 2D structure as

$$G_0 = \int_{\mathcal{H}_i} \omega_0 (\boldsymbol{\xi})\, dV_j = \frac{2}{3}\omega_0 (\boldsymbol{\xi})\, \delta^3 t_h \mapsto \omega_0 (\boldsymbol{\xi}) = \frac{3G_0}{2\delta^3 t_h} \tag{1.100}$$

  where t_h is the thickness of the structure, where at every time t, if $\mathbf{x}$ and $\mathbf{p}$ are the points connected by the bond $\boldsymbol{\xi}$

$$\omega_t (\boldsymbol{\xi}) = \omega_{t_0} (\boldsymbol{\xi}) + \int_{t_0}^{t} \{(\underline{T}\,[\mathbf{p}, t] - \underline{T}\,[\mathbf{x}, t]) \cdot \underline{U}\}\, dt \tag{1.101}$$

- a limit energy ω_0 storable in the bond computed as in the previous criterion, but the amount of energy stored in the bond is considered to be equal to an

average energy between the two nodes connected by it

$$\omega_t(\boldsymbol{\xi}) = \frac{\mathcal{W}[\mathbf{p}, t] + \mathcal{W}[\mathbf{x}, t]}{2} \tag{1.102}$$

where $\mathcal{W}$ is the energy density associated to each node, computed as expressed in Eq. (1.70) for plane strain analyses and in Eq. (1.65) for plane stress ones.

Chapter 2

Numerical aspects

2.1 Spatial integration

The body $\mathcal{B}$ is discretized into a grid of nodes (Figure 2.1) associated to small finite volumes V_j in the reference configuration. The problems for the proper definition of the geometric boundary of the body are the same of FEM: the approximation of real boundaries by the the geometrical representation defined by the finite volumes improves as the finite volumes decreases and, thus, the number of nodes increases.

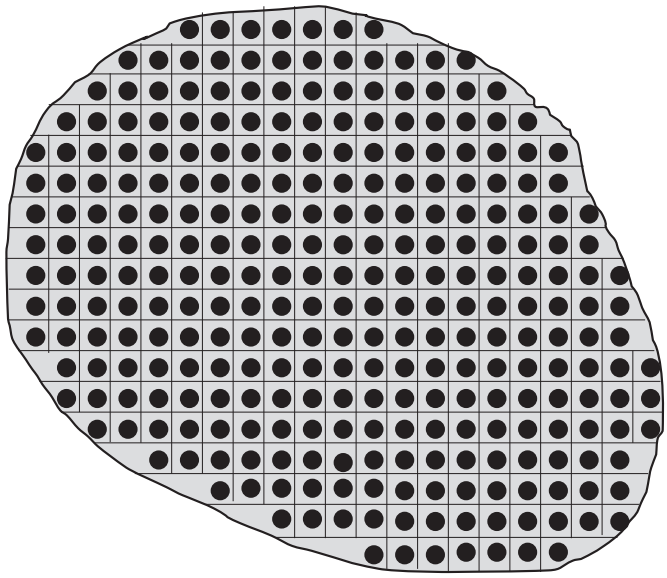


Figure 2.1: Uniform discretization of the body $\mathcal{B}$ into nodes.

In all the examples, the meshfree method proposed by Silling and Askari in [68] is adopted: no elements or other geometrical connections between nodes are

present. In addition, a 1-point Gaussian quadrature rule is adopted, as it leads to a simpler and unambiguous way of defining the crack path (Figure 2.2). More complex algorithms can be used to ensure faster and more robust convergence of peridynamic models [9, 124]. However, Seleson shows that this is quite accurate. As a consequence, the non-monotonic behavior appearing in the results may be partly due to the approximate partial volume integration algorithm and partly to the nonlocal nature of the theory.

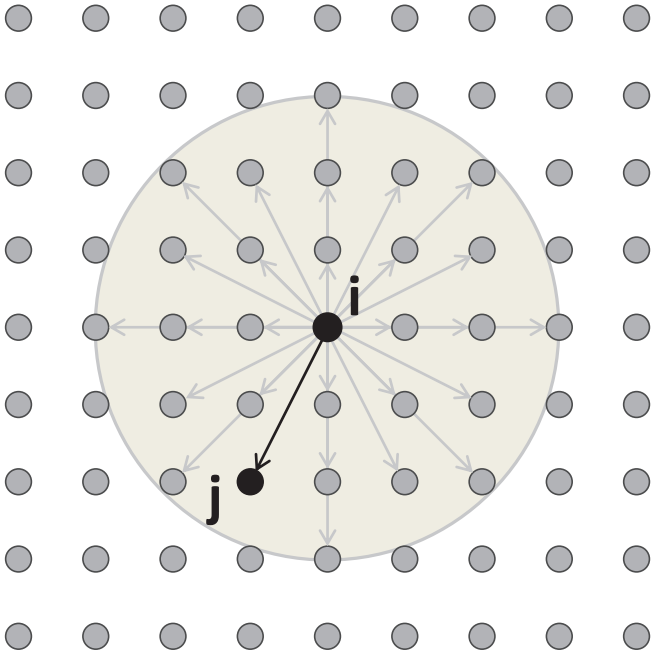


Figure 2.2: Uniform discretization of the body $\mathcal{B}$ into nodes.

In the discretized form of the equation of motion in BBP (Eq. (1.2)), the integral is replaced by a finite sum:

$$\rho_i \ddot{\mathbf{y}}_i^n = \sum_j \mathbf{f}_{ij}^n \left(\mathbf{u}_j^n - \mathbf{u}_i^n, \mathbf{x}_j - \mathbf{u}_i \right) f_{V_j} V_j + \mathbf{b}_i^n \tag{2.1}$$

where ρ_i is the density of the node, $\mathbf{f}_{ij}^n$ the pairwise force function exerted by j on i, n the time step, the subscript i or j the node ID numbers (i is the source node while j the family node), f_{V_j} the integration weight associated to point j, V_j the volume of node located at $\mathbf{x}_j$, and $\ddot{\mathbf{y}}_i^n$ is a concise expression for $\ddot{\mathbf{y}}\left(\mathbf{x}_i, t^n\right)$. In all the simulated cases, a constant grid spacing in all directions $\Delta x = \Delta y = \Delta z$ is assumed, so that the volume associated to each node is $V = \Delta x \cdot \Delta y \cdot \Delta z = \Delta x^3$.

The algorithm adopted for the computation of weight f_{V_j} is the one adopted by EMU[1] [125] for the BBP code and the *PA-HHB algorithm* in [9] for the SBP code.

The body can be visualized as a structure made by linear springs[2], where the maximum initial length of the springs is equal to δ.

The classical linear spring behavior may be expressed in matrix form in the local reference system as

$$\begin{Bmatrix} f_1^{loc} \\ f_2^{loc} \end{Bmatrix} = k_{spring} \cdot \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1^{loc} \\ u_2^{loc} \end{Bmatrix} \qquad or \qquad \left\{ f^{loc} \right\} = \left[k^{loc} \right] \left\{ u^{loc} \right\} \tag{2.2}$$

In the local reference system, the nodes move in the x direction (see Figure 2.3).

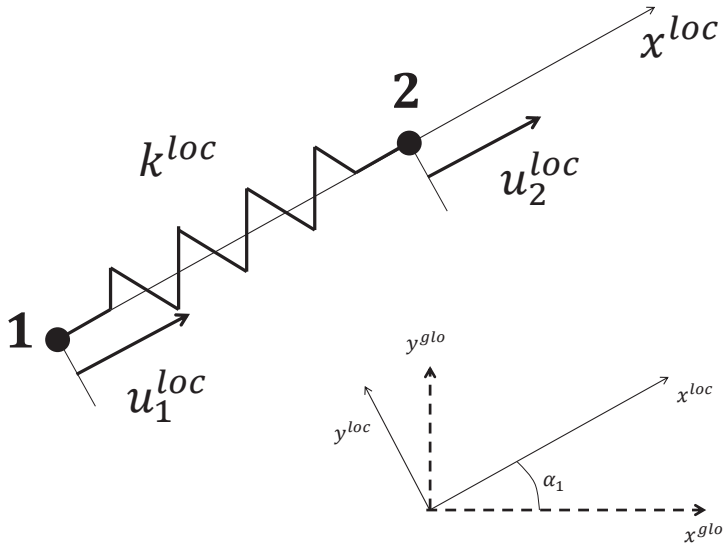


Figure 2.3: Representation of a spring with its two degrees of freedom associated to its two nodes in the local reference system and rotation angle from the local to the global reference system in a 2D case.

The local stiffness matrix of each spring can be expressed in the global coordinate system, by an appropriate rotation matrix (i.e. in 2D, one angle α_1 is enough to define the rotation, in 3D two angles are required), and add to a global stiffness matrix composed by a superposition of the individual element stiffness matrices for the same element nodal displacement.

[1] the BBP code developed at Sandia National Laboratories by Silling.

[2] This is valid for the PMB model, the springs can be nonlinear if other models are adopted.

In particular, Eq. (2.1) has to be linearized with respect to the displacement, so, employing the linearized PMB model, at every time step n, it can be written as

$$\rho_i \left(\ddot{\mathbf{u}}_i^n + \ddot{\mathbf{y}}_i^{n-1} \right) = \sum_j \mathbf{C}_{ij}^{n-1} \boldsymbol{\eta}_{ij}^n f_{V_j} V_j + \mathbf{b}_i^n + I_i^{n-1} \tag{2.3}$$

where $\ddot{\mathbf{u}}_i^n$ is the incremental acceleration at time step n, $\mathbf{b}_i^n$ the prescribed force density at n, $\mathbf{C}_{ij}^{n-1}$ the micromodulus of Eq. (1.28) at time step $n-1$, $\boldsymbol{\eta}_{ij}^n$ the relative displacement vector at n, and I_i^{n-1} is the sum of internal forces at the previous time step[3].

The stiffness matrix in the global reference system can be easily computed from this formulation of the force vector in Eq. (1.37): a first contribution may be interpreted, after an appropriate rearrangement, as an entry of the stiffness matrix and the relative displacement vector as an entry of the displacement vector.

$$\mathbf{C}\left(\boldsymbol{\xi}\right)\boldsymbol{\eta} f_V V = \underbrace{\frac{c\left(\boldsymbol{\xi}\right)\mu\left(\boldsymbol{\xi}\right)}{|\boldsymbol{\xi}|} \begin{bmatrix} \frac{\xi_x^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_x \xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_x \xi_z}{|\boldsymbol{\xi}|^2} \\ \frac{\xi_y \xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_y^2}{|\boldsymbol{\xi}|^2} & \frac{\xi_y \xi_z}{|\boldsymbol{\xi}|^2} \\ \frac{\xi_z \xi_x}{|\boldsymbol{\xi}|^2} & \frac{\xi_z \xi_y}{|\boldsymbol{\xi}|^2} & \frac{\xi_z^2}{|\boldsymbol{\xi}|^2} \end{bmatrix} f_V V}_{[\mathrm{K}]_{bond}^{glo}} \underbrace{\begin{bmatrix} \eta_x \\ \eta_y \\ \eta_z \end{bmatrix}}_{\{\mathrm{U}\}_{bond}^{glo}} \tag{2.4}$$

where the subscripts and superscripts are omitted for simplicity. In particular, $\boldsymbol{\eta}$ is the current relative displacement vector $\mathbf{u}_j^n - \mathbf{u}_i^n$, so $[K]_{bond}^{glo}$ for each bond connecting node j to node i has to be added with the proper sign:

$$\begin{array}{c} \\ 3j-2 \\ 3j-1 \\ 3j \\ 3i-2 \\ 3i-1 \\ 3i \end{array} \begin{array}{c} \begin{array}{ccc} (3j-2 & 3j-1 & 3j) \end{array} \\ \left[\begin{array}{ccc} & +[K]^{glo} & \\ & & \\ & & \\ & -[K]^{glo} & \\ & & \\ & & \end{array}\right. \end{array} \begin{array}{c} \begin{array}{ccc} (3i-2 & 3i-1 & 3i) \end{array} \\ \left.\begin{array}{ccc} & -[K]^{glo} & \\ & & \\ & & \\ & +[K]^{glo} & \\ & & \\ & & \end{array}\right] \end{array} \tag{2.5}$$

The dynamic solver does not involved the computation of the stiffness matrix, while the static solver algorithm does involve it and the method employed to solve

[3]It does not depend on the displacement vector at n, thus it can be separated from the sum of internal forces at the current time step n

Eq. (3.2) is the *biconjugate gradient stabilized*, simply abbreviated as *BiCGSTAB*. It is an iterative method developed by Van der Vorst [126] in order to find numerical solution of non-symmetrical linear systems and it does not require for the system matrix to be invertible. If the system is nonlinear, a Newton-Raphson approach is adopted for which the tangent stiffness matrix is computed. In section 3.2 the static solver and the tangent stiffness matrix are illustrated in detail.

As for what concerns SBP, Eq. (1.44) may be expressed with Riemann sums over the total number of nodes N of the discretized structure:

$$\rho_{n_x}\ddot{\mathbf{y}}^n_{n_x} = \sum_{n_p=1}^{N} \left\{ \underline{T}^n_{n_x}\langle \mathbf{p}-\mathbf{x}\rangle - \underline{T}^n_{n_p}\langle \mathbf{x}-\mathbf{p}\rangle \right\} f_{V_p} V_p + \mathbf{b}^n_{n_x} \tag{2.6}$$

where ρ_{n_x} is the density of the node n_x, $\ddot{\mathbf{y}}^n_{n_x}$ is the acceleration of node n_x at time t^n, $\underline{T}^n_{n_x}\langle \mathbf{p}-\mathbf{x}\rangle$ (i.e. concise expression for $\underline{T}[\mathbf{x}, t^n]\langle \mathbf{p}-\mathbf{x}\rangle$) is the force exerted by n_x on n_p at time t^n, $\underline{T}^n_{n_p}\langle \mathbf{x}-\mathbf{p}\rangle$ is the force exerted by n_p on n_x at time t^n, n is the time step number, f_{V_p} is the integration volume weight associated to n_p, V_p is the volume associated to node n_p, and $\mathbf{b}^n_{n_x}$ the force density at time t^n.

As for the linearized version, the integral at node $\mathbf{x}$ in Eq. (1.77) becomes a finite sum:

$$\begin{aligned} \rho_{n_x}\ddot{\mathbf{y}}^n_{n_x} &= \rho_{n_x}\left(\ddot{\mathbf{y}}^{n-1}_{n_x} + \ddot{\mathbf{u}}^n_{n_x}\right) \\ &= \sum_{n_p=1}^{N} \{ (\underline{T}^{n-1}_{n_x} + \underline{\mathbb{K}}^{n-1}_{n_x} \bullet \underline{U}^n_{n_x})\langle \mathbf{p}-\mathbf{x}\rangle \\ &\quad - (\underline{T}^{n-1}_{n_p} + \underline{\mathbb{K}}^{n-1}_{n_p} \bullet \underline{U}^n_{n_p})\langle \mathbf{x}-\mathbf{p}\rangle \} f_{V_p} V_p + \mathbf{b}^n_{n_x} \end{aligned} \tag{2.7}$$

where $\ddot{\mathbf{y}}^n_{n_x}$ is the acceleration of node n_x at time t^n, $\ddot{\mathbf{u}}^n_{n_x}$ is the change in the acceleration, $\underline{\mathbb{K}}^{n-1}_{n_x}$ and $\underline{\mathbb{K}}^{n-1}_{n_p}$ the double state evaluated at n_x and n_p at the time step t^{n-1}, $\underline{U}^n_{n_x}$ and $\underline{U}^n_{n_p}$ the displacement state evaluated at n_x and n_p at the time step t^n, and $\mathbf{b}^n_{n_x}$ the body force density at t^n.

In this thesis, this formulation is adopted for the static solver where Eq. (2.7) can be simplified. Therefore, assuming that the body is equilibrated at t^{n-1}, Eq. (1.79) becomes

$$\begin{aligned} \rho_{n_x}\left(\ddot{\mathbf{y}}^{n-1}_{n_x} + \ddot{\mathbf{u}}^n_{n_x}\right) = \sum_{n_p=1}^{N}\sum_{n_q=1}^{N} \{ &\underline{\mathbb{K}}^{n-1}_{n_x}\langle \mathbf{p}-\mathbf{x}, \mathbf{q}-\mathbf{x}\rangle \cdot \underline{U}^n_{n_x}\langle \mathbf{q}-\mathbf{x}\rangle \\ &- \underline{\mathbb{K}}^{n-1}_{n_p}\langle \mathbf{x}-\mathbf{p}, \mathbf{q}-\mathbf{p}\rangle \cdot \underline{U}^n_{n_p}\langle \mathbf{q}-\mathbf{p}\rangle \} f_{V_p} V_p f_{V_q} V_q + \mathbf{b}^n_{n_x} + I^{n-1}_{n_x} \end{aligned} \tag{2.8}$$

where $I_{n_x}^{n-1}$ is the sum of internal forces at the previous time step, similarly to what was done in the BBP formulation. Most of the contributions are actually zero, the double state is not zero only within the horizon spheres of n_x or n_p, which means that the nodes considered for each node n_x in the algorithm are located at a distance of less than 2δ from n_x (the maximum distance of a family node n_p is δ and δ is the maximum distance of a family node n_q from n_p, see Figure 1.15).

Notice that, dimensionally, $\underline{\underline{\mathbb{K}}}$ is a force per length to the power of 10 ($[\frac{N}{m^{10}}]$), since it has to multiply by a displacement once and by a volume three times to have the dimension of the force $\rho(\mathbf{x})\,\ddot{\mathbf{u}}(\mathbf{x},t^n)\,V_x$; but this numerical integration can be tricky in the discretized form, since there's a Dirac Δ-function in the formula. In fact, from the definition of Eq. (1.87) the contributions which contain a Dirac Δ-function have to be multiplied by a volume just twice (not V_{n_q}) for integrating in a correct way.

Since all the example presented in this thesis are 2D, the relative vector equation has two components[4]. Thus, suppressing the explicit bond dependence formality, it can be written for each node in matrix form as:

$$\sum_{n_p=1}^{N}\sum_{n_q=1}^{N}\left\{\begin{bmatrix}\underline{\underline{\mathbb{K}}}^{n-1}_{n_x,xx} & \underline{\underline{\mathbb{K}}}^{n-1}_{n_x,xy}\\ \underline{\underline{\mathbb{K}}}^{n-1}_{n_x,yx} & \underline{\underline{\mathbb{K}}}^{n-1}_{n_x,yy}\end{bmatrix}\begin{Bmatrix}\underline{U}^{n}_{n_x,x}\\ \underline{U}^{n}_{n_x,y}\end{Bmatrix}-\begin{bmatrix}\underline{\underline{\mathbb{K}}}^{n-1}_{n_p,xx} & \underline{\underline{\mathbb{K}}}^{n-1}_{n_p,xy}\\ \underline{\underline{\mathbb{K}}}^{n-1}_{n_p,yx} & \underline{\underline{\mathbb{K}}}^{n-1}_{n_p,yy}\end{bmatrix}\begin{Bmatrix}\underline{U}^{n}_{n_p,x}\\ \underline{U}^{n}_{n_p,x}\end{Bmatrix}\right\} f_{V_p}V_p f_{V_q}V_q + \begin{Bmatrix}b^{n}_{n_x,x}\\ b^{n}_{n_x,y}\end{Bmatrix}+\begin{Bmatrix}I^{n-1}_{n_x,x}\\ I^{n-1}_{n_x,y}\end{Bmatrix}=\rho_{n_x}\begin{Bmatrix}\ddot{y}^{n}_{n_x,x}\\ \ddot{y}^{n}_{n_x,y}\end{Bmatrix} \tag{2.9}$$

As the method aforementioned for BBP, rows $2n_x-1$ and $2n_x$ are associated with the horizontal and vertical degrees of freedom of each node n_x to write all the terms of the equations in matrix form: all the $\underline{\underline{\mathbb{K}}}$ terms can be collected in a $2N\times 2N$ matrix (the *stiffness matrix*), all the $\underline{U}$ contributions in a displacement vector ($2N\times 1$) and all the $\mathbf{b}(\mathbf{x})+I(\mathbf{x})$ components in the known force vector ($2N\times 1$). The detailed algorithm of the stiffness matrix computation for SBP is illustrated in section 3.3.

[4]In a general 3D case, there are 3 components, the equation becomes more complex, although the extension can be done intuitively.

2.2 Time integration

For what concerns time integration for the dynamic solver, Silling and Askari adopted an explicit central difference formula for the acceleration

$$\ddot{\mathbf{u}}_i^n = \frac{\mathbf{u}_i^{n+1} - 2\mathbf{u}_i^n + \mathbf{u}_i^{n-1}}{\Delta t^2} \tag{2.10}$$

where Δt is the time step size.

The time integration scheme adopted for all the dynamic simulations in this thesis is an explicit algorithm with a velocity-Verlet time integration scheme for consistency with most existing publications about Peridynamics: in fact, this scheme is commonly used in molecular dynamics for particle trajectories [127] and Peridynamics is strongly related to it (molecular dynamics is the discrete version of Peridynamics and its length scale δ is infinite), so it has spontaneously inherited it, at least after the implementation of Peridynamics within LAMMPS [62, 61]. Velocity-Verlet has a good numerical stability and is similar to the leapfrog method [128], because velocity is computed not only at the step size but also at half step size. Time steps for stable solutions are suggested in several papers, such as [108]. The algorithm is the following (see Figure 2.4), provided the initial conditions for each node $(\mathbf{u}_i^0, \dot{\mathbf{u}}_i^0, \ddot{\mathbf{u}}_i^0)$, the time step Δt and the time sampling $t^{n+1} = t^n + \Delta t$,

1. compute $\dot{\mathbf{u}}_i^{n+\frac{1}{2}} = \dot{\mathbf{u}}_i^n + \frac{1}{2}\ddot{\mathbf{u}}_i^n \Delta t$
2. compute $\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \dot{\mathbf{u}}_i^{n+\frac{1}{2}} \Delta t$
3. compute $\ddot{\mathbf{u}}_i^{n+1}$ from the total potential of the system (internal and external forces) employing the updated time configuration $\mathbf{u}_i^{n+1}$
4. compute $\dot{\mathbf{u}}_i^{n+1} = \dot{\mathbf{u}}_i^{n+\frac{1}{2}} + \frac{1}{2}\ddot{\mathbf{u}}_i^{n+1} \Delta t$

The critical time step for the prototype microelastic brittle model was evaluated by Silling and Askari [68] as follows

$$\Delta t_{crit} = \sqrt{\frac{2\rho}{\sum_p V_p C_p}} \tag{2.11}$$

where ρ is the density, p iterates over all the neighbors of the given material point, V_p is the volume associated with neighbor p, and C_p is the micromodulus between the given material point and neighbor p.

Another option commonly found in the literature is the maximum critical time step given by the Courant-Friedrichs-Lewy (CFL) approach [108],

$$\Delta t_{crit} = \frac{\Delta x}{c_w} \tag{2.12}$$

where Δx is the grid spacing and c_w is the wave speed as

$$c_w = \sqrt{\frac{K}{\rho}} \tag{2.13}$$

with K as the bulk modulus. Thus,

$$\Delta t_{crit} = \Delta x \sqrt{\frac{\rho}{K}} \tag{2.14}$$

This is a very conservative estimate of the critical time step for peridynamic models, since the wave speed is related to the horizon, not to the grid spacing. This critical value is employed in all the explicit dynamic simulations. However, the sudden release of energy in some cases and local feature may produce the unrealistic effect of breaking bonds connected to loaded nodes. To prevent rupture in this area, all the bonds connected to loaded nodes cannot be broken, creating a "no-fail" zone [67, 71, 113]: damage initiation is not allowed in this zone and damage cannot spread further. This no-fail zone should be limited and far away from the crack propagation area, according to Saint-Venant's Principle.

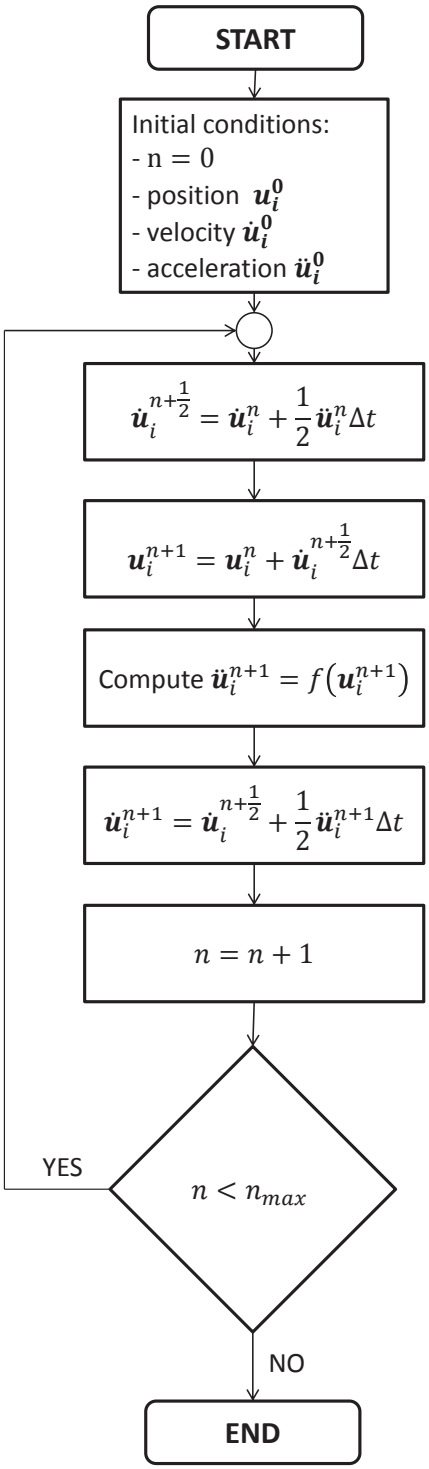


Figure 2.4: Flow chart of the velocity-Verlet time integration scheme.

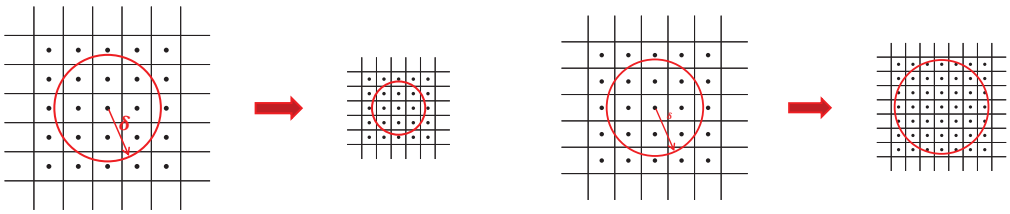
2.3 Numerical convergence

The parameters related to the number of interactions that have to be considered in a numerical simulation for each node are two: the horizon and the m-ratio $= \delta/\Delta x$, that is related the number of nodes within a horizon sphere[5].

Bobaru et al. [130] and Bobaru et Duangpanya [131] have introduced three different types of convergence that can be used to compare the numerical peridynamic solutions with the classical elasticity solutions for regular problems (without discontinuities).

The peridynamic model is nonlocal and its nonlocality is related to the horizon δ. Since the classical equations of elasticity have no intrinsic length scale, a first type of convergence for Peridynamics is *δ-convergence* (Figure 2.5a), in which the peridynamic horizon goes to zero. More precisely, δ-convergence is carried out by keeping the m-ratio constant while decreasing the horizon ($\delta \rightarrow 0$). The numerical solution converges to an approximation of the local classical solution, although not uniformly.

The second type is related to the m-ratio, thus called *m-convergence* (Figure 2.5b), carried out by keeping the horizon fixed while increasing the m-ratio (m $\longrightarrow \infty$). By increasing the m-ratio, the number of nodes within the horizon neighborhood increases as well. The results converge to the exact nonlocal peridynamic solution for the given δ.



(a) (b)

Figure 2.5: Two types of convergence in Peridynamics: (a) δ-convergence, (b) m-convergence.

Figure 2.6 shows the third type which is a combination of the previous ones:

[5]It can be correctly defined only for uniform discretized grids. It is not employed for non-uniform grids, such as in [78, 129].

(δm-*convergence*, where the m-ratio increases with a higher rate than the rate of decrease of δ). In this case, the numerical peridynamic approximation converges not only to the exact peridynamic solution but also to the local classical solution in a uniform way.

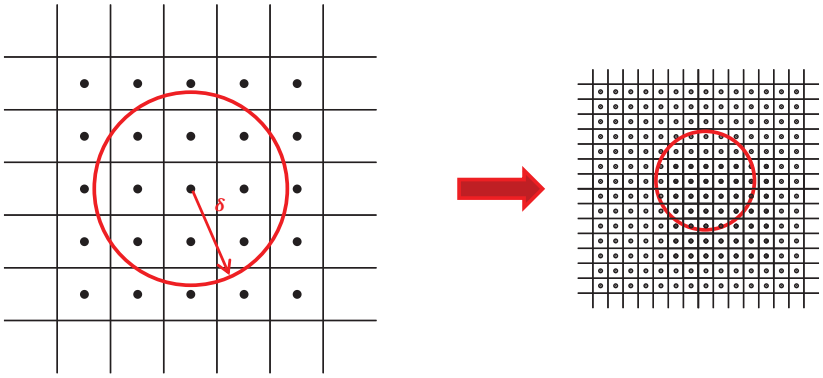


Figure 2.6: mδ-convergence in Peridynamics: δ decreases while the m-ratio increases with a higher rate of change.

Chapter 3

Algorithms for solvers

All the codes have been developed in Matlab®. During the three years of PhD research, an important task was to develop a robust and efficient code, since Peridynamics, being a nonlocal method, is computationally expensive.

Peridynamic solutions in elasticity or dynamic fracture problems use, almost exclusively[1], explicit solver code (e.g. [67, 83]). Most works dealing with peridynamic static solutions utilize quasi-dynamic solution methods, like dynamic relaxation ([114, 130]) or energy minimization ([121]), methods that avoid building the stiffness matrix. Explicit solvers are easier to implement and parallelize, but small time steps are required for stability reasons especially whenever nonlinear problems are considered. Besides, equilibrium is not usually imposed in explicit methods and the numerical solution may be subjected to a drift, as the computation progresses.

Implicit solvers offer some benefits in terms of stability over a wide range of time steps and of accuracy of the converged equilibrated solution. On the other hand, implicit solvers usually require a high computational time because they are based on the inversion of the stiffness matrix. When using Newton-type schemes (see e.g. [132, 133]) for implicit solutions of classical elastostatic and elastodynamic problems, building the stiffness matrix might be one of the main steps.

In section 3.1 the algorithm for the dynamic solver is illustrated, while the algorithm for static analyses is written in section 3.2, where the stiffness matrix assembling is investigated in detail in section 3.2.2. These algorithms are presented for the SBP formulation: the BBP version is basically the same, even though

[1]The exceptions are some examples in Sandia Reports [8, 7, 108].

slightly simpler.

Lastly, section 3.3 focuses on the development of the code from the first implementation to the vectorized notation which has produced a faster code to execute.

3.1 Dynamic Solver

The algorithm for the dynamic solver is made of the following steps:

1. *Acquire Material and Geometric Data,*
2. *Associate Load and Boundary Conditions to the appropriate set of nodes,*
3. *Define the initial conditions for every node:*
   (a) *Set the initial position* $\mathbf{u}(\mathbf{x}, t_0)$ *and velocity* $\dot{\mathbf{u}}(\mathbf{x}, t_0)$ *for each node located at* $\mathbf{x}$*,*
   (b) *Compute two nested loops for all nodes located at* $\mathbf{x}$ *and for all their family nodes* $\mathbf{p}$*, then compute, in the following order these states or matrices:*
      i. $\underline{X}$, $|\underline{X}|$ *the reference state and its scalar state (this state has to be computed just at this point for the all simulation),*
      ii. m *(or* q *in 2D simulations) the weighted volume computed just at* t_0 *as expressed in Eq.* (1.61)*,*
      iii. $\underline{Y}$, $|\underline{Y}|$ *the deformation state and its scalar state, which at the initial time step* t_0 *are respectively equal to* $\underline{X}$ *and* $|\underline{X}|$*,*
      iv. $\underline{e}(\underline{Y})$ *the scalar extension state, which is null at* t_0*, if the configuration is undeformed,*
      v. θ *the dilatation, a scalar value associated to each node* $\mathbf{x}$ *computed as Eq.* (1.62) *for 3D cases, Eq.* (1.66) *for 2D plane stress cases and Eq.* (1.71) *for 2D plane strain cases.*
   (c) *Compute two nested loops for all nodes located at* $\mathbf{x}$ *and for all their family nodes* $\mathbf{p}$*: all those states, which require* θ *to be computed beforehand, can be now calculated:*
      i. $\underline{e}^d$ *the scalar deviatoric state, corresponding to the deviatoric component of the extension scalar state, as expressed in Eq.* (1.63)*,*
      ii. $\underline{t}$ *the modulus force state as expressed in Eq.* (1.54) *for 3D cases, Eq.* (1.67) *for 2D plane stress cases and Eq.* (1.72) *for 2D plane strain cases,*

iii. $\underline{T}$ *the force state as expressed in Eq.* (1.53).

(d) Compute the initial acceleration $\ddot{\mathbf{u}}(\mathbf{x}, t_0)$ *for every node located at* $\mathbf{x}$.

4. Time step integration loop for each time step t_n*: the parameters are computed for each iteration, starting from* $t_1 = t_0 + \Delta t$ *in the following order (using a velocity-Verlet explicit algorithm):*

(a) Compute $\dot{\mathbf{u}}_{n+\frac{1}{2}}$ *and* $\mathbf{u}_{n+1}$ *from the previous step node accelerations,*

(b) Compute two nested loops for all nodes located at $\mathbf{x}$ *and for all their family nodes* $\mathbf{p}$*, in which calculate the following states:*

i. $\underline{Y}(t_n)$, $|\underline{Y}|(t_n)$*, which require the displacement of all the nodes calculated at the previous time step,*

ii. $\underline{e}(\underline{Y})(t_n)$,

iii. $\theta(t_n)$.

(c) Once θ *is known for every node, compute two nested loops for all nodes located at* $\mathbf{x}$ *and for all their family nodes* $\mathbf{p}$ *for all those states depending on the dilatation:*

i. $\underline{e}^d(t_n)$ *the current scalar deviatoric state,*

ii. $\underline{t}(t_n)$ *the current modulus force state,*

iii. $\underline{T}(t_n)$ *the current force state.*

(d) Compute the integral of all the force state,

(e) Update the acceleration $\ddot{\mathbf{u}}_{n+1}$ *and velocity* $\dot{\mathbf{u}}_{n+1}$ *for all the nodes.*

5. Save data.

The overall algorithm is shown in Figure 3.1

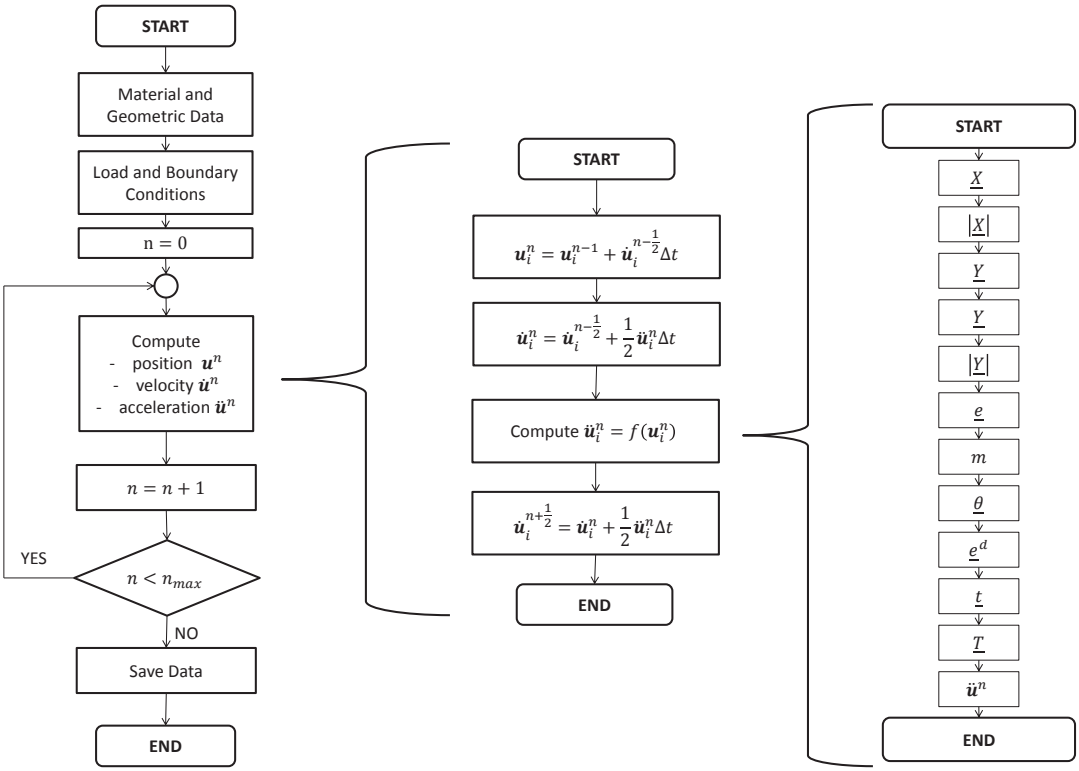


Figure 3.1: Flow chart of the algorithm of the dynamic solver.

3.2 Static Solver

Among the numerous static solvers employed in classical mechanics, the chosen one for static analyses has been a stiffness matrix approach. Thus, the motion equation of Peridynamics have been manipulated into the form:

$$\{F_{ext}\} = [K]\{U\} \tag{3.1}$$

where $\{F_{ext}\}$ is the external force vector, $[K]$ the stiffness matrix and $\{U\}$ the nodal displacement vector. The internal forces can be expressed as the stiffness matrix of the system $[K]$ times the nodal displacement vector $\{U\}$. In particular, if the load condition is divided into load steps then for a generic step n, Eq. (3.1) becomes

$$\{F_{ext}^n\} = [K(U^n)]\{U^n\} \tag{3.2}$$

When nonlinearities are taken into consideration, the equation of external and internal forces may be obtained by employing an iterative method. The reason is

due to the stiffness matrix dependence on the nodal displacements. In this case the tangent stiffness matrix $[K(U^n)]_T$ is needed to reach the convergent values of nodal displacements (i.e. the out of balance force $\{g^n\}$ becomes zero). The tangent stiffness matrix is described in section 3.2.1.

For this formulation, the stiffness matrix of the system is required and two steps have to be carried out in order to compute it:

1. to compute the linearized model with respect to the displacement state
2. to assemble the contributions, related to the bonds in the right position within the stiffness matrix components.

The former step is described in section 1.2.2. The latter step is shown in this section, in the following lines, explaining the structure of the algorithm.

The static solver includes the following steps:

1. Acquire Material and Geometric Data,

2. Associate Load and Boundary Conditions to the appropriate set of nodes,

3. Define the initial conditions for every node:

- *(a) Set the initial position and compute* $\underline{X}$, $|\underline{X}|$ *for each node located at* $\mathbf{x}$,
- *(b) Compute the weighted volume* m *as expressed in Eq.* (1.61),
- *(c) Compute a first stiffness matrix; some steps are required:*
  - *i. Compute* $\underline{Y}$, $|\underline{Y}|$ *the deformation state and its scalar state equal to the reference state and its scalar,*
  - *ii. Compute a loop for all the nodes: when analyzing a node* $\mathbf{x}$*, all the terms related to its motion equation is inputed in the stiffness matrix rows corresponding to node* $\mathbf{x}$ *degrees of freedom; the stiffness matrix is thus built as a column vector of pairs of rows. A detailed algorithm for the stiffness matrix is described in section 3.2.2.*

4. Load step integration for-loop, for every step n*, the steps to be followed are:*

- *(a) Building the external force vector* $\{F^n_{ext}\}$ *from the load conditions,*
- *(b) Reducing the stiffness matrix* $[K(U^n)]$ *removing the rows and columns corresponding to the degree of freedom corresponding to those nodes for which the displacements are* a priori *known,*

(c) Employing a solving method to solve Eq. (3.1)*, where the unknown vector is the reduced displacement vector ,*

(d) Building the complete incremental displacement vector $\{\Delta U^n\}$ *inserting the rows corresponding to the known degrees of freedom and sum it up to the previous total displacement vector* $\{U^{n-1}\}$*,*

(e) Update the stiffness matrix in order to check the residual vector and if needed iterate with a method to reach convergence or to move to the next load step.

5. Save data.

The algorithm for the static code is shown in the flow chart in Figure 3.2. the detail algorithm for the assembling of the stiffness matrix is shown in section 3.3.2.

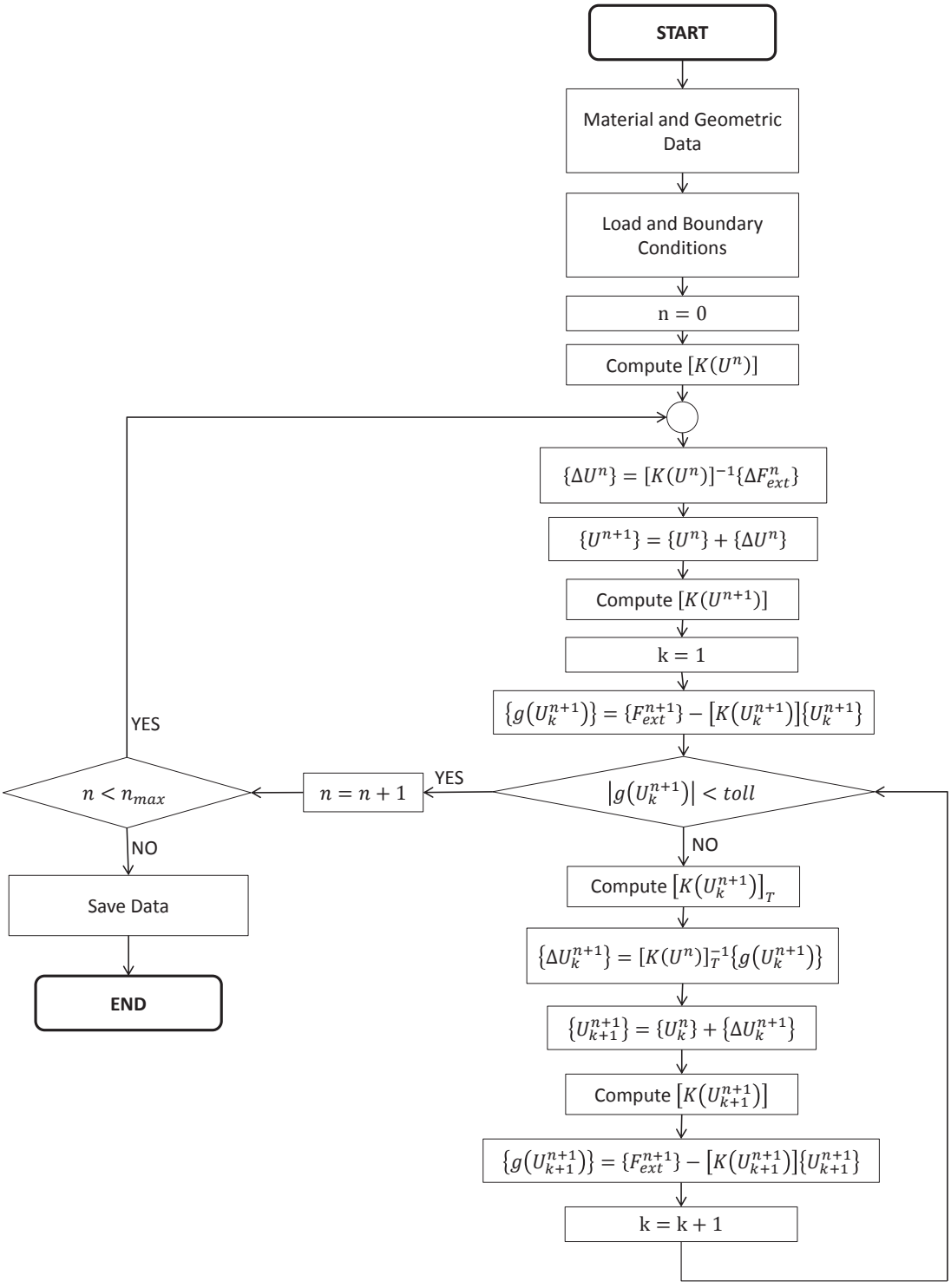


Figure 3.2: Flow chart of the algorithm of the static solver.

3.2.1 The tangent stiffness matrix

In order to solve nonlinear problems, such as those where crack initiates and propagates, a Newton-Raphson integration scheme may be employed. In this case, the tangent stiffness matrix is needed. In a discretized system, the equation system is

$$[K(U)]\{U\} = \{F_{ext}\} \tag{3.3}$$

The out of balance residual force **g** is, therefore, written as

$$\{g(U)\} = \{F_{ext}\} - [K(U)]\{U\} \tag{3.4}$$

It is equal to zero when the system is equilibrated.

The Newton-Raphson scheme becomes:

$$\{g(U)\} = 0 \rightarrow \{U_{k+1}\} = \{U_k\} - \frac{\{g(U_k)\}}{\frac{d\{g(U_k)\}}{d\{U_k\}}} \tag{3.5}$$

The derivative of the out of balance force with respect to the nodal displacement is the slope of the stiffness matrix, the *tangent stiffness matrix*:

$$\begin{aligned} \frac{d\{g(U)\}}{d\{U\}} &= \frac{d}{d\{U\}}(\{F_{ext}\} - [K(U)]\{U\}) \\ &= [K(U)] + \frac{d[K(U)]}{d\{U\}}\{U\} = [K(U)]_T \end{aligned} \tag{3.6}$$

In particular, if N is the total number of nodes in the system, the (i, j) component of the tangent stiffness matrix is

$$K_{T,ij} = K_{ij} + \sum_{m=1}^{N} \frac{dK_{im}}{du_j} u_m \tag{3.7}$$

If the current iteration is k and the next iteration to be computed is $k + 1$, the Newton-Raphson formulation is derived as

$$\{U_{k+1}\} = \{U_k\} - [K(U_k)]_T^{-1}\{g(U_k)\} \tag{3.8}$$

Numerically speaking, the condition for convergence is that its norm is less than a given tolerance ($|g(U)| < tol$).

In section 3.2.2, the assembling of the stiffness matrix is illustrated. A similar procedure is adopted to assemble the tangent stiffness matrix.

3.2.2 The assembling of the Stiffness Matrix

The detailed algorithm to compute the stiffness matrix is described in the following lines where three nested loops are set in the stiffness matrix rows corresponding to the degrees of freedom of the considered node:

- n_x is the number of the node located at $\mathbf{x}$. If N is the total number of nodes of a three dimensional structure, the stiffness matrix is a $3N \times 3N$ matrix. Perform a cycle on all the nodes n_x of the structure.
- For each n_x, perform a cycle on all the nodes $n_p \in \mathcal{H}_x$, where $\mathcal{H}_x$ is the set of family nodes of n_x located at $\mathbf{p}$.
- Within the inner loop, a third loop is computed for all the nodes $n_q \in (\mathcal{H}_x \cup \mathcal{H}_p)$ located at $\mathbf{q}$, where $\mathcal{H}_p$ is the set of family nodes of n_p.
- Compute $\underline{\underline{\mathbb{K}}}[\mathbf{x}]\langle \mathbf{p}-\mathbf{x}, \mathbf{q}-\mathbf{x}\rangle$ and $\underline{\underline{\mathbb{K}}}[\mathbf{p}]\langle \mathbf{x}-\mathbf{p}, \mathbf{q}-\mathbf{p}\rangle$, which are 3×3 matrices, employing the appropriate terms in the formula.
- Add these contributions to the stiffness matrix in the right position. The assembly method can be summarized as:

$$
\begin{array}{l}
[K]^{\begin{matrix}3n_x-2,3n_x-2 & 3n_x-2,3n_x-1 & 3n_x-2,3n_x\\ 3n_x-1,3n_x-2 & 3n_x-1,3n_x-1 & 3n_x-1,3n_x\\ 3n_x,3n_x-2 & 3n_x,3n_x-1 & 3n_x,3n_x\end{matrix}} = [K]^{\begin{matrix}3n_x-2,3n_x-2 & 3n_x-2,3n_x-1 & 3n_x-2,3n_x\\ 3n_x-1,3n_x-2 & 3n_x-1,3n_x-1 & 3n_x-1,3n_x\\ 3n_x,3n_x-2 & 3n_x,3n_x-1 & 3n_x,3n_x\end{matrix}} \\
\qquad\qquad - \underline{\underline{\mathbb{K}}}[\mathbf{x}]\langle \mathbf{p}-\mathbf{x}, \mathbf{q}-\mathbf{x}\rangle
\end{array}
$$

$$
\begin{array}{l}
[K]^{\begin{matrix}3n_x-2,3n_p-2 & 3n_x-2,3n_p-1 & 3n_x-2,3n_p\\ 3n_x-1,3n_p-2 & 3n_x-1,3n_p-1 & 3n_x-1,3n_p\\ 3n_x,3n_p-2 & 3n_x,3n_p-1 & 3n_x,3n_p\end{matrix}} = [K]^{\begin{matrix}3n_x-2,3n_p-2 & 3n_x-2,3n_p-1 & 3n_x-2,3n_p\\ 3n_x-1,3n_p-2 & 3n_x-1,3n_p-1 & 3n_x-1,3n_p\\ 3n_x,3n_p-2 & 3n_x,3n_p-1 & 3n_x,3n_p\end{matrix}} \\
\qquad\qquad + \underline{\underline{\mathbb{K}}}[\mathbf{p}]\langle \mathbf{x}-\mathbf{p}, \mathbf{q}-\mathbf{p}\rangle
\end{array} \tag{3.9}
$$

$$
\begin{array}{l}
[K]^{\begin{matrix}3n_x-2,3n_q-2 & 3n_x-2,3n_q-1 & 3n_x-2,3n_q\\ 3n_x-1,3n_q-2 & 3n_x-1,3n_q-1 & 3n_x-1,3n_q\\ 3n_x,3n_q-2 & 3n_x,3n_q-1 & 3n_x,3n_q\end{matrix}} = [K]^{\begin{matrix}3n_x-2,3n_q-2 & 3n_x-2,3n_q-1 & 3n_x-2,3n_q\\ 3n_x-1,3n_q-2 & 3n_x-1,3n_q-1 & 3n_x-1,3n_q\\ 3n_x,3n_q-2 & 3n_x,3n_q-1 & 3n_x,3n_q\end{matrix}} \\
\qquad\qquad - \underline{\underline{\mathbb{K}}}[\mathbf{p}]\langle \mathbf{x}-\mathbf{p}, \mathbf{q}-\mathbf{p}\rangle \\
\qquad\qquad + \underline{\underline{\mathbb{K}}}[\mathbf{x}]\langle \mathbf{p}-\mathbf{x}, \mathbf{q}-\mathbf{x}\rangle
\end{array}
$$

In this way, every step of the most external cycle, concerning a given node **x**, fills out three rows corresponding to the three degrees of freedom of node **x**.

$$[K] = \begin{bmatrix} & \cdots & \\ & \cdots & \\ & \cdots & \\ \begin{bmatrix} k_{3n_x-2,1} & k_{3n_x-2,2} & \cdots & k_{3n_x-2,3N-1} & k_{3n_x-2,3N} \\ k_{3n_x-1,1} & k_{3n_x-1,2} & \cdots & k_{3n_x-1,3N-1} & k_{3n_x-1,3N} \\ k_{3n_x,1} & k_{3n_x,2} & \cdots & k_{3n_x,3N-1} & k_{3n_x,3N} \end{bmatrix} & & \\ & \cdots & \\ & \cdots & \\ & \cdots & \end{bmatrix} \tag{3.10}$$

The code for the computation of the stiffness matrix is shown in Figure 3.3. Assembling the stiffness matrix is illustrated in section 3.3.

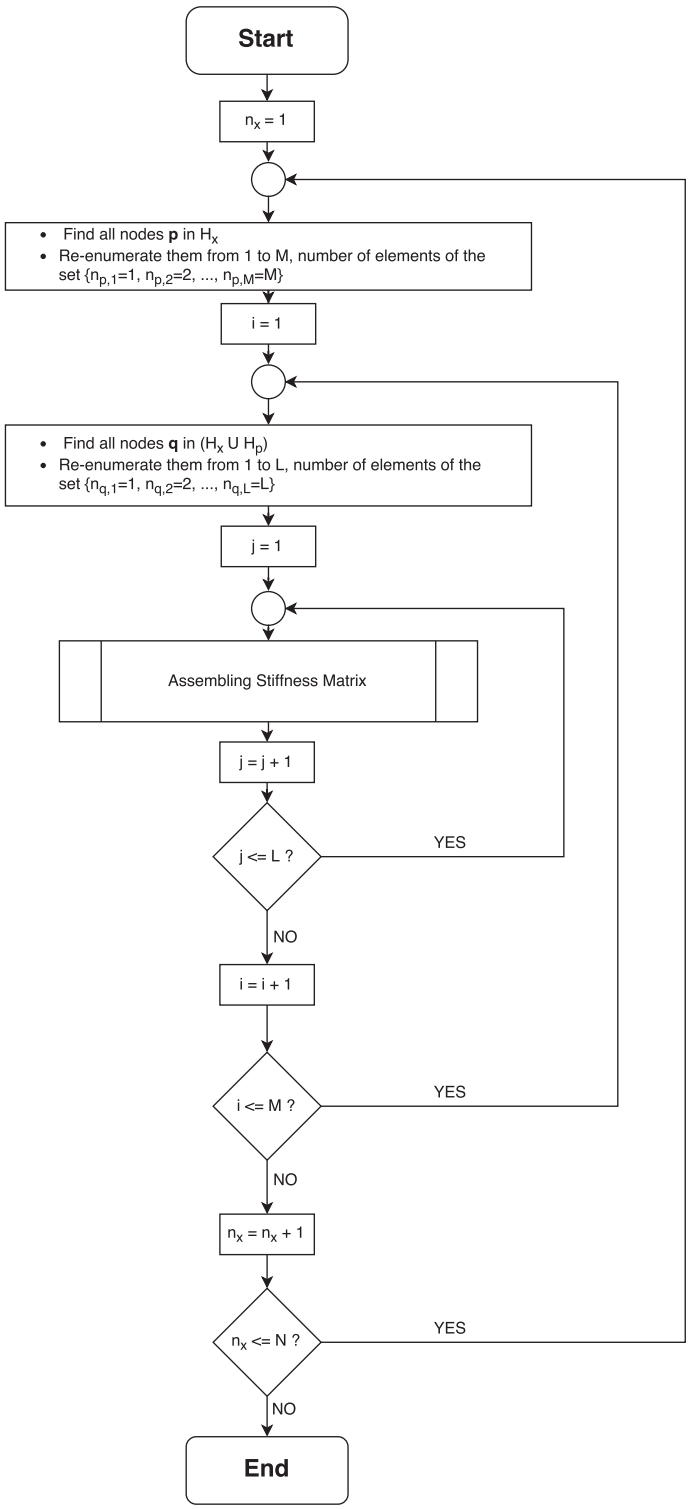


Figure 3.3: Flow chart of the computation of the stiffness matrix.

3.3 Code Optimization

In this section, the data structure of the developed code is described. This is useful for the development of further codes. Pseudo-codes of the employed algorithms are shown. Section 3.3.1 refers to the different command lines for BBP code, while section 3.3.2 introduces further optimization with additional details related to SBP code.

3.3.1 BBP Formulation

The first basic code and the final vectorized code are compared in Appendix A for BBP formulation. The vectorization takes advantage of Matlab® environment and Appendix B shows in detail the data structure for the assembling of the stiffness matrix in the BBP code.

After initializing the geometric and material properties, the grid of nodes must be defined: in all simulations of this thesis, the adopted grid is uniform, thus $\Delta x = \Delta y = \Delta z$. A matrix collecting all the absolute position data is generated, through nested loops; for convenience the volume of the node is collected in the last column of the matrix. The algorithm shown in Figure 3.4 is for a rectangular specimen, but it can be generalized to more complex shapes.

Algorithm for building the geometric matrix

{Initialize the node ID number}
$n_x = 1$
{Compute the maximum number of nodes in x, y and z direction}
$num_x = L_x/\Delta x$
$num_y = L_y/\Delta y$
$num_z = L_z/\Delta z$
{Sweep the three directions and define the positions of all nodes}
for i$=-num_x/2 : num_x/2$
  for j$=-num_y/2 : num_y/2$
    for k$=-num_z/2 : num_z/2$
      $position(n_x, 1:3) = [i \cdot \Delta x \quad j \cdot \Delta y \quad k \cdot \Delta z]$
      $position(n_x, 4) = \Delta x \cdot \Delta y \cdot \Delta z$
      $n_x = n_x + 1$
  end for
  end for
end for

Figure 3.4: Algorithm for building the position matrix for all the nodes of the grid.

Note that there is no need to store the ID number, since it is identified by the index of the rows in which data are stored. This is a strategy that is employed throughout all codes to avoid unnecessary memory consumption.

The algorithm for the initial bond matrix has been developed by a colleague of mine and it is reported here in Figure 3.5 to be thorough: it contains the ID numbers of family nodes in the row corresponding to the source node. For source nodes belonging to boundary layers, the rows are completed with zero element (i.e. for a simulation of a plate where the parameters are set to have a maximum number of family node M, a corner node row has $1/4 \cdot M$ nonzero entries and $3/4 \cdot M$ zero elements).

Algorithm for building the initial bond matrix

{Estimate the maximum value of the bonds of an internal point}
$c_b = \frac{\pi\delta^2}{\Delta x \Delta y} + 10$
{Initialize the bond matrix I_{bond} to a zero matrix}
$I_{bond} = zeros(N, c_b)$
{Compute a loop within the node ID numbers}
for $n_{x,1} = 1 : N$
  {Compute the distance between $n_{x,1}$ and all the other points}
  $pos_r(:, 1:2) = position(:, 1:2) - position(n_{x,1}, 1:2)$
  $d(:, 1) = (pos_r(:, 1).^2 - pos_r(:, 2).^2).^{0.5}$
  {Find the points within the horizon disk}
  $n_{p_j} = find(d(:, 1) \leq \delta \ \ \& \ \ d(:, 1) > 0)$
  $n = numel(n_{p_j})$
  if $n > c_b$
    $I_{bond} = [I_{bond}, zeros(N, n - c_b)]$
    $c_b = n$
  endif
  $I_{bond}(n_{x,1}, 1 : n) = n_{p_j}$
endfor
{Eliminate possible excess null column in I_{bond}}
$[\ , c] = find(I_{bond})$
$max_c = max(c)$
if $max_c < c_b$
  $I_{bond}(:, (max_c + 1) : c_b) = [\,]$
endif

Figure 3.5: Algorithm for building the Initial bond matrix of the structure for 2D systems.

However, the matrix I_{bond} containing the pristine bonds is conveniently com-

puted with a slightly different data structure, in which the number of source node (i.e. the row index) is inputed instead of null elements, because a simple sequence of operations can be maintained with a small additional memory and computational cost (for example, I don't need "if" conditions to consider exceptions in the algorithm).

$$I_{bond} = \begin{matrix} 1 \\ 2 \\ \vdots \\ \vdots \end{matrix} \begin{bmatrix} 2 & 3 & \cdots & 1 & 1 & 1 \\ 1 & 3 & \cdots & \cdots & 2 & 2 \\ & & \vdots & & & \\ & & \vdots & & & \end{bmatrix} \tag{3.11}$$

While in the original code every property and quantity was computed "bond-wise", in the optimized developed code, every quantity has been stored in matrices which, not only contain the relative value for all the bonds, but also all the operations are performed element-wise. The matrices are built in this way:

- if the property is associated to each node, then the matrix containing that property is a column vector of length equal to the total number of nodes N in the structure, for example the density $\rho(\mathbf{x})$ is

$$\varrho = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_{N-1} \\ \rho_N \end{bmatrix} \tag{3.12}$$

- if the property is associated to each bond and it is a scalar, the matrix has the same size (N, M) of the bond matrix I_{bond}, like the scalar reference matrix $|X|$

$$I_{bond} = \begin{matrix} 1 \\ 2 \\ 3 \\ \vdots \\ \vdots \end{matrix} \begin{bmatrix} 2 & 3 & 4 & \cdots & 0 & 0 \\ 1 & 3 & 4 & \cdots & \cdots & 0 \\ 1 & 2 & 4 & \cdots & \cdots & 0 \\ & & & \vdots & & \\ & & & \vdots & & \end{bmatrix} \mapsto |X| = \begin{bmatrix} |\boldsymbol{\xi}|_{2-1} & |\boldsymbol{\xi}|_{3-1} & |\boldsymbol{\xi}|_{4-1} & \cdots & 0 & 0 \\ |\boldsymbol{\xi}|_{1-2} & |\boldsymbol{\xi}|_{3-2} & |\boldsymbol{\xi}|_{4-2} & \cdots & \cdots & 0 \\ |\boldsymbol{\xi}|_{1-3} & |\boldsymbol{\xi}|_{2-3} & |\boldsymbol{\xi}|_{4-3} & \cdots & \cdots & 0 \\ & & \vdots & & & \\ & & \vdots & & & \end{bmatrix} \tag{3.13}$$

- if the property is associated to each bond and it is a vector, three (for $3D$ cases) or two (for $2D$ cases) matrices are built of the same size (N, M) of I_{bond} containing (x, y, z) components separately.

$$I_{bond} = \begin{matrix} 1 \\ 2 \\ 3 \\ \vdots \\ \vdots \end{matrix} \begin{bmatrix} 2 & 3 & 4 & \cdots \\ 1 & 3 & 4 & \cdots \\ 1 & 2 & 4 & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix} \quad \mapsto \quad X_x = \begin{bmatrix} \xi_{x,(2-1)} & \xi_{x,(3-1)} & \xi_{x,(4-1)} & \cdots \\ \xi_{x,(1-2)} & \xi_{x,(3-2)} & \xi_{x,(4-2)} & \cdots \\ \xi_{x,(1-3)} & \xi_{x,(2-3)} & \xi_{x,(4-3)} & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix}$$

$$X_y = \begin{bmatrix} \xi_{y,(2-1)} & \xi_{y,(3-1)} & \xi_{y,(4-1)} & \cdots \\ \xi_{y,(1-2)} & \xi_{y,(3-2)} & \xi_{y,(4-2)} & \cdots \\ \xi_{y,(1-3)} & \xi_{y,(2-3)} & \xi_{y,(4-3)} & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix}$$

$$X_z = \begin{bmatrix} \xi_{z,(2-1)} & \xi_{z,(3-1)} & \xi_{z,(4-1)} & \cdots \\ \xi_{z,(1-2)} & \xi_{z,(3-2)} & \xi_{z,(4-2)} & \cdots \\ \xi_{z,(1-3)} & \xi_{z,(2-3)} & \xi_{z,(4-3)} & \cdots \\ & \vdots & & \\ & \vdots & & \end{bmatrix} \tag{3.14}$$

The most advanced code for these matrices is shown in Figures 3.6, 3.7 and 3.8.

These algorithms take advantage of some built-in function of Matlab®, such as:

- *reshape*, a function that changes the size of a matrix by keeping the total number of elements constant.
- *repmat*, a function that replicate matrix, so that the original matrix is a sub-matrix of the final one
- "." operator, an operator allowing for element-wise operations on matrices
- $bsxfun$, a function that apply or verify element-by-element conditions in a matrix
- $sparse$, a function that automatically add contributions to a given entry

Algorithm for building the X_x, X_y , X_z and $|X|$ matrices

1: {Initialize X_x, X_y, X_z and $|X|$ to zero matrices}
2: $X_x = zeros(N, M)$
3: $X_y = zeros(N, M)$
4: $X_z = zeros(N, M)$
5: $|X| = zeros(N, M)$
6: {Compute a vector calculation of relative position}
7: $X_x = reshape(position(I_{bond}, 1) - position(repmat([1:1:N]', M, 1), 1), N, [\,])$
8: $X_y = reshape(position(I_{bond}, 2) - position(repmat([1:1:N]', M, 1), 2), N, [\,])$
9: $X_z = reshape(position(I_{bond}, 3) - position(repmat([1:1:N]', M, 1), 3), N, [\,])$
10: $|X| = (X_x.^2 + X_y.^2 + X_z.^2).^{0.5}$

Figure 3.6: Algorithm for building the $|X|$ and its component for all the nodes of the grid.

Where source node ID numbers are present in the corresponding rows of I_{bond}, the components in X_x, X_y, X_z and $|X|$ matrices are null.

Algorithm for building the volume correction factor f_V matrix

1: {Initialize f_V to a zero matrix}
2: $f_V = zeros(N, M)$
3: $f_V = bsxfun(@lte, |X|, 0.0) \cdot 0.0 +$
$bsxfun(@and, |X| \leq \delta - \frac{1}{2}\Delta x, |X| > 0.0) \cdot 1.0 +$
$bsxfun(@and, |X| \leq \delta + \frac{1}{2}\Delta x, |X| > \delta - \frac{1}{2}\Delta x) \cdot (\frac{\delta + \Delta x/2 - |X|}{\Delta x})$

Figure 3.7: Algorithm for building the volume correction factor f_V for all the nodes of the grid. This algorithm is shown in [9] as HBB.

It is convenient for computational purposes and for code readability to have a volume matrix containing all the volumes of the nodes specified in the correspondent element of I_{bond} (Figure 3.8).

Algorithm for building the volume matrix V

1: {Initialize V to a zero matrix}
2: $V = zeros(N, M)$
3: $V = reshape(position(I_{bond}, 4), N, [\,])$

Figure 3.8: Algorithm for building the volume matrix V for all the nodes of the grid.

Employing these matrices, the stiffness matrix of the system can be easily assembled as shown in Figure 3.9 where a stiffness matrix for a $2D$ system is computed. The motivation for the algorithm, especially for command lines 14, 15 and 16 is clarified through a small example in Appendix B.

Algorithm for building the initial stiffness matrix

{Initialize the state of the stiffnesses of all the bonds to a zero state}
$rig_{mod} = zeros(N, M)$
{Compute all the stiffnesses in the local coordinate system at once}
$rig_{mod} = \frac{1}{2}\frac{V_\cdot \cdot f_V}{|X|}. \cdot repmat(position(:, 4), 1, M)$
{Compute the components in the global coordinate system}
$rig_{xx} = rig_{mod}. \cdot (X_x.^2./|X|.^2)$
$rig_{xy} = rig_{mod}. \cdot (X_x. \cdot X_y./|X|.^2)$
$rig_{yy} = rig_{mod}. \cdot (X_y.^2./|X|.^2)$
{Initialize the three inputs of *sparse* function.}
$R_1 = zeros(1, 16N \cdot M)$
$C_1 = zeros(1, 16N \cdot M)$
$V_1 = zeros(1, 16N \cdot M)$
{Compute their values}
$R_1 = [repmat(1:2:2N, 1, M)';\ repmat(1:2:2N, 1, M)';\ \ldots$
$repmat(2:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$repmat(1:2:2N, 1, M)';\ repmat(1:2:2N, 1, M)';\ \ldots$
$repmat(2:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:);\ 2 \cdot I_{bond}(:);\ \ldots$
$2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:);\ 2 \cdot I_{bond}(:)]$
$C_1 = [repmat(1:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$repmat(1:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:);\ 2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:);\ \ldots$
$repmat(1:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$repmat(1:2:2N, 1, M)';\ repmat(2:2:2N, 1, M)';\ \ldots$
$2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:);\ 2 \cdot I_{bond}(:) - 1;\ 2 \cdot I_{bond}(:)]$
$V_1 = [rig_{xx}(:), rig_{xy}(:), rig_{xy}(:), rig_{yy}(:), \ldots$
$-rig_{xx}(:), -rig_{xy}(:), -rig_{xy}(:), -rig_{yy}(:), \ldots$
$-rig_{xx}(:), -rig_{xy}(:), -rig_{xy}(:), -rig_{yy}(:), \ldots$
$rig_{xx}(:), rig_{xy}(:), rig_{xy}(:), rig_{yy}(:)]$
$[K] = sparse(R_1, C_1, V_1)$

Figure 3.9: Algorithm for building the stiffness matrix of the structure.

The tangent stiffness matrix employed an algorithm based on the same approach, taking into account the additional term of Eq. (3.7).

3.3.2 SBP Formulation

The old algorithm for the SBP code is described in Appendix C.

If the property is associated to each node, then the matrix containing that property is a column vector of length equal to the total number N of nodes in the

structure, like in BBP code:

$$\underline{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{N-1} \\ \theta_N \end{bmatrix} \qquad m = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_{N-1} \\ m_N \end{bmatrix} \tag{3.15}$$

If they are vector properties, three (for 3D cases) or two (2D) matrices are built, one for each components. If the property is associated to a node, the component matrices are columns, otherwise, if associated to a bond, the component matrices have $N \times M$ elements. There are some states that don't get updated during the simulation, for example the reference state, which is the state containing the initial length of all the bonds in the structure, the influence function $\underline{\omega}$, the partial volume correction factor f_V and the weight m in Eq. (1.61). After building these states the same approach of the BBP formulation (see for example Figures 3.10 and 3.11), the stiffness matrix can be computed as explained in Figure 3.12. Note that in the algorithm $[Kxp] = \mathbb{K}\langle \mathbf{x} - \mathbf{p}, \mathbf{q} - \mathbf{p} \rangle$ and $[Kxx] = \mathbb{K}\langle \mathbf{p} - \mathbf{x}, \mathbf{q} - \mathbf{x} \rangle$.

Algorithm for building the influence function $\underline{\omega}$ state

1: {Initialize $\underline{\omega}$ state to a zero state }
2: $\underline{\omega} = zeros(N, M)$
3: $\underline{\omega} = f.(\underline{X})$

Figure 3.10: Algorithm for building the $\underline{\omega}$ state for all the nodes of the grid, considering that $\underline{\omega} = f(\underline{X})$.

Algorithm for building m weight

1: {Initialize m to a zero state }
2: $m = zeros(N, 1)$
3: $m = sum(\underline{\omega}. \cdot |\underline{X}|.^2. \cdot \underline{f_V}. \cdot \underline{V}, 2)$

Figure 3.11: Algorithm for building the m weight for all the nodes of the grid, where $\underline{V}$ is a state containing all the V_{n_p} of the nodes listed in the correspondent position of I_{bond} and the $sum(\cdot, 2)$ function sums along the column dimension.

Algorithm for building the stiffness matrix

1: {Initialize the stiffness matrix K to a zero state }
2: $K = zeros(3 \times N, 3 \times M)$
3: {Compute a loop within the node ID numbers}
4: **for** $n_x = 1 : N$
5: {Compute a loop within the family nodes of node n_x}
6: **for** j= $1 : M$
7: $n_p = I_{bond}(n_x, j)$
8: {Compute a loop within the family nodes of node n_p and n_x}
9: $I_q = [I_{bond}(n_p, :), I_{bond}(n_x, :)]$
10: **for** k= $1 : 2M$
11: $n_q = I_q(1, k)$
12: {Identify to which set of Figure 3.15 n_q belongs}
13: PP=find($n_q \in I_{bond}(n_p, :)$)
14: QQ=find($n_q \in I_{bond}(n_x, :)$)
15: **if** $QQ = \emptyset$
16: {Compute $[Kxp]$ with Eq. (1.81) without the Dirac term}
17: {Add $[Kxp]$ to the stiffness matrix}
18: $K_{3n_x-2:3n_x,3n_p-2:3n_p} = K_{3n_x-2:3n_x,3n_p-2:3n_p} + [Kxp]$
19: $K_{3n_x-2:3n_x,3n_q-2:3n_q} = K_{3n_x-2:3n_x,3n_q-2:3n_q} - [Kxp]$
20: **elseif** $PP = \emptyset$
21: {Compute $[Kxx]$ with Eq. (1.81) without the Dirac term}
22: {Add $[Kxx]$ to the stiffness matrix}
23: $K_{3n_x-2:3n_x,3n_x-2:3n_x} = K_{3n_x-2:3n_x,3n_x-2:3n_x} - [Kxx]$
24: $K_{3n_x-2:3n_x,3n_q-2:3n_q} = K_{3n_x-2:3n_x,3n_q-2:3n_q} + [Kxx]$
25: **else**
26: {Compute $[Kxx]$ and $[Kxp]$ with or without the Dirac term with Eq. (1.81)}
27: {Add $[Kxx]$ and $[Kxp]$ to the stiffness matrix}
28: $K_{3n_x-2:3n_x,3n_x-2:3n_x} = K_{3n_x-2:3n_x,3n_x-2:3n_x} - [Kxx]$
29: $K_{3n_x-2:3n_x,3n_p-2:3n_p} = K_{3n_x-2:3n_x,3n_p-2:3n_p} + [Kxp]$
30: $K_{3n_x-2:3n_x,3n_q-2:3n_q} = K_{3n_x-2:3n_x,3n_q-2:3n_q} + [Kxx] - [Kxp]$
31: **end if**
32: **end for**
33: **end for**
34: **end for**

Figure 3.12: Algorithm for building the stiffness matrix of the structure. The complete algorithm is shown in the flow chart of Figure 3.3, while command lines 12-30 are explained in more details in Figure 3.15.

The optimized code can be seen in Figures 3.13 and 3.14 ad it is built with the same data structure of the BBP code. Nevertheless, it becomes so complex to understand that the initial algorithm for the stiffness matrix is shown as well, for a faster comprehension of the assembly of the stiffness matrix in SBP.

Algorithm for building the initial stiffness matrix in SBP(1^{st} part)

1: {Initialize the stiffness matrix K to a zero state }
2: $K = zeros(3 \times N, 3 \times M)$
3: {Compute the γ constant}
4: {Computed loop within the node ID numbers}
5: **for** $n_x = 1 : N$
6: $q_{ij} = I_{bond}(I_{bond}(n_x, :), :)$;
7: $KKx_{xx_1} = \frac{\gamma_{\cdot} pos(n_x,4)}{m(n_x).^2} . \cdot [\underline{\omega}(n_x, :)' \cdot \underline{\omega}(n_x, :)] . \cdot [\underline{x}(n_x, :)' \cdot \underline{x}(n_x, :)] . \cdot \ldots$
$. \cdot [\underline{V}(n_x, :)' \cdot \underline{V}(n_x, :)] . \cdot \left[\underline{f_V}(n_x, :)' \cdot \underline{f_V}(n_x, :)\right] . \cdot \ldots$
$. \cdot [\underline{M}_x(n_x, :)' \cdot \underline{M}_x(n_x, :)]$;
8: $KKx_{xy_1} = \frac{\gamma_{\cdot} pos(n_x,4)}{m(n_x).^2} . \cdot [\underline{\omega}(n_x, :)' \cdot \underline{\omega}(n_x, :)] . \cdot [\underline{x}(n_x, :)' \cdot \underline{x}(n_x, :)] . \cdot \ldots$
$. \cdot [\underline{V}(n_x, :)' \cdot \underline{V}(n_x, :)] . \cdot \left[\underline{f_V}(n_x, :)' \cdot \underline{f_V}(n_x, :)\right] . \cdot \ldots$
$. \cdot \left[\underline{M}_x(n_x, :)' \cdot \underline{M}_y(n_x, :)\right]$;
9: $KKx_{yx_1} = \frac{\gamma_{\cdot} pos(n_x,4)}{m(n_x).^2} . \cdot [\underline{\omega}(n_x, :)' \cdot \underline{\omega}(n_x, :)] . \cdot [\underline{x}(n_x, :)' \cdot \underline{x}(n_x, :)] . \cdot \ldots$
$. \cdot [\underline{V}(n_x, :)' \cdot \underline{V}(n_x, :)] . \cdot \left[\underline{f_V}(n_x, :)' \cdot \underline{f_V}(n_x, :)\right] . \cdot \ldots$
$. \cdot \left[\underline{M}_y(n_x, :)' \cdot \underline{M}_x(n_x, :)\right]$;
10: $KKx_{yy_1} = \frac{\gamma_{\cdot} pos(n_x,4)}{m(n_x).^2} . \cdot [\underline{\omega}(n_x, :)' \cdot \underline{\omega}(n_x, :)] . \cdot [\underline{x}(n_x, :)' \cdot \underline{x}(n_x, :)] . \cdot \ldots$
$. \cdot [\underline{V}(n_x, :)' \cdot \underline{V}(n_x, :)] . \cdot \left[\underline{f_V}(n_x, :)' \cdot \underline{f_V}(n_x, :)\right] . \cdot \ldots$
$. \cdot \left[\underline{M}_y(n_x, :)' \cdot \underline{M}_y(n_x, :)\right]$;
11: $KKp_{xx_1} = \frac{\gamma_{\cdot} repmat(pos(I_{bond}(n_x,:),4),1,M)}{.repmat(m(I_{bond}(n_x,:)).^2,1,M)} . \cdot \ldots$
$. \cdot \left[repmat(\underline{\omega}(I_{bond}(n_x, :), :)', 1, M) \cdot \underline{\omega}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{x}(n_x, :)', 1, M) \cdot \underline{x}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{V}(n_x, :)', 1, M) \cdot \underline{V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{f_V}(n_x, :)', 1, M) \cdot \underline{f_V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(-\underline{M}_x(n_x, :)', 1, M) \cdot \underline{M}_x(I_{bond}(n_x, :), :)\right]$;
12: $KKp_{xy_1} = \frac{\gamma_{\cdot} repmat(pos(I_{bond}(n_x,:),4),1,M)}{.repmat(m(I_{bond}(n_x,:)).^2,1,M)} . \cdot \ldots$
$. \cdot \left[repmat(\underline{\omega}(I_{bond}(n_x, :), :)', 1, M) \cdot \underline{\omega}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{x}(n_x, :)', 1, M) \cdot \underline{x}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{V}(n_x, :)', 1, M) \cdot \underline{V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{f_V}(n_x, :)', 1, M) \cdot \underline{f_V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(-\underline{M}_x(n_x, :)', 1, M) \cdot \underline{M}_y(I_{bond}(n_x, :), :)\right]$;
13: $KKp_{yx_1} = \frac{\gamma_{\cdot} repmat(pos(I_{bond}(n_x,:),4),1,M)}{.repmat(m(I_{bond}(n_x,:)).^2,1,M)} . \cdot \ldots$
$. \cdot \left[repmat(\underline{\omega}(I_{bond}(n_x, :), :)', 1, M) \cdot \underline{\omega}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{x}(n_x, :)', 1, M) \cdot \underline{x}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{V}(n_x, :)', 1, M) \cdot \underline{V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{f_V}(n_x, :)', 1, M) \cdot \underline{f_V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(-\underline{M}_y(n_x, :)', 1, M) \cdot \underline{M}_x(I_{bond}(n_x, :), :)\right]$;
14: $KKp_{yy_1} = \frac{\gamma_{\cdot} repmat(pos(I_{bond}(n_x,:),4),1,M)}{.repmat(m(I_{bond}(n_x,:)).^2,1,M)} . \cdot \ldots$
$. \cdot \left[repmat(\underline{\omega}(I_{bond}(n_x, :), :)', 1, M) \cdot \underline{\omega}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{x}(n_x, :)', 1, M) \cdot \underline{x}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{V}(n_x, :)', 1, M) \cdot \underline{V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(\underline{f_V}(n_x, :)', 1, M) \cdot \underline{f_V}(I_{bond}(n_x, :), :)\right] . \cdot \ldots$
$. \cdot \left[repmat(-\underline{M}_y(n_x, :)', 1, M) \cdot \underline{M}_y(I_{bond}(n_x, :), :)\right]$;
...

Figure 3.13: Algorithm for building the stiffness matrix of the structure: 1^{st} part.

Algorithm for building the initial stiffness matrix in SBP (2^{nd} part)

15: $KKx_{xx_2} = \frac{8\mu \cdot pos(n_x,4)}{m(n_x)}. \cdot \underline{\omega}(n_x,:)'. \cdot \underline{V}(n_x,:)'. \cdot \underline{f_V}(n_x,:)'. \cdot \cdot \cdot \cdot$
$[\underline{M}_x(n_x,:)' \cdot \underline{M}_x(n_x,:)]$;

16: $KKx_{xy_2} = \frac{8\mu \cdot pos(n_x,4)}{m(n_x)}. \cdot \underline{\omega}(n_x,:)'. \cdot \underline{V}(n_x,:)'. \cdot \underline{f_V}(n_x,:)'. \cdot \cdot \cdot \cdot$
$\left[\underline{M}_x(n_x,:)' \cdot \underline{M}_y(n_x,:)\right]$;

17: $KKx_{yx_2} = \frac{8\mu \cdot pos(n_x,4)}{m(n_x)}. \cdot \underline{\omega}(n_x,:)'. \cdot \underline{V}(n_x,:)'. \cdot \underline{f_V}(n_x,:)'. \cdot \cdot \cdot \cdot$
$\left[\underline{M}_y(n_x,:)' \cdot \underline{M}_x(n_x,:)\right]$;

18: $KKx_{yy_2} = \frac{8\mu \cdot pos(n_x,4)}{m(n_x)}. \cdot \underline{\omega}(n_x,:)'. \cdot \underline{V}(n_x,:)'. \cdot \underline{f_V}(n_x,:)'. \cdot \cdot \cdot \cdot$
$\left[\underline{M}_y(n_x,:)' \cdot \underline{M}_y(n_x,:)\right]$;

19: $KKp_{xx_2} = \frac{8\mu \cdot pos(n_x,4)}{m(I_{bond}(n_x,:))}. \cdot \underline{\omega}(I_{bond}(n_x,:),:)'. \cdot \underline{V}(I_{bond}(n_x,:),:)'. \cdot \cdot \cdot \cdot$
$\underline{f_V}(I_{bond}(n_x,:),:)'. \cdot [-\underline{M}_x(I_{bond}(n_x,:),:)' \cdot -\underline{M}_x(I_{bond}(n_x,:),:)]$;

20: $KKp_{xy_2} = \frac{8\mu \cdot pos(n_x,4)}{m(I_{bond}(n_x,:))}. \cdot \underline{\omega}(I_{bond}(n_x,:),:)'. \cdot \underline{V}(I_{bond}(n_x,:),:)'. \cdot \cdot \cdot \cdot$
$\underline{f_V}(I_{bond}(n_x,:),:)'. \cdot \left[-\underline{M}_x(I_{bond}(n_x,:),:)' \cdot -\underline{M}_y(I_{bond}(n_x,:),:)\right]$;

21: $KKp_{yx_2} = \frac{8\mu \cdot pos(n_x,4)}{m(I_{bond}(n_x,:))}. \cdot \underline{\omega}(I_{bond}(n_x,:),:)'. \cdot \underline{V}(I_{bond}(n_x,:),:)'. \cdot \cdot \cdot \cdot$
$\underline{f_V}(I_{bond}(n_x,:),:)'. \cdot \left[-\underline{M}_y(I_{bond}(n_x,:),:)' \cdot -\underline{M}_x(I_{bond}(n_x,:),:)\right]$;

22: $KKp_{yy_2} = \frac{8\mu \cdot pos(n_x,4)}{m(I_{bond}(n_x,:))}. \cdot \underline{\omega}(I_{bond}(n_x,:),:)'. \cdot \underline{V}(I_{bond}(n_x,:),:)'. \cdot \cdot \cdot \cdot$
$\underline{f_V}(I_{bond}(n_x,:),:)'. \cdot \left[-\underline{M}_y(I_{bond}(n_x,:),:)' \cdot -\underline{M}_y(I_{bond}(n_x,:),:)\right]$;

23: $R_1 = [2n_x - 1, 2n_x - 1, 2n_x, 2n_x, 2n_x - 1, 2n_x - 1, 2n_x, 2n_x, \cdots$
$repmat(2n_x - 1, 1, 2M), repmat(2n_x, 1, 2M), \cdots$
$repmat(2n_x - 1, 1, 2M), repmat(2n_x, 1, 2M), \cdots$
$repmat(2n_x - 1, 1, 2M^2), repmat(2n_x, 1, 2M^2), \cdots$
$2n_x - 1, 2n_x - 1, 2n_x, 2n_x, repmat(2n_x - 1, 1, 2M), \cdots$
$repmat(2n_x, 1, 2M), repmat(2n_x - 1, 1, 2M), \cdots$
$repmat(2n_x, 1, 2M),]$;

24: $C_1 = [2n_x - 1, 2n_x, 2n_x - 1, 2n_x, 2n_x - 1, 2n_x, 2n_x - 1, 2n_x, \cdots$
$2I_{bond}(n_x,:) - 1, 2I_{bond}(n_x,:), 2I_{bond}(n_x,:) - 1, 2I_{bond}(n_x,:), \cdots$
$2I_{bond}(n_x,:) - 1, 2I_{bond}(n_x,:), 2I_{bond}(n_x,:) - 1, 2I_{bond}(n_x,:), \cdots$
$2q_{ij}(:)' - 1, 2q_{ij}(:)', 2q_{ij}(:)' - 1, 2q_{ij}(:)', 2I_{bond}(n_x,:) - 1, \cdots$
$2I_{bond}(n_x,:), 2I_{bond}(n_x,:) - 1, 2I_{bond}(n_x,:)]$;

25: $V_1 = [sum(sum(-KKx_{xx_1})), sum(sum(-KKx_{xy_1})), \cdots$
$sum(sum(-KKx_{yx_1})), sum(sum(-KKx_{yy_1})), \cdots$
$sum(-KKx_{xx_2}), sum(-KKx_{xy_2}), sum(-KKx_{yx_2}), \cdots$
$sum(-KKx_{yy_2}), sum(KKx_{xx_1}, 1), sum(KKx_{xy_1}, 1), \cdots$
$sum(KKx_{yx_1}, 1), sum(KKx_{yy_1}, 1), KKx_{xx_2}, KKx_{xy_2}, \cdots$
$KKx_{yx_2}, KKx_{yy_2}, -KKp_{xx_1}(:)', -KKp_{xy_1}(:)', -KKp_{yx_1}(:)', \cdots$
$-KKp_{yy_1}(:)', sum(sum(-KKp_{xx_2})), sum(sum(-KKp_{xy_2})), \cdots$
$sum(sum(-KKp_{yx_2})), sum(sum(-KKxp_{yy_2})), \cdots$
$sum(KKp_{xx_1}, 2))', sum(KKp_{xy_1}, 2))', sum(KKp_{yx_1}, 2))', \cdots$
$sum(KKp_{yy_1}, 2))', KKp_{xx_2}, KKp_{xy_2}, KKp_{yx_2}, KKp_{yy_2}]$;

26: **end for**

27: $[K] = sparse(R_1, C_1, V_1)$;

Figure 3.14: Algorithm for building the stiffness matrix of the structure: 2^{nd} part.

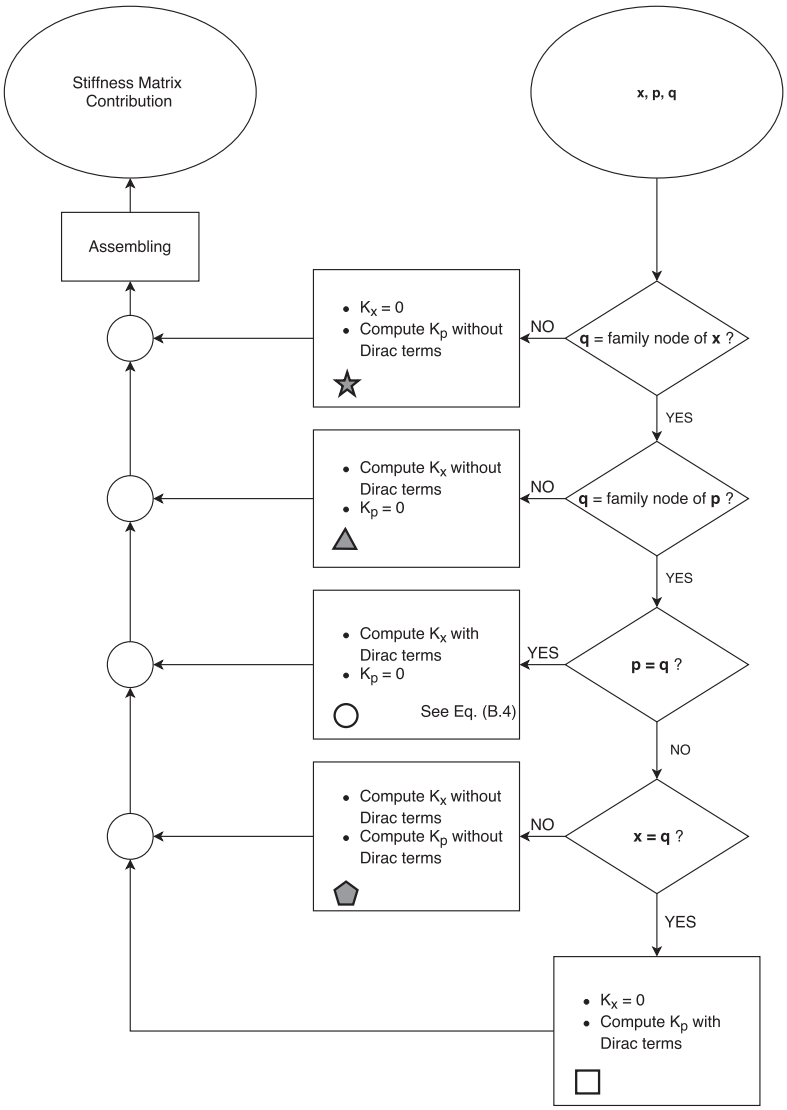


Figure 3.15: Detailed stiffness matrix contribution algorithm. The symbols involved in each choice are graphically presented in Figure 3.16.

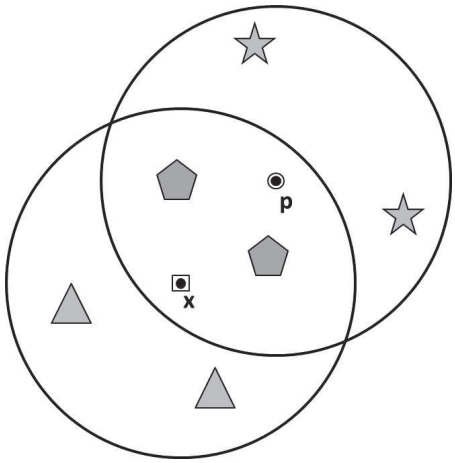
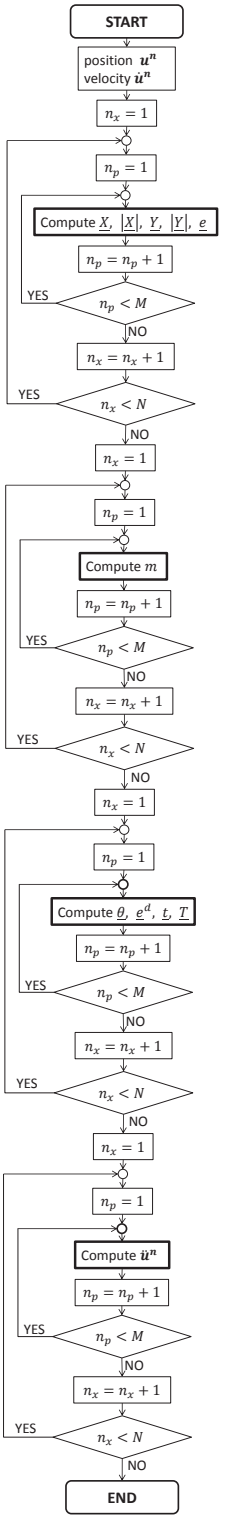


Figure 3.16: Symbols of Figure 3.15 explaining the different sets of **q** nodes which leads to different terms in the formulas.

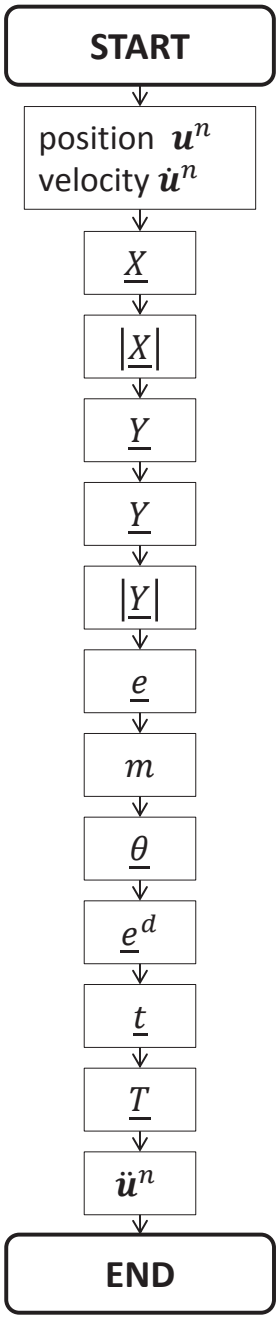
After computing the global stiffness matrix and building the external force vector, they are reduced [134] and the system shown in Eq. (3.1) is solved by the biconjugate gradients stabilized method built-in function of Matlab®.

The incremental reduced displacement vector is then integrated with the *a priori* known displacements. Therefore, the out of balance force is computed and if it is smaller than a tolerance value, the next load step can be added, otherwise the updated stiffness matrix related to the new configuration is computed and a loop employing a Newton type convergence scheme is used until the tolerance is reached (equilibrium configuration) and the next load step is subsequently applied.

The other parameters needed for the algorithms are optimized following what was done for the BBP code. For example, Figure 3.17a shows the original code with quantities requiring two nested loops for being computed. Thanks to the code optimization, the number of nested loops as well as the computational time drastically decrease (Figure 3.17b).



(a)



(b)

Figure 3.17: Flow chart of the initial algorithm for the dynamic code: (a) first algorithm, (b) new algorithm.

Chapter 4

Comparison with Analytical Results

Analytical results can be compared for linear elastic cases to the classical continuum theory exact results which may be obtained by finite element software with very high accuracy. The simulations of this chapter are all performed through a static solver.

4.1 BBP Analysis Results

Initially the proposed static solution strategy has been applied to three classic linear elastic cases, shown in Figure 4.1, to verify and illustrate the performance of Peridynamics applied to linear problems. The beam dimensions are $L_x = 30mm$ and $L_y = 1.5mm$. The material properties are $E = 70GPa$ and $\nu = 0.33$. The solutions for the plane stress conditions have been obtained assuming a linear elastic behavior for the material. The results of the static analysis have been compared to those obtained for the same problem solved with the finite element method (FEM) using the MSC.Nastran solver. The sensitivity of the peridynamic solution with respect to the horizon and the m-ratio has been investigated for case 1.

Figure 4.1: Case 1: cantilever beam with tip force; case 2: cantilever beam with tip moment; case 3: simply supported beam with uniformly distributed load.

The classical case results obtained with the static solution of Peridynamics have

been reported in Figures 4.3, 4.4 and 4.5 together with the relevant results obtained with finite element method: the comparison was made between peridynamic and FEM model both with 20x400 nodes.

As previously mentioned, the forces have to be applied as force densities on one or more boundary layers. In particular, the tip moment of case 2 is obtained by a linear distribution of forces (see Figure 4.2).

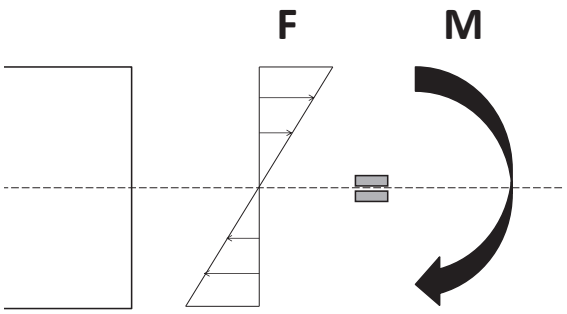


Figure 4.2: The moment is applied as a linear distribution of force density along the cross sectional area of the tip of the moment.

Figures 4.6, 4.7 and 4.8 show the relative error of the peridynamic solution with respect to that obtained with FEM.

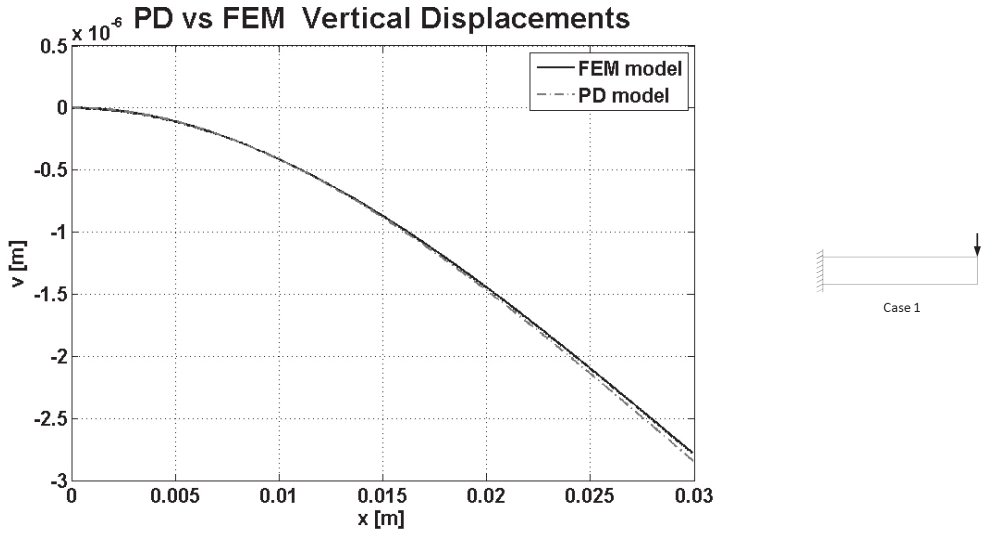


Figure 4.3: Case 1 of Figure 4.1: comparison between Peridynamics and FEM results.

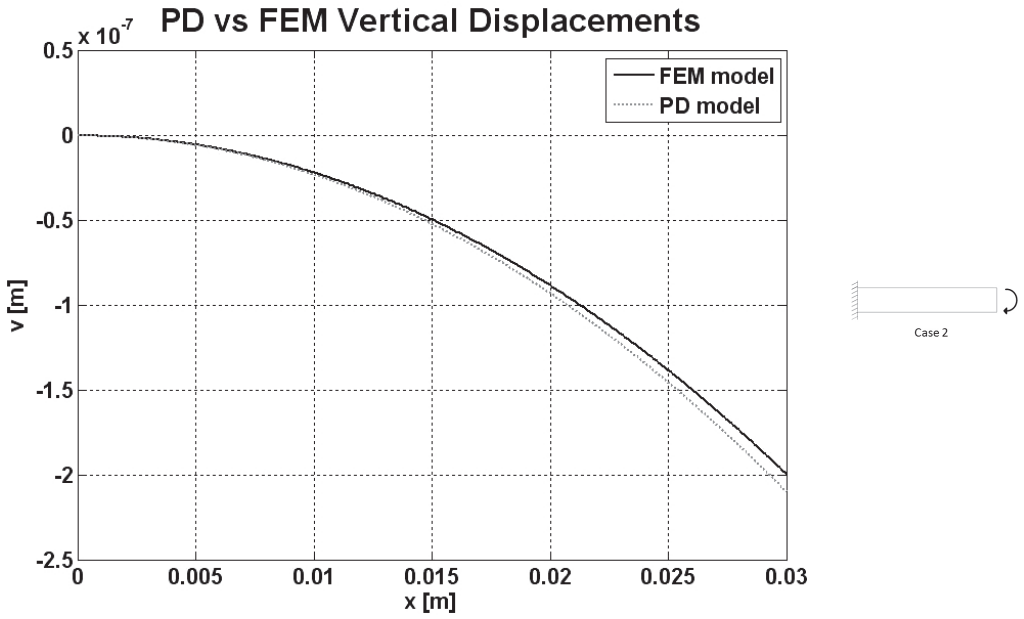


Figure 4.4: Case 2 of Figure 4.1: comparison between Peridynamics and FEM results.

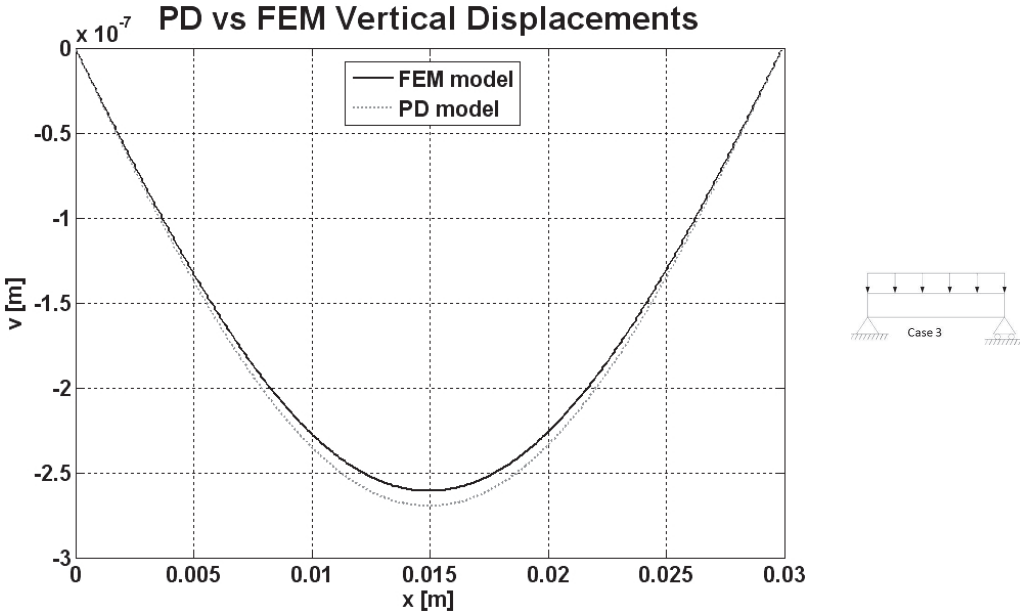


Figure 4.5: Case 3 of Figure 4.1: comparison between Peridynamics and FEM results.

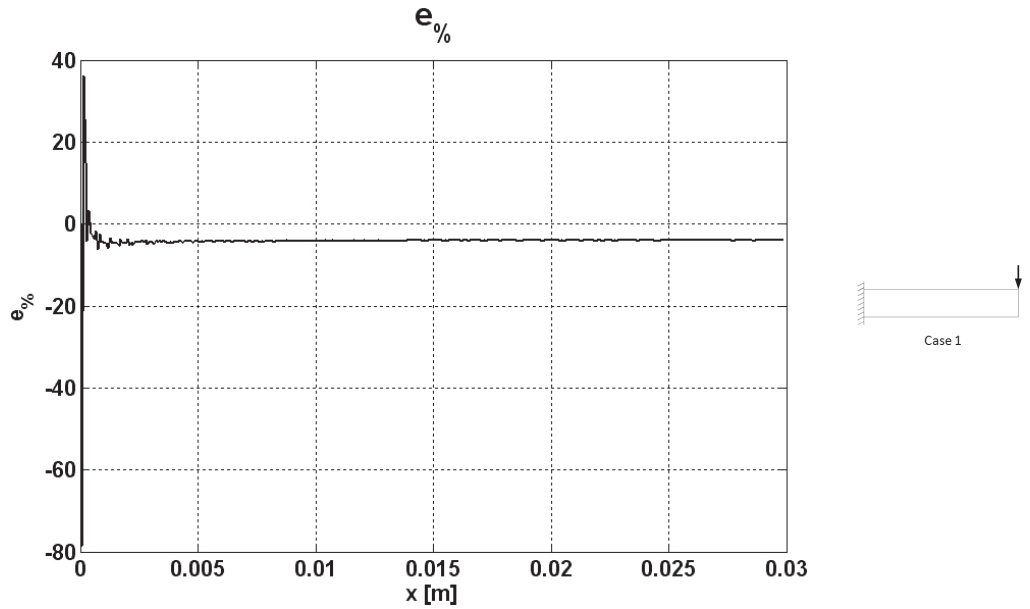


Figure 4.6: Case 1: relative error for the displacements of peridynamic versus FEM results.

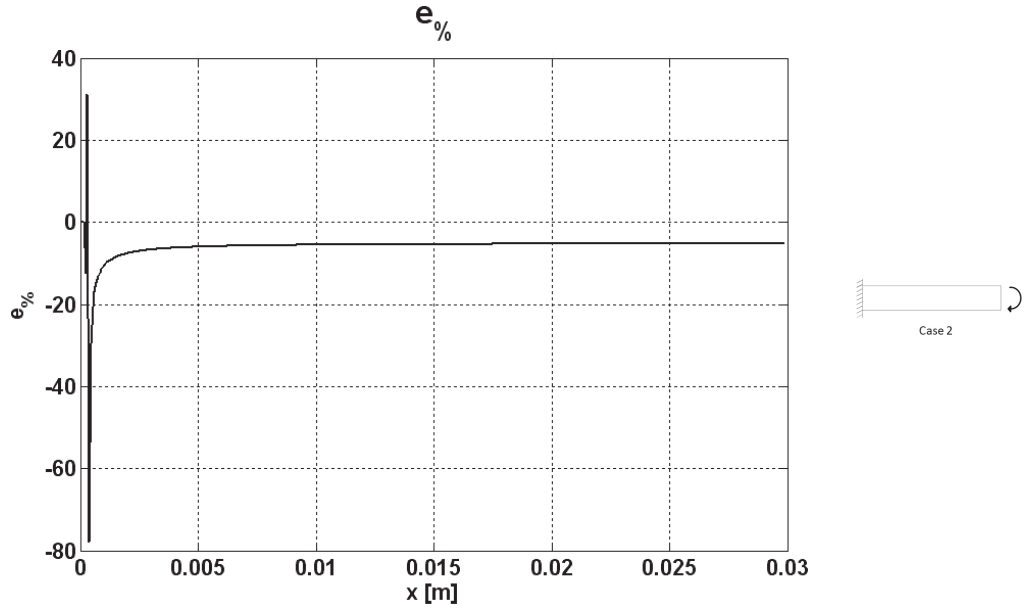


Figure 4.7: Case 2: relative error for the displacements of peridynamic versus FEM results.

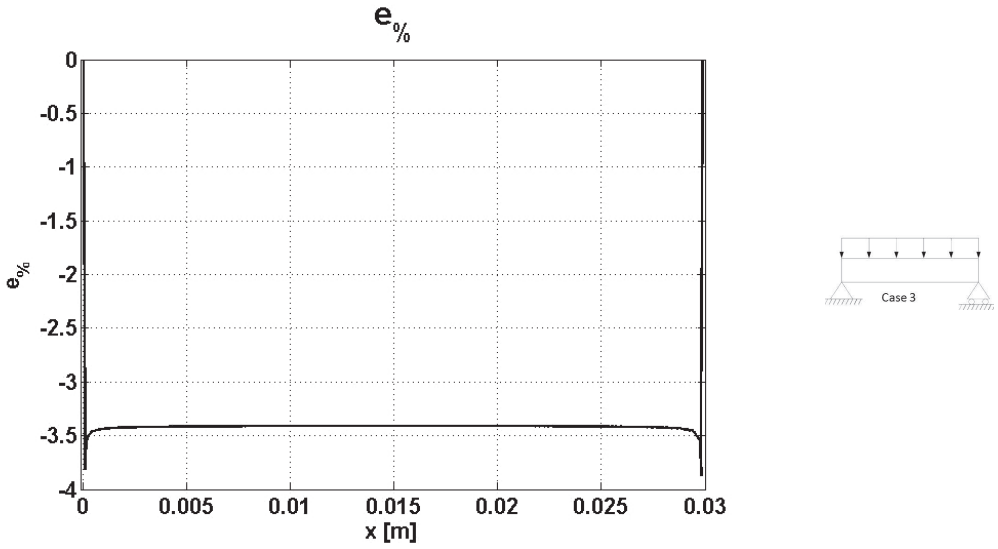


Figure 4.8: Case 3: relative error for the displacements of peridynamic versus FEM results.

The peridynamic solution is overall accurate, except in the zone of the structure where the boundary conditions are applied, where displacements are rather small and small absolute errors may increase in terms of relative errors. If we disregard those zones the maximum error for case 1 is about 4.5%, for case 2 is about 5.5% and for case 3 the maximum relative error is less than 3.5%. The observed error where the boundary conditions are applied is due to the way the peridynamic theory deals with external loads and constraints [1, 135]: they are applied within a horizon distance. The error along the span of the beam is due as well to the skin softening effect [68] related to the material points that are less than one horizon away from the system edges: this effect can be controlled by reducing the horizon. The error depends, in fact, on the horizon and, since the m-ratio is constant, on the grid size as shown Figure 4.9: the absolute error of the tip displacement for case 1 is shown as a function of the uniform grid size Δx and the corresponding percentage error is presented. The studied peridynamic models have the dimensions given in Table 4.1:

Δx	num_x	num_y
$0.3 \cdot 10^{-4}$	100	5
$0.15 \cdot 10^{-4}$	200	10
$0.1 \cdot 10^{-4}$	300	15
$0.075 \cdot 10^{-4}$	400	20

Table 4.1: Grid sizes and numbers of nodes along x and y directions used in the sensitivity analysis.

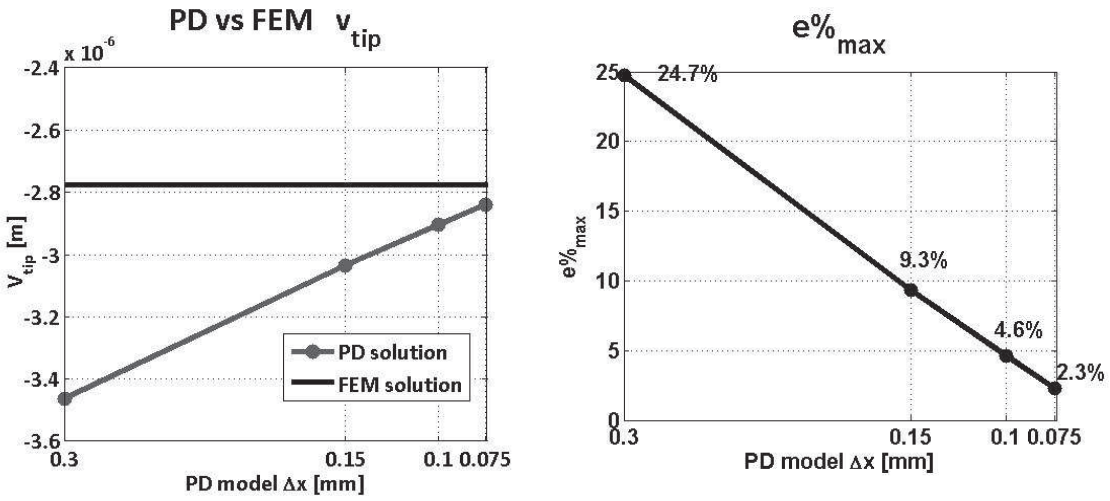


Figure 4.9: Absolute error for the tip displacements of case 1 versus the peridynamic horizon (δ-convergence) and the corresponding percentage error.

The proposed static approach for the peridynamic analysis is able to capture the effective elastic behavior of the studied systems. The numerical accuracy of the results depends on the horizon: decreasing this parameter means increasing the total number of nodes of the model, since the grid size decreases as well. In addition, it should be pointed out that a large number of nodes reduces (but not eliminates) the skin softening effect and allows a better comparison between the boundary conditions of the peridynamic models and those of the finite element models. A grid size sensitivity study has been conducted also on the FEM solu-

tion of case 1 and its results are shown in Figure 4.10, next to the corresponding Peridynamics results. The comparison between the curves of Figure 4.10 suggests that the convergence of the FEM solution is faster than that of the peridynamic solution and that increasing the number of nodes for a peridynamic analysis makes the model more rigid, whereas for a finite element analysis the model becomes softer if the mesh is refined.

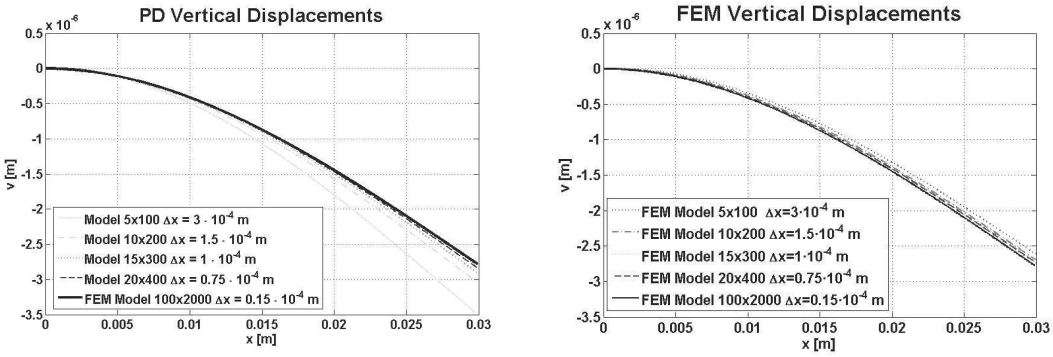


Figure 4.10: Sensitivity study on the horizon for the case 1 using FEM results.

In all peridynamic simulations $\delta = 3\Delta x$. Figure 4.11 shows the absolute and the percentage errors for the m-convergence: the convergence, in this case, is not to the local exact solution [130] identified by FEM results (black horizontal line), but to the nonlocal solution for the employed horizon ($\delta = 0.6mm$), so readers should pay attention to the fact that this error is not expected to go to zero even with an infinite m-ratio. Figure 4.12 presents a comparison between peridynamic and FEM results.

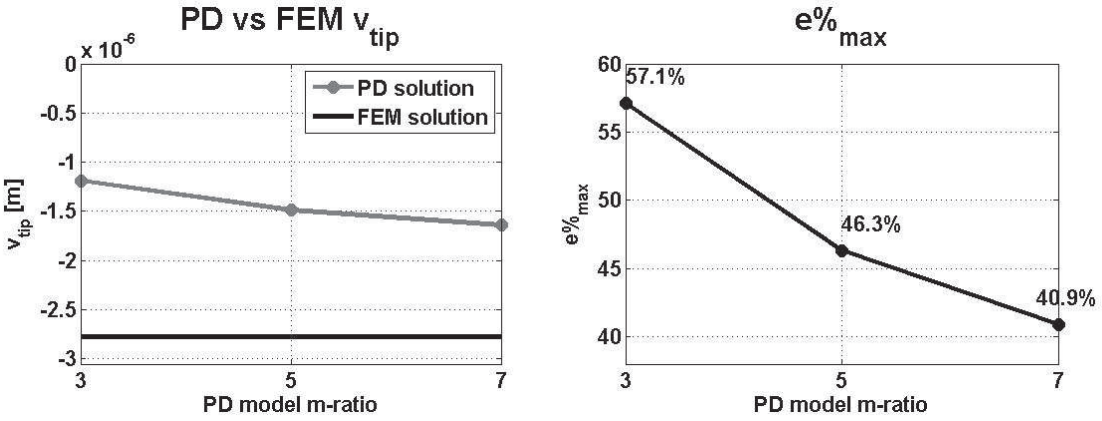


Figure 4.11: Absolute error for the tip displacements of case 1 versus the peridynamic horizon (m-convergence) and the corresponding percentage error.

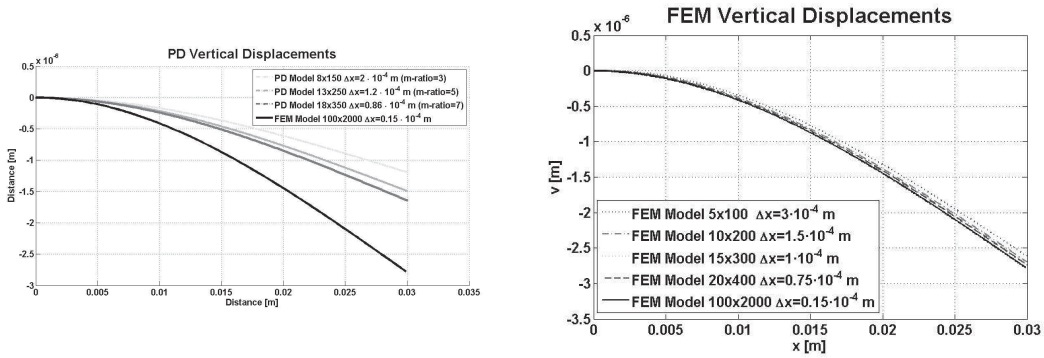


Figure 4.12: Sensitivity study on m-ratio for the case 1 using FEM results.

4.2 SBP Analysis Results

SBP results are compared to FEM results not only in terms of displacement fields, but also in terms of recovered material properties.

Since the BBP model (see [2]) is limited to a fixed Poisson's ratio, it is interesting to see how the SBP model is able to reproduce the behavior of an elastic material with any Poisson's ratio. For this purpose, Poisson's ratios varying in the $[0.1 - 0.45]$ range and a constant Young's modulus of $71GPa$ are assumed as input parameters of the peridynamic model for simulations employing the linearized formula.

In addition, an analysis on influence functions is carried out to see their influence on displacement fields and on errors.

The example is related to the computation of the peridynamic solution of a 2D plate under tension in plane stress conditions (see Figure 4.13).

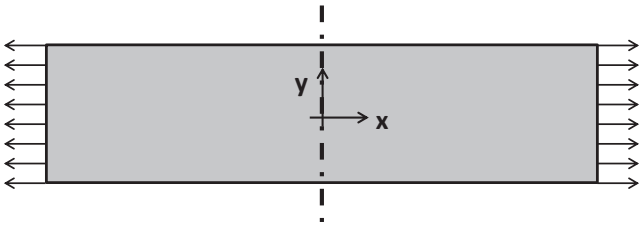


Figure 4.13: Simulated system for the recovery of the mechanical properties (Young's modulus and Poisson's ratio).

The simulated structure is a rectangular plate under tension as shown in Figure 4.13. The material properties are: density $\rho = 2780\,kg/m^3$ and Young's modulus

$E = 71\,GPa$. The Poisson's ratio is varied, the plate dimensions are $8 \times 2mm^2$. All simulations are carried out with the plane stress model and the forces are applied to one layer of nodes.

4.2.1 The effective E and ν modeled by SBP

One of the issues is related to the recovery of properties. How to recover Young's modulus and Poisson's ratio when the body is discretized into a grid of points instead of mesh of elements?

The effective elastic properties of the plate are evaluated by using the algorithms described later in this section which makes use of the computed displacement values. Static analyses have been performed varying the main parameters of the Peridynamic discretization, the maximum length of the nonlocal interaction δ and the number of nodes inside the horizon region given by the spatial discretization used.

Young's modulus and Poisson's ratio can be evaluated *a posteriori* in several ways; three strategies have been adopted considering different sets of nodes for the computation:

- the first method, called *double node method*, considers the central node of the plate, which is fixed for the symmetry of the problem and two nodes of its family are considered for the displacements along the two axes;

- the second method, called *internal node method*, consists in averaging the deformation of all the internal nodes, which are considered to be those which are at a distance 2δ from the border;

- the last method, called *tensor method*, takes into consideration the deformation gradient at the central node of the plate and the definition of the collapsed peridynamic stress tensor (see [5]).

A detailed description of the three procedures are given in the following sections.

Double node method

Consider the two nodes (node number 1 and node number 2 in Figure 4.14) of the family nodes of the central node (the red one in Figure 4.14). For the symmetry of the problem the central node is fixed along the two axes, node number 1 can only move in horizontal direction and in a similar way node number 2 can only move in the vertical direction.

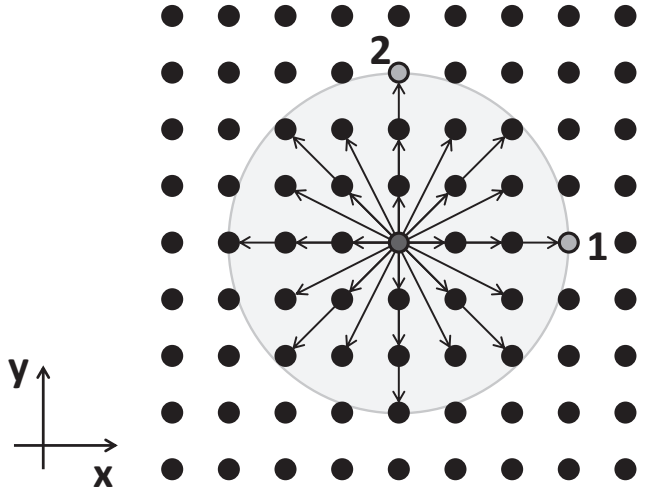


Figure 4.14: Family nodes used for computing the strains: with respect to the central node, node 1 is used for computing ε_{xx}, while node 2 is used for computing ε_{yy}.

The engineering strain ε_{xx} is computed as

$$\varepsilon_{xx} = \frac{u_{x,1} - u_{x,center}}{\Delta L_{x,1}} \tag{4.1}$$

where $u_{x,1}$ is the horizontal displacement of point 1, $u_{x,center}$ is the horizontal displacement of the central point and $\Delta L_{x,1}$ is their initial horizontal distance. The resultant reaction force F_x at the ends of the bar is computed, σ_{xx} is estimated as

$$\sigma_{xx} = \frac{F_x}{A} \tag{4.2}$$

where A is the cross section of the bar, Young's modulus E is computed as

$$E = \frac{\sigma_{xx}}{\varepsilon_{xx}} \tag{4.3}$$

As for the Poisson's ratio, the engineering strain ε_{yy} is computed as

$$\varepsilon_{yy} = \frac{u_{y,2} - u_{y,center}}{\Delta L_{y,2}} \tag{4.4}$$

where $u_{y,2}$ is the vertical displacement of point 2, $u_{y,center}$ is the vertical displacement of the central point, $\Delta L_{y,2}$ is their initial vertical distance and the Poisson's ratio is computed as

$$\nu = -\frac{\varepsilon_{yy}}{\varepsilon_{xx}} \tag{4.5}$$

Internal node method

Another way to compute the material properties is to average them for all the nodes which are in the internal part of the plate to avoid introducing the error related to the surface effect. The distance of these nodes from the border is at least twice the horizon, since the simulations analyzing the surface effect show that it affects in a less relevant way these nodes (Figure 4.15).

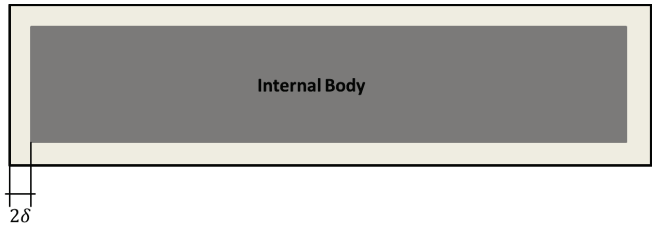


Figure 4.15: The set of nodes for the computation of material properties is composed of those contained in the internal body.

So, consider that the number of nodes of the internal part of the body is N^-, the strain is computed in the x direction as the horizontal displacement from

$$\varepsilon_{xx} = \frac{1}{N^-}\sum_{i=1}^{N^-}\frac{u_{x,i} - u_{x,center}}{\Delta L_{x,i}} \tag{4.6}$$

while in the y direction is computed as

$$\varepsilon_{yy} = \frac{1}{N^-}\sum_{i=1}^{N^-}\frac{u_{y,i} - u_{y,center}}{\Delta L_{y,i}} \tag{4.7}$$

Their values are employed in Eqs. (4.3) and (4.5) to get the properties.

Tensor method

Reference [5] shows that SBP reproduces the classical mechanical local model highlighting that if the motion and the model are sufficiently smooth, i.e. integrable on the whole body $\mathcal{B}$ [120] as in the present case, the peridynamic stress tensor

converges as $\delta \to 0$ to the Piola-Kirchoff stress tensor in the classical sense of a stress tensor (i.e. a stress tensor that is a function of the local displacement gradient tensor). This result comes from the equality of the peridynamic equation of motion (Eq. (1.80)), to the following partial differential equation:

$$\rho(\mathbf{x})\ddot{\mathbf{y}}(\mathbf{x},t) = \nabla \cdot \boldsymbol{\nu}(\mathbf{x},t) + \mathbf{b}(\mathbf{x},t) \tag{4.8}$$

where $\nabla\cdot$ identifies the divergence operator and $\boldsymbol{\nu}(\mathbf{x},t)$ is the peridynamic stress tensor field. The Peridynamic stress tensor [63] is:

$$\boldsymbol{\nu}(\mathbf{x},t) = \int_{\mathcal{S}} \int_0^{\delta} \int_0^{\delta} (y+z)^2 \, \underline{T}\left[\mathbf{x} - z\mathbf{m}, t\right] \langle (y+z)\,\mathbf{m} \rangle \otimes \mathbf{m} \, dz \, dy \, d\Omega_m \tag{4.9}$$

where $\mathcal{S}$ is the unit sphere, $d\Omega_m$ is a differential solid angle in the direction of any unit vector $\mathbf{m}$, which is the dummy variable of integration in the outer integral (Figure 4.16).

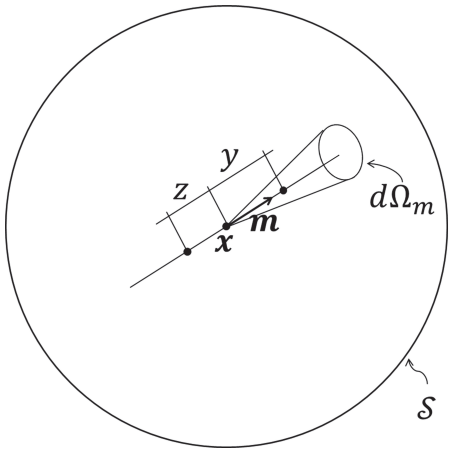


Figure 4.16: Variables used for the computation of the peridynamic stress tensor.

The peridynamic stress tensor as the horizon tends to zero tends to the collapsed peridynamic stress tensor which is expressed as [5]:

$$\lim_{\delta \to 0} \nu(\mathbf{x},t) = \int_{\mathcal{H}} \underline{T}\left(\mathbf{F}\left(\mathbf{x},t\right) \underline{X}, \mathbf{x}\right) \langle \boldsymbol{\xi} \rangle \otimes \boldsymbol{\xi} \, dV_{\xi} \tag{4.10}$$

where $\mathbf{F}\left(\mathbf{x},t\right) \underline{X}$ is the deformation gradient tensor evaluated in the node located at $\mathbf{x}$ and applied to the vector reference state. It is expressed as

$$\mathbf{F}\left(\mathbf{x},t\right) = \left(\int_{\mathcal{H}} \underline{\omega} \langle \boldsymbol{\xi} \rangle \, \underline{Y} \langle \boldsymbol{\xi} \rangle \otimes \boldsymbol{\xi} \, dV_{\xi} \right) \mathbf{K}^{-1}\left(\mathbf{x}\right) \tag{4.11}$$

where $\mathbf{K}(\mathbf{x})$ is the shape tensor defined by

$$\mathbf{K}(\mathbf{x}) = \int_{\mathcal{H}} \underline{\omega}\langle\boldsymbol{\xi}\rangle\, \boldsymbol{\xi} \otimes \boldsymbol{\xi}\, dV_{\xi} \tag{4.12}$$

and it includes the influence function $\underline{\omega}\langle\boldsymbol{\xi}\rangle$. The authors in [5] also mention that this stress tensor is equivalent to the Piola-Kirchhoff stress tensor of the classical elasticity theory, so, if we assumed that the displacement field is continuously differentiable, the corresponding Cauchy stress tensor is defined as

$$\boldsymbol{\sigma} = J^{-1}\, \boldsymbol{\nu}(\mathbf{x}, t)\, \mathbf{F}^{T}(\mathbf{x}, t) \qquad J = det\,(\mathbf{F}(\mathbf{x}, t)) \tag{4.13}$$

in which it is assumed that $J \neq 0$.

This nonlocal peridynamic strain tensor can be evaluated [63], considering the definition of the deformation gradient, as

$$\begin{aligned} \boldsymbol{\varepsilon}(\mathbf{x}, t) &= \frac{1}{2}\left(\mathbf{F}(\mathbf{x}, t) + \mathbf{F}^{T}(\mathbf{x}, t)\right) - I \\ &= \frac{1}{2}\left(\int_{\mathcal{H}} \underline{\omega}\langle\boldsymbol{\xi}\rangle\, \underline{Y}\langle\boldsymbol{\xi}\rangle \otimes \boldsymbol{\xi} + \underline{\omega}\langle\boldsymbol{\xi}\rangle\, \boldsymbol{\xi} \otimes \underline{Y}\langle\boldsymbol{\xi}\rangle\, dV_{\xi}\right) \mathbf{K}^{-1}(\mathbf{x}) - I \end{aligned} \tag{4.14}$$

where $\mathbf{K}^{-1}(\mathbf{x})$ is the shape tensor previously defined. So the four components of the strain tensor ($\varepsilon_{xx}, \varepsilon_{xy}, \varepsilon_{yx}, \varepsilon_{yy}$ in 2D) can be computed.

Using Hooke's Law for 2D plane stress, we can write:

$$\begin{aligned} \varepsilon_{xx} &= \frac{1}{E}\left(\sigma_{xx} - \nu\sigma_{yy}\right) \\ \varepsilon_{yy} &= \frac{1}{E}\left(\sigma_{yy} - \nu\sigma_{xx}\right) \end{aligned} \tag{4.15}$$

which can be rearranged in the unknowns E and ν as following

$$\begin{bmatrix} \sigma_{yy} & \varepsilon_{xx} \\ \sigma_{xx} & \varepsilon_{yy} \end{bmatrix} \begin{Bmatrix} \nu \\ E \end{Bmatrix} = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \end{Bmatrix} \tag{4.16}$$

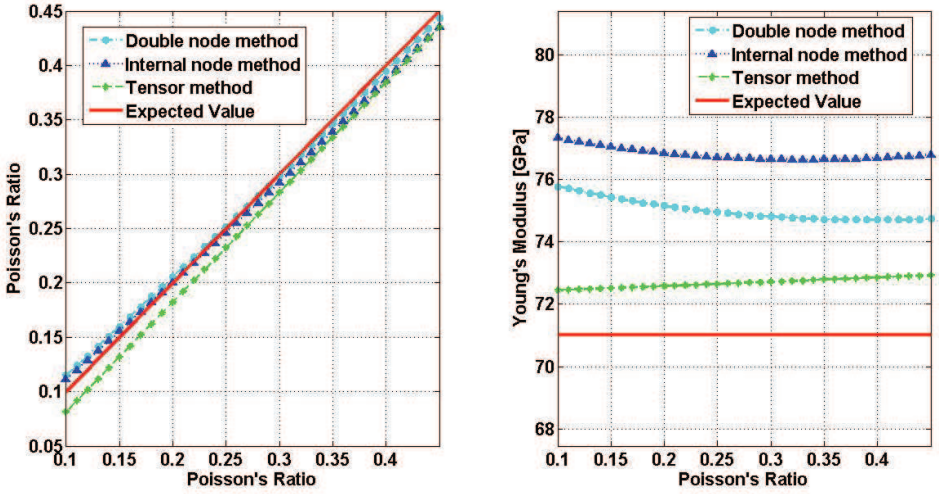
from which

$$\begin{Bmatrix} \nu \\ E \end{Bmatrix} = \begin{bmatrix} \sigma_{yy} & \varepsilon_{xx} \\ \sigma_{xx} & \varepsilon_{yy} \end{bmatrix}^{-1} \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \end{Bmatrix} \tag{4.17}$$

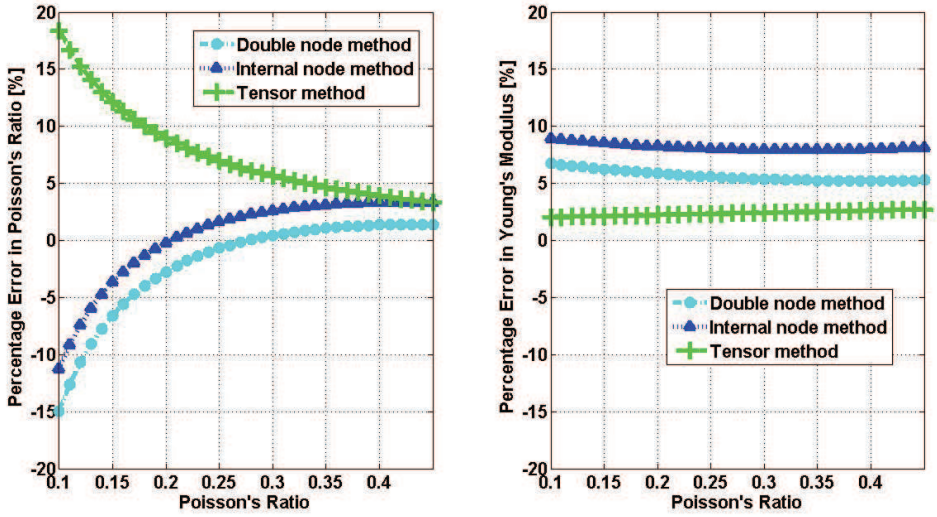
These methods are used to calculate the local values of Young's modulus and Poisson's ratio in this section. The relative results are shown in the following section.

4.2.2 Linear Static Analyses

Every case study is solved with the developed peridynamic solver. The value of ν and E are computed *a posteriori* using the displacement data as explained in section 4.2.1. Then, the computed values are compared to the relevant input values and a convergence study is carried out by varying the value of the m-ratio and the horizon. The influence function is fixed for all the simulation in this section. For some simulations, Poisson's ratio is fixed; in these cases, it is chosen to be equal to 0.1, for two main reasons: it is different from the fixed Poisson's ratio of the BBP theory for 2D cases and, comparing the results to differing Poisson's ratios (in Figure 4.17 Poisson's ratio is changed from 0.1 to 0.45), it's possible to see a slight increase in the error in Poisson's ratio estimation when Poisson's ratio decreases.



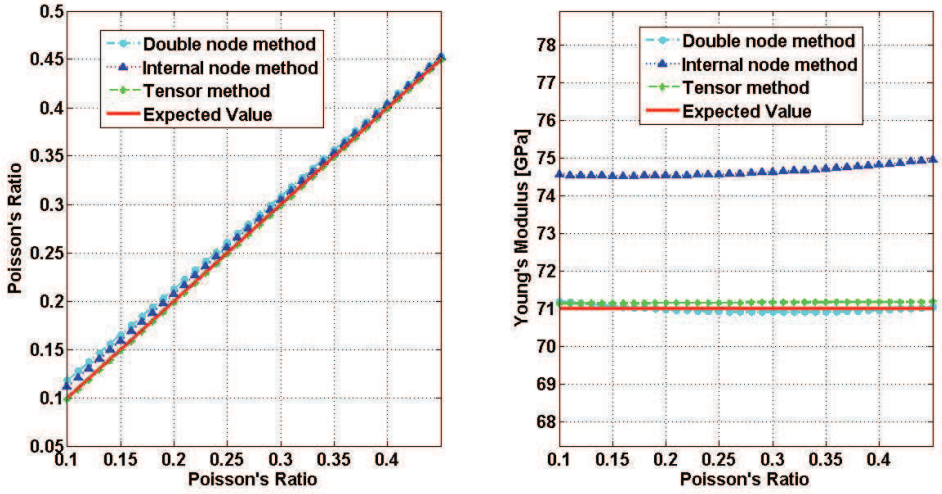
(a)



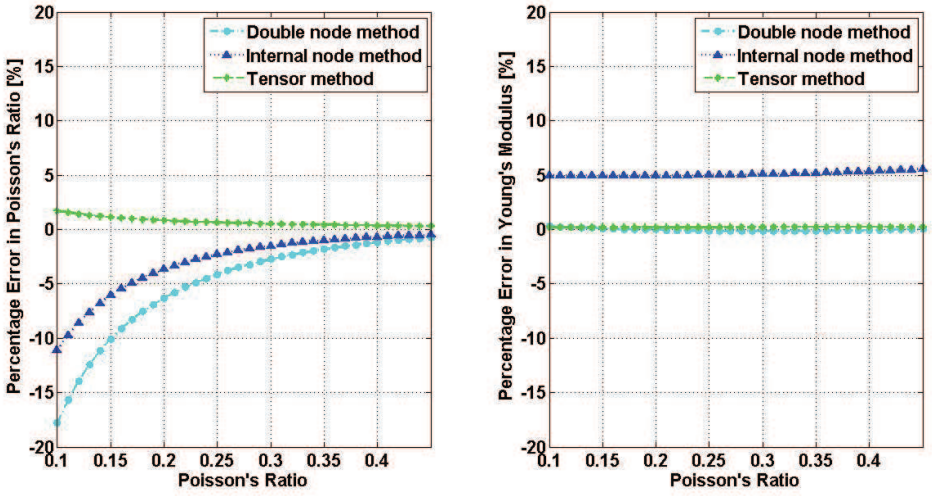
(b)

Figure 4.17: Values of Poisson's ratio and Young's modulus computed with the three methods are plotted and compared with the expected values (continuous lines) in (a) while the percentage errors are shown in (b). For these simulations: m-ratio=3 and $\delta = 0.3mm$.

As it is expected these errors decrease as soon as m-ratio increases (this can be seen from Figure 4.18 in which the results are obtained with m-ratio=5 and can be compared to those of Figure 4.17). For small values of Poisson's ratio, the tensor method shows the best results.



(a)

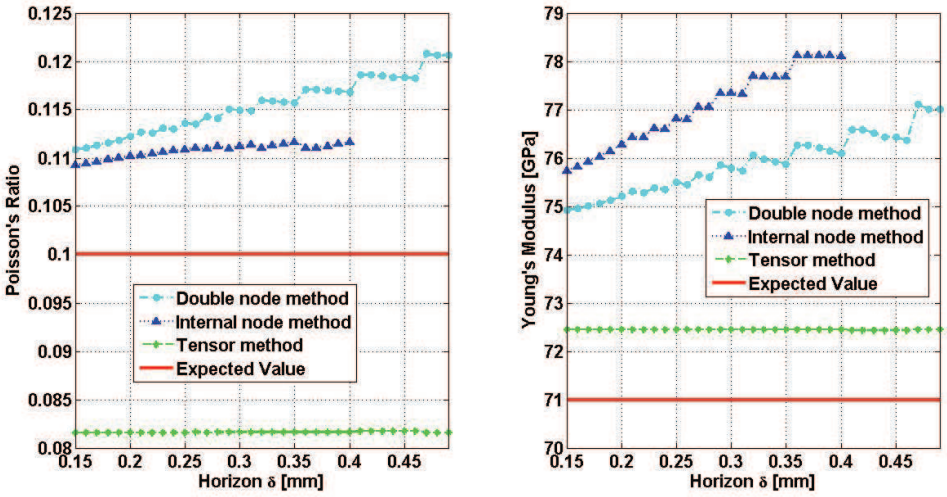


(b)

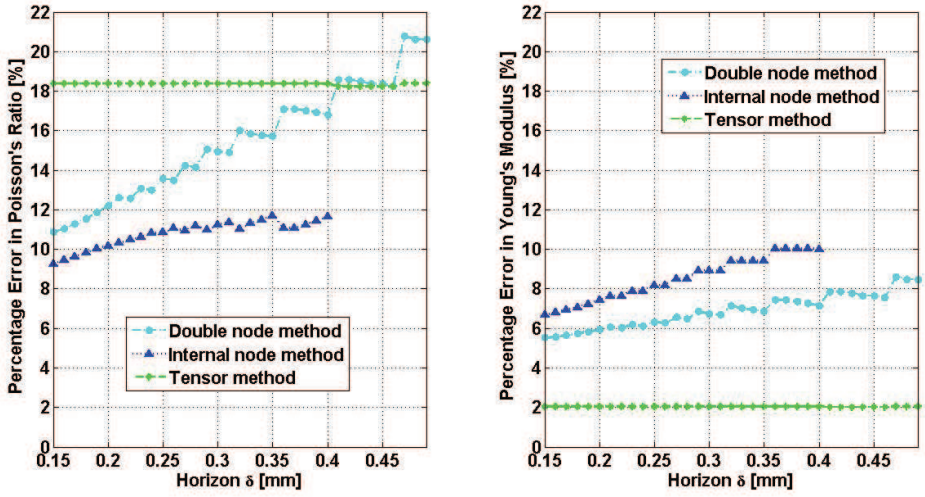
Figure 4.18: Values of Poisson's ratio and Young's modulus computed with the three methods are plotted and compared with the expected values (continuous lines) in (a) while the percentage errors are shown in (b). For these simulations: m-ratio=5 and $\delta = 0.3mm$.

The next results show the δ-convergence [130] of the solution in linear elasticity (Figure 4.19): if the m-ratio is kept constant, the solution converges (it's more appropriate to write δ-converges) to the theoretical value for the double node method and the internal node method, while, unexpectedly, the solution does not change significantly for the tensor method (it δ-converges with a horizon as big as 1/4 of the smaller dimension of the plate). The single node and the internal

node computations are affected by large errors, which decrease as the horizon gets smaller. The error for the tensor method can be reduced only if m-ratio increases, so the results are relative to the worst case scenario of the entire range.



(a)

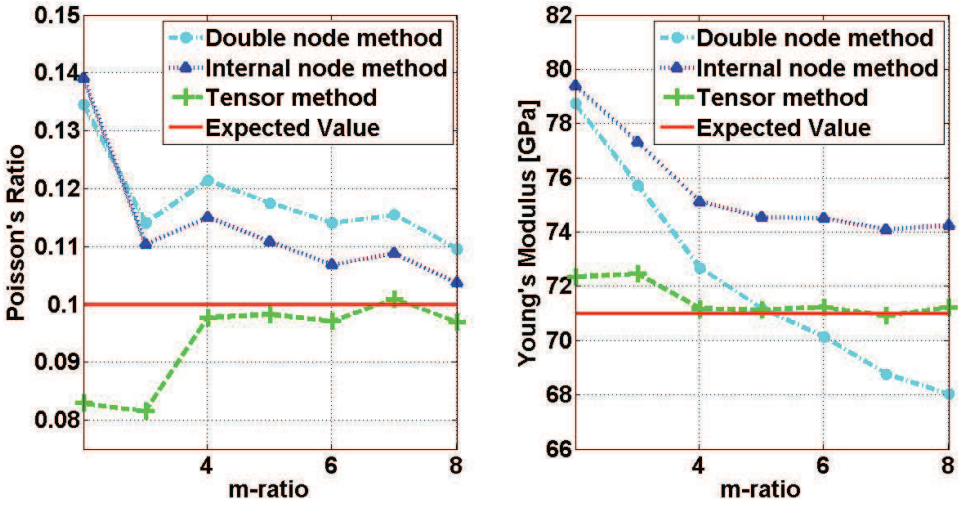


(b)

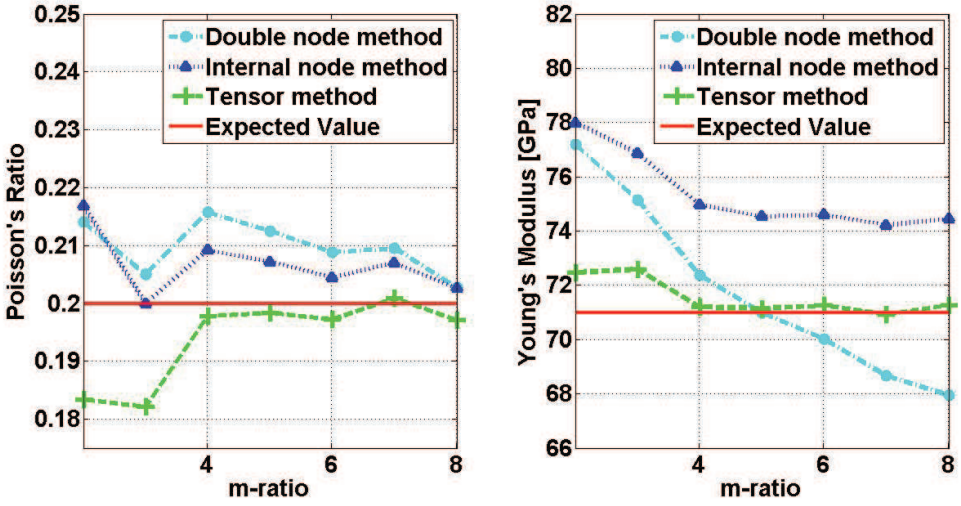
Figure 4.19: δ-convergence: values of Poisson's ratio and Young's modulus computed with the three methods are compared to the theoretical values in (a), while the percentage errors are presented in (b). The simulations are carried out with m-ratio=3.

Similarly, with a fixed horizon, as the m-ratio increases, the results tend to the theoretical ones. As it is possible to see from Figures 4.20 and 4.21, when the m-convergence is analyzed, the results tend to the expected ones as m-ratio

increases, even though not monotonically: in particular, this is always verified for the tensor method, while, for the double node and the internal node methods, this is not evident (in fact, Figures 4.22 and 4.23 show that the errors of the last two methods are generally higher than the tensor method error).

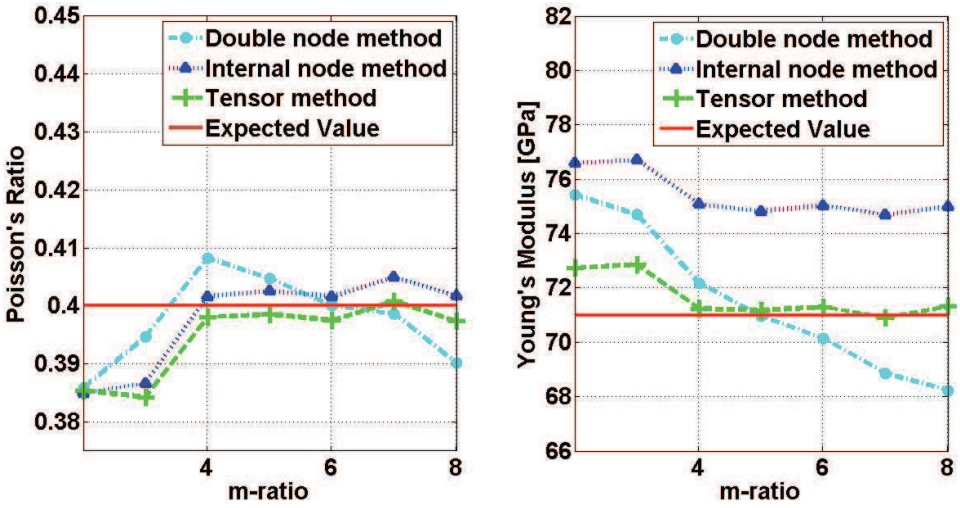


(a)

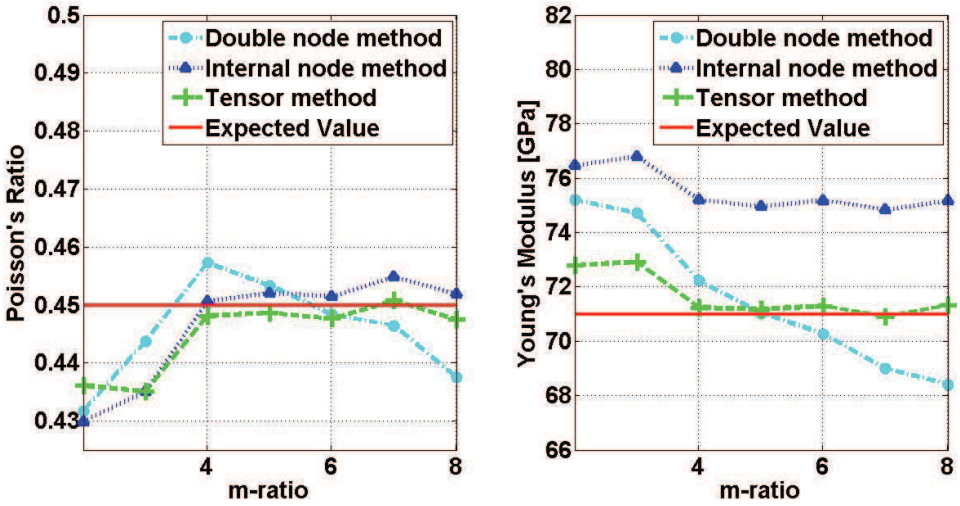


(b)

Figure 4.20: m-convergence simulations for different value of Poisson's ratio ($\nu =$ 0.1 (a), 0.2(b)) with the comparison among the three methods. The theoretical values are identified by continuous lines. For these simulations, $\delta = 0.3mm$.

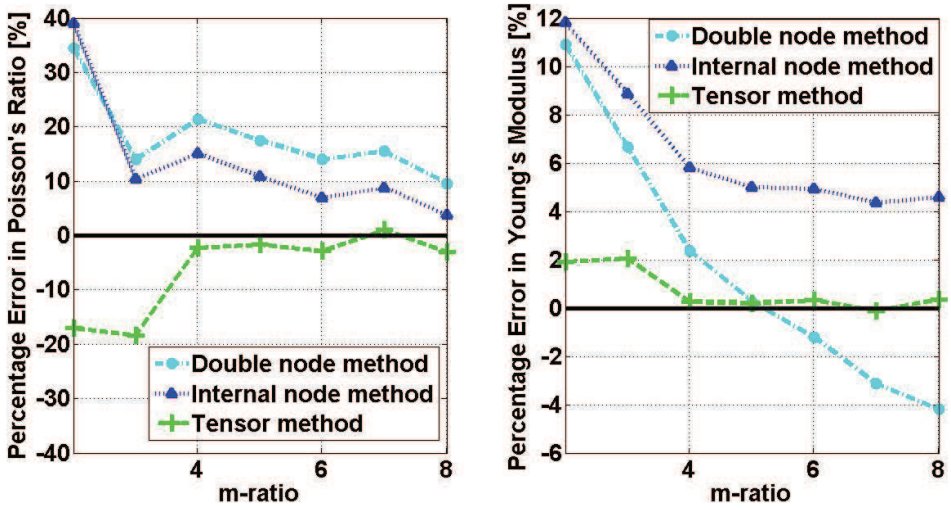


(a)

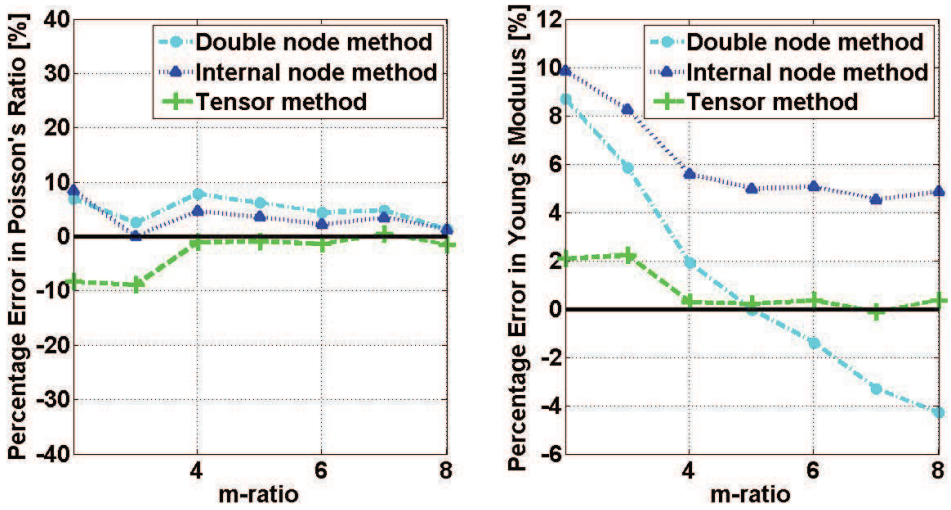


(b)

Figure 4.21: m-convergence simulations for different value of Poisson's ratio ($\nu =$ 0.4(a), 0.45(b)) with the comparison among the three methods. The theoretical values are identified by continuous lines. For these simulations, $\delta = 0.3mm$.

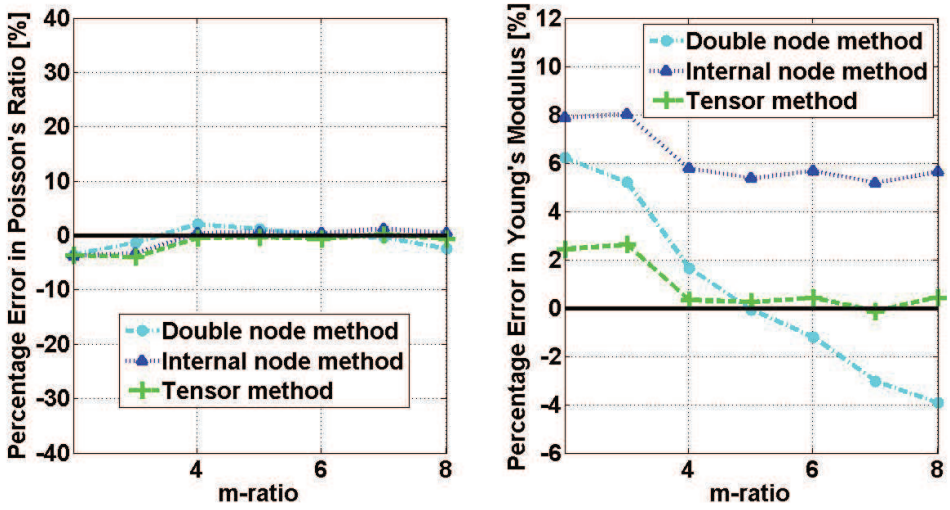


(a)

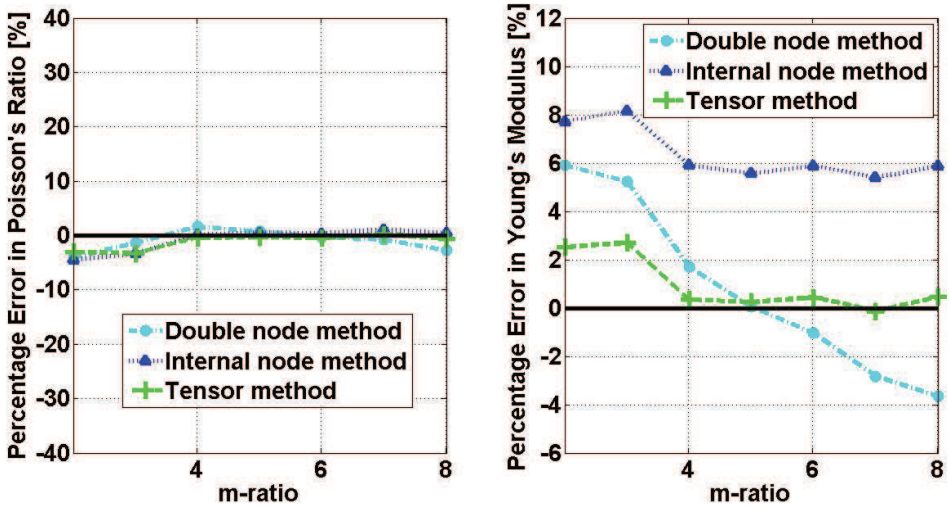


(b)

Figure 4.22: m-convergence simulation percentage errors for different value of Poisson's ratio ($\nu = 0.1$ (a), 0.2(b)) with the comparison among the three methods. For these simulations, $\delta = 0.3mm$.



(a)



(b)

Figure 4.23: m-convergence simulation percentage errors for different value of Poisson's ratio ($\nu = 0.4$(a), 0.45(b)) with the comparison among the three methods. For these simulations, $\delta = 0.3mm$.

The displacement field also has a good correspondence to the one computed by FEM software MSC.NASTRAN, as it can be seen from Figures 4.24, 4.25 and 4.26: the simulations are carried out with the same grid spacing of $\Delta x = 0.05mm$ so that each node in the FEM model corresponds to a node in the peridynamic model; for the peridynamic simulation, the parameters are chosen to be m-ratio= 6 and $\delta = 0.3mm$. The main differences in the displacement field are near the boundary: this is due to the surface effect, since in these simulations no technique is adopted

to take it into account.

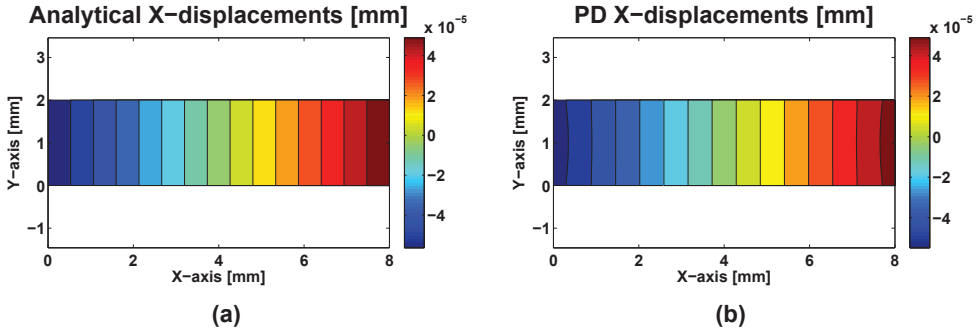


Figure 4.24: Comparison of the horizontal displacements computed with (a) the analytical solution and (b) Peridynamics.

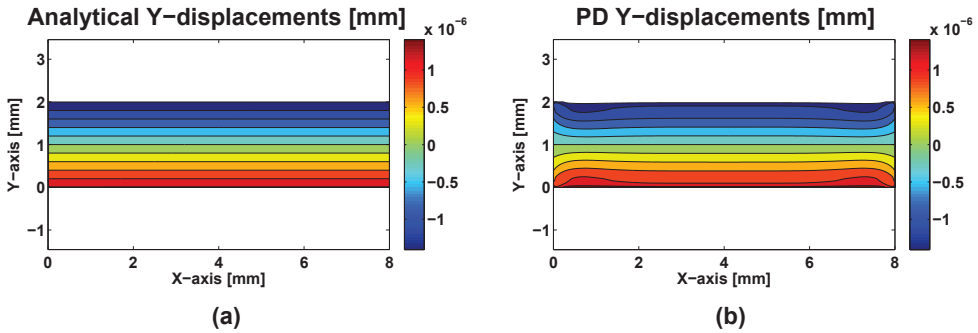


Figure 4.25: Comparison of the vertical displacements computed with (a) the analytical solution and (b) Peridynamics.

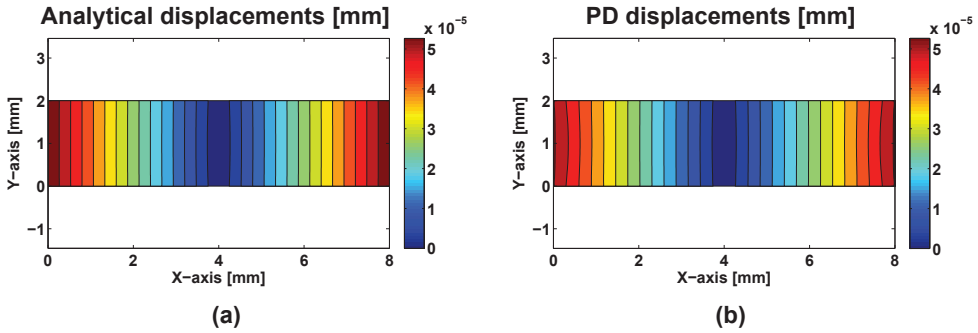


Figure 4.26: Comparison of the magnitude of the displacement vector field computed with (a) the analytical solution and (b) Peridynamics.

4.2.3 Influence Functions

The simulations in the previous sections are all carried out with a constant influence function, $\underline{\omega} = 1$, in Eq. (1.92). Spherical influence functions, which means that their value depends only on the length of the bond and not on the directionality ($\underline{\omega} : |\boldsymbol{\xi}| \in [0, \delta] \mapsto \mathbb{R}$), have been studied. The most common influence function to be implemented is the constant (it requires less computational resources) and the Gaussian; a family of influence functions was simulated in [122] using a dynamic solver to see if any non-negative influence function is acceptable for an accurate numerical solution, as it is supposed to be theoretically [2, 122]: the non-negativity limitation is related to the fact that the influence function is a weight for the bond interaction in the overall computation. In this section, several influence functions are considered to see the dependence of the static solution on them: the considered influence functions are

- the constant $\underline{\omega} = 1$,
- the Gaussian $\underline{\omega} = e^{-\frac{|\boldsymbol{\xi}|^2}{\delta^2}}$,
- the linear proportional $\underline{\omega} = |\boldsymbol{\xi}|$,
- the conical $\underline{\omega} = \delta - |\boldsymbol{\xi}|$,
- the hyperbole $\underline{\omega} = |\boldsymbol{\xi}|^{-1}$,
- a power function with a negative exponent $\underline{\omega} = |\boldsymbol{\xi}|^{-4}$,
- a discontinuous function with a vertical asymptote at $\delta + \frac{\Delta x}{2}$, $\underline{\omega} = \left(\delta + \frac{\Delta x}{2} - |\boldsymbol{\xi}|\right)^{-1}$,
- a discontinuous function with a vertical asymptote at $\frac{\delta}{2}$, $\underline{\omega} = \frac{\delta}{2|\boldsymbol{\xi}| - \delta}$,
- a discontinuous function with a vertical asymptote at $\frac{\delta}{4}$, $\underline{\omega} = e^{\frac{1}{|\boldsymbol{\xi}| - \frac{\delta}{4}}}$

Eq. (1.80) can be arranged as a Fredholm linear integral equation of the second kind, which is shown in [120]:

$$\int_{\mathcal{B}} \mathbf{C}\left(\mathbf{x}, \mathbf{q}\right) \mathbf{u}\left(\mathbf{x}\right) dV_q - \mathbf{P}\left(\mathbf{x}\right) \mathbf{u}\left(\mathbf{x}\right) + \mathbf{b}\left(\mathbf{x}\right) = \mathbf{0} \tag{4.18}$$

where $\mathbf{C}(\mathbf{x}, \mathbf{q})$ is the integral kernel and depends on the source node located at $\mathbf{x}$ and all the other nodes in the body $\mathcal{B}$, $\mathbf{u}(\cdot)$ is the displacement vector field, $\mathbf{P}(\mathbf{x})$ is the tensor obtained by the integral of all the contributions multiplying $\mathbf{u}(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ is the prescribed body force. Since the kernel depends on the influence function, in the following figures the normalized kernel is plotted as well. It is computed from the four rows of the stiffness matrix corresponding to the degrees of freedom of two internal nodes located at $\mathbf{x}$ (rows $2n_{x-1}$ and $2n_x$, corresponding to the horizontal and vertical directions respectively) and at $\mathbf{p}$: these four rows are composed of tensors of dimension 2×2 which take into consideration the interaction between nodes (see also Appendix B). The traces of these tensors are the value of the kernel for the single bond connecting the two nodes, in a similar way of what is computed for the finite element method (see Eq. (4.19)). Consider the stiffness matrix of a spring element connecting point n_x to point n_p in the global coordinate system (see Figure 2.3): it's a 4×4 matrix computed as

$$[K]_{glo} = k_{spring} R^T \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} R = \tag{4.19}$$

$$= k_{spring} \begin{bmatrix} \begin{bmatrix} \cos^2\alpha_1 & \cos\alpha_1 \sin\alpha_1 \\ \cos\alpha_1 \sin\alpha_1 & \sin^2\alpha_1 \end{bmatrix} & \begin{bmatrix} -\cos^2\alpha_1 & -\cos\alpha_1 \sin\alpha_1 \\ -\cos\alpha_1 \sin\alpha_1 & -\sin^2\alpha_1 \end{bmatrix} \\ \begin{bmatrix} -\cos^2\alpha_1 & -\cos\alpha_1 \sin\alpha_1 \\ -\cos\alpha_1 \sin\alpha_1 & -\sin^2\alpha_1 \end{bmatrix} & \begin{bmatrix} \cos^2\alpha_1 & \cos\alpha_1 \sin\alpha_1 \\ \cos\alpha_1 \sin\alpha_1 & \sin^2\alpha_1 \end{bmatrix} \end{bmatrix}$$

where k_{spring} is the stiffness of the spring along its local axis and R is the rotation matrix from local to global coordinates and in the second line of the equation the attention is focused on its 2×2 sub-matrices. The stiffness can be computed as the trace of any of them. In particular, rows $2n_{x-1}$ and $2n_x$ contain two of the four sub-matrices in Eq. (4.19) for node n_p and two sub-matrices for every other bond connected with n_x. The traces are computed for all the sub-matrices of rows $2n_{x-1}$ and $2n_x$. Every value is then divided by the maximum value (normalization), so that the curve in Figure 4.27 is plotted. In this section several influence function are taken into consideration and the corresponding kernels are computed and normalized, as shown in Figure 4.27: every node is influenced by

nodes that are as far as 2δ, particularity that can also be mathematically seen from Eq. (1.80), even if this influence is limited compared to that of the nodes within a distance δ from the source node.

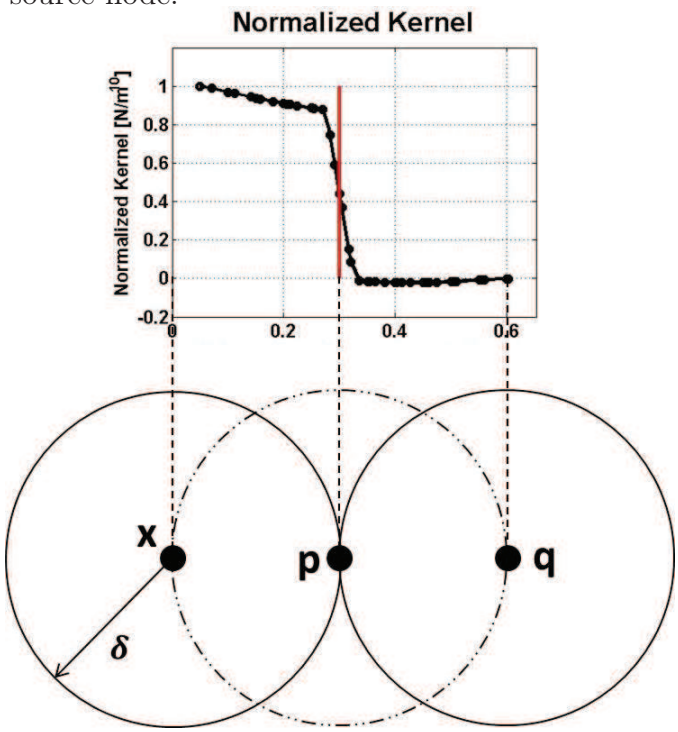


Figure 4.27: The kernels are computed for the constant influence function $\underline{\omega} = 1$: the red vertical line marks a horizon distance from the source node located at $\mathbf{x}$ (the origin of the horizontal axis), $\mathbf{p}$ is the position of a node at a δ distance from $\mathbf{x}$ and so the node located at $\mathbf{q}$ is from $\mathbf{p}$; a node at $\mathbf{q}$ affects the behavior of nodes that are as far as 2δ.

The following plots summarize the results for different influence functions.

1. Constant influence function $\underline{\omega} = 1$

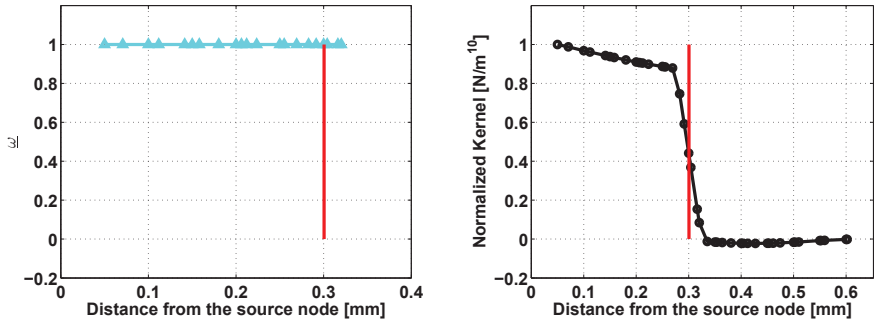


Figure 4.28: Influence function ($\underline{\omega} = 1$) vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.28 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.29; only the right half plate is shown due to the symmetry of the problem.

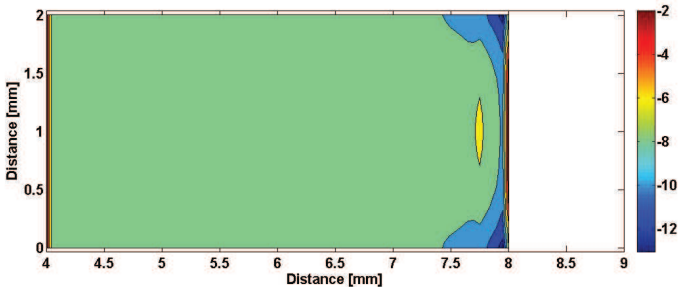


Figure 4.29: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = 1$.

2. Gaussian influence function $\underline{\omega} = e^{-\frac{|\boldsymbol{\xi}|^2}{\delta^2}}$

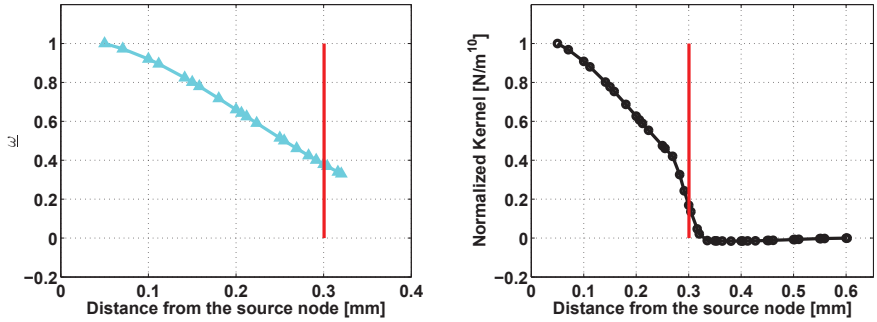


Figure 4.30: Influence function $\left(\underline{\omega} = e^{-\frac{|\boldsymbol{\xi}|^2}{\delta^2}}\right)$ vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.30 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.31.

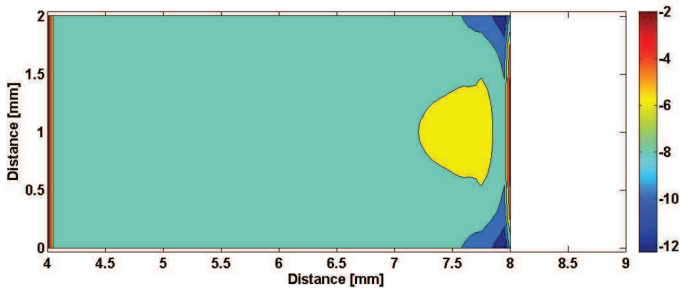


Figure 4.31: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = e^{-\frac{|\boldsymbol{\xi}|^2}{\delta^2}}$.

3. Linear proportional influence function $\underline{\omega} = |\boldsymbol{\xi}|$

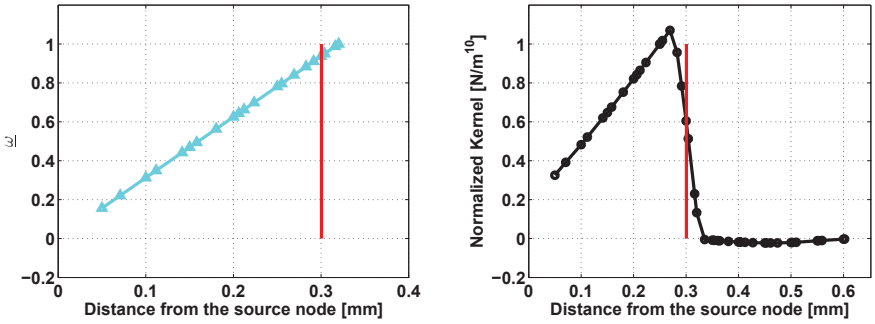


Figure 4.32: Influence function ($\underline{\omega} = |\boldsymbol{\xi}|$) vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.32 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.33.

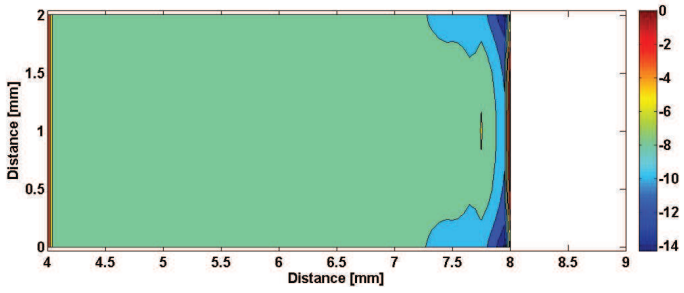


Figure 4.33: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = |\boldsymbol{\xi}|$.

4. Conical influence function $\underline{\omega} = \delta - |\boldsymbol{\xi}|$

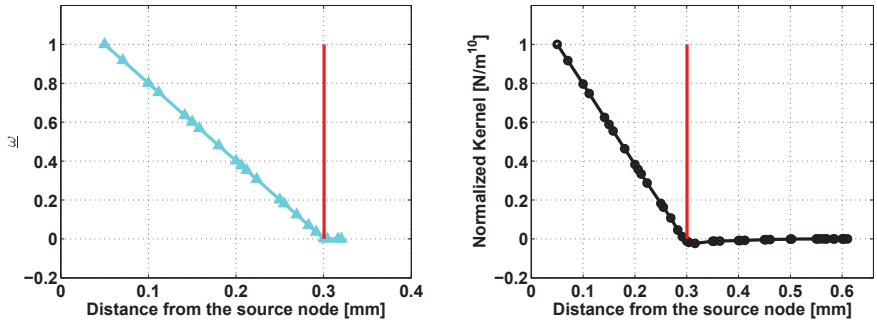


Figure 4.34: Influence function ($\underline{\omega} = \delta - |\boldsymbol{\xi}|$) vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.34 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.35.

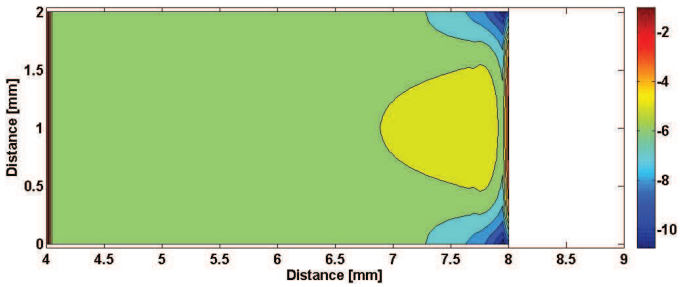


Figure 4.35: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = |\boldsymbol{\xi}|$.

5. Hyperbolic influence function $\underline{\omega} = |\boldsymbol{\xi}|^{-1}$

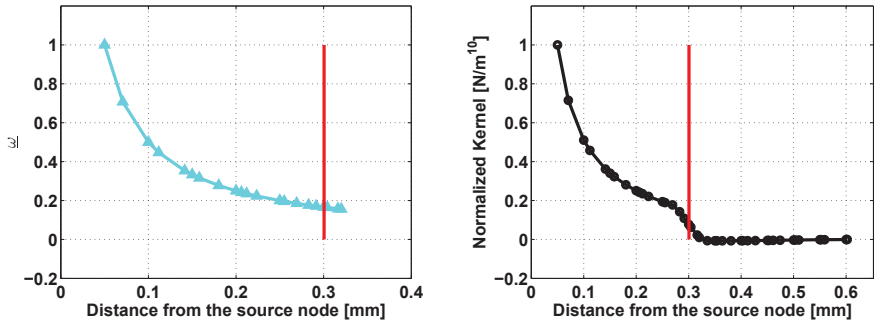


Figure 4.36: Influence function ($\underline{\omega} = |\boldsymbol{\xi}|^{-1}$) vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.36 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.37.

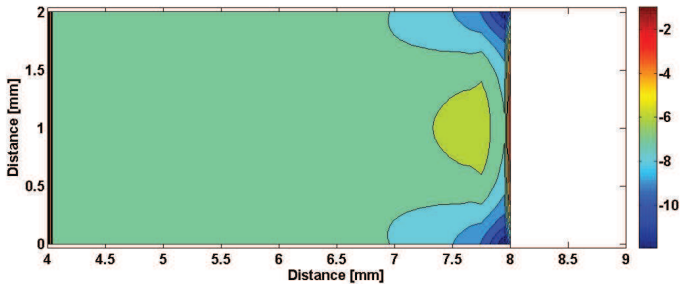


Figure 4.37: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = |\boldsymbol{\xi}|^{-1}$.

6. A highly local influence function $\underline{\omega} = |\boldsymbol{\xi}|^{-4}$

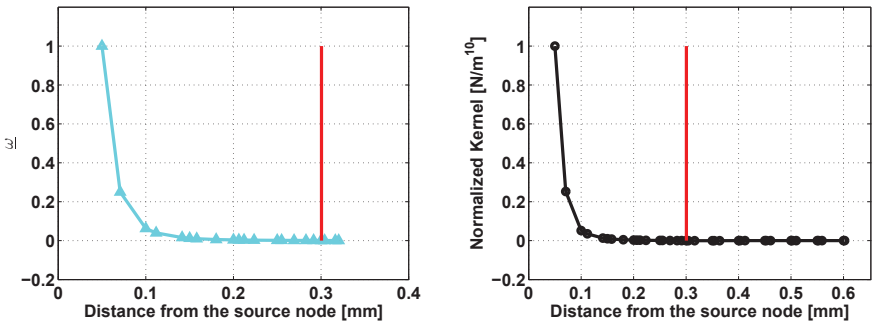


Figure 4.38: Influence function ($\underline{\omega} = |\boldsymbol{\xi}|^{-4}$) vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.38 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.39.

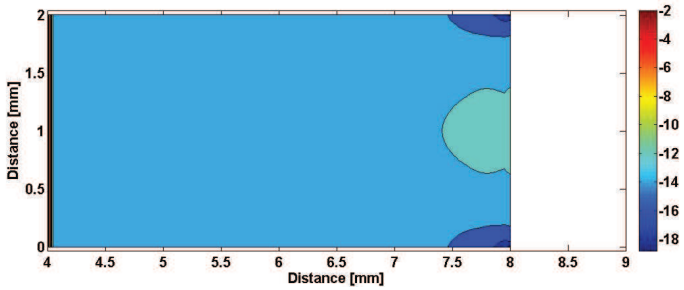


Figure 4.39: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = |\boldsymbol{\xi}|^{-4}$.

7. An influence function with an asymptote at its domain boundary $\underline{\omega} = \left(\delta + \frac{\Delta x}{2} - |\boldsymbol{\xi}|\right)^{-1}$

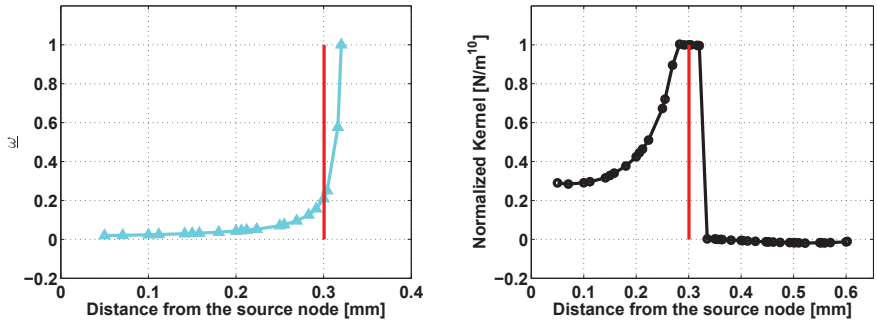


Figure 4.40: Influence function $\left(\underline{\omega} = \left(\delta + \frac{\Delta x}{2} - |\boldsymbol{\xi}|\right)^{-1}\right)$ vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.40 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.41.

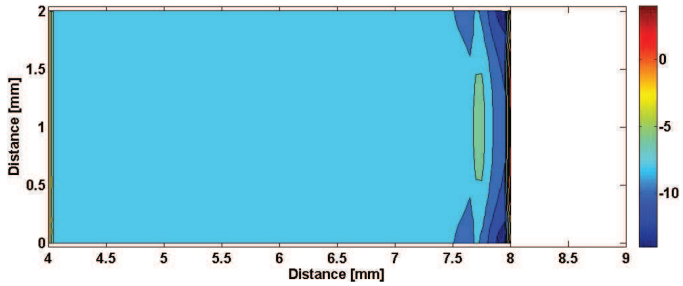


Figure 4.41: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = \left(\delta + \frac{\Delta x}{2} - |\boldsymbol{\xi}|\right)^{-1}$.

8. Influence function with an asymptote inside its domain $\underline{\omega} = |\frac{\delta}{2|\boldsymbol{\xi}|-\delta}|$

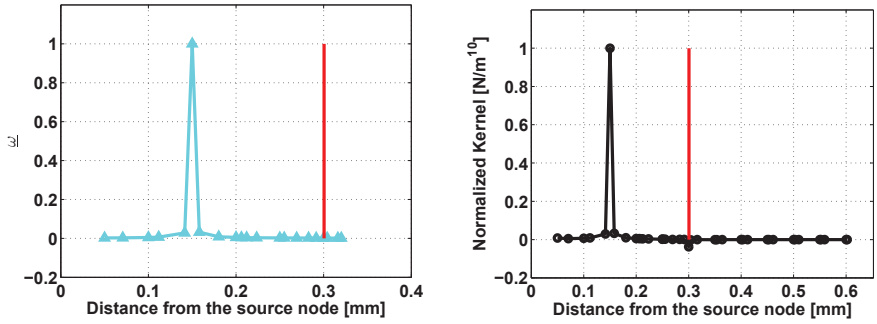


Figure 4.42: Influence function $\left(\underline{\omega} = |\frac{\delta}{2|\boldsymbol{\xi}|-\delta}|\right)$ vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.42 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.43.

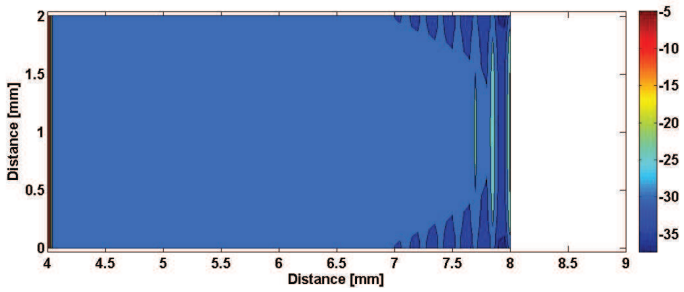


Figure 4.43: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = |\frac{\delta}{2|\boldsymbol{\xi}|-\delta}|$.

9. Non-integrable influence function with an asymptote inside its domain $\underline{\omega} = e^{\frac{1}{|\boldsymbol{\xi}|-\frac{\delta}{4}}}$

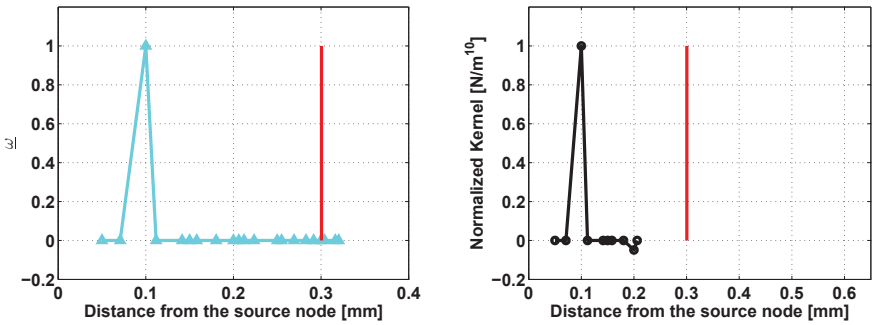


Figure 4.44: Influence function $\left(\underline{\omega} = e^{\frac{1}{|\boldsymbol{\xi}|-\frac{\delta}{4}}}\right)$ vs bond length and normalized kernel; the red vertical line identifies the horizon distance.

In Figure 4.44 the influence function and the kernel are shown. The percentage errors of the horizontal displacement field are plotted in Figure 4.45.

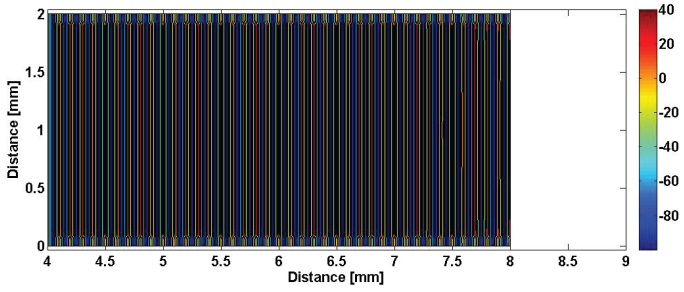


Figure 4.45: Percentage error [%] in the horizontal displacement field for half plate in a simulation with $\underline{\omega} = e^{\frac{1}{|\boldsymbol{\xi}|-\frac{\delta}{4}}}$.

The accuracy of the static solution is highly dependent on the influence function: in fact, for some influence functions the modeled plate is too stiff, the displacements are smaller than the expected ones. This happens with the influence functions for which the number of nodes contributing to the source node displacement is limited

and it doesn't change significantly as m-ratio increases (it's possible to see it in Figure 4.39). This effect is due to the numeric integration error with respect to the theoretical integral value. Unstable behavior may be caused by an influence function extremely localized (see Figure 4.45).

	Percentage Error on Young's Modulus [%]			**Percentage Error on Poisson's Ratio** [%]		
	Double node method	**Internal node method**	**Tensor method**	**Double node method**	**Internal node method**	**Tensor method**
$\underline{\omega} = \lvert\boldsymbol{\xi}\rvert^{-1}$	9.49	5.74	-2.87	4.55	0.58	0.32
$\underline{\omega} = 1$	14.27	7.13	-2.90	4.94	-1.20	0.32
$\underline{\omega} = e^{-\frac{\lvert\boldsymbol{\xi}\rvert^2}{\delta^2}}$	12.16	6.61	-1.47	4.37	-0.39	0.16
$\underline{\omega} = \delta - \lvert\boldsymbol{\xi}\rvert$	16.32	7.34	-4.08	5.39	-2.62	0.45
$\underline{\omega} = \lvert\boldsymbol{\xi}\rvert$	8.53	5.22	2.13	2.98	0.05	-0.24
$\underline{\omega} = \lvert\boldsymbol{\xi}\rvert^{-4}$	-86.20	-79.56	-83.03	11.78	11.28	9.23
$\underline{\omega} = \left(\delta + \frac{\Delta x}{2} - \lvert\boldsymbol{\xi}\rvert\right)^{-1}$	42.37	33.52	11.68	4.93	-5.60	-1.30
$\underline{\omega} = \lvert\frac{\delta}{2\lvert\boldsymbol{\xi}\rvert - \delta}\rvert$	243.61	164.66	-172.33	-21.49	3.63	19.15
$\underline{\omega} = e^{\frac{1}{\lvert\boldsymbol{\xi}\rvert - \frac{\delta}{4}}}$	142.26	143.21	-341.38	40.01	-30.62	37.93

Table 4.2: The percentage error on both Poisson's ratio and Young's modulus computation are reported with respect to the three estimation procedure previously explained (m-ratio=6, $\delta = 0.3mm$).

In the previous table (Table 4.2) the results on the errors in the computation of Poisson's ratio and Young's modulus are collected: the best results are given by the tensor node method for the majority of the studied influence function, while influence functions that have discontinuity in the domain generate errors independently of the employed method (see the last three influence function analyses). Even if the only restriction to the influence function is its non-negativity in the domain, for numerical reasons, it has to be regular: it might have jumps, but not vertical asymptote within it domain. In fact, being nonlocal, the peridynamic approach cannot describe material behavior even in very simple cases, such as a mono-dimensional traction test, when the behavior of a node depends on very few nodes (i.e. the influence function is extremely localized, like with asymptotes).

Chapter 5

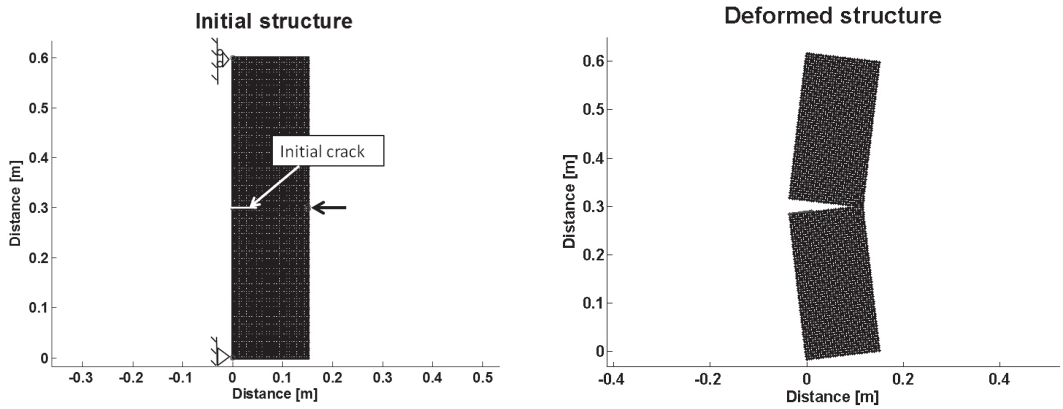
Comparison with Experimental Data

For analyses of fracture mechanics patterns, the comparison is carried out between experimental data found in literature and numerical results given by the peridynamic code. In this thesis, brittle fracture is the type of fracture taken into consideration, since it is one of the common types of mechanical failure in components and structures. It is quite interesting, since it can occur under pure mode I, pure mode II or mixed mode (I and II). In particular, since the 1960s mixed-mode fracture has been studied numerically, using different failure criteria in simulations (for example in [136]), or experimentally, using appropriate test methods and carefully designed specimens. The numerical tests are mainly concerned with δ and m-convergences.

5.1 BBP Numerical Results

The numerical tests simulated with the BBP code are employing both the static and the dynamic code.

The example is a three-point bending test as reported in Carpinteri [137]. The specimen is $150mm \times 600mm$, the fracture energy $G_0 = 9.8J/m^2$, Young's modulus $E = 35.77GPa$, Poisson's ratio $\nu = 0.33$. For the simulation, m-ratio=3 and $\delta = 15mm$ are adopted, so the total node number is $N = 3720$ in a plane stress static formulation. Load and boundary conditions are presented in Figure 5.1a, while Figure 5.1b shows the deformed configuration.



(a) (b)

Figure 5.1: Results of the BBP static code: (a) Deformed configuration (b) Force versus opening displacement measured at the nodes where the force is applied.

Figure 5.2 shows that the peridynamic solution is able to identify very well the maximum value of the applied force and the displacement at which the load capability drops.

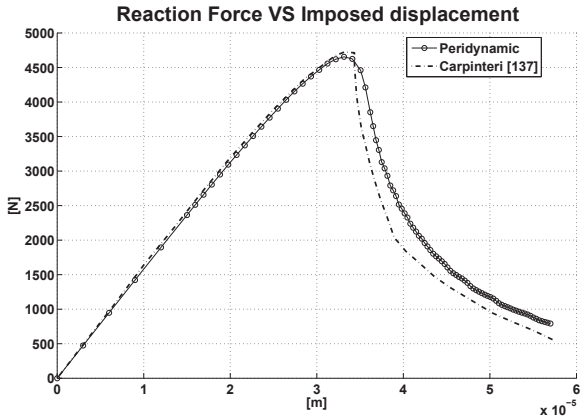


Figure 5.2: Comparison between experimental data and numerical results in terms of load vs displacement curve obtained from the three-point bending test.

The dynamic code was employed for analyzing some benchmark problems, such as the well documented experiment of Kalthoff-Winkler [138]. The experiment consists of a pre-cracked specimen impacted by a projectile. Young's modulus is $E = 190GPa$, Poisson's ratio $\nu = 0.25$. The geometric properties of the specimen an the load condition are shown in Figure 5.3: the specimen is $50mm \times 100mm$ with two horizontal pre-cracks $25mm$ long. The projectile is simulated by a velocity load condition. The time step is $\Delta t = 125nm$.

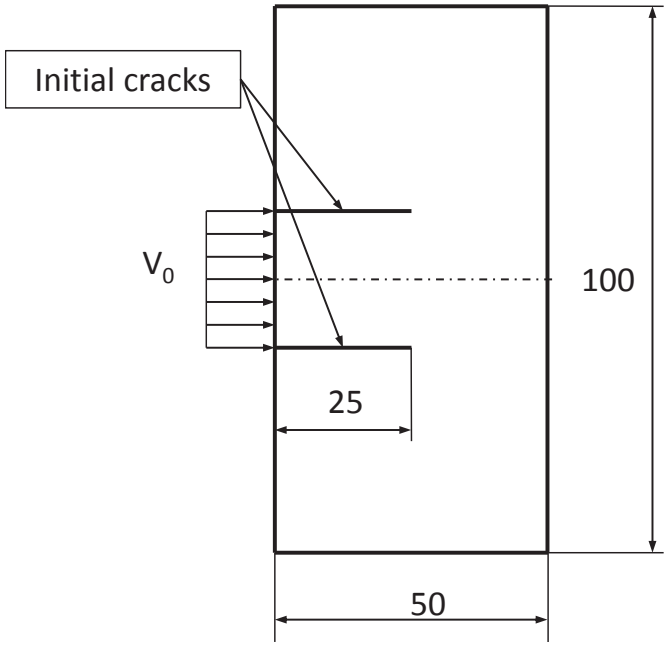


Figure 5.3: Kalthoff-Winkler experiment: the pre-crack specimen dimensions are expressed in $[mm]$ and the impacted projectile is simulated by a velocity load condition.

Since the structure is symmetric, only half of the specimen is simulated, imposing a rolling constraint in the symmetry axis (i.e. zero X-displacement for those nodes). The simulated grid of nodes and bonds are shown in Figure 5.4

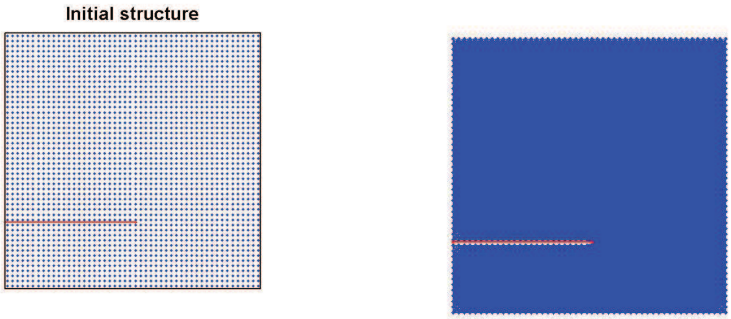


Figure 5.4: Simulated grid of nodes and bonds for the Kalthoff-Winkler experiment.

The body is discretized in $N = 2500$ nodes, the horizon is $\delta = 4mm$ and m-ratio is equal to 4.

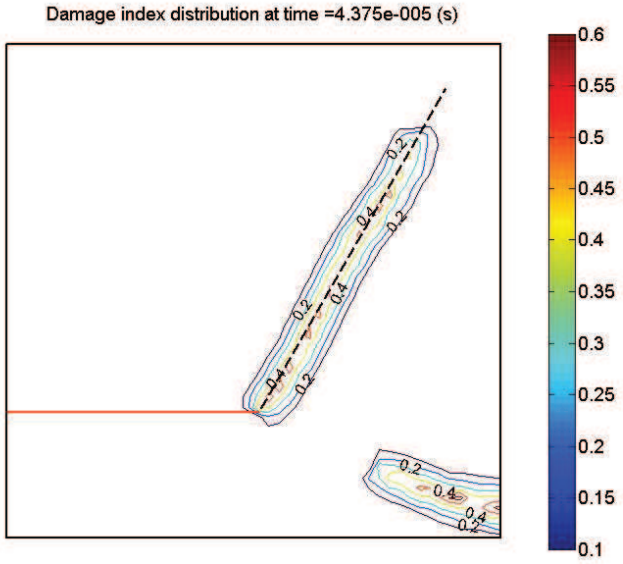


Figure 5.5: Damage index plot for the Kalthoff-Winkler experiment.

The crack propagates at an angle of approximately 70° counterclockwise with the notch-axis. In Figure 5.5 the damage index expressed in Eq. (1.43) is plotted. The secondary crack pattern are related to the coarse grid and they can be seen propagating from the right edge of the sample. This crack is due to numerical approximations and it does not propagate with a completely finer grid of with a grid to which an adaptive refinement strategy has been applied [139]. The model, with a rather coarse grid of nodes, can capture the propagation direction of the primary cracks, since the results of the original experiment indicated that the brittle failure cracks extended from the notch tip at an angle of 70° from the horizontal [138].

5.2 SBP Numerical Results

For the simulations in this section, a dynamic solver is employed. These analyses are some of the first dynamic analyses employing the two dimensional model in a dynamic code for fracture dynamics.

The first numerical tests employing the SBP code are aimed at the verification of the effective influence of the Poisson's ratio on the reproduced behavior of the material. In [79], the authors show an example of how SBP can take into considera-

tion the effect of different Poisson's ratio on the propagation direction of the crack. The sample in consideration is in Figure 5.6: a thick square plate is pre-notched in its symmetry axis where the notch length is half the size of the sample; the material is linear elastic; the plate is subjected to combined tension and shear and the failure criterion applied is a maximum stretch criterion, describing a brittle behavior.

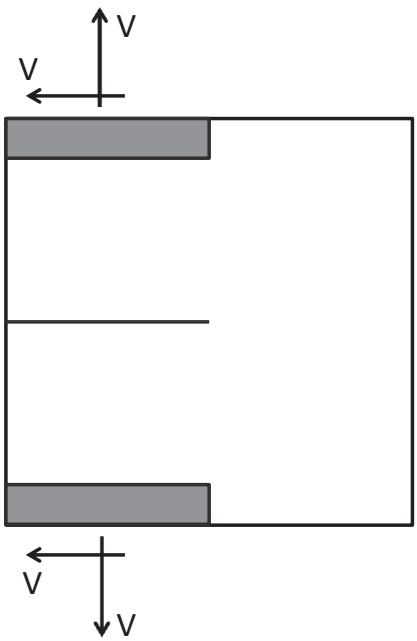


Figure 5.6: System under study for the crack propagation depending on Poisson's ratio.

Approximate analysis near the crack tip based on Mohr's stress circle results in a maximum principal stress, and so an initial crack direction, at an angle

$$\psi = \frac{1}{2}\arctan\left(\frac{1}{1+\nu}\right) \tag{5.1}$$

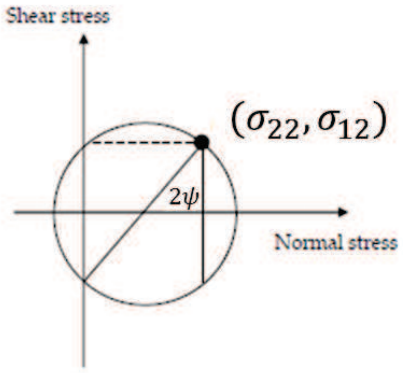


Figure 5.7: Mohr's circles for the studied case.

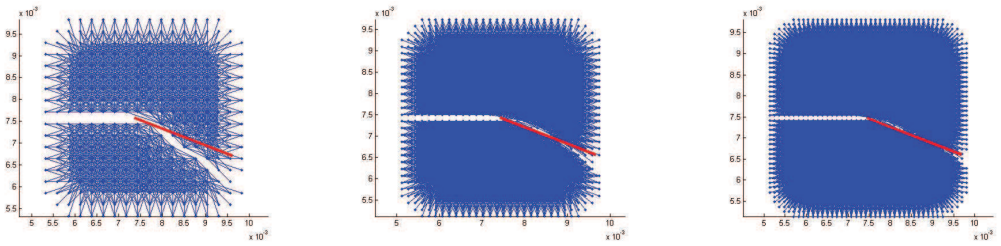
In fact, referring to Figure 5.7, $\sigma_{11} = 0$, $\sigma_{22} = E\mathsf{v}t/a$ and $\sigma_{12} = \mu\mathsf{v}t/a$, where E is Young's modulus, μ shear modulus, v the applied velocity, t the time of application and a the size of the plate. The angle of maximum principal stress is identified by σ_{22} and σ_{12}, in particular

$$\psi = \frac{1}{2}\arctan\left(\frac{2\sigma_{12}}{\sigma_{22}}\right) \tag{5.2}$$

and since $\mu = E/\left[2\left(1+\nu\right)\right]$, Eq. (5.2) becomes Eq. (5.1).

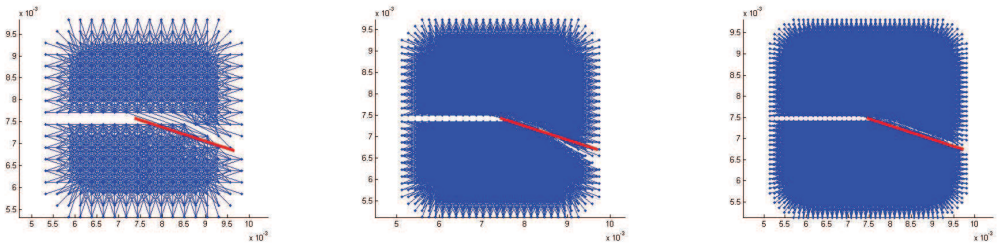
Differently from the authors of [79], where the 3D LPS model has been used, the plane stress model has been employed, so to take into account only one layer of nodes z direction. these results are the first numerical results for this case. The critical stretch is assumed $s_0 = 0.1\%$.

The results are shown in Figures 5.8 and 5.9: as the m-ratio increases the initial propagation direction tends to overlap with the theoretical one for different Poisson's ratios ($\nu = 0.1,\ 0.4$).



(a) (b) (c)

Figure 5.8: Results of the crack propagation direction compared to the theoretical one, highlighted by the continuous line in the SBP dynamic code with $\nu = 0.1$: (a) m-ratio=3 (b) m-ratio=5 (c) m-ratio=7. The theoretical value is $\psi = 21.17°$.



(a) (b) (c)

Figure 5.9: Results of the crack propagation direction compared to the theoretical one, highlighted by the continuous line in the SBP dynamic code with $\nu = 0.4$: (a) m-ratio=3 (b) m-ratio=5 (c) m-ratio=7. The theoretical value is $\psi = 17.77°$.

These preliminary simulations are carried out in order to verify the code and the potentiality of the theory. Afterwards, several fracture analyses are carried out involving different failure criteria.

One of these analyses has concerned a test specimen proposed by Ayatollahi, Aliya and Hassani in [10]. The experimental test is as shown in Figure 5.10. A semi-circular bending (SCB) specimen was employed for the mixed mode fracture test. The specimen is $5mm$ thick, its radius is $R_s = 50mm$, the initial length of the crack is $a = 15mm$ and the semi-distance between the two bottom supports is $S = 26mm$. A vertical compression force P is applied at the top of the specimen. The material chosen for the test is Polymethylmethacrylate (PMMA or Plexiglas®) for its brittle behavior at room temperature. Similarly to what was done by Ayatollahi et al., different initial crack angles β are simulated in order to investigate different mode mixities fracture patterns, from pure mode I ($\beta = 0°$) to pure mode II ($\beta = 50°$).

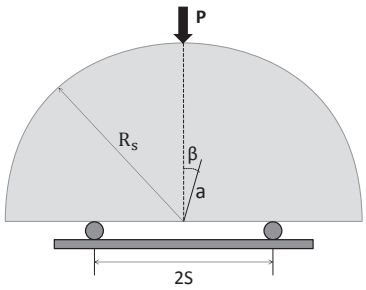


Figure 5.10: Semi-circular bend (SCB) specimen with the relative geometric parameters in [10].

For these simulations, the plane stress model (Eq. (1.67)) is adopted to carry out both δ and m-convergences, where a maximum stretch failure criterion has been employed; the critical value is assumed to be equal to that of the BBP formulation (see Eq. (1.41)). Crack patterns are qualitatively compared to the experimental ones.

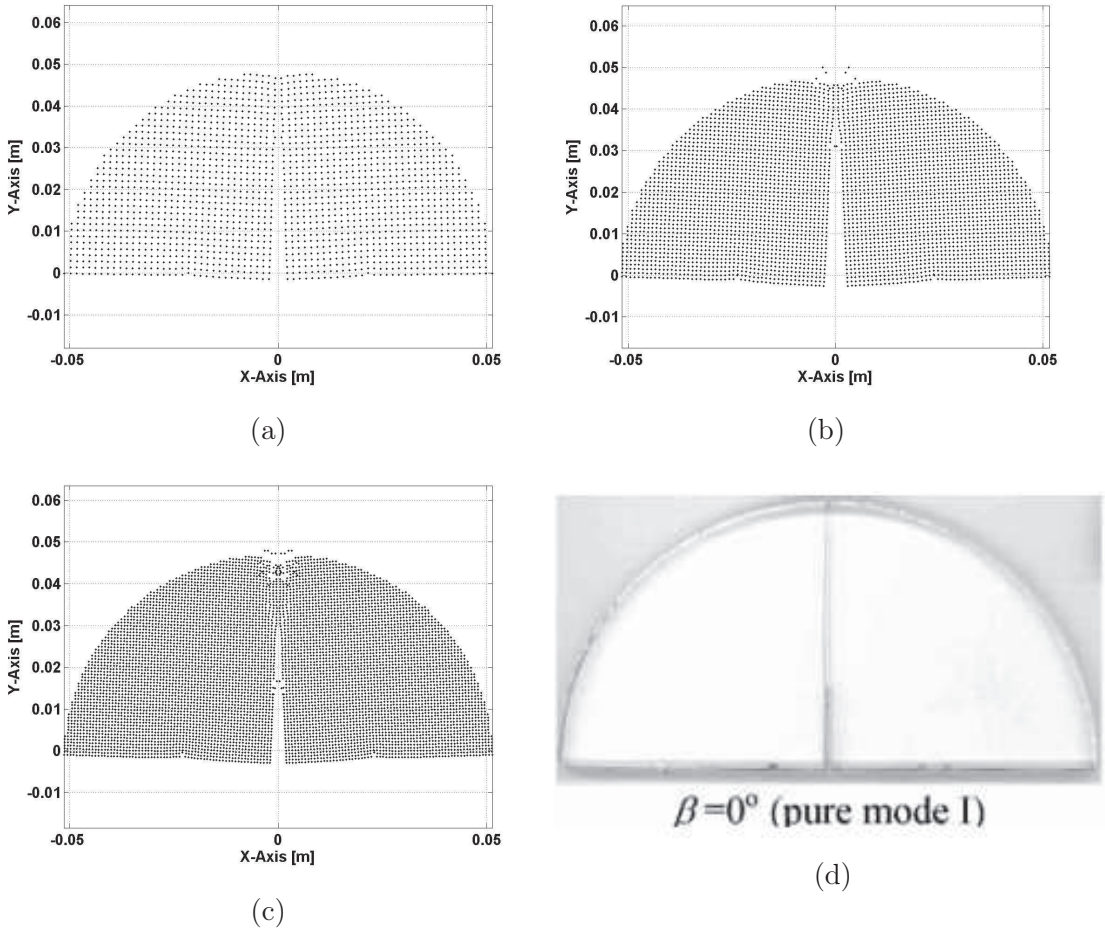


Figure 5.11: m-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, $\delta = 3mm$ and $\beta = 0°$: (a) m-ratio=2, (b) m-ratio=3, (c) m-ratio=4, (d) experimental results.

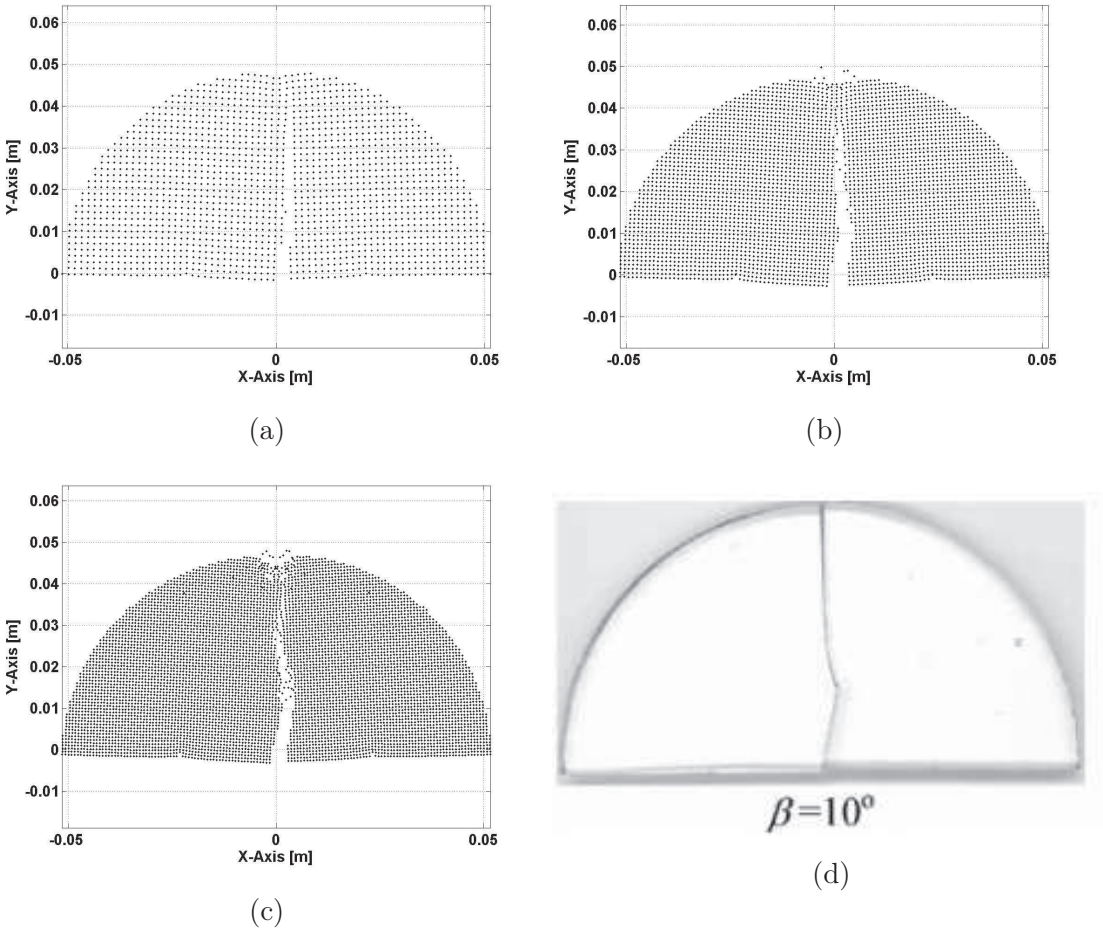


Figure 5.12: m-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, $\delta = 3mm$ and $\beta = 10°$: (a) m-ratio=2, (b) m-ratio=3, (c) m-ratio=4, (d) experimental results.

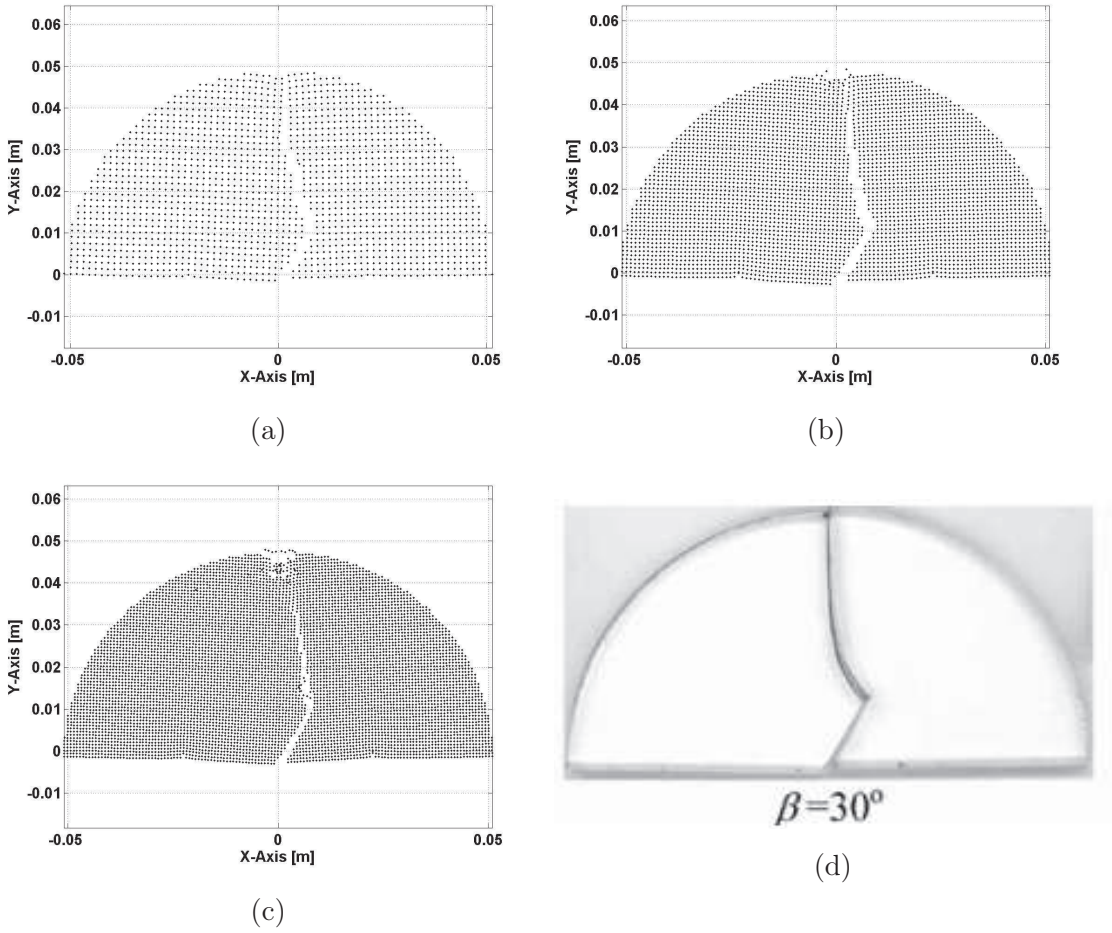


Figure 5.13: m-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, $\delta = 3mm$ and $\beta = 30°$: (a) m-ratio=2, (b) m-ratio=3, (c) m-ratio=4, (d) experimental results.

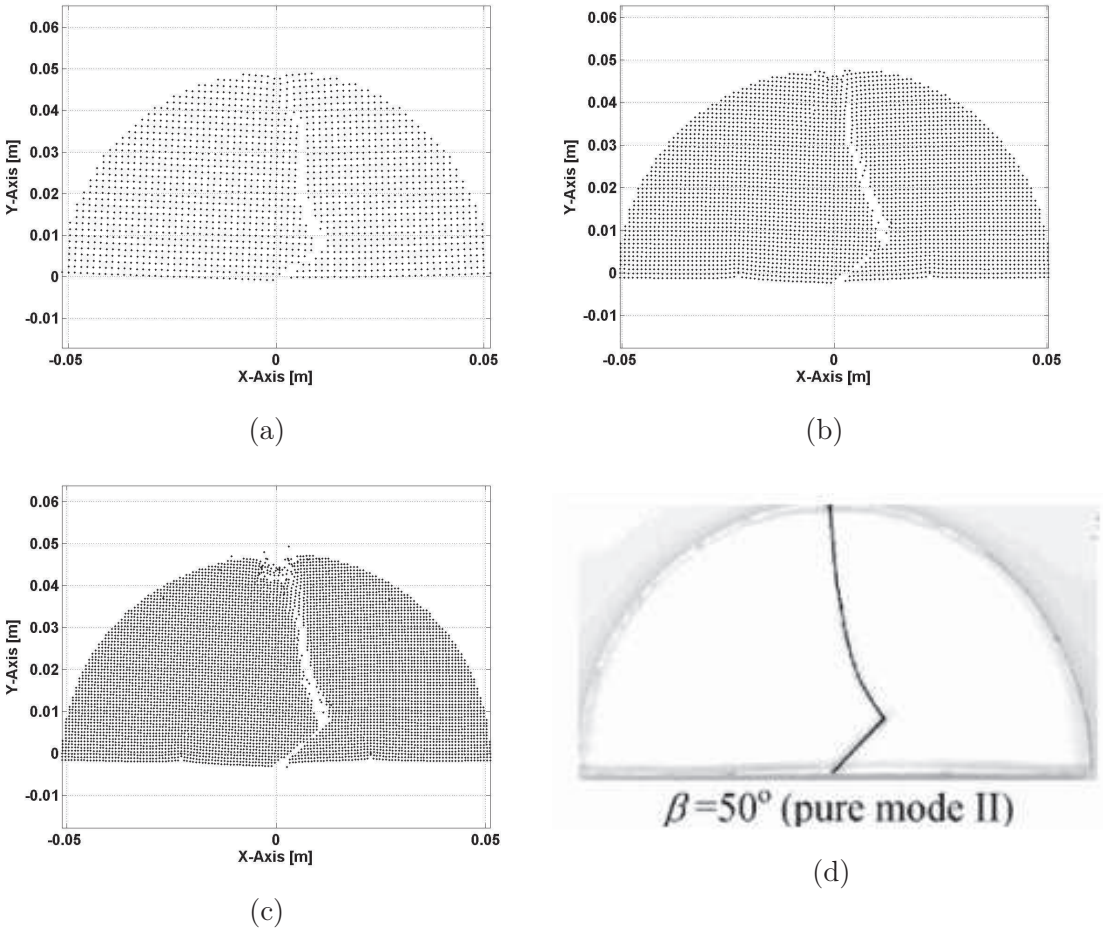


Figure 5.14: m-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, $\delta = 3mm$ and $\beta = 50°$: (a) m-ratio=2, (b) m-ratio=3, (c) m-ratio=4, (d) experimental results.

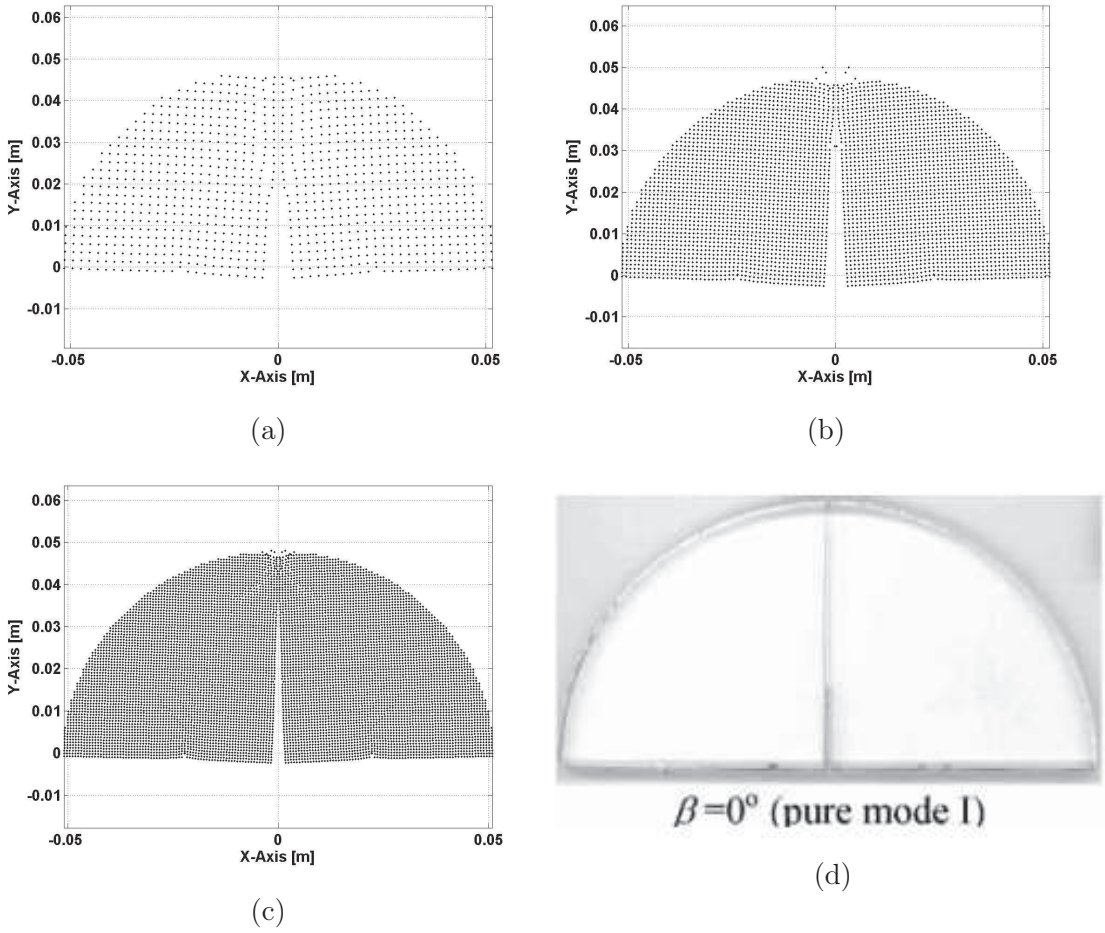


Figure 5.15: δ-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, m-ratio=3 and $\beta = 0°$: (a) $\delta = 6mm$, (b) $\delta = 3mm$, (c) $\delta = 2mm$, (d) experimental results.

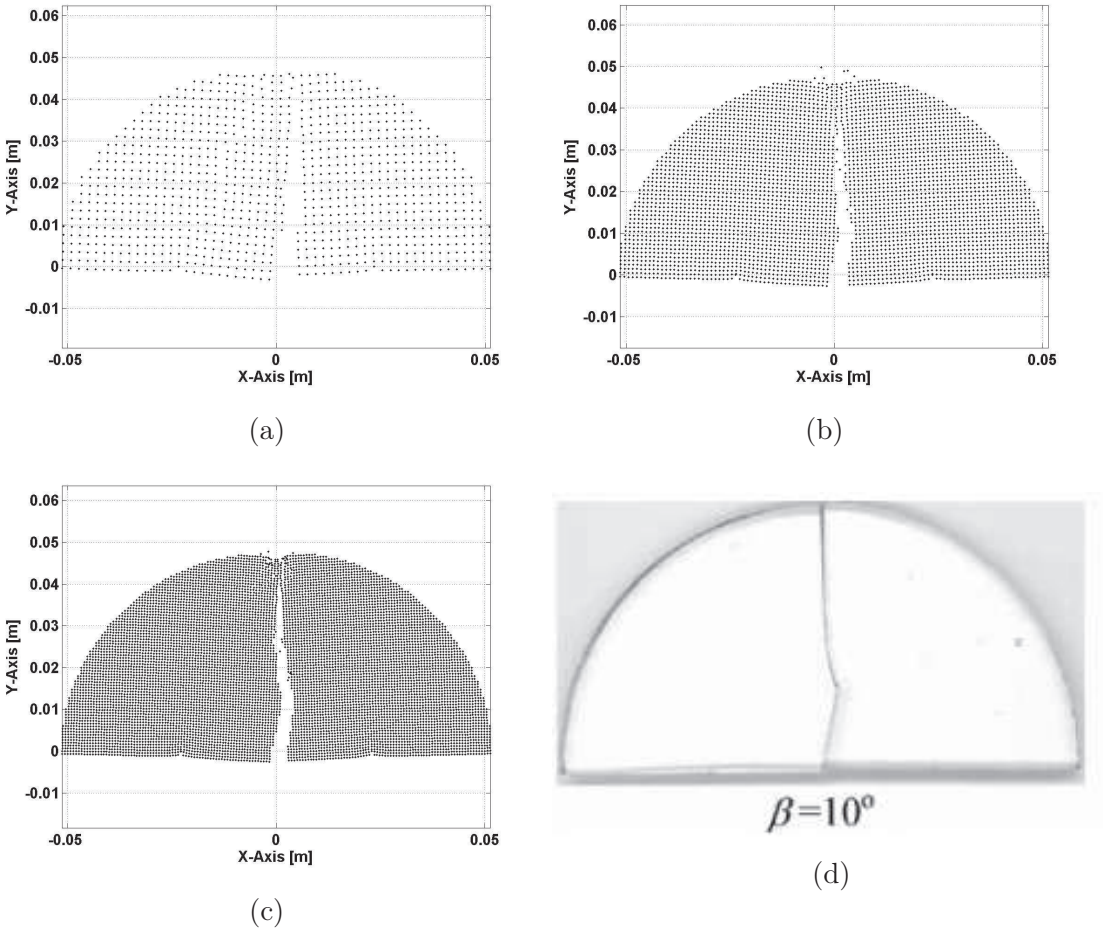


Figure 5.16: δ-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, m-ratio=3 and $\beta = 10°$: (a) $\delta = 6mm$, (b) $\delta = 3mm$, (c) $\delta = 2mm$, (d) experimental results.

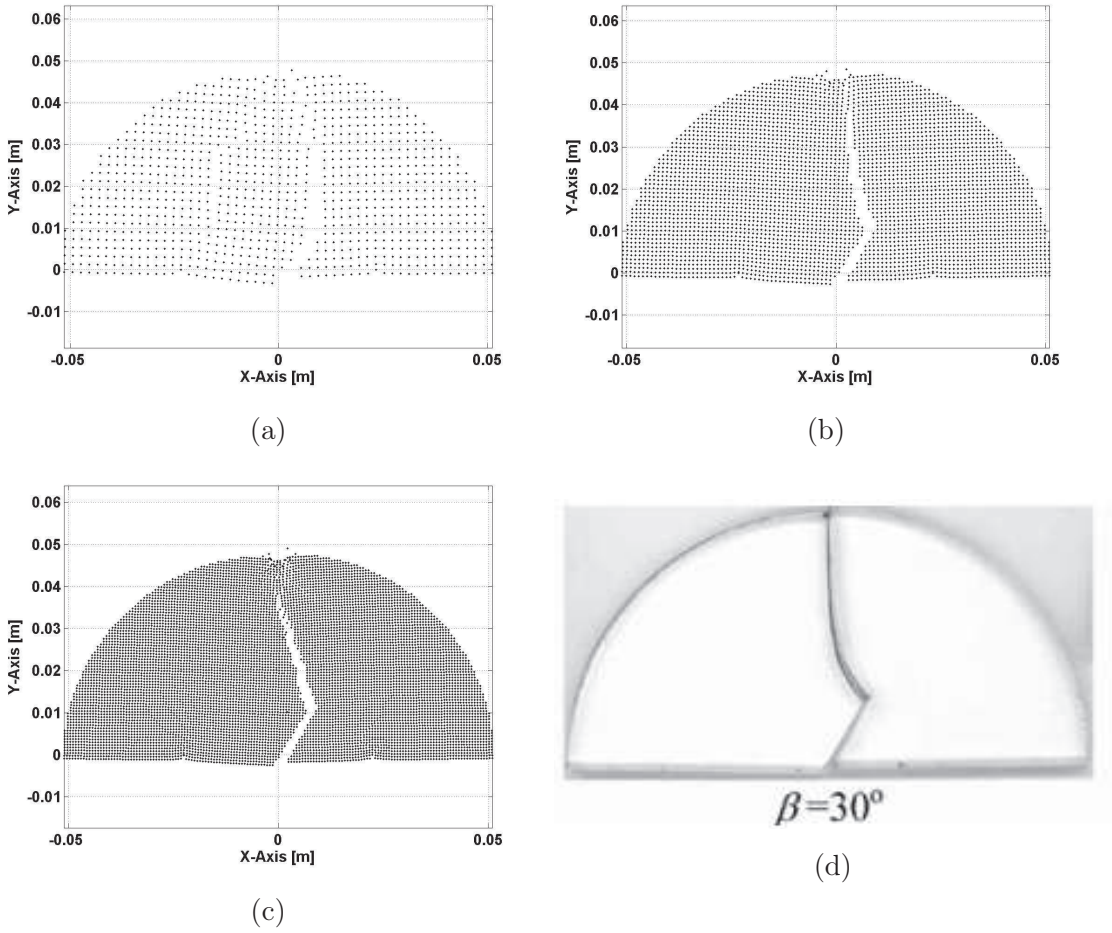


Figure 5.17: δ-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, m-ratio=3 and $\beta = 30°$: (a) $\delta = 6mm$, (b) $\delta = 3mm$, (c) $\delta = 2mm$, (d) experimental results.

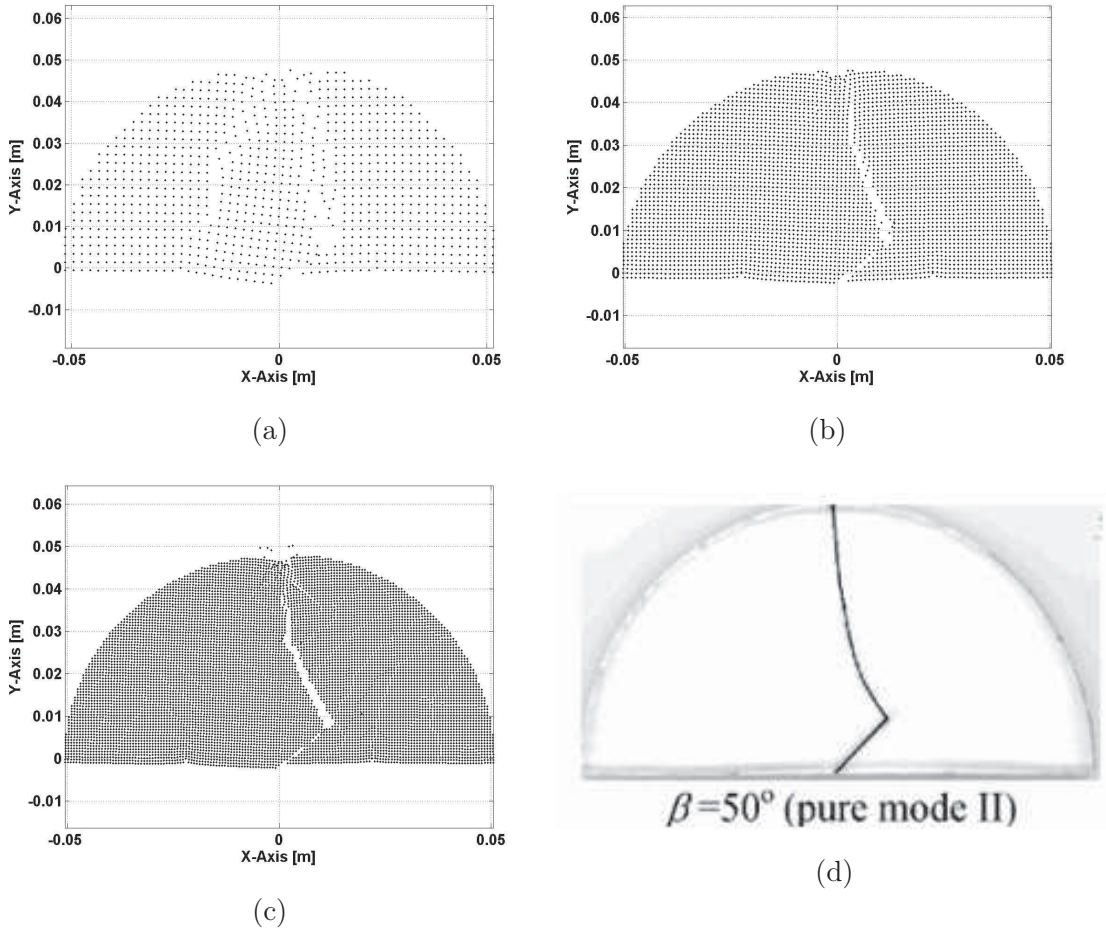


Figure 5.18: δ-convergence of the Peridynamic solutions compared to the experimental ones. For these simulations, m-ratio=3 and $\beta = 50°$: (a) $\delta = 6mm$, (b) $\delta = 3mm$, (c) $\delta = 2mm$, (d) experimental results.

Figures 5.11, 5.12, 5.13 and 5.14 show the m-convergence for different β, respectively 0°, 10°, 30°and 50°, and for a horizon length $\delta = 3mm$. As m increases, the theoretical crack path is reproduced more accurately, but an unrealistic behavior is present especially near the nodes where the load is applied. These numerical instabilities arise partly because of a nonlocal damage effect which is related to the horizon length and partly due to the no-fail zone near the nodes where the force is applied. No surface correction is adopted.

These effects are lighter when carrying out a δ-convergence (Figures 5.15, 5.16, 5.17 and 5.18), where it is clear that big horizons correspond to incorrect behavior (see Figure 5.18a)This aspect turns out to be really convenient, because it seems that for this type of problem (SCB test) and for this type of materials (homoge-

neous, isotropic and brittle), the important parameter to be tuned is the horizon δ, while the m-ratio (i.e. the number of bonds per node) doesn't need to be increased dramatically.

The following example is related to a test proposed by Ayatollahi and Aliya in [11]. The experimental test is as shown in Figure 5.19.

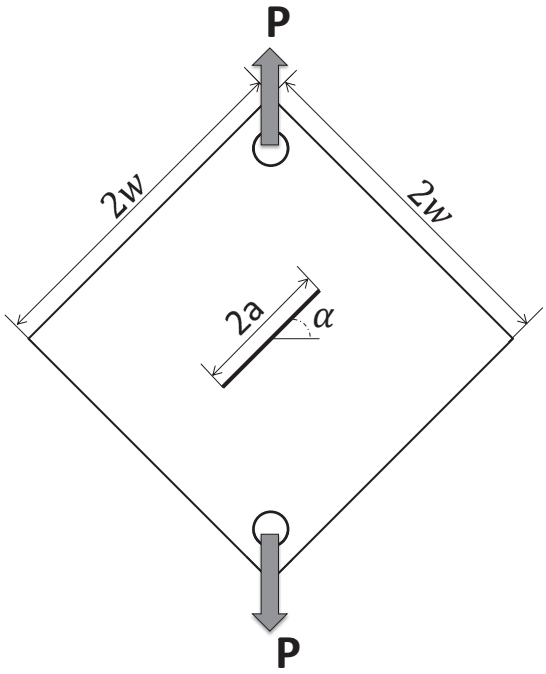


Figure 5.19: The diagonally loaded square plate (DLSP) specimen in [11].

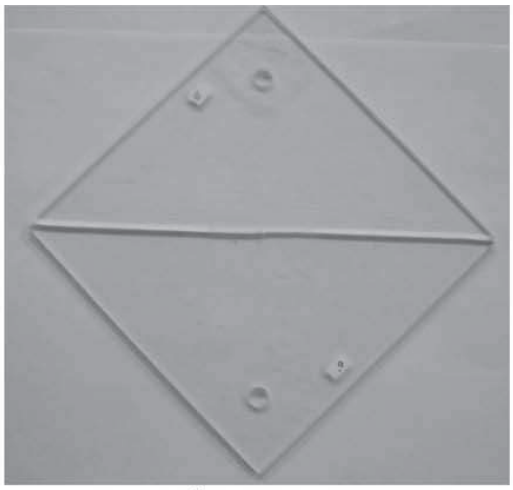
In this test configuration, a square plate of edge length $2\mathsf{w} = 150mm$ and of thickness $t = 5mm$ characterized by an inclined center crack of length $2\mathsf{a} = 45mm$ is loaded diagonally by two opposite concentrated loads. The concentrated load P was applied to the boundary of a small hole with $4mm$ radius, while the second hole was pinned. The elastic material properties of the brittle polymer Plexiglas (PMMA, Polymethylmethacrylate) are $E = 2940\,MPa$ and $\nu = 0.38$.

Pure mode I is at $\alpha = 0°$ while pure mode II is at $\alpha = 62.5°$. In [11], several inclinations are taken into consideration $\alpha = 0°,\ 15°,\ 45°,\ 62.5°$; the authors adopted the maximum tangential stress (MTS) criterion [140] and a modifies MTS criterion [141]. The crack path is illustrated for different initial crack inclinations in Figure 5.20 and their numerical simulations are carried out in plane stress con-

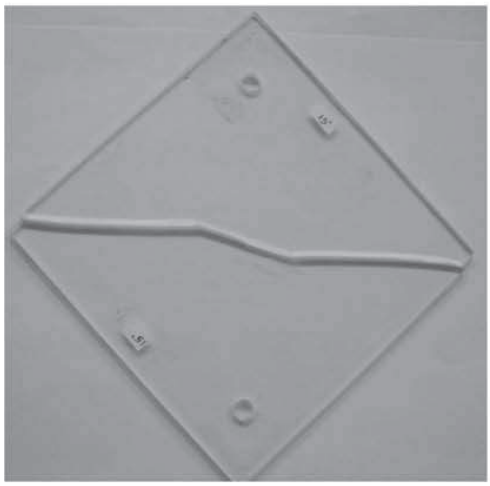
ditions.

This test has been numerically simulated through the SBP code implemented in Matlab, in which the different failure criteria explained in section 1.2.3 are taken into consideration for the 45° inclination case.

Figures 5.21, 5.23 and 5.25 show the δ-convergence, while Figures 5.22, 5.24 and 5.26 show the m-convergence. The crack line is the black continuous line in the left plot, while the red line represent the pre-crack in the specimen. The experimental results are reported on the right picture of each figure.

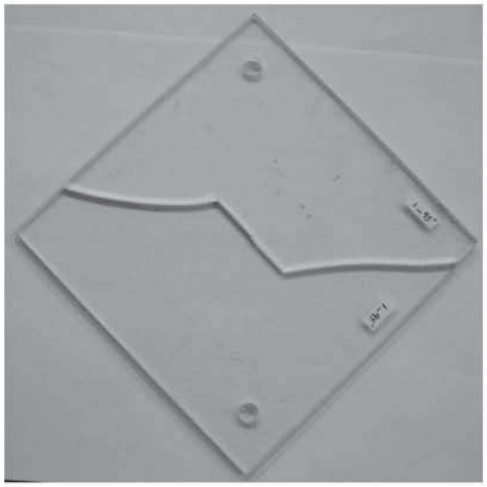


$\alpha = 0^{o}$ (pure mode I)



$\alpha = 15^{o}$

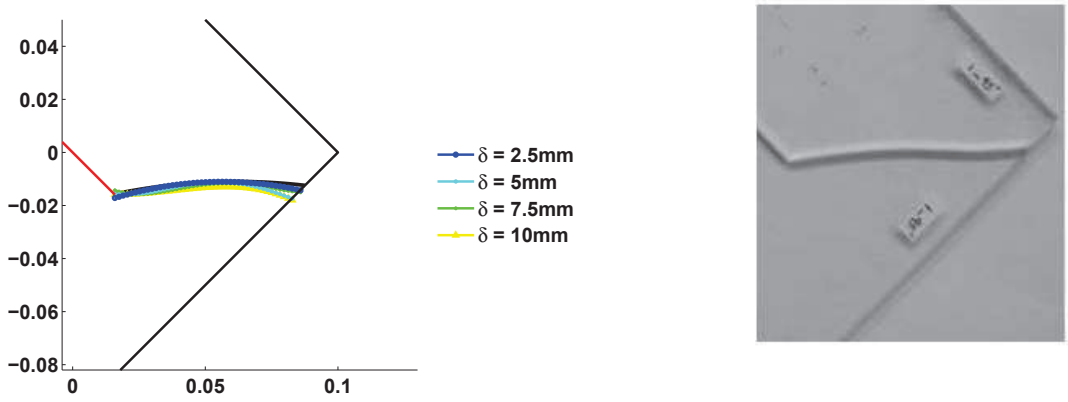
$\alpha = 30^{o}$



$\alpha = 45^{o}$

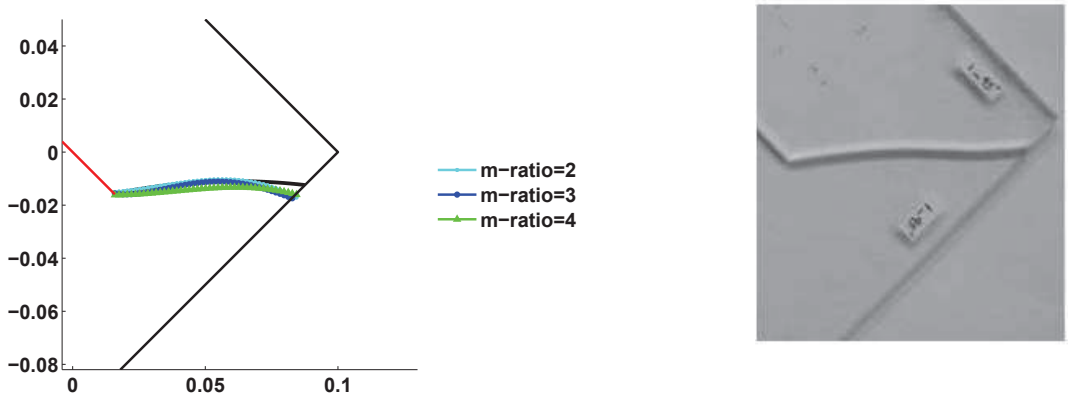
$\alpha = 62.5^{o}$ (pure mode II)

Figure 5.20: Fracture pattern in the DLSP specimens made of PMMA and for different crack inclination angles in [11].



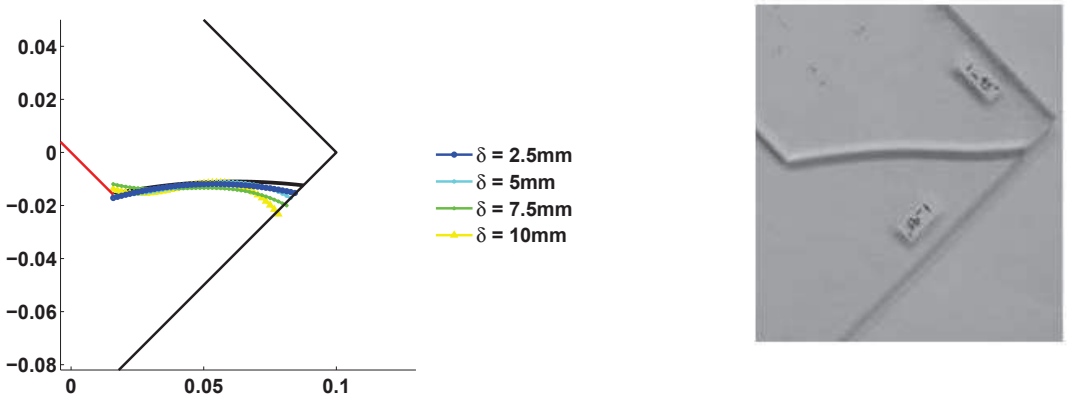
(a) (b)

Figure 5.21: Results for the maximum stretch criterion computed as in the BBP formulation: (a) δ-convergence, m-ratio=3, (b) experimental results.



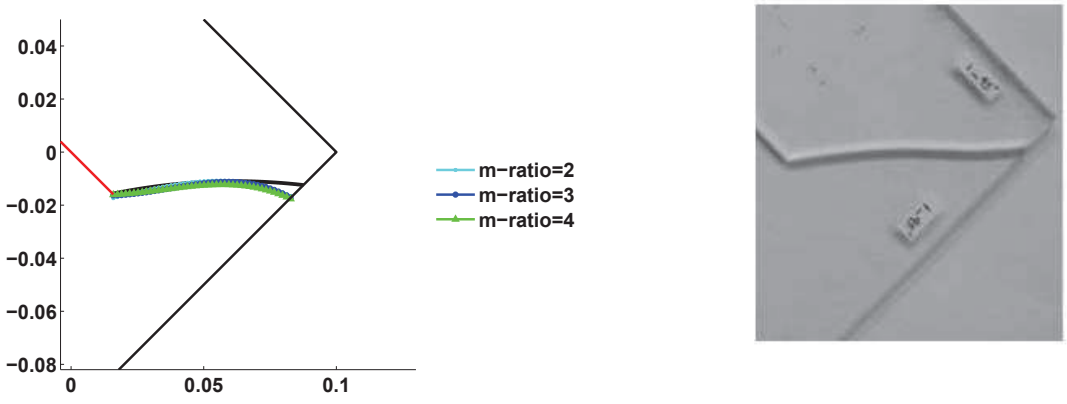
(a) (b)

Figure 5.22: Results for the maximum stretch criterion computed as in the BBP formulation: (a) m-convergence, $\delta = 5mm$, (b) experimental results.



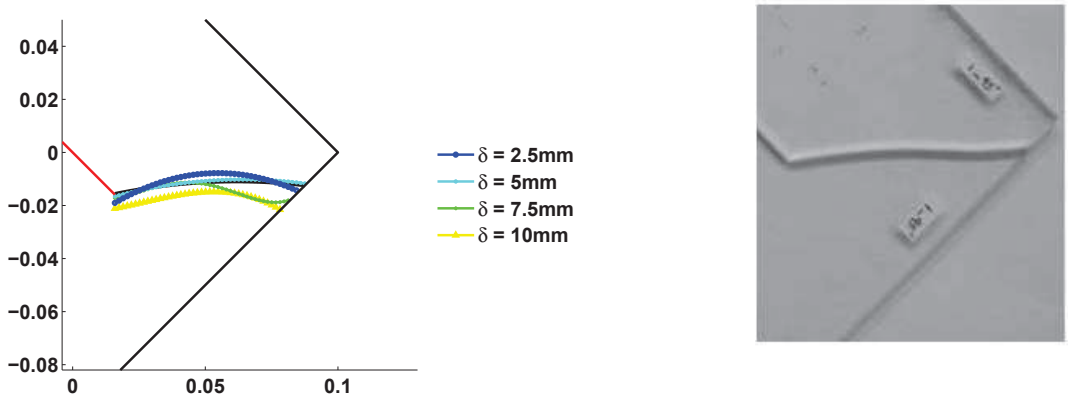
(a) (b)

Figure 5.23: Results for the maximum constant energy criterion computed incrementally: (a) δ-convergence, m-ratio=3, (b) experimental results.



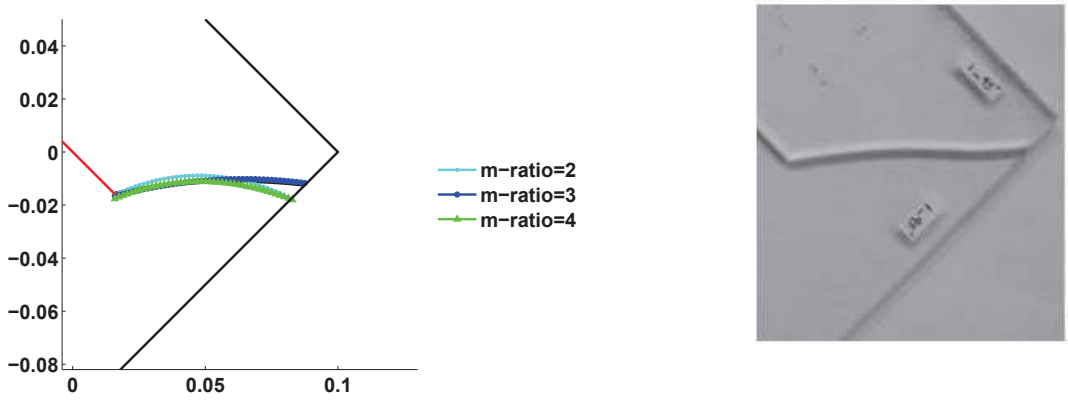
(a) (b)

Figure 5.24: Results for the maximum constant energy criterion computed incrementally: (a) m-convergence, $\delta = 5mm$, (b) experimental results.



(a) (b)

Figure 5.25: Results for the maximum constant energy criterion computed as an average between the energy density associated to the nodes at the ends of the bonds: (a) δ-convergence, m-ratio=3, (b) experimental results.



(a) (b)

Figure 5.26: Results for the maximum constant energy criterion computed as an average between the energy density associated to the nodes at the ends of the bonds: (a) m-convergence, $\delta = 5mm$, (b) experimental results.

As the m-ratio increases the slope of the fracture pattern is in general more accurate, but the position of the simulated crack and the experimental one may be

shifted. Differently, as δ decreases the numerically simulated crack tend to overlap the experimental one, but the curvature may be inaccurate. This analysis shows that δ is the main parameter controlling the propagation direction of the crack, while the m-ratio is mainly affecting the slope of the crack pattern, according to what was drawn for the SCB numerical example. Figure 5.25 and 5.26 show an unexpected behavior due to the spread of damage in a relatively large zone: this criterion may be more appropriate for describing a ductile fracture; in the model for this type of fracture, two main characteristics have to be considered: clearly, plasticity have to be implemented and, probably, the horizon have to be related to the plastic core size at the crack tip.

The last presented study case is the benchmark problem of mixed mode crack propagation carried out by Nooru-Mohamed et al. in [12], where a double-edge-notched square specimen made of concrete was subjected to a complex stress condition, due to shear forces and opening displacement loads. The specimen and load conditions are shown in Figure 5.27: it is a $200mm \times 200mm \times 50mm$ specimen and the notch length is $25mm$. Young's modulus is $E = 30GPa$, Poisson's ratio $\nu = 0.2$ and the fracture energy is $G_0 = 110J/m^2$. The analysis was performed with the 2D plane strain model and the time step of the simulation is $\Delta t = 500nm$. The numerical results are compared to the red line in Figure 5.28, where the experimental crack pattern is shown for the front face (red line) and the rear face (blue line) of the specimen.

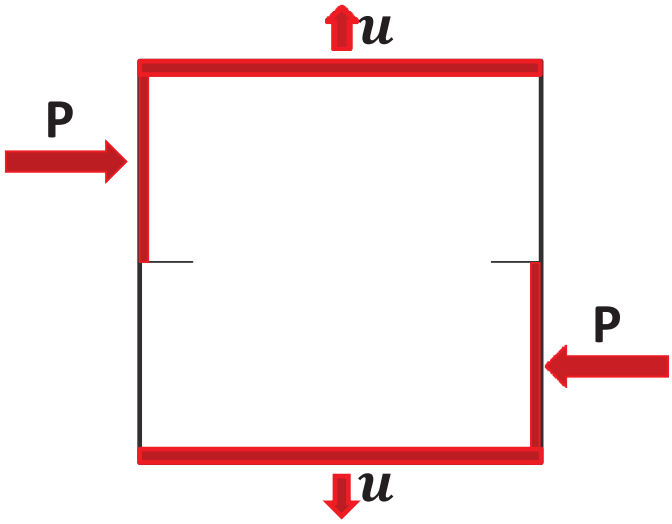


Figure 5.27: Load conditions applied to the pre-cracked specimen in Nooru-Mohamed's experiment in [12].

The convergences are shown in the following figures: Figures 5.29, 5.31 and 5.33 show the δ-convergence, while Figures 5.30, 5.32 and 5.34 show the m-convergence. The experimental crack line is the black continuous line in the plot.

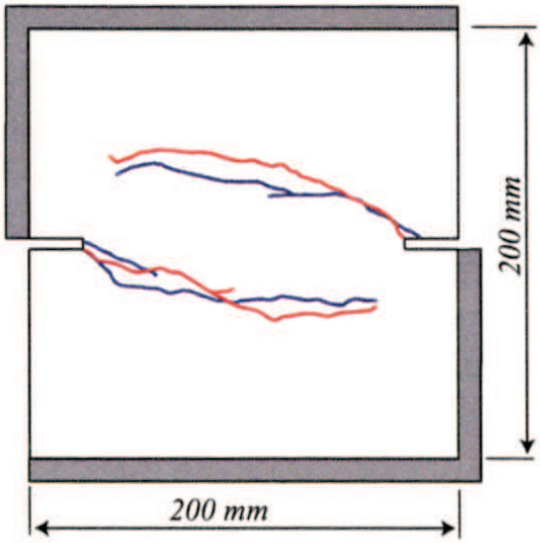


Figure 5.28: Crack pattern shown in [12].

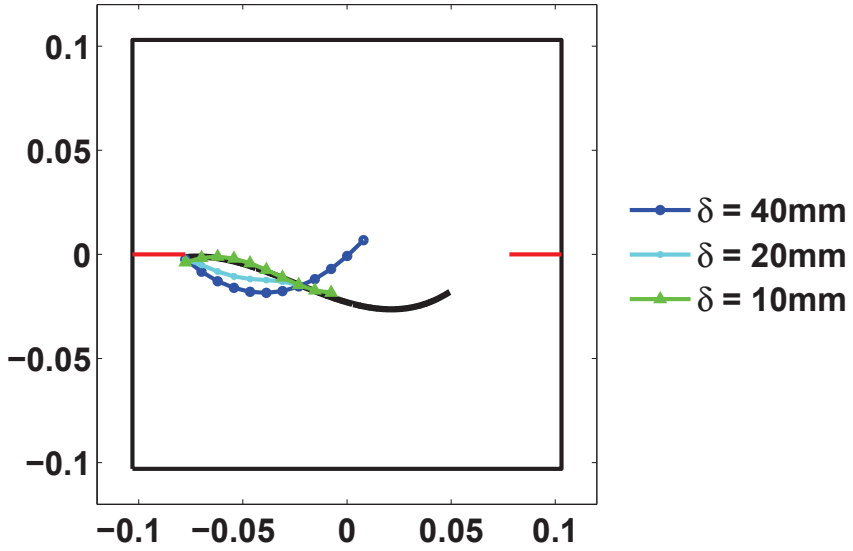


Figure 5.29: δ-convergence results for the maximum stretch criterion computed as in the BBP formulation (m-ratio=3).

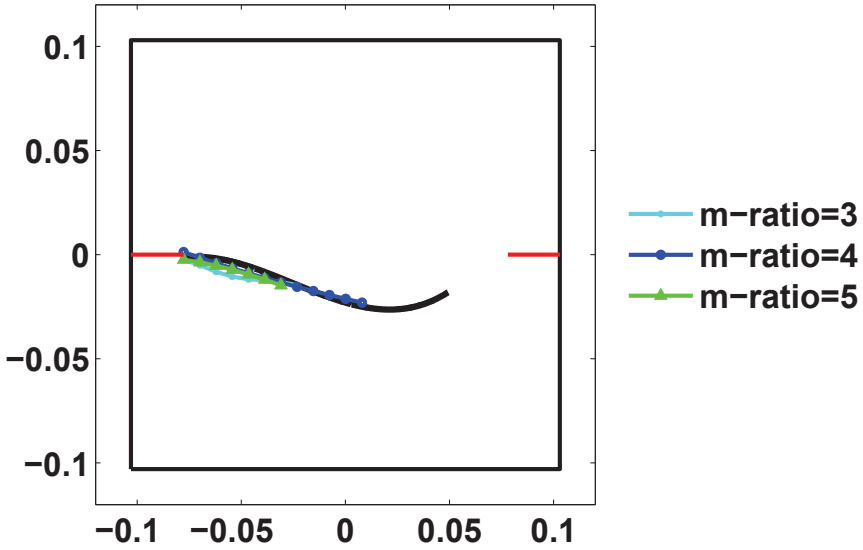


Figure 5.30: m-convergence results for the maximum stretch criterion computed as in the BBP formulation ($\delta = 20mm$).

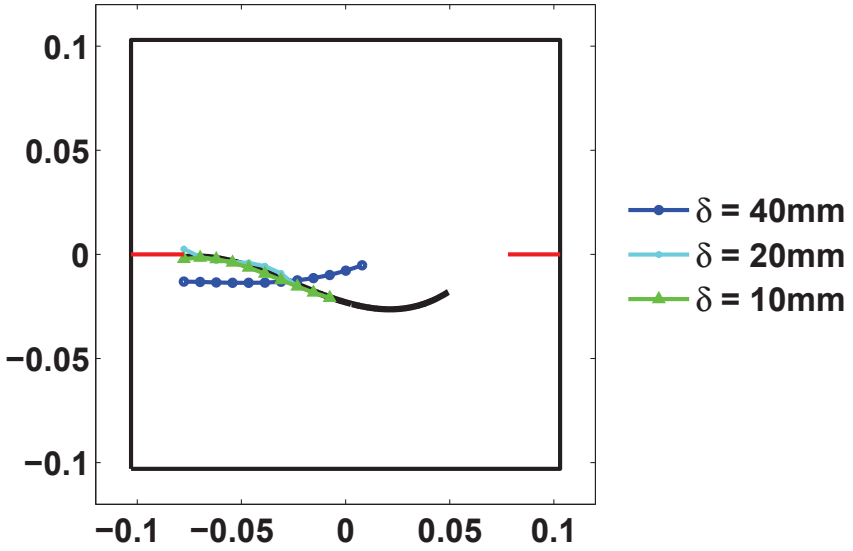


Figure 5.31: δ-convergence results for the maximum constant energy criterion computed incrementally (m-ratio=3).

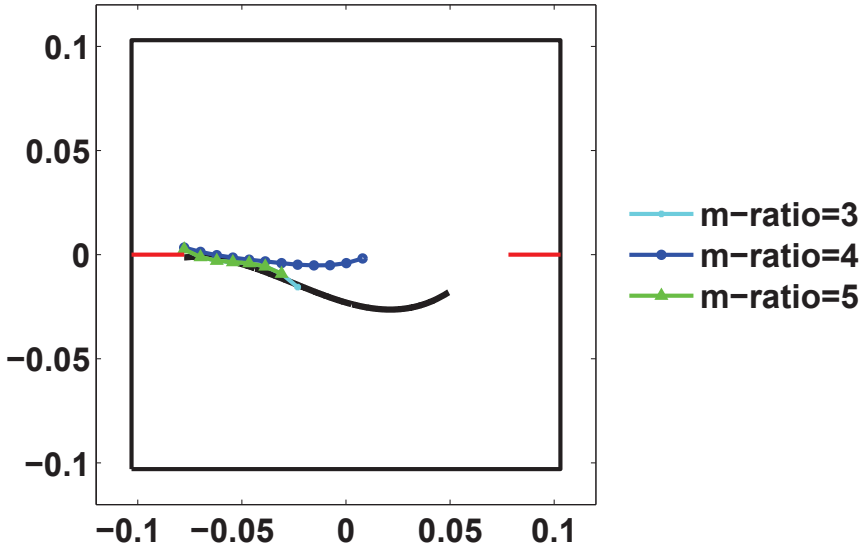


Figure 5.32: m-convergence results for the maximum constant energy criterion computed incrementally ($\delta = 20mm$).

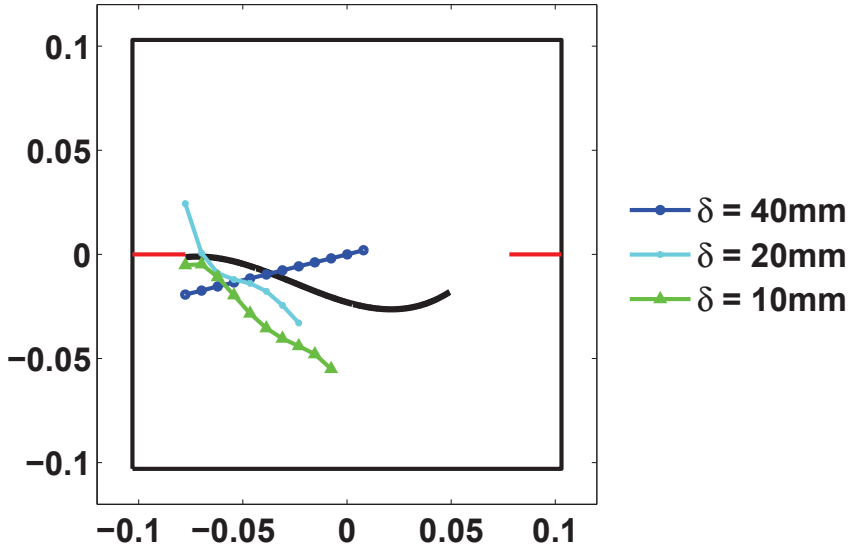


Figure 5.33: δ-convergence results for the maximum constant energy criterion computed as an average between the energy density associated to the nodes at the ends of the bonds (m-ratio=3).

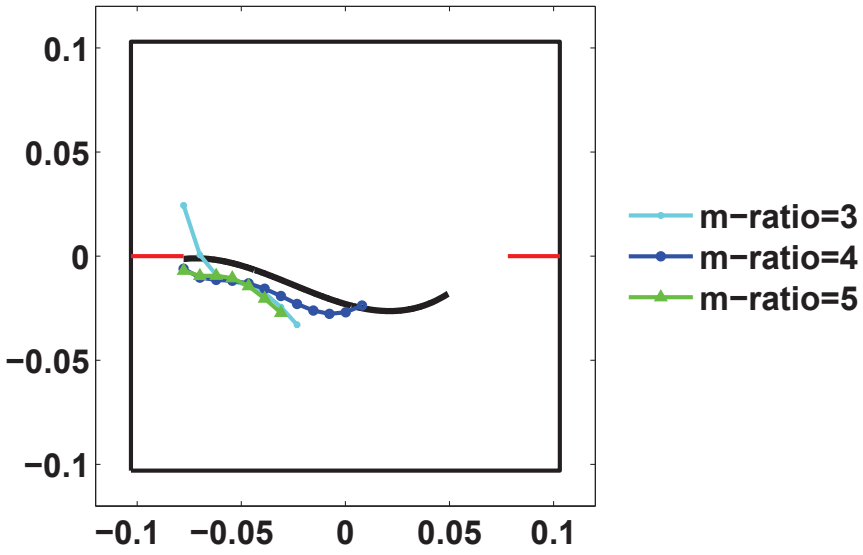


Figure 5.34: m-convergence results for the maximum constant energy criterion computed as an average between the energy density associated to the nodes at the ends of the bonds ($\delta = 20mm$).

For this example, the m-ratio is linked to the slope of the fracture pattern and as it increases, the slope becomes more accurate; δ is linked to the actual

position of the crack pattern: as it decreases the simulated crack overlaps the experimental one. Figures 5.33 and 5.34 show the unexpected behavior as it was for the previous study case (Figures 5.25 and 5.26) case. It is related to a spread of damage in a relatively large zone and may be inappropriate for brittle materials where directionality of crack propagation is a clear feature of the crack pattern. Intuitively, the last criterion is based on an averaged energy over a volume that is almost the same for two neighbor nodes (see Figure 5.35); for this reason, if one bond reaches the critical value, then the neighbor nodes are in the verge of reaching it as well, causing a spread of damage.

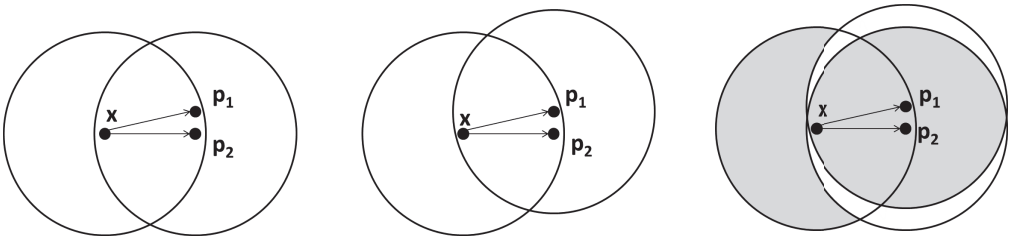


Figure 5.35: The energy associated every bond is an average value measured on the volume of the horizon spheres of the nodes linked by the bonds, thus for neighbor nodes the energy is comparable.

Chapter 6

Comments and Conclusions

Peridynamics employs integration for the calculation of forces on material points, so to avoid any problem arising from partial derivatives. In fact, in the classical theory of continuum mechanics, derivatives are involved in the governing equation, and cannot be applied whenever singularities occur. An integral formulation is not affected by this problem and can use only one mathematical framework to describe material behavior both where the displacement field is singular and where it is differentiable.

In this thesis, Peridynamics theory is analyzed and implemented in a Matlab® code to investigate its potentialities. Several analyses are carried out either with static or dynamic solvers: both linear and nonlinear static simulations are performed with an algorithm involving the assembly of the stiffness matrix and the biconjugate gradient stabilized method as solution scheme, while dynamic simulations are performed by using a solver that employs a velocity-Verlet time integration scheme.

The former approach allows the solution to be computed in one step, differently from dynamic relaxation or energy minimization methods. It becomes advantageous for the application of Peridynamics to nonlinear problems such as fatigue crack growth and quasi-static crack propagation, by employing a Newton-Raphson scheme with the relevant tangent stiffness matrix. The latter is straightforward and requires modest storage resources compared to the former one. However, it is explicit and the equilibrium is not verified so that the solution may be subjected to a drift as the time steps increase.

The original contributions presented in this thesis are the following:

- the derivation of the plane stress and the plane strain formulations for linearized 2D State-Based Peridynamics;
- the study of a set of proposed fracture criteria for the State-Based Peridynamics;
- the definition of a strategy to drastically reduce the computational time required for the assembly of stiffness and tangent matrix;
- the study of convergence of the peridynamic solution to the classical elasticity by means of static analyses;
- the study of a set of influence functions and of their effects on peridynamic solutions;
- numerous numerical analyses (both static and dynamic) have been performed to show the capabilities of Peridynamics to reproduce experimentally obtained crack paths (also in mixed mode fracture).

6.1 Peridynamics Linearized Models

One of the main original contributions to the existing literature in Peridynamics is the study of the linearized model for 2D plane stress and strain assumptions and the stiffness matrix in the state-based formulation. In the SBP formulation, the double tensor $\underline{\underline{\mathbb{K}}}$ (i.e. the derivative of the force state) necessary for the direct stiffness matrix method has been derived here for the 2D cases, while [7] illustrates the double tensor for 3D simulations. The model converges to the classical one in the limit-to-zero horizon. The verification of the peridynamic model is assessed by numerical results of linear elastic problems involving homogeneous isotropic brittle material.

The approach involves the stiffness matrix when the problem is linear, but for nonlinear solutions the tangent stiffness matrix is required [108]: in current applications, it is mainly elaborated numerically, because of the high computational

resources required. However, the present approach may be helpful to compute the tangent stiffness matrix analytically.

$\underline{\mathbb{K}}$ can be employed in an implicit dynamic solver as suggested by Mitchell in [7, 8]. To the author's knowledge, studies on implicit dynamic solvers have yet to be investigated in detail. Some attempts have been carried out by Mitchell in the aforementioned reports, but he applied his implicit dynamic solver only to a single bond.

6.2 Failure Criteria for SBP

This thesis presents failure criteria based on the elastic micropotential of each interaction. The maximum energy storable in a bond is considered to be independent or proportional to the initial length of the bond (similarly to what has been done in bond-based Peridynamics [68]) or a value related to the average energy associated to the nodes linked by the bond. The first two criteria do not relate the maximum energy of the single bond to anything not directly linked to the bond itself. Therefore, the directionality of cracks in brittle fracture is well reproduced. They are useful to identify the main roles played by the horizon and the m-ratio in the overall crack pattern: the horizon controls the crack position (the smaller the more accurate is the location of the crack in the sample with respect to the experimental one) while the m-ratio affects the curvature and shape of the cracks. A noticeable feature of these criteria is that the fracture energy employed is the fracture energy in mode I, but the results show good correspondence to the experimental data, even in the presence of mixed-mode failures. Besides, a critical stretch criterion implicitly neglects the contributions from the deviatoric strain tensor to the total stored energy. This is probably due to the type of described materials (homogeneous and isotropic). Differently, the last criterion gives rise to a spread of damage that may be unrealistic in some cases, such as in cracks where directionality is apparent. These criteria may be applied to specific types of materials, which are homogeneous and isotropic. Therefore, limits for using them have to be discussed and further criteria should be investigated.

6.3 Future works

Future activities concern an efficient and robust software implementation of the code in a supercomputer, by HCP computing techniques such as parallelization and GPU tools. Particular attention has to be paid to estimate numerical approximation errors, such as the round-off errors and truncation errors [142].

It is rather complex to compute analytically the tangent stiffness matrix, because it means not only to derive the formula for the second derivative of the force state $\underline{\mathbb{J}}\langle \boldsymbol{\xi}, \boldsymbol{\zeta}, \boldsymbol{\psi} \rangle$, but also to implement it correctly. In fact, several terms of the second derivative of force state will contain more than one Dirac delta functions (for example terms with $\Delta(\boldsymbol{\xi} - \boldsymbol{\zeta}) \cdot \Delta(\boldsymbol{\xi} - \boldsymbol{\psi})$) and have to be reduced to only one Dirac delta function. Besides, the resulting state will require three vectors as input. Thus, it will be represented numerically by a $3 \times 3 \times 3$ matrix[1] that has to be handled for every possible combination of three bonds.

$\underline{\mathbb{K}}$ can be employed in an implicit dynamic solver as suggested by Mitchell in [7, 8]. To the author's knowledge, studies on implicit dynamic solvers have yet to be investigated in detail. Some attempts have been carried out by Mitchell in the aforementioned reports, but he applied his implicit dynamic solver only to a single bond. When a problem is nonlinear, the solution strategy may be incremental so that it consists of finding an increment Δu^n to the displacement u^n to obtain the displacement u^{n+1} at load step $n+1$. Δu^n is found through an iterative process, such as Newton-Raphson scheme. At iteration k, Δu^n_k is known from the previous iteration and Δu^n_{k+1} can be computed by the first order approximation (i.e. the double tensor $\underline{\mathbb{K}}$) of the integral of internal force vectors at iteration $k+1$. Implicit dynamic solver employing a Newton-Raphson technique would be useful for the accurate prediction of crack patterns, but the costs in terms of time and resources may become intolerable for every day applications.

Another important future work is related to the implementation in the SBP code of additional failure criteria. The criteria employed in this thesis give accurate results because of the mechanisms of failure described and because of the type of described materials. In fact, the proposed failure criteria are applicable to brit-

[1] In 2D cases, it will be a $2 \times 2 \times 2$ matrix.

tle material (the damage spread seen in the energy average criterion is still under study) when the structure fails in mode I, mode II or mixed mode I/II. Failure modes as mode III (tearing), environmental effects as corrosion and phenomena like thermal shocks, impacts or fatigue are not analyzed. To date, most applications and studies for these failure modes are relative to the BBP formulation. Besides, the materials of the samples analyzed in this thesis are homogeneous and isotropic, so no effects related to directional properties (for example in crystal or composites) are present, so no delamination or damage related to grain boundary can be describe with these criteria. Appropriate considerations and analyses have to be performed to be able to reproduce with state-based Peridynamics these phenomena.

Bibliography

[1] Stewart A Silling. Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids*, 48(1):175–209, Jan 2000.

[2] Stewart A Silling, M Epton, O Weckner, J Xu, and E Askari. Peridynamic states and constitutive modeling. *Journal of Elasticity*, 88(2):151–184, Jul 2007.

[3] Bahattin Kilic and Erdogan Madenci. Coupling of peridynamic theory and the finite element method. *Journal of Mechanics of Materials and Structures*, 5(5):707–733, 2010.

[4] Walter Gerstle, Nicolas Sau, and SA Silling. Peridynamic modeling of plain and reinforced concrete structures. In *SMiRT18, Int. Conf. Structural Mech. Reactor Technol*, 2005.

[5] Stewart A Silling and Richard B Lehoucq. Convergence of peridynamics to classical elasticity theory. *Journal of Elasticity*, 93(1):13–37, 2008.

[6] Thomas L Warren, Stewart A Silling, Abe Askari, Olaf Weckner, Michael A Epton, and Jifeng Xu. A non-ordinary state-based peridynamic method to model solid material deformation and fracture. *International Journal of Solids and Structures*, 46(5):1186–1195, Mar 2009.

[7] John A Mitchell. A nonlocal, ordinary, state-based plasticity model for peridynamics. *SAND Report*, 7597, 2011.

[8] John A Mitchell. A non-local, ordinary-state-based viscoelasticity model for peridynamics. *Sandia National Lab Report*, pages 1–28, Oct 2011.

[9] Pablo Seleson. Improved one-point quadrature algorithms for two-dimensional peridynamic models based on analytical calculations. *Computer Methods in Applied Mechanics and Engineering*, 282:184–217, 2014.

[10] MR Ayatollahi, MRM Aliha, and MM Hassani. Mixed mode brittle fracture in pmma—an experimental study using scb specimens. *Materials Science and Engineering: A*, 417(1):348–356, 2006.

[11] MR Ayatollahi and MRM Aliha. Analysis of a new specimen for mixed mode fracture tests on brittle materials. *Engineering Fracture Mechanics*, 76(11):1563–1573, 2009.

[12] MB Nooru-Mohamed, Erik Schlangen, and Jan GM van Mier. Experimental and numerical study on the behavior of concrete subjected to biaxial tension and shear. *Advanced cement based materials*, 1(1):22–37, 1993.

[13] James R. Rice. Solid mechanics. `http://esag.harvard.edu/rice/e0_Solid_Mechanics_94_10.pdf`. Online Lecture Notes; accessed January 10, 2016.

[14] Clifford Truesdell. *The Elements of Continuum Mechanics: Lectures given in August-September 1965 for the Department of Mechanical and Aerospace Engineering Syracuse University Syracuse, New York*. Springer Science & Business Media, 2012.

[15] J. Merodio and G. Saccomandi. *Continuum Mechanics - Volume I*. 2011.

[16] Walter Noll. A new mathematical theory of simple materials. *Archive for Rational Mechanics and Analysis*, 48(1):1–50, 1972.

[17] Dominik Rogula. *Nonlocal theory of material media*, volume 268. Springer, 2014.

[18] Ronald E Miller and Vijay B Shenoy. Size-dependent elastic properties of nanosized structural elements. *Nanotechnology*, 11(3):139, 2000.

[19] R Maranganti and P Sharma. Length scales at which classical elasticity breaks down for various materials. *Physical review letters*, 98(19):195504, 2007.

[20] X Zhang and P Sharma. Size dependency of strain in arbitrary shaped anisotropic embedded quantum dots due to nonlocal dispersive effects. *Physical Review B*, 72(19):195345, 2005.

[21] F Leonforte, R Boissiere, A Tanguy, JP Wittmer, and J-L Barrat. Continuum limit of amorphous elastic bodies. iii. three-dimensional systems. *Physical Review B*, 72(22):224206, 2005.

[22] Youping Chen, James D Lee, and Azim Eskandarian. Examining the physical foundation of continuum theories from the viewpoint of phonon dispersion relation. *International journal of engineering science*, 41(1):61–83, 2003.

[23] Zdeněk P Bažant and Milan Jirásek. Nonlocal integral formulations of plasticity and damage: survey of progress. *Journal of Engineering Mechanics*, 2002.

[24] R. Pyrz and B. Bochenek. Discrete-continuum transition in modelling nanomaterials in proceedings of the iutam symposium on modelling nanomaterials and nanosystems held in aalborg, denmark, 19-22 may, 2008, 2008.

[25] Marcel Arndt and Michael Griebel. Derivation of higher order gradient continuum models from atomistic models for crystalline solids. *Multiscale Modeling & Simulation*, 4(2):531–562, 2005.

[26] Nicolas Triantafyllidis and S Bardenhagen. On higher order gradient continuum theories in 1-d nonlinear elasticity. derivation from and comparison to the corresponding discrete models. *Journal of Elasticity*, 33(3):259–293, 1993.

[27] Raymond David Mindlin. Second gradient of strain and surface-tension in linear elasticity. *International Journal of Solids and Structures*, 1(4):417–438, 1965.

[28] NA Fleck and JW Hutchinson. A reformulation of strain gradient plasticity. *Journal of the Mechanics and Physics of Solids*, 49(10):2245–2271, 2001.

[29] P.A. Deymier, K. Runge, and K. Muralidharan. *Multiscale Paradigms in Integrated Computational Materials Science and Engineering: Materials Theory, Modeling, and Simulation for Predictive Design*. Springer Series in Materials Science. Springer International Publishing, 2015.

[30] Y Mi, MA Crisfield, GAO Davies, and HB Hellweg. Progressive delamination using interface elements. *Journal of composite materials*, 32(14):1246–1272, 1998.

[31] JJ Munoz, U Galvanetto, and P Robinson. On the numerical simulation of fatigue driven delamination with interface elements. *International journal of fatigue*, 28(10):1136–1146, 2006.

[32] Goangseup Zi and Ted Belytschko. New crack-tip elements for xfem and applications to cohesive cracks. *International Journal for Numerical Methods in Engineering*, 57(15):2221–2240, 2003.

[33] M Nakane, K Shizawa, and K Takahashi. Microscopic discussions of macroscopic balance equations for solids based on atomic configurations. *Archive of Applied Mechanics*, 70(8-9):533–549, 2000.

[34] Attila Askar. *Lattice dynamical foundations of continuum theories*. World Scientific, 1986.

[35] Isaak Abramovich Kunin. Elastic media with microstructure i: One-dimensional models, vol. 26 of springer series in solid state sciences, 1982.

[36] Youping Chen, James D Lee, and Azim Eskandarian. Atomistic viewpoint of the applicability of microcontinuum theories. *International journal of solids and structures*, 41(8):2085–2097, 2004.

[37] Jerome Harris Weiner. *Statistical mechanics of elasticity*. Courier Corporation, 2012.

[38] JH Irving and John G Kirkwood. The statistical mechanical theory of transport processes. iv. the equations of hydrodynamics. *The Journal of chemical physics*, 18(6):817–829, 1950.

[39] James K Knowles and Eli Sternberg. On the failure of ellipticity and the emergence of discontinuous deformation gradients in plane finite elastostatics. *Journal of Elasticity*, 8(4):329–379, 1978.

[40] Morton E Gurtin and A Ian Murdoch. A continuum theory of elastic material surfaces. *Archive for Rational Mechanics and Analysis*, 57(4):291–323, 1975.

[41] Isaak Abramovich Kunin. Elastic media with microstructure ii: Three-dimensional models, vol. 44 of springer series in solid state sciences, 1982.

[42] Zdeněk P Bažant. Instability, ductility, and size effect in strain-softening concrete. *Journal of the Engineering Mechanics Division*, 102(2):331–344, 1976.

[43] E. Madenci and E. Oterkus. *Peridynamic Theory and Its Applications*. SpringerLink : Bücher. Springer New York, 2013.

[44] T Belytschko, SP Xiao, GC Schatz, and RS Ruoff. Atomistic simulations of nanotube fracture. *Physical Review B*, 65(23):235430, 2002.

[45] Fabrizio Cleri, Simon R Phillpot, Dieter Wolf, and Sidney Yip. Atomistic simulations of materials fracture and the link between atomic and continuum length scales. *J. Am. Ceram. Soc.*, 81(3):501–516, 1998.

[46] WJ Drugan and JR Willis. A micromechanics-based nonlocal constitutive equation and estimates of representative volume element size for elastic composites. *Journal of the Mechanics and Physics of Solids*, 44(4):497–524, 1996.

[47] Ilaria Monetto and WJ Drugan. A micromechanics-based nonlocal constitutive equation for elastic composites containing randomly oriented spheroidal heterogeneities. *Journal of the Mechanics and Physics of Solids*, 52(2):359–393, 2004.

[48] Ilaria Monetto and WJ Drugan. A micromechanics-based nonlocal constitutive equation and minimum rve size estimates for random elastic composites containing aligned spheroidal heterogeneities. *Journal of the Mechanics and Physics of Solids*, 57(9):1578–1595, 2009.

[49] Walter Noll. The foundations of classical mechanics in the light of recent advances in continuum mechanics. *The Axiomatic Method, with Special Reference to Geometry and Physics (Berkeley, 1957/58)*, pages 266–281, 1959.

[50] D Rogula. *Nonlocal Theory of Material Media. CISM Courses and Lectures*. Springer, Udine, 1982.

[51] E Kröner. Interrelations between various branches of continuum mechanics. In *Mechanics of Generalized Continua*, pages 330–340. Springer, 1968.

[52] A Cemal Eringen and DGB Edelen. On nonlocal elasticity. *International Journal of Engineering Science*, 10(3):233–248, 1972.

[53] A Cemal Eringen. Edge dislocation in nonlocal elasticity. *International Journal of Engineering Science*, 15(3):177–183, 1977.

[54] A Cemal Eringen. Screw dislocation in non-local elasticity. *Journal of Physics D: Applied Physics*, 10(5):671, 1977.

[55] Gilles Pijaudier-Cabot and Zdeněk Bažant. Nonlocal damage theory. *Journal of engineering mechanics*, 113(10):1512–1533, 1987.

[56] A Cemal Eringen. On nonlocal plasticity. *International Journal of Engineering Science*, 19(12):1461–1474, 1981.

[57] Zdeněk Bažant, Feng-Bao Lin, et al. Non-local yield limit degradation. *International Journal for Numerical Methods in Engineering*, 26(8):1805–1823, 1988.

[58] A Cemal Eringen, CG Speziale, and BS Kim. Crack-tip problem in non-local elasticity. *Journal of the Mechanics and Physics of Solids*, 25(5):339–355, 1977.

[59] Pablo Seleson, Michael L Parks, Max Gunzburger, and Richard B Lehoucq. Peridynamics as an upscaling of molecular dynamics. *Multiscale Modeling & Simulation*, 8(1):204–227, 2009.

[60] E Kröner. Elasticity theory of materials with long range cohesive forces. *International Journal of Solids and Structures*, 3(5):731–742, 1967.

[61] Michael L Parks, Richard B Lehoucq, Steven J Plimpton, and Stewart A Silling. Implementing peridynamics within a molecular dynamics code. *Computer Physics Communications*, 179(11):777–783, 2008.

[62] Michael L Parks, Steven J Plimpton, Richard B Lehoucq, and Stewart A Silling. Peridynamics with lammps: A user guide. Technical report, 2008.

[63] Richard B Lehoucq and Stewart A Silling. Force flux and the peridynamic stress tensor. *Journal of the Mechanics and Physics of Solids*, 56(4):1566–1577, Apr 2008.

[64] Richard W Macek and Stewart A Silling. Peridynamics via finite element analysis. *Finite Elements in Analysis and Design*, 43(15):1169–1178, 2007.

[65] Michael L Parks, David J Littlewood, John A Mitchell, and Stewart A Silling. Peridigm Users' Guide v1. 0.0. Technical report, 2012.

[66] Augustus Edward Hough Love. *A Treatise on the Mathematical Theory of Elasticity*. Courier Corporation, 1944.

[67] Youn Doh Ha and Florin Bobaru. Studies of dynamic crack propagation and crack branching with peridynamics. *International Journal of Fracture*, 162(1-2):229–244, Jan 2010.

[68] Stewart A Silling and Ebrahim Askari. A meshfree method based on the peridynamic model of solid mechanics. *Computers & Structures*, 83(17-18):1526–1535, Jun 2005.

[69] Youn Doh Ha and Florin Bobaru. Characteristics of dynamic brittle fracture captured with peridynamics. *Engineering Fracture Mechanics*, 78(6):1156–1168, Apr 2011.

[70] Wenke Hu, Yenan Wang, Jian Yu, Chian-Fong Yen, and Florin Bobaru. Impact damage on a thin glass plate with a thin polycarbonate backing. *International Journal of Impact Engineering*, 62:152–165, Dec 2013.

[71] Wenke Hu, Youn Doh Ha, and Florin Bobaru. Peridynamic model for dynamic fracture in unidirectional fiber-reinforced composites. *Computer Methods in Applied Mechanics and Engineering*, 217-220:247–261, Apr 2012.

[72] Wenke Hu, Youn Doh Ha, and Florin Bobaru. Modeling dynamic fracture and damage in a fiber-reinforced composite lamina with peridynamics. 9(6):707–726, 2011.

[73] Florin Bobaru. Influence of Van der Waals forces on increasing the strength and toughness in dynamic fracture of nanofibre networks: a peridynamic approach. *Modelling and Simulation in Materials Science and Engineering*, 15(5):397–417, Jun 2007.

[74] Erkan Oterkus, Ibrahim Guven, and Erdogan Madenci. Fatigue failure model with peridynamic theory. In *2010 12th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*. IEEE, Jun 2010.

[75] M Zaccariotto, F Luongo, G Sarego, D Dipasquale, and U Galvanetto. Fatigue crack propagation with peridynamics: a sensitivity study of paris law parameters. *CEAS2013, Innov Eur Sweden, Linkoping*, 2013.

[76] Ebrahim Askari, Jifeng Xu, and Stewart Silling. Peridynamic analysis of damage and failure in composites. In *44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada. Reston, VA: AIAA*, 2006.

[77] SA Silling, O Weckner, E Askari, and Florin Bobaru. Crack nucleation in a peridynamic solid. *International Journal of Fracture*, 162(1-2):219–227, 2010.

[78] E Askari, F Bobaru, RB Lehoucq, ML Parks, SA Silling, and O Weckner. Peridynamics for multiscale materials modeling. In *Journal of Physics: Conference Series*, volume 125, page 012078. IOP Publishing, 2008.

[79] Paul Demmie Silling, S. A. and Thomas L. Warren. Peridynamic simulation of high-rate material failure. ASME applied mechanics and materials conference, Austin, TX, USA, 2007.

[80] Florin Bobaru and Stewart A Silling. Peridynamic 3D models of nanofiber networks and carbon nanotube-reinforced composites. In *MATERIALS PROCESSING AND DESIGN: Modeling, Simulation and Applications-NUMIFORM 2004-Proceedings of the 8th International Conference on Numerical Methods in Industrial Forming Processes*, volume 712, pages 1565–1570. AIP Publishing, 2004.

[81] MS Breitenfeld, PH Geubelle, O Weckner, and SA Silling. Non-ordinary state-based peridynamic analysis of stationary crack problems. *Computer Methods in Applied Mechanics and Engineering*, 272:233–250, Apr 2014.

[82] S Sun and V Sundararaghavan. A peridynamic implementation of crystal plasticity. *International Journal of Solids and Structures*, 51(19-20):3350–3360, Oct 2014.

[83] John T Foster, Stewart Andrew Silling, and WAYNE W Chen. Viscoplasticity using peridynamics. *International journal for numerical methods in engineering*, 81(10):1242–1258, 2009.

[84] Stewart A Silling and Abe Askari. Peridynamic model for fatigue cracking. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), Oct 2014.

[85] James O'Grady and John Foster. Peridynamic plates and flat shells: A non-ordinary, state-based model. *International Journal of Solids and Structures*, 51(25-26):4572–4579, Dec 2014.

[86] James O'Grady and John Foster. Peridynamic beams: A non-ordinary, state-based model. *International Journal of Solids and Structures*, 51(18):3177–3183, Sep 2014.

[87] Michael Breitenfeld. *Quasi-static non-ordinary state-based peridynamics for the modeling of 3D fracture*. PhD thesis, Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, 2014.

[88] Ted Belytschko, Yun Yun Lu, Lei Gu, et al. Element free galerkin methods. *International journal for numerical methods in engineering*, 37(2):229–256, 1994.

[89] Wing Kam Liu, Sukky Jun, Yi Fei Zhang, et al. Reproducing kernel particle methods. *International journal for numerical methods in fluids*, 20(8-9):1081–1106, 1995.

[90] Robert A Gingold and Joseph J Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977.

[91] Leon B Lucy. A numerical approach to the testing of the fission hypothesis. *The astronomical journal*, 82:1013–1024, 1977.

[92] Rade Vignjevic and James Campbell. Review of development of the smooth particle hydrodynamics (sph) method. In *Predictive Modeling of Dynamic Processes*, pages 367–396. Springer, 2009.

[93] MA Bessa, JT Foster, T Belytschko, and Wing Kam Liu. A meshfree unification: reproducing kernel peridynamics. *Computational Mechanics*, 53(6):1251–1264, 2014.

[94] R Vignjevic, J Campbell, and L Libersky. A treatment of zero-energy modes in the smoothed particle hydrodynamics method. *Computer methods in Applied mechanics and Engineering*, 184(1):67–85, 2000.

[95] Stephen Beissel and Ted Belytschko. Nodal integration of the element-free galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 139(1):49–74, 1996.

[96] DP Flanagan and T Belytschko. A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. *International journal for numerical methods in engineering*, 17(5):679–706, 1981.

[97] David J Littlewood. A nonlocal approach to modeling crack nucleation in aa 7075-t651. In *ASME 2011 International Mechanical Engineering Congress and Exposition*, pages 567–576. American Society of Mechanical Engineers, 2011.

[98] MR Tupek, JJ Rimoli, and R Radovitzky. An approach for incorporating classical continuum damage models in state-based peridynamics. *Computer Methods in Applied Mechanics and Engineering*, 263:20–26, 2013.

[99] MR Tupek and R Radovitzky. An extended constitutive correspondence formulation of peridynamics based on nonlinear bond-strain measures. *Journal of the Mechanics and Physics of Solids*, 65:82–92, 2014.

[100] Sonia Fernández-Méndez and Antonio Huerta. Imposing essential boundary conditions in mesh-free methods. *Computer methods in applied mechanics and engineering*, 193(12):1257–1275, 2004.

[101] Gregory J Wagner and Wing Kam Liu. Application of essential boundary conditions in mesh-free methods: a corrected collocation method. *International Journal for Numerical Methods in Engineering*, 47(8):1367–1379, 2000.

[102] Gregory J Wagner and Wing Kam Liu. Hierarchical enrichment for bridging scales and mesh-free boundary conditions. *International Journal for Numerical Methods in Engineering*, 50(3):507–524, 2001.

[103] Kun Zhou and Qiang Du. Mathematical and numerical analysis of linear peridynamic models with nonlocal boundary conditions. *SIAM Journal on Numerical Analysis*, 48(5):1759–1780, 2010.

[104] CT Wu. Kinematic constraints in the state-based peridynamics with mixed local/nonlocal gradient approximations. *Computational Mechanics*, 54(5):1255–1267, 2014.

[105] Selda Oterkus. *Peridynamics For The Solution Of Multiphysics Problems*. PhD thesis, Department of Aerospace and Mechanical Engineering, University of Arizona, 2015.

[106] David J. Littlewood, Stewart A. Silling, John A. Mitchell, Pablo D. Seleson, Stephen D. Bond, Michael L. Parks, Daniel Z. Turner, Damon J. Burnett, Jakob Ostien, and Max Gunzburger. Strong local-nonlocal coupling for integrated fracture modeling. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States).

[107] Mirco Zaccariotto, Fabio Luongo, Giulia Sarego, and Ugo Galvanetto. Examples of applications of the peridynamic theory to the solution of static equilibrium problems. *The Aeronautical Journal*, 119(1216):677–700, June 2015.

[108] David J. Littlewood. Roadmap for peridynamic software implementation. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States).

[109] A Cemal Eringen. *Nonlocal continuum field theories*. Springer Science & Business Media, 2002.

[110] SA Silling and RB Lehoucq. Peridynamic theory of solid mechanics. *Advances in Applied Mechanics*, 44(1):73–166, 2010.

[111] Shaofan Li and Wing Kam Liu. Meshfree and particle methods and their applications. *Applied Mechanics Reviews*, 55(1):1–34, 2002.

[112] Wenyang Liu and Jung-Wuk Hong. A coupling approach of discretized peridynamics with finite element method. *Computer Methods in Applied Mechanics and Engineering*, 245:163–175, 2012.

[113] Erkan Oterkus, Erdogan Madenci, Olaf Weckner, Stewart Silling, Philip Bogert, and Alexander Tessler. Combined finite element and peridynamic analyses for predicting failure in a stiffened composite curved panel with a central slot. *Composite Structures*, 94(3):839–850, 2012.

[114] B Kilic and E Madenci. An adaptive dynamic relaxation method for quasi-static simulations using the peridynamic theory. *Theoretical and Applied Fracture Mechanics*, 53(3):194–204, Jun 2010.

[115] B Kilic and E Madenci. Structural stability and failure analysis using peridynamic theory. *International Journal of Non-Linear Mechanics*, 44(8):845–854, 2009.

[116] Quang Van Le and Florin Bobaru. Surface corrections in peridynamic models in elasticity and around cracks. submitted to International Journal of Solids and Structures, 2015.

[117] Huilong Ren, Xiaoying Zhuang, Yongchan Cai, and Timon Rabczuk. Dual-horizon peridynamics. *arXiv preprint arXiv:1506.05146*, 2015.

[118] Xi Chen and Max Gunzburger. Continuous and discontinuous finite element methods for a peridynamics model of mechanics. *Computer Methods in Applied Mechanics and Engineering*, 200(9):1237–1250, 2011.

[119] Alan A Griffith. The phenomena of rupture and flow in solids. *Philosophical transactions of the royal society of london. Series A, containing papers of a mathematical or physical character*, pages 163–198, 1921.

[120] Stewart A Silling. Linearized theory of peridynamic states. *Journal of Elasticity*, 99(1):85–111, 2010.

[121] QV Le, WK Chan, and J Schwartz. A two-dimensional ordinary, state-based peridynamic model for linearly elastic solids. *International Journal for Numerical Methods in Engineering*, 98(8):547–561, 2014.

[122] Pablo Seleson and Michael Parks. On the role of the influence function in the peridynamic theory. *International Journal of Multiscale Computational Engineering*, 9(6):689–706, 2011.

[123] Giulia Sarego, Quang V. Le, Florin Bobaru, Mirco Zaccariotto, and Ugo Galvanetto. Linearized state-based peridynamics for 2D problems. *International Journal for Numerical Methods in Engineering*, in press.

[124] Xiaochuan Tian and Qiang Du. Asymptotically compatible schemes and applications to robust discretization of nonlocal models. *SIAM Journal on Numerical Analysis*, 52(4):1641–1665, 2014.

[125] K Yu, XJ Xin, and KB Lease. A new adaptive integration method for the peridynamic theory. *Modelling and Simulation in Materials Science and Engineering*, 19(4):045003, 2011.

[126] Henk A Van der Vorst. Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM Journal on scientific and Statistical Computing*, 13(2):631–644, 1992.

[127] Numerical integration schemes. `http://www2.mpip-mainz.mpg.de/~andrienk/journal_club/integrators.pdf`. Accessed: 05-10-2015.

[128] Loup Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Phys. Rev.*, 159:98–103, Jul 1967.

[129] Joseph E Bishop. Simulating the pervasive fracture and fragmentation of materials and structures using randomly close-packed voronoi tessellations. Technical report, Sandia National Laboratories, 2008.

[130] Florin Bobaru, Mijia Yang, Leonardo Frota Alves, Stewart A Silling, Ebrahim Askari, and Jifeng Xu. Convergence, adaptive refinement, and scaling in 1d peridynamics. *International Journal for Numerical Methods in Engineering*, 77(6):852–877, 2009.

[131] Florin Bobaru and Monchai Duangpanya. The peridynamic formulation for transient heat conduction. *International Journal of Heat and Mass Transfer*, 53(19-20):4047–4059, Sep 2010.

[132] J Zhu, ZRL Taylor, and OC Zienkiewicz. *The finite element method: its basis and fundamentals*. Butterworth-Heinemann, 2005.

[133] K.J. Bathe and E.L. Wilson. *Numerical methods in finite element analysis*. Prentice-Hall civil engineering and engineering mechanics series. Prentice-Hall Englewood Cliffs, NJ, 1976.

[134] M.R. Hatch. *Vibration Simulation Using MATLAB and ANSYS*. Number v. 10 in Mechanical engineering / Applied mathematics. Taylor & Francis, 2000.

[135] René De Borst, Mike A Crisfield, Joris JC Remmers, and Clemens V Verhoosel. *Nonlinear finite element analysis of solids and structures*. John Wiley & Sons, 2012.

[136] George C Sih. Strain-energy-density factor applied to mixed mode crack problems. *International Journal of fracture*, 10(3):305–321, 1974.

[137] Alberto Carpinteri and Giovanni Colombo. Numerical analysis of catastrophic softening behaviour (snap-back instability). *Computers & structures*, 31(4):607–636, 1989.

[138] JF Kalthoff and S Winkler. Failure mode transition at high rates of shear loading. *DGM Informationsgesellschaft mbH, Impact Loading and Dynamic Behavior of Materials*, 1:185–195, 1988.

[139] Daniele Dipasquale, Mirco Zaccariotto, and Ugo Galvanetto. Crack propagation with adaptive grid refinement in 2d peridynamics. *International Journal of Fracture*, 190(1-2):1–22, 2014.

[140] Fazil Erdogan and GC Sih. On the crack extension in plates under plane loading and transverse shear. *Journal of Fluids Engineering*, 85(4):519–525, 1963.

[141] MR Ayatollahi and MRM Aliha. Cracked brazilian disc specimen subjected to mode ii deformation. *Engineering fracture mechanics*, 72(4):493–503, 2005.

[142] A. Ralston and P. Rabinowitz. *A First Course in Numerical Analysis: Second Edition*. Dover Books on Mathematics. Dover Publications, 2012.

[143] W. Hu, Y. D. Ha, and Florin Bobaru. Numerical integration in peridynamics. technical report, Department of Mechanical & Materials Engineering, University of Nebraska-Lincoln.

Appendix A

Old algorithms for BBP code

The initial code was more immediate and clear for a first approach to Peridynamics, but rather slow and memory consuming. A more advantageous code has been developed in the PhD course.

In the initial code, I_{bond} was a sparse matrix of the dimension $N \times N$, where the (n_x, n_p) component contains 1 if node n_x and node n_p had a connection, a "bond", or 0 if otherwise (see Figure A.1). It was needed as a reference matrix containing information on the existence of bonds. The data structure has been changed. The new algorithm is shown in Figure 3.5.

Algorithm for building the initial bond matrix

{Initialize the bond matrix I_{bond} to a zero matrix}
$I_{bond} = zeros(N, N)$
{Compute the distance between two generic nodes $n_{x,1}$ and node $n_{x,2}$}
{Compute a loop within the node ID numbers}
for $n_{x,1} = 1 : N$
{Compute a nested loop within the node ID numbers}
    for $n_{x,2} = 1 : N$
        $d(n_{x,1}, n_{x,2}) = norm(position(n_{x,2}, 1 : 3) - position(n_{x,1}, 1 : 3))$
    endfor
endfor
{Associate a unit value to the position in I_{bond} corresponding to an existing bond}
for $n_{x,1} = 1 : N$
$II = find(d(n_{x,1}, :) - \delta \leq 0)$
    for $tt = 1 : length(II)$
        $I_{bond}(i, II(tt)) = 1$
    endfor
endfor

Figure A.1: Algorithm for building the Initial bond matrix of the structure.

In the first code, the search for the family nodes was carried out not only when building I_{bond}, but also whenever a family node was recalled (i.e if n_x is the node in consideration, then the family nodes are the element of the set $n_{pj} = find(I_{bond}(n_x, :) > 0)$) in the computation, as you can see in Figure A.2. Note that the volume correction factor (*HHB algorithm* in [9]) has not been described in detail in the algorithm, even if present.

Algorithm for building the initial stiffness matrix

{Initialize the stiffness matrix K to a zero state}
$K = zeros(3 \times N, 3 \times N)$
{Compute a loop within the node ID numbers}
for $n_x = 1 : N$
{Find its family nodes}
$n_{pj} = find(I_{bond}(n_x, :) > 0)$
{Compute a loop within the family nodes n_{pj} of node n_x}
  for j$= 1 : length(n_{pj})$
    $n_p = n_{pj}(j)$
    {Compute the initial length of the bond}
    $\boldsymbol{\xi} = position(n_x, 1:3) - position(n_p, 1:3)$
    {Compute the volume correction factor of the bond}
    $f_V = f\left(\delta - |\boldsymbol{\xi}|\right)$
    {Compute the stiffness of the bond in the local coordinate system}
    $rig = \frac{1}{2}\frac{f_V \cdot c}{|\boldsymbol{\xi}|} \cdot V_{n_p} \cdot V_{n_x}$
    {Compute the rotation angles between the local and the global coordinate systems}
    $\alpha = |\arctan\left(\frac{\xi_2}{\xi_1}\right)|$
    $\phi = |\frac{\pi}{2} - \arctan\left(\frac{\xi_3}{\sqrt{\xi_1^2+\xi_2^2}}\right)|$
    {Compute the stiffness of the bond in the global coordinate system}
    $lx = \cos\alpha \cos\left(\frac{\pi}{2} - \phi\right)$
    $mx = \cos\left(\frac{\pi}{2} - \phi\right) \cos\left(\frac{\pi}{2} - \alpha\right)$
    $nx = \cos\phi$
    $[A] = rig \cdot \begin{bmatrix} lx^2 & lx \cdot mx & lx \cdot nx \\ mx \cdot lx & mx^2 & mx \cdot nx \\ nx \cdot lx & nx \cdot mx & nx^2 \end{bmatrix}$
    {Add $[A]$ to the stiffness matrix}
    $indices = [3n_x - 2 \quad 3n_x - 1 \quad 3n_x \quad 3n_p - 2 \quad 3n_p - 1 \quad 3n_p]$
    $c_{row} = [1 \quad 1 \quad 1 \quad -1 \quad -1 \quad -1]$
    $c_{column} = [1 \quad 1 \quad 1 \quad -1 \quad -1 \quad -1]$
    $qa = 1$
    for $j = 1 : 6$
      if $qa > 3$
        $qa = 1$
      endif
      $qb = 1$
      for $t = 1 : 6$
        if $qb > 3$
          $qb = 1$
        endif
        $K_{indices(j),indices(t)} = K_{indices(j),indices(t)} + c_{row}(j) \cdot c_{column}(t) \cdot A(qa, qb)$
        $qb = qb + 1$
      endfor
      $qa = qa + 1$
    endfor
  endfor
endfor

Figure A.2: Algorithm for building the stiffness matrix of the structure.

.

Appendix B

Assembling the stiffness matrix in BBP code

All the xx components of the stiffnesses of the bonds in the global reference systems are computed and stored in only one matrix (rig_{xx}); other two matrices are needed in order to stored the xy and yy components separately. rig_{xx}, rig_{xy} and rig_{yy} can be computed as explained in command lines 6, 7, 8 of Figure 3.9. The contributes have to be assembled as shown in section 3.2.2. The assembling can be carried out by the built-in function *sparse*, requiring three vectors as input: a vector containing the row indices, a vector containing the column indices and a vector containing the value of the stiffness to add. *Sparse* automatically add contributions when they belong to the same component in the final global matrix. For example,

$$\begin{cases} R_1(8) = 23 \\ C_1(8) = 15 \\ V_1(8) = 10^4 \end{cases} \quad \mapsto \quad K_{23,15} = K_{23,15} + 10^4 \tag{B.1}$$

Note that since they are vector, only one index is necessary to identify the element. So considering a bond, the related stiffness matrix is 4×4 in $x - y$ plane. Its components must be added to the appropriate rows ans columns, as shown in Eq. (B.2), where the relative bond connects point i to point j.

$$[K]_{bond} = \begin{array}{c} \\ 2i-1 \\ 2i \\ 2j-1 \\ 2j \end{array} \begin{array}{c} \begin{array}{cc} 2i-1 & 2i \end{array} \\ \begin{bmatrix} \begin{bmatrix} rig_{xx} & rig_{xy} \\ rig_{xy} & rig_{yy} \end{bmatrix} & \begin{bmatrix} -rig_{xx} & -rig_{xy} \\ -rig_{xy} & -rig_{yy} \end{bmatrix} \\ \begin{bmatrix} -rig_{xx} & -rig_{xy} \\ -rig_{xy} & -rig_{yy} \end{bmatrix} & \begin{bmatrix} rig_{xx} & rig_{xy} \\ rig_{xy} & rig_{yy} \end{bmatrix} \end{bmatrix} \end{array} \begin{array}{c} \\ \\ \begin{array}{cc} 2j-1 & 2j \end{array} \\ \\ \\ \end{array} \tag{B.2}$$

Consider the component $(1,1)$ of the $[K]_{bond}$, corresponding to the rig_{xx} value for the bond: for every source node i, it must be added to the position $(2i-1, 2i-1)$. In Matlab® a matrix can be seen as a juxtaposition of column vectors, that can be aligned to be one column vector by the ":" colon command:

$$V_1(1 : N \times M) = rig_{xx}(:) = \begin{bmatrix} \begin{bmatrix} rig_{xx}(1,1) \\ rig_{xx}(2,1) \\ \vdots \\ rig_{xx}(N,1) \end{bmatrix} \\ \begin{bmatrix} rig_{xx}(1,2) \\ rig_{xx}(2,2) \\ \vdots \\ rig_{xx}(N,2) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} rig_{xx}(N-1,M) \\ rig_{xx}(N,M) \end{bmatrix} \end{bmatrix} \tag{B.3}$$

Since rig_{xx} is a $N \times M$ matrix, V_1 is a $(N \cdot M) \times 1$ vector. The corresponding row and column indices for these values are then computed as:

$$R_1(1:N\times M)=\begin{bmatrix}2\times(1)-1\\2\times(2)-1\\\vdots\\2\times(N)-1\\2\times(1)-1\\2\times(2)-1\\\vdots\\2\times(N)-1\\\vdots\\2\times(N-1)-1\\2\times(N)-1\end{bmatrix}\quad C_1(1:N\times M)=\begin{bmatrix}2\times(1)-1\\2\times(2)-1\\\vdots\\2\times(N)-1\\2\times(1)-1\\2\times(2)-1\\\vdots\\2\times(N)-1\\\vdots\\2\times(N-1)-1\\2\times(N)-1\end{bmatrix} \tag{B.4}$$

which can be written as

$$\begin{aligned}R1(1:N\times M)&=repmat(1:2:2N,1,M)'\\C1(1:N\times M)&=repmat(1:2:2N,1,M)'\end{aligned} \tag{B.5}$$

The component $(1,2)$ of the $[K]_{bond}$ corresponds to the rig_{xy} value for the bond and it has to be added to the position $(2i-1,2i)$, which can be computed as

$$\begin{cases}R_1((N\times M)+1:2\,(N\times M))=repmat(1:2:2N,1,M)'\\C_1((N\times M)+1:2\,(N\times M))=repmat(2:2:2N,1,M)'\\V_1((N\times M)+1:2\,(N\times M))=rig_{xy}(:)\end{cases}$$

The component $(1,3)$ of the $[K]_{bond}$ corresponds to the $-rig_{xx}$ value for the bond and it has to be added to the position $(2i-1,2j-1)$, where j is the family bond. V_1 and R_1 can be computed as previously explained, while

$$C_1\left(2\,(N\times M)+1:3\,(N\times M)\right)=2\cdot I_{bond}(:)-1 \tag{B.6}$$

This approach is applied to all the components so that R_1 , C_1 and V_1 become $(16\,(N\times M),1)$ vectors to give as inputs to *sparse* function.

Appendix C

Old algorithms for SBP code

If the property is associated to each bond and it is a vector, the first attempts matrices had the same number of column, M, of the bond matrix I_{bond} and a number of rows that was twice (for 2D cases) or three times (for 3D cases) N, the number of rows of I_{bond}, since there is a row for every component of the property for every source node, like the reference state $\underline{X}$. For example in a two-dimensional simulation, for node $n_x = 1$ the associated rows are 1 and 2, for node $n_x = 2$ the associated rows are 3 and 4, so for a generic node n_x the associated rows are $2n_x - 1$ and $2n_x$.

$$I_{bond} = \begin{matrix} 1 \\ 2 \\ \vdots \\ \vdots \end{matrix} \begin{bmatrix} 2 & 3 & \cdots \\ 1 & 3 & \cdots \\ & \vdots & \\ & \vdots & \end{bmatrix} \quad \mapsto \quad \underline{X} = \begin{bmatrix} \xi_{x,(2-1)} & \xi_{x,(3-1)} & \cdots \\ \xi_{y,(2-1)} & \xi_{y,(3-1)} & \cdots \\ \xi_{x,(1-2)} & \xi_{x,(3-2)} & \cdots \\ \xi_{y,(1-2)} & \xi_{y,(3-2)} & \cdots \\ & \vdots & \\ & \vdots & \end{bmatrix} \tag{C.1}$$

In the first attempt, these states were computed by the command lines in Figures C.1, C.2, C.3 and C.4.

Algorithm for building the $\underline{X}$ state

1: {Initialize $\underline{X}$ state to a zero state }
2: $\underline{X} = zeros(3 \times N, M)$
3: {Compute a loop within the node ID numbers}
4: **for** $n_x = 1 : N$
5: {Compute a loop within the family nodes of node n_x}
6: **for** j$= 1 : M$
7: $n_p = I_{bond}(n_x, j)$
8: $\underline{X}(3n_x - 2, j) = position(n_p, 1) - position(n_x, 1)$
9: $\underline{X}(3n_x - 1, j) = position(n_p, 2) - position(n_x, 2)$
10: $\underline{X}(3n_x, j) = position(n_p, 3) - position(n_x, 3)$
11: **end for**
12: **end for**
13: **end for**

Figure C.1: Algorithm for building the $\underline{X}$ state for all the nodes of the grid.

Algorithm for building the influence function $\underline{\omega}$ state

1: {Initialize $\underline{\omega}$ state to a zero state }
2: $\underline{\omega} = zeros(N, M)$
3: {Compute a loop within the node ID numbers}
4: **for** $n_x = 1 : N$
5: {Compute a loop within the family nodes of node n_x}
6: **for** j$= 1 : M$
7: $n_p = I_{bond}(n_x, j)$
8: $\underline{\omega}(n_x, j) = f\left(\underline{X}(n_x, j)\right)$
9: **end for**
10: **end for**

Figure C.2: Algorithm for building the $\underline{\omega}$ state for all the nodes of the grid, considering that $\underline{\omega} = f\left(\underline{X}\right)$.

Algorithm for building the volume correction factor $\underline{f_V}$ state and the scalar reference state $|\underline{X}|$

{Initialize $\underline{f_V}$ and $|\underline{X}|$ states to zero states }
$|\underline{X}| = \overline{zeros}(N, M)$
$\underline{f_V} = zeros(N, M)$
{Compute a loop within the node ID numbers}
for $n_x = 1 : N$
{Compute a loop within the family nodes of node n_x}
    for j= $1 : M$
        $n_p = I_{bond}(n_x, j)$
        $\xi = \sqrt{\sum_{k=1}^{3} \left[position(n_x, k) - position(n_p, k)\right]^2}$
        if $\xi <= 0.0$
            $\underline{f_V}(n_x, j) = 0.0$
            $\overline{|\underline{X}|}(n_x, j) = 0.0$
        elseif $\xi <= \delta - \frac{1}{2}\Delta x$
            $\underline{f_V}(n_x, j) = 1.0$
            $\overline{|\underline{X}|}(n_x, j) = \xi$
        elseif $\xi <= \delta + \frac{1}{2}\Delta x$
            $\underline{f_V}(n_x, j) = \frac{\delta + \Delta x/2 - \xi}{\Delta x}$
            $|\underline{X}|(n_x, j) = \delta - \frac{\Delta x}{2} \cdot \underline{f_V}(n_x, j)$
        end if
    end for
end for

Figure C.3: Algorithm for building the volume correction factor $\underline{f_V}$ and the scalar reference state $|\underline{X}|$ for all the nodes of the grid.

In particular, in Figure C.3 the algorithm explained by Bobaru et al. in [143] is employed.

Algorithm for building m weight

1: {Initialize m to a zero state }
2: $m = zeros(N, 1)$
3: {Compute a loop within the node ID numbers}
4: **for** $n_x = 1 : N$
5: {Compute a loop within the family nodes of node n_x}
6: **for** j$= 1 : M$
7: $n_p = I_{bond}(n_x, j)$
8: $m(n_x, 1) = m(n_x, 1) + \underline{\omega}(n_x, j) \cdot |\underline{X}|^2(n_x, j) \cdot \underline{f_V}(n_x, j) \cdot V_{n_p}$
9: **end for**
10: **end for**

Figure C.4: Algorithm for building the m weight for all the nodes of the grid.