

Fall 11-14-2017

State-based Peridynamic Particle Method

Siavash Nikravesh Kazeroni
University of New Mexico

Follow this and additional works at: https://digitalrepository.unm.edu/ce_etds

 Part of the [Computational Engineering Commons](#), and the [Structural Engineering Commons](#)

Recommended Citation

Nikravesh Kazeroni, Siavash. "State-based Peridynamic Particle Method." (2017). https://digitalrepository.unm.edu/ce_etds/189

This Thesis is brought to you for free and open access by the Engineering ETDs at UNM Digital Repository. It has been accepted for inclusion in Civil Engineering ETDs by an authorized administrator of UNM Digital Repository. For more information, please contact disc@unm.edu.

Siavash Nikravesh Kazeroni
Candidate

Civil Engineering
Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

Dr. Walter Gerstle, Chairperson

Dr. Yu-Lin Shen

Dr. Fernando Moreu

State-based Peridynamic Particle Method

by

Siavash Nikravesh Kazeroni

M.Sc., Civil Engineering-Hydraulic Structures/Computational Mechanics,
Iran University of Science and Technology, Tehran, Iran, 2014

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science in Civil Engineering

The University of New Mexico

Albuquerque, New Mexico

December, 2017

DEDICATION

*To my dear parents, Susan and Reza, and my dear brother, Arash,
for their support and love.*

ACKNOWLEDGMENTS

I heartily acknowledge Dr. Walter Gerstle, my advisor and thesis chair, for his support throughout this research. His guidance and professional style will remain with me as I continue my career. I also thank my committee members, Dr. Yu-Lin Shen, and Dr. Fernando Moreu for their valuable recommendations pertaining to this study and assistance in my professional development.

I want to thank the Center for Advanced Research Computing (CARC) for the computational resources used in this work.

I want to thank my friends, Dr. Saman Rashidyan, and Mojgan Madandar, for their help and support.

State-Based Peridynamic Particle Method

by

Siavash Nikravesi Kazeroni

*M.Sc., Civil Engineering-Hydraulic Structures/Computational Mechanics,
Iran University of Science and Technology, Tehran, Iran, 2014*

*Second M.Sc., Civil/Structural Engineering.
Certificate degree, Computational Science and Engineering (CSE),
University of New Mexico, Albuquerque, NM, USA, 2017*

ABSTRACT

In this study, a novel discrete Peridynamics framework called the “State-Based Peridynamic Particle Model (SPPM)” is introduced. In this approach, a solid body is simulated by neither solving differential equations nor integral equations; instead, the simulation is accomplished by directly solving discrete systems of equations using finite summations. SPPM is formulated for a random distribution of particles, hence, it can be considered as a meshfree method. The assumptions of continuity and homogeneity are not necessary in this approach. The SPPM is a generalization of the “State-Based Peridynamic Lattice Model (SPLM)”. In the SPLM formulation, for sake of simplicity and computational efficiency, a lattice of particles is employed and the horizon size is fixed. The proposed SPLM approach differs from the previous versions in that the procedures for calculating the bond forces, damage and plasticity are improved. A novel and robust damage approach called the “Two Spring Damage Model”, with the capability of modeling partial damage, is also proposed and developed for the SPPM and the SPLM.

The re-formulated SPLM method is then calibrated and employed to simulate concrete structures. The obtained results are compared with the previous SPLM versions, experimental tests, and the commercial finite element software, Abaqus. The advantages and difficulties of each modeling approach are described. The re-formulated SPLM demonstrates significant improvements over the previous versions. The obtained simulation results indicate that the SPLM approach produces similar, and in some ways more realistic results than the well-developed Abaqus methods, but is much simpler to understand and use. The obtained results also reasonably replicate the available laboratory data.

TABLE OF CONTENTS

Chapter 1 Introduction.....	1
Chapter 2 Literature Review	4
2.1. Abaqus Brittle Cracking Model	4
2.2. Abaqus Plasticity Damage Model	5
2.3. Abaqus Smeared Cracking Model.....	9
2.4. Summary and Conclusions regarding the Abaqus Concrete Models	9
2.5. Peridynamic Models.....	10
Chapter 3 State-Based Peridynamics Particle Model.....	11
3.1. Introduction	11
3.2. SPPM Linear-Elastic Formulation	13
3.3. SPPM Two Spring Damage Model.....	18
3.4. SPPM Plasticity Model	22
3.5. Numerical Implementation of SPPM	24
Chapter 4 Re-formulated State-Based Peridynamic Lattice Model	28
4.1. Introduction	28
4.2. Linear-Elastic SPLM.....	29
4.3. SPLM Damage-Plasticity Model for Concrete	32
Chapter 5 Numerical Results.....	39
5.1. Introduction	39
5.2. Re-formulated SPLM versus the older versions	39
5.2.1. Material properties and parameters.....	39
5.2.2. Uniaxial Tension Problem	41
5.2.3. Uniaxial Compression Problem	44
5.2.4. Split Brazilian Cylinder	47
5.3. Re-formulated SPLM versus Abaqus and Experimental data.....	50
5.3.1. Material properties and parameters.....	50
5.3.2. Dog-Bone Specimens Under Uniaxial Tension.....	52
5.3.3. Brazilian Split Cylinder	58
5.4. Convergence Study	61
Chapter 6 Discussion and Conclusions	63

6.1. Summary	63
6.2. Discussion and Remarks	64
6.3. Final Thoughts.....	66
6.4. Future Studies.....	66
References	68

Chapter 1

Introduction

A reliable simulation tool for concrete structures, with the ability to predict damage and fracture, would aid structural engineers.

In recent decades, computers have greatly improved. Consequently, various computational tools and numerical techniques have been developed by engineering professionals and academics. Among the proposed numerical approaches, the finite element method (FEM) has been the most successful. Nowadays, numerous FEM-based applications and codes are available for both industrial and academic use. Despite significant achievements, the FEM, and more generally, classical mechanics, have been rather unsuccessful in simulating strain-softening and damage in solids. Although methods have been developed for simulating damage and fracture, the FEM generally has limited capability with respect to damage and fracture. Other difficulties with FEM models can be addressed as the necessity for having a well-structured mesh, and problems regarding modelling moving boundary and large deformation problems.

Recently, some of the difficulties with FEM, related to mesh generation, are overcome by introducing new types of methods called “meshless or mesh-free” approaches [2-4]. It has been claimed that meshless techniques provide more accuracy, decrease discretization costs, produce more flexibility in modelling complicated boundaries, and facilitate advanced adaptive refinement methods [5]. However, mesh-free approaches typically require more computational effort than conventional FEM approaches due to the expense of meshless shape function construction [6]. Modelling progressive damage and cracking are also approached by application of specific constitutive material models such as nonlocal microplane models [7] and implementation of smeared crack models [8], etc., within the continuum mechanics framework.

The commercial code “Simulia Abaqus” is one of the most prominent commercial nonlinear FEM software packages available. Abaqus includes three different concrete constitutive models. All three Abaqus standard concrete cracking models are based upon smeared cracking approaches. The Abaqus concrete models are named the “brittle cracking model”, the “smeared cracking model”, and the “damaged plasticity model” [9, 10]. The Abaqus product suite contains the Abaqus/Standard Solver (an implicit solver that is designed to efficiently solve static and low speed dynamic problems), and the Abaqus/Explicit Solver (an explicit solver designed for efficient solution of nonlinear dynamic problems and recommended for nonlinear analysis). The brittle cracking model is implemented on Abaqus/Explicit, the damage plasticity model is implemented on both Abaqus/Explicit and Abaqus/Standard, and the smeared cracking model is only available in Abaqus/Standard [9, 10]. The Abaqus concrete cracking models are studied in this research.

All the mentioned approaches (FEM, Meshless, etc.) are designed to solve spatio-temporal partial differential equations and are based upon continuum mechanics theory. On the other

hand, in 2000, an alternative approach, called “peridynamic” theory, a re-formulation of continuum mechanics in terms of nonlocal forces, was proposed. Peridynamics was initially introduced by Silling [11]. The peridynamic model avoids an assumption of differentiability of the displacement field. In this theory, the concept of stress is replaced by a nonlocal pairwise force, which is a function of particle positions. Two different peridynamics approaches were proposed. In the first approach, called the “bond-based peridynamic model [11]” the pairwise force function between two interacting particles is assumed to be a function only of the relative initial position and the relative displacement between the interacting pair of particles. The bond-based model was found to be insufficiently general, requiring a non-arbitrary Poisson’s ratio and lack of the capability to adequately model the plastic volumetric deformations. In a later paper, the issue of the non-arbitrary Poisson’s ratio was resolved by the development of the “micropolar bond-based peridynamic model [12]” which includes the rotational and moment degrees of freedom. In 2007, a second continuum peridynamics approach, called the “state-based peridynamic model”, was published [13]. In the state-based approach, the pairwise force function is not only a function of the positions of the two adjacent interacting particles, but is also a function of other neighboring particles. The state-based model allowed for more general solid models, without the mentioned limitations of the bond-based method. However, compared to the bond-based method, the state-based approach is more complex. In addition, due to having more particles involved in computation of the pairwise force function, the computational cost of the state based approach is higher than that of the bond-based model.

It should be also noted that in both the bond-based and the state-based peridynamic approaches, the reference material space is treated as a continuum. Hence, the mentioned Silling’s methods are continuum peridynamics models. In 2015, Gerstle re-formulated the state-based peridynamic in a non-continuum, integer Cartesian, solid material space. The proposed method is called the “state-based peridynamic lattice model (SPLM) [14]”, in which the material geometry is discretized by a finite number of particles with a lattice configuration. By employing the lattice topology for particles, the number of neighboring particles and also the reference locations of the neighboring particles in every peridynamic horizon are fixed. At each time step of the analysis, each particle “knows” which particles it should interact with (without need for any extra calculations). The topological neighborhood of each particle remains invariant throughout the simulation. The computational implementation of SPLM is much simpler than conventional continuum peridynamic methods in the sense that storing and loading the data is easier and there is no need for numerical integration and other complex numerical operations. Therefore, SPLM can be more efficient, in terms of computation effort and accuracy, than continuum peridynamic approaches.

Despite the novelty and the addressed advantages, further studies showed that the initial version of SPLM [14] was insufficiently objective for models with damage. The simulations including damage did not match the classical solutions, and did not accurately

replicate experimental behavior; did not converge well with lattice refinement, and were sensitive to lattice rotation and translation.

In this thesis, firstly, a new formulation of discrete Peridynamics called the “State-Based Peridynamics Particle Model (SPPM)” is introduced. In this approach, which can be considered as a mesh-free method in a peridynamic framework; the assumption of a continuum domain is relaxed, and the classical spring theory is combined and strengthened with the concepts of state-based peridynamics. In addition, a damage model with the capability of modelling partial damage, named the “two-spring damage model”, is developed within the SPPM framework. The two-spring damage model employs the concepts of combined spring systems as well as state based peridynamic ideas. This approach is considered a particle-based damage approach since the damage and failure are associated with the particles instead of bonds. A suitable plasticity method is also proposed for the SPPM framework. Afterwards, the improved SPLM (which is a special case of the SPPM) is also proposed. The improved SPLM method is then demonstrated, by solving several plain stress concrete problems. The results are compared with those of FEM, experimental, and theoretical solutions.

In this thesis, we investigate and evaluate the capabilities of the improved SPLM in the simulation of plain concrete. We demonstrate conformity with laboratory test results, and compare the SPLM results with the various Abaqus concrete cracking models, as well as the classical solutions.

We investigate and evaluate the advantages and disadvantages of each modelling approach.

The obtained results demonstrate promising capability and efficiency of the SPLM method in modelling concrete structures. The SPLM/SPPM approach might be more computationally efficient and more accurate than the continuum peridynamics approach. Despite the relatively simple algorithm, the introduced SPLM/SPPM method may provide more realistic damage and cracking simulations than those of well-developed commercial FEM codes like Abaqus.

Chapter 2

Literature Review

In this chapter, the key features of Abaqus concrete cracking models, including assumptions, limitations, brief theory, and input setup are presented. More details can be found in the Abaqus user's manuals [9, 10]. In addition, a brief discussion of the available peridynamic models is presented.

2.1. Abaqus Brittle Cracking Model

This method, applicable for plain or reinforced concrete, is the simplest of the Abaqus cracking approaches to understand, implement and calibrate. This approach is for the applications in which the concrete behavior is dominated by tensile cracking. The compressive behavior is assumed to be linear elastic [9]. This model is claimed to be applicable for modelling any kind of concrete structure (beam, truss, solid, etc.). The main idea of this approach is that when the maximum principal tensile stress exceeds the tensile strength of the brittle material, the crack forms. The formed crack is assumed to be irrecoverable (remains constant) and the constant crack surface is defined normal to the direction of maximum tensile principal stress. These are questionable assumptions because of the changing the direction of principal stresses over time.

In this approach, Modes I and II fracture (tension and shear softening/retention) can be implemented [9, 10]. Tension softening (Mode I fracture) is based on the fictitious fracture concept of Hillerborg [15], and can be defined by tabulating the tensile strength of concrete (post peak behavior) as a function of either crack opening strain, ε^{cr} , or crack opening displacement, COD . The fracture process zone is large in concrete, and the assumption of the crack opening strain being normal to the fixed crack surface is questionable. The relation between the COD and crack opening strain, ε^{cr} , is defined as follows.

$$COD = \varepsilon^{cr} L_e, \quad (2.1)$$

where L_e is the characteristic length of the cracked elements. The definition of the element size, and subsequently the characteristic length of elements, are vague in all the Abaqus models. Abaqus manual define the characteristic length of the first order square elements as $L_e = \bar{L}\sqrt{2}$, (\bar{L} is the element size: side of the square elements); however, even for the structured mesh construction, having a perfectly square element may not be possible due to the shape of the geometry. Consequently, choosing a correct value of characteristic length is almost impossible for most of the cases. Moreover, based on Eq. (2.1), the crack opening strain is essentially a function of characteristic length of the cracked element; hence, by changing the element size, the crack opening strain should be re-calculated. Consequently, using stress-crack opening strain option makes the results mesh dependent, and thus non-objective. However, defining stress-crack opening displacement relation (which is an alternative option to stress-crack opening strain) can decrease, although not eliminate, the mesh dependency of the model.

Shear softening, necessary to model Mode II fracture, can be also implemented via the shear retention model [10]. In this arguable approach, the reduction in shear modulus due to crack opening is specified by defining the post-cracked shear stiffness, G_c , as a function of uncracked shear modulus, G , as

$$G_c = \rho G, \quad (2.2)$$

where ρ is the shear retention factor which is a function of crack opening strain. In the software, the value of shear retention factor can be assigned as a tabular function of crack opening strain across the crack. In this thesis, a bilinear relation between the shear retention factor and the crack opening strain is specified, in correspondence with Fig. (4.4) in chapter 4. As illustrated, defining Mode II fracture behavior is optional in the program, however, employing it will make the model mesh dependent due to the need for defining the crack opening strain (due to the mentioned issues with re-calculating the characteristic length of elements for different element sizes).

Besides of the mesh dependency and other mentioned limitations, although the Abaqus brittle cracking model seems to have the capability of modelling partial tensile damage, partial damage cannot be visualized, as no contour plots are available for partial damage. Instead, Abaqus can show only complete damage, by removing the fully-cracked elements based on a failure criterion [9]. Element removal is another controversial issue with this model and seems is not rational in the cases where transverse compression exists (since failed materials under tension are expected to withstand some compressive stress). When cracking strain or displacement at a material point reaches the user-defined failure value, the material point fails and all the stress components are set to zero. Element removal then takes place when all of the material points in an element have failed [9]. Element removal can be disabled and the Abaqus manual refers the effective use of element removal to the user and their own knowledge of structural behavior (which is questionable and disqualifies the objectivity of the approach). More details about this model can be found elsewhere [10].

The brittle cracking approach can only allow the user to visualize the ultimate damage through element removal via the STATUS contours [9]. Therefore, the STATUS contours are used in Chapter 4 of this thesis.

2.2. Abaqus Plasticity Damage Model

The plasticity damage model implemented in Abaqus is based upon the damage-plasticity models proposed by Lubliner [16] and by Lee and Fenves [17]. This model claims to provide a comprehensive capability for modeling plasticity and damage in all kinds of concrete structures. However, defining the input parameters and calibrating the model is more complicated than the other proposed concrete cracking methods in Abaqus. The description of the model in the Abaqus manual is not very clear; hence, some of the key formulations are simplified and illustrated as follows.

The plasticity damage model essentially assumes that the main two failure mechanisms are tensile cracking and compressive crushing of the material [9]. This method assumes a reduction in the material's elastic stiffness, \mathbf{D}^{el} , by defining the scalar damage degradation parameter, d_p , as

$$d_p = 1 - (1 - s_t d_c)(1 - s_c d_t), \quad (2.3)$$

and

$$\mathbf{D}^{el} = (1 - d_p)\mathbf{D}_0^{el}, \quad (2.4)$$

where \mathbf{D}_0^{el} is the uncracked elastic stiffness, and d_c and d_t are, respectively, the compressive and tensile damage parameters. Note that d_p , d_c , and d_t vary between 0 (no damage) to 1 (full damage). s_c and s_t are the stiffness recovery parameters (essentially for cyclic loading) which are defined as functions of a "stress weight factor", $r=r(\boldsymbol{\sigma})$, and the recovery weight factors, w_c and w_t , as follows:

$$s_t = 1 - w_t r, \quad (2.5)$$

$$s_c = 1 - w_c(1 - r), \quad (2.6)$$

where

$$r = \frac{\frac{1}{2}(\sum_{i=1}^n (|\sigma_i| + \sigma_i))}{\sum_{i=1}^n |\sigma_i|}; \quad (2.7)$$

and w_c , w_t , and r vary between 0 and 1, σ_i are the principal stress components, and n is the number of principal stresses ($n=2$ for the plane stress case and $n=3$ for the three-dimensional multiaxial condition). In this damage model, there are four user input parameters, as well as tension softening and compressive hardening inputs, that must be defined by user: w_c , w_t , d_c , and d_t . In this thesis, since the cyclic loading steps are not considered, w_t and w_c are, respectively, assumed to be zero and one (default values). Note that per Eq. (2.7), in the case of uniaxial tension, $\sigma_i = \sigma_1 > 0$, hence, $r=1$; and for uniaxial compression $\sigma_i = \sigma_1 < 0$, hence, $r=0$.

Accordingly, the stress-strain relation for a multiaxial condition is defined as

$$\boldsymbol{\sigma} = \mathbf{D}^{el}(\boldsymbol{\varepsilon} - \tilde{\boldsymbol{\varepsilon}}^{pl}) = (1 - d)\mathbf{D}_0^{el}(\boldsymbol{\varepsilon} - \tilde{\boldsymbol{\varepsilon}}^{pl}) = (1 - d)\bar{\boldsymbol{\sigma}}, \quad (2.8)$$

where, $\boldsymbol{\varepsilon}$ and $\tilde{\boldsymbol{\varepsilon}}^{pl}$ are, respectively, the total strain and equivalent plastic strain vectors, and $\bar{\boldsymbol{\sigma}}$ is the effective cohesion stress vector. In case of uniaxial loading, Eq. (2.8) can be simplified for uniaxial tension, Eq. (2.9), and compression, Eq. (2.10), as follows

$$\begin{aligned}\sigma_t &= E(\varepsilon_t - \tilde{\varepsilon}_t^{pl}) = (1 - d_t)E_0^{el}(\varepsilon_t - \tilde{\varepsilon}_t^{pl}) \\ &= (1 - d_t)\bar{\sigma}_t,\end{aligned}\quad (2.9)$$

$$\begin{aligned}\sigma_c &= E(\varepsilon_c - \tilde{\varepsilon}_c^{pl}) = (1 - d_c)E_0^{el}(\varepsilon_c - \tilde{\varepsilon}_c^{pl}) \\ &= (1 - d_c)\bar{\sigma}_c,\end{aligned}\quad (2.10)$$

where E_0^{el} is the uncracked elastic modulus. Therefore, from Eqs. (2.9) and (2.10), the following relationships for $\tilde{\varepsilon}_c^{pl}$ and $\tilde{\varepsilon}_t^{pl}$ are derived as follows.

$$\tilde{\varepsilon}_t^{pl} = \varepsilon_t - \frac{\sigma_t}{(1 - d_t)E_0^{el}}, \quad (2.11)$$

$$\tilde{\varepsilon}_c^{pl} = \varepsilon_c - \frac{\sigma_c}{(1 - d_c)E_0^{el}}. \quad (2.12)$$

In the damage-plasticity model, the tension softening behavior can be defined exactly in the same way as in the brittle cracking model (shown in Fig. (4.4)). The tensile damage parameter, d_t can be specified as a function of either cracking strain, $\tilde{\varepsilon}_t^{cr}$, or u_t^{cr} (crack opening displacement). The software automatically converts $\tilde{\varepsilon}_t^{cr}$ to $\tilde{\varepsilon}_t^{pl}$. The cracking strain, $\tilde{\varepsilon}_t^{cr}$, is defined as

$$\tilde{\varepsilon}_t^{cr} = \varepsilon_t - \frac{\sigma_t}{E_0^{el}}. \quad (2.13)$$

By substituting ε_t from Eq. (2.13) into Eq. (2.11),

$$\tilde{\varepsilon}_t^{pl} = \tilde{\varepsilon}_t^{cr} - \frac{d_t \sigma_t}{(1 - d_t)E_0^{el}}. \quad (2.14)$$

In terms of plastic displacement, u_t^{pl} , Eq. (2.14) can be rewritten as

$$u_t^{pl} = u_t^{cr} - \frac{d_t \sigma_t l_0}{(1 - d_t)E_0^{el}}, \quad (2.15)$$

where l_0 is the specimen length (assumed to be one unit length, $l_0 = 1$). Also, d_t , is specified, from Eq. (2.14), as

$$d_t = \frac{(\tilde{\varepsilon}_t^{cr} - \tilde{\varepsilon}_t^{pl})E_0^{el}}{\sigma_t + (\tilde{\varepsilon}_t^{cr} - \tilde{\varepsilon}_t^{pl})E_0^{el}} = \frac{(\tilde{\varepsilon}_t^{cr} - \tilde{\varepsilon}_t^{pl})}{(\sigma_t/E_0^{el} + \tilde{\varepsilon}_t^{cr} - \tilde{\varepsilon}_t^{pl})}, \quad (2.16)$$

and from Eq. (2.16) as

$$d_t = \frac{(u_t^{cr} - u_t^{pl})}{(\sigma_t/E_0^{el} + u_t^{cr} - u_t^{pl})}. \quad (2.17)$$

If the calculated equivalent plastic strain (or plastic displacement), based on Eqs. (2.14, and 2.15), values become negative, Abaqus will give an error message and the analysis will terminate. Hence, in specifying d_t , the user should make sure that $\tilde{\varepsilon}_t^{pl} \geq 0$ (or $u_t^{pl} \geq 0$).

Moreover, as can be seen from Eqs. (2.11) and (2.12), we have $(1 - d_t)$ in the denominator; therefore, the user cannot specify the damage parameter exactly as one. It is recommended to avoid using values of the damage parameters above 0.99, which corresponds to a 99% reduction of the stiffness [9]. The user may face a material data regularization error if the smallest defined interval (in this case for the damage parameters) is small compared to the range of the independent variable. In such a condition, the user should either redefine the material data, in this case redefine damage parameters, (which may not be practical due to possibility of getting the other mentioned errors) or change the tolerance value in the property modulus [9]. The mentioned issues make the definition of damage parameters extremely complicated.

Similarly, for compressive crushing behavior, the user must tabulate the compressive strength as a function of so the called inelastic (or crushing) strain, $\tilde{\varepsilon}_c^{in}$, which is defined as

$$\tilde{\varepsilon}_c^{in} = \varepsilon_c - \frac{\sigma_c}{E_0^{el}}. \quad (2.18)$$

The compressive damage parameter should also be specified as a function of $\tilde{\varepsilon}_c^{in}$, and the software will automatically convert the inelastic strain values to compressive plastic strain, $\tilde{\varepsilon}_c^{pl}$. Following a similar procedure as with the tensile damage parameter, the following equations are obtained.

$$\tilde{\varepsilon}_c^{pl} = \tilde{\varepsilon}_c^{in} - \frac{d_c \sigma_c}{(1 - d_c) E_0^{el}}, \quad (2.19)$$

and

$$d_c = \frac{(\tilde{\varepsilon}_c^{in} - \tilde{\varepsilon}_c^{pl})}{(\sigma_c / E_0^{el} + \tilde{\varepsilon}_c^{in} - \tilde{\varepsilon}_c^{pl})}, \quad (2.20)$$

Note that we have the same issues with analysis errors in defining the compressive damage parameter as in the tensile case.

In this thesis, to keep the models simple, elastic-perfectly plastic behavior is assumed for concrete in compression. In addition, the tensile and compressive damage parameters are specified similar to [18] and corresponding to the assumed tension softening and compressive crushing mechanisms. In order to avoid getting the error of negative converted plastic strain value, Eqs. (2.17) and (2.20) are checked in the process of defining the damage parameters.

In the damage plasticity model, the yield surface is defined based on the yield function, $F(\bar{\sigma}, \tilde{\varepsilon}^{pl})$ [10]; and the Drucker-Prager hyperbolic flow potential function is used for the plastic flow rule [10]. In Abaqus, there are five input parameters related to plasticity and the yield surface that must be set by user [9]. These parameters are: biaxial/uniaxial compressive yield stress ratio, σ_{b0}/σ_{c0} , with the default value of 1.16; dilation angle, ψ ,

with the default value of 15 degrees; the ratio of the second stress invariant on the tensile meridian to compressive meridian at initial yield, K_c , with a default value of 2/3; flow potential eccentricity, ϵ , with a default value of 0.1; and viscosity, μ , with a default value of zero. In this thesis, due to insufficient experimental data, all the mentioned plasticity input parameters are set to the default values. In addition, the viscoplastic regularization feature [10] is not considered in this thesis.

Among the Abaqus approaches, only the damage plasticity method has the capability of contouring the partial damage directly using the software's visualization package. In this thesis, SDEG contours, which is the damage parameter (d_p), introduced in Eq. (2.3), are presented.

2.3. Abaqus Smeared Cracking Model

The Abaqus smeared cracking model is another proposed Abaqus approach that claims to provide a general capability to simulate the post-cracking response of plain and reinforced concrete structures.

The smeared cracking model has the capability to model Mode I fracture (tension softening), Mode II fracture (via shear retention), and the compressive behavior of concrete [9, 10]. In the smeared cracking model, two independent yield functions of F_t and F_c are defined for the tensile and compressive yield surfaces [10]. When the state of stress is predominantly tensile, the model uses the "crack detection" or "failure" surface (based on F_t), and in the case of having a dominantly compressive stress state, it uses the "compressive" or "yield" surface (based on F_c). Both F_t and F_c are defined as functions of first and second stress tensor invariants (p and q). Four failure ratios must be specified by the user in Abaqus [9].

Unlike the other Abaqus models, this model is only implemented in Abaqus/Standard, which is an implicit solver. Although the dynamic-implicit solver claims to use automatic time incrementation; the time integration is not fully automatic and the user must define the maximum number of time increments as well as initial and minimum increment size. Defining a proper value for the mentioned parameters is not easy for the user, and affects the obtained results, total computational time, and convergence behavior of the simulation. The mentioned difficulties cause a lack of objectivity of this approach. In addition, this approach is essentially applicable only for monotonic loadings under low confining pressures (less than $5f'_c$) [10]. Therefore, due to the limited simulation capabilities, the smeared cracking approach is not considered in this study.

2.4. Summary and Conclusions regarding the Abaqus Concrete Models

As discussed, the commercial FEM software package, Abaqus, presents three standard methods particularly for modeling damage and fracture in concrete structures. The main difficulties of the Abaqus concrete cracking models are summarized as follows.

- 1) Limited modelling capabilities and complexity in defining the input parameters. As discussed, for calibrating some of the approaches (for instance the damage plasticity method) it is required from the user to define more than ten input parameters; while, there are no such experimental data for some of those parameters. The authenticity of the main theories of the developed methods (in accommodating with the real structural behavior of concrete) are questionable. On the other hand, the simpler models (like brittle cracking) have limited simulation abilities and simplified assumptions which do not capture the real physical behavior of concrete.
- 2) Mesh dependency. All the Abaqus concrete methods are mesh dependent; in that the tension softening behavior is modeled using the fictitious crack approach which requires the characteristic length of the cracked elements. Consequently, in the case of using non-rectangular elements, the difference in the characteristic length of the elements and what is defined in input data leads to some inaccuracies in the results.
- 3) Convergence issues. Mesh refinement study is not recommended since it leads to narrower crack bands and, therefore, different results. In other words, the results will not converge to a unique solution by performing the convergence study. This issue is explicitly mentioned in the Abaqus manual [9].

In conclusion, none of the Abaqus concrete cracking models are satisfactory for modelling real structural behavior of concrete; which is addressed as lack of objectivity of the proposed models. Nevertheless, among the presented methods, the Abaqus damage plasticity model seems to be more powerful and more realistic. The Abaqus brittle cracking model is handy and easy to calibrate, however, the limitations of this method should be clear for the user before starting to employ this approach.

2.5. Peridynamic Models

The continuum peridynamics theory is introduced by Silling [11, 13] and developed for modelling solid mechanics [19]. Different forms of discretization are employed such as finite element discretization [20], and meshfree discretization [21, 22] to solve the continuum peridynamics integral equations numerically. The issues related to continuum peridynamic approaches are discussed in detail in the following chapters.

About the research in the field of simulation of concrete structures using peridynamics, there are a limited number of publications in the literature. The first effort in modelling concrete structures with the bond-based peridynamics theory is performed by Gerstle and colleagues [12, 23, 24]. Some more recent studies based on the state-based theory are also available in the literature [25].

Chapter 3

State-Based Peridynamics Particle Model

3.1. Introduction

Continuum state-based peridynamic theory was initially proposed by Silling [13] as an alternative to classical continuum mechanics methods; which was expressed as

$$\rho(\mathbf{x})\ddot{\mathbf{u}}(\mathbf{x}, t) = \int_{H_x} \mathbf{f}_{xx'} dV_{x'} + \mathbf{B}(\mathbf{x}, t), \quad (3.1)$$

where \mathbf{x} and \mathbf{x}' are, respectively, the position vector of two neighboring particles in the spherical horizon of particle \mathbf{x} . ρ is the mass density associated with particle \mathbf{x} in the reference configuration, \mathbf{u} is the displacement vector field, \mathbf{B} is the body force vector, and $\mathbf{f}_{xx'}$ is a function called “pairwise force function”, introduced in form of Eq. (3.2) as

$$\mathbf{f}_{xx'} = \underline{\mathbf{T}}[\mathbf{x}, t]\langle \mathbf{x}' - \mathbf{x} \rangle - \underline{\mathbf{T}}[\mathbf{x}', t]\langle \mathbf{x} - \mathbf{x}' \rangle. \quad (3.2)$$

Here, $\underline{\mathbf{T}}$ is a function namely “force vector state field” or “force state”; which could be defined as a function of the deformation of all the bonds connected to particle \mathbf{x} in the neighborhood, H_x , of particle \mathbf{x} . The mathematical notation $\underline{\mathbf{T}}[\mathbf{x}, t]\langle \mathbf{x}' - \mathbf{x} \rangle$ means that the force state, $\underline{\mathbf{T}}$, is acting on particle \mathbf{x} , at time t , and in the bond direction of $\langle \mathbf{x}' - \mathbf{x} \rangle$. Note that \mathbf{f} should always satisfy Newton’s second and third laws so that Eq. (3.1) could be valid; however, there is no necessity for $\underline{\mathbf{T}}$ to be defined in such a way that it satisfies Newton’s laws.

The mentioned constitutive peridynamic equation of motion (Eq. (3.1)) is defined based on the following assumptions.

- (1) The peridynamic horizon of particle \mathbf{x} is apparently defined in a continuum domain. In other words, the reference material domain is assumed to be a continuum.
- (2) The pairwise force function (\mathbf{f}), or essentially force state ($\underline{\mathbf{T}}$), should be Reimann integrable; and the integral in Eq. (3.1) should converge uniformly [13].
- (3) As can be understood from Eq. (3.2), both $\underline{\mathbf{T}}$ and \mathbf{f} are apparently assumed to be calculated at time t (in other words, the in same time step).

Although the state-based peridynamics approach, proposed by Silling, is a superior formulation compared to the classical continuum mechanics, because it can directly deal with discontinuities and singularities, it is considered as an alternative continuum approach, since it is developed based on the mentioned questionable assumptions. From a physical behavior point of view, the validity of Eq. (3.1) for the materials without a continuous physical domain (such as concrete) is questionable. Therefore, similar to the issue with the classical continuum approaches, continuum peridynamics models, based on Eq. (3.1), generally do not have a good correspondence with the real physical behavior of non-continuum materials. Moreover, with no regard to the method of discretization of the problem domain (mesh-free or mesh-based), the integral equation shown in Eq. (3.1) must

be discretized using quadrature methods or other numerical integration approaches. Performing such discretizations are necessary, and may increase the numerical errors and computational costs; since to reach the desired accuracy is specific cases, more integration points may be needed than the initial discretization nodes.

In light of the mentioned issues, the continuum formulations of state-based peridynamics seems to be unsatisfactory to simulate the realistic behavior of non-continuum materials. It is also computationally inefficient.

In this study, the discrete framework of the state-based peridynamic theory is proposed. The method is called “state-based peridynamic particle method (SPPM)”. In this approach, the assumption of the continuity of the problem domain is relaxed and a finite number of particles is assumed for each peridynamic horizon; the integrals are substituted by finite summations, and the effect of overlapping the horizons on each other are considered more carefully by calculating $\underline{\mathbf{T}}$ and \mathbf{f} in two different consecutive time steps. The general scheme of the SPPM formulation is shown in Eqs. (3.3) and (3.4) as

$$m(\mathbf{x})\ddot{\mathbf{u}}(\mathbf{x}, t^n) = \sum_{H_x} \mathbf{f}_{xx'} + \mathbf{B}(\mathbf{x}, t^n), \quad (3.3)$$

and

$$\mathbf{f}_{xx'} = \underline{\mathbf{T}}[\mathbf{x}, t^{n-1}]\langle \mathbf{x}' - \mathbf{x} \rangle - \underline{\mathbf{T}}[\mathbf{x}', t^{n-1}]\langle \mathbf{x} - \mathbf{x}' \rangle, \quad (3.4)$$

where m is the mass of particle \mathbf{x} , and n is the time step number. Another criticism to the published references of the continuum state-based peridynamics methods is that, for unknown reason, the simple theory of peridynamics is proposed using unnecessary and complicated mathematical definitions and notations; which makes it hard for the researchers to follow the math and study the theory. Hence, in this study, a simpler, common mathematical formulation is employed. A precise form of SPPM formulation is proposed in the following sections of this chapter. In the following sections, an elastoplastic damage method is developed.

3.2. SPPM Linear-Elastic Formulation

In this section, the 3D formulation of SPPM is proposed and discussed. The introduced formulation can be then specialized to 2D/1D cases.

Consider an arbitrary problem domain, Ω , (can be a continuum or non-continuum) with its boundaries, Γ , which is discretized with a random finite number of particles, $N_{p\Omega}$, shown in Fig. (3.1). Note that, the definition of analytical boundaries do not exist in this approach, and the desired physical domain will substitute with integer number of particles; consequently, the resolution of the modelled boundaries depends upon the number of the particles used for boundary discretization.

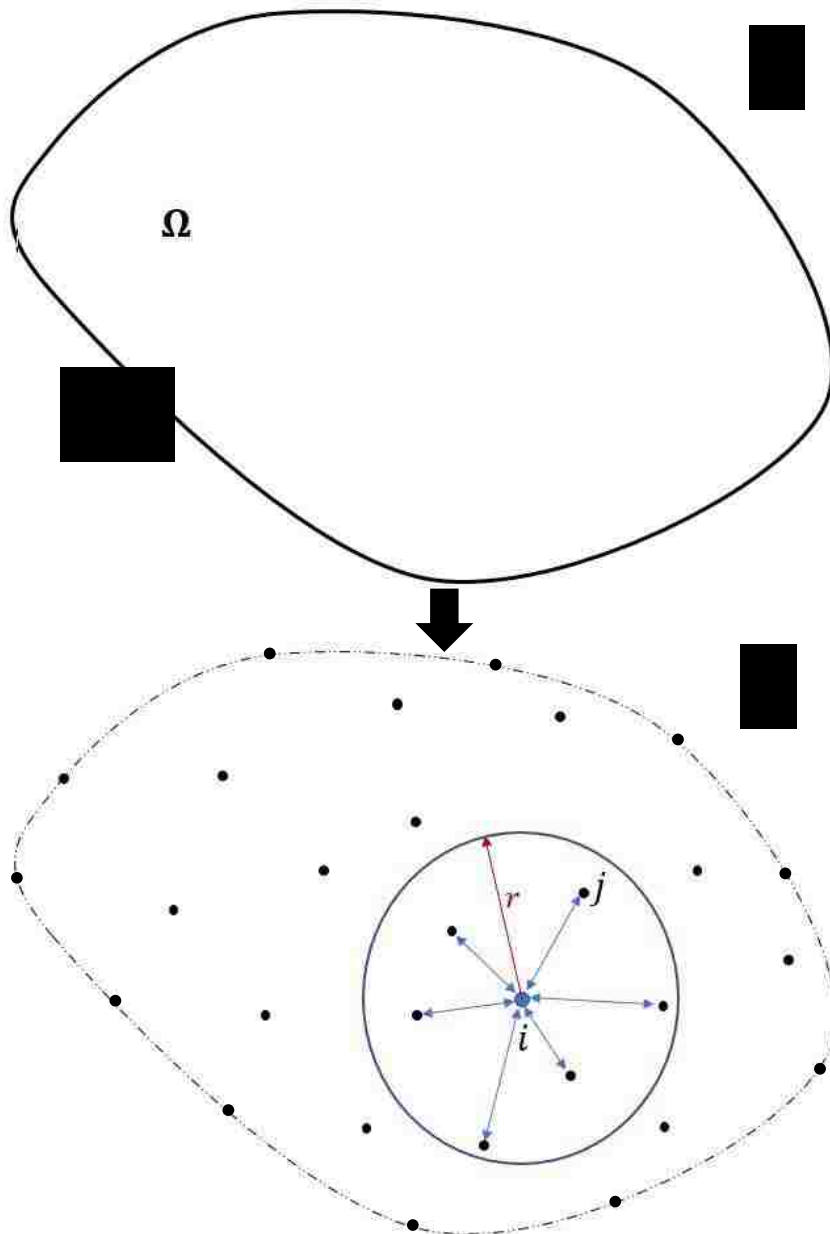


Figure 3.1. Assuming problem domain substituted by randomly-distributed finite number of particles. (a) before discretization, (b) after discretization.

Now consider a spherical peridynamic horizon (H_i), centered upon particle i , and with the radius of r_i , shown in Fig. (3.1). Depending the radius size, the horizon of particle i will include a finite number of particles, N_{P_i} , interacting with i , so called the “neighboring particles” or “neighbor-list of particle i ” (N_{P_i} does not include particle i itself). Moreover, it is assumed that the particles located outside of the horizon of particle i will not have any interaction with particle i . In other words, particle i only interacts particles within its horizon. Choosing a satisfactory horizon radius is an important and controversial issue in the literature. The horizon size could depend on the material type or even loading rate; and it could also affect the computational efficiency of the method. One can assume a fixed number of particles for each particle horizon and calculate r based on the average distance of those particles from i . Another approach which is frequently used in the literature would be considering a fixed radius for all the particle horizons [14, 21, 24]. This approach would be more applicable for the regularly distributed particle configurations. The proposed method for calculating the horizon size for SPPM approach is discussed in section 3. 5.

Coming back to Fig. (3.1), in SPPM approach it is assumed that particle i is interacting with each particle inside its horizon, via a bond (shown by arrows in Fig. (3.1)). Therefore, for each horizon, the total number of bonds would be equal to the number of particles, N_{P_i} , in that horizon. The acceleration of particle i at time step n , $\ddot{u}_i^n = \ddot{u}(x_i, t^n)$, can be calculated by the simplest form of equation of motion (similar to Eq. (3.3)) as

$$m_i \ddot{u}_i^n = \sum_{j=1}^{N_{P_i}} (F_b)_{ij}^n + B_i^n, \quad (3.5)$$

where m_i is the mass of particle i , N_{P_i} is the total number of particles in the neighbor list of particle i , B is the body force acting on particle i , and $(F_b)_{ij}^n$ is the bond force acting on particle i in the direction of particle j ; can be generally defined as a function of force states, $(F_s)_{ij}^n$, as

$$(F_b)_{ij}^n = \Phi((F_s)_{ij}^{n-1}, (F_s)_{ji}^{n-1}). \quad (3.6)$$

The function Φ in SPPM should be defined so that satisfies the following conditions:

- (1) Newton’s third law should be satisfied $((F_b)_{ij}^n = -(F_b)_{ji}^n)$.
- (2) The force states, $(F_s)_{ij}$, should be calculated in the previous time step (time step $n - 1$).
- (3) The force states, $(F_s)_{ij}$, may be not only a function of the stretch of bond ij , but also all the other bonds in the horizon of particle i .

Any combination of force states fulfilling the first condition (for instance minimum, maximum, average, etc.) for defining Φ could be valid. Furthermore, note that the classical continuum peridynamics approaches [11, 13] does not consider the second mentioned condition; instead, they calculate the bond forces and the force states in the same time step. The third condition is obligatory to satisfy the state-based peridynamics theory. In the

following, it is shown that considering the second condition will significantly improve the accuracy of the results in proposed SPLM approach (see chapter 4).

In this study, the simplest form of the function Φ , the average of the force states, is proposed for SPPM approach. Therefore, Eq. (3.6) can be re-written as

$$(F_b)_{ij}^n = \frac{1}{2}((F_s)_{ij}^{n-1} + (F_s)_{ji}^{n-1}), \quad (3.7)$$

In linear-elastic SPPM, the Force State, $(F_s)_{ij}$, acting on particle i in direction of particle j , can be assumed to be a linear function of the elastic stretch between i and j , as well as a linear function of the summation of the stretches of all the other bonds in the horizon of particle i . Hence, $(F_s)_{ij}$ is defined, in a general form, as a summation of a bond-based term plus state-based terms, as

$$(F_s)_{ij} = a_i(S_e)_{ij} + b_i \sum_{m=1}^{N_{Pi}} (S_e)_{im} + c_i \sum_{\bar{m}=1}^{N_{II}} (S_e)_{i\bar{m}}, \quad (3.8)$$

where the constants a_i is defined as a bond-based micro-elastic modulus, and b_i and c_i are defined as state-based peridynamics micro-elastic modules associated with particle i ; N_{II} is the total number of particles that located in the second-half of the horizon of particle i (second nearest neighbors) defined so that $\{particle\ j \in N_{II} \mid L_0 > r_i/2\}$, where L_0 is defined in Eq. (3.10). $(S_e)_{ij}$ is the elastic stretch in bond ij , generally defined as follows.

$$(S_e)_{ij} = (S_T)_{ij} - (S_P)_{ij}. \quad (3.9)$$

In Eq. (3.9), $(S_T)_{ij}$ and $(S_P)_{ij}$ are, respectively, the total stretch and plastic stretch between particles i and j . Note that for the linear elastic case, the plastic term of Eq. (3.9) will cancel out. The total stretch, $(S_T)_{ij}$, can be defined as

$$(S_T)_{ij} = \left(\frac{L - L_0}{L_0} \right)_{ij}, \quad (3.10)$$

where L and L_0 are, respectively, the current and reference length of the bond between particles i and j , calculating as $L = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$, and $L_0 = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2}$; where (x, y) and (X, Y) are, respectively, the coordinates of the particles in current and reference configuration. The force-stretch relation can be shown in a matrix form as

$$\mathbf{F}_{S_i} = \mathbf{K}_i \mathbf{S}_{e_i}, \quad (3.11)$$

where

$$\mathbf{F}_{S_i} = [(F_s)_{ij}]_{N_{Pi} \times 1} = [(F_s)_{i1}, \dots, (F_s)_{ij}, \dots, (F_s)_{iN_{Pi}}]^T,$$

$$\mathbf{S}_{e_i} = [(S_e)_{ij}]_{N_{P_i} \times 1} = [(S_e)_{i1}, \dots, (S_e)_{ij}, \dots, (S_e)_{iN_{P_i}}]^T,$$

And the micro-elastic stiffness matrix associated with particle i , \mathbf{K}_i , (would be square and symmetric) can be shown in a typical form as

$$\mathbf{K}_i =$$

$$\begin{bmatrix} a_i + b_i & b_i & \dots & & b_i & b_i & \dots & \dots & \dots & b_i \\ b_i & a_i + b_i & \square & \square & & \vdots & & \square & \square & \vdots \\ \vdots & & \ddots & & \vdots & & & \ddots & & \vdots \\ b_i & \square & \square & a_i + b_i & b_i & \vdots & & \square & & \vdots \\ b_i & & \dots & b_i & a_i + b_i & b_i & \dots & \dots & \dots & b_i \\ \hline b_i & & \dots & b_i & a_i + b_i + c_i & b_i + c_i & \dots & b_i + c_i & b_i + c_i & b_i + c_i \\ & & \square & \square & b_i + c_i & a_i + b_i + c_i & \square & \square & b_i + c_i & b_i + c_i \\ \vdots & & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ b_i & \square & \square & b_i & b_i + c_i & \square & \square & a_i + b_i + c_i & b_i + c_i & b_i + c_i \\ & & \dots & b_i & b_i + c_i & b_i + c_i & \dots & b_i + c_i & a_i + b_i + c_i & a_i + b_i + c_i \end{bmatrix}_{N_{P_i} \times N_{P_i}} \quad (3.12)$$

Given a global XY cartesian coordinate system, The SPPM kinematic stretch-strain relationship can be defined as

$$S_{ij} = (N_{xij})^2 \varepsilon_x + (N_{yij})^2 \varepsilon_y + N_{xij}N_{yij}\gamma_{xy}, \quad (\text{for 2D case})$$

$$S_{ij} = (N_{xij})^2 \varepsilon_x + (N_{yij})^2 \varepsilon_y + (N_{zij})^2 \varepsilon_z + N_{xij}N_{yij}\gamma_{xy} + N_{yij}N_{zij}\gamma_{yz} + N_{xij}N_{zij}\gamma_{xz}, \quad (\text{for 3D case}) \quad (3.13)$$

And in matrix form as

$$\mathbf{S}_i = \mathbf{N}_i \boldsymbol{\varepsilon}_i, \quad (3.14)$$

where $\mathbf{S}_i = [S_{ij}]_{N_{P_i} \times 1} = [S_{i1}, \dots, S_{ij}, \dots, S_{iN_{P_i}}]^T$, $\boldsymbol{\varepsilon}_i = [\varepsilon_x, \varepsilon_y, \gamma_{xy}, \varepsilon_z, \gamma_{yz}, \gamma_{xz}]_i^T$, (T denotes the transpose operation) and the transformation matrix, \mathbf{N}_i , is defined as follows.

$$\mathbf{N}_i = \begin{bmatrix} N_{xi1} & N_{yi1} & N_{xi1}N_{yi1} & N_{zi1} & N_{yi1}N_{zi1} & N_{xi1}N_{zi1} \\ \vdots & & \vdots & & & \vdots \\ N_{xij} & N_{yij} & N_{xij}N_{yij} & N_{zij} & N_{yij}N_{zij} & N_{xij}N_{zij} \\ \vdots & & \vdots & & & \vdots \\ N_{xiN_{P_i}} & N_{yiN_{P_i}} & N_{xiN_{P_i}}N_{yiN_{P_i}} & N_{ziN_{P_i}} & N_{yiN_{P_i}}N_{ziN_{P_i}} & N_{xiN_{P_i}}N_{ziN_{P_i}} \end{bmatrix}_{N_{P_i} \times 6} \quad (3.15)$$

Here, N_{xij} is defined the direction cosine between the bond ij and x -axis in the reference configuration (and so on).

In SPPM, in order to find the micro-elastic constants (a_i , b_i and c_i) for a finite volume (ΔV_i) of a linear-elastic solid, associated with particle i in the reference configuration (ΔV_i is defined in section 3.5); the internal virtual work of the classical linear-elastic model and SPPM approach are assuming to be identical:

$$\delta W_{Classical} = \delta W_{SPPM} \rightarrow \boldsymbol{\sigma}_i^T \delta \boldsymbol{\varepsilon}_i \Delta V_i = \frac{1}{2} \mathbf{F}_{S_i}^T \mathbf{L}_{0_i} \delta \mathbf{S}_{e_i}, \quad (3.16)$$

where $\delta \mathbf{S}_{e_i}$ and $\delta \boldsymbol{\varepsilon}$ are, respectively, infinitesimal virtual elastic stretch and strain matrixes, $\boldsymbol{\sigma}_i = [\sigma_x, \sigma_y, \tau_{xy}, \sigma_z, \tau_{yz}, \tau_{xz}]_i^T$, and \mathbf{L}_{0_i} is the diagonal reference bond length matrix, shown as

$$\mathbf{L}_{0_i} = \begin{bmatrix} (L_0)_{i1} & 0 & \dots & 0 \\ 0 & & & \vdots \\ \vdots & \ddots & & 0 \\ 0 & \dots & 0 & (L_0)_{iN_{P_i}} \end{bmatrix}_{N_{P_i} \times N_{P_i}}, \quad (3.17)$$

where $(L_0)_{i1}$ is defined in Eq. (3.10). By substituting Eq. (3.14) into Eq. (3.16) and canceling $\delta \boldsymbol{\varepsilon}$ from both side of the equation, the following relation for calculating the global stresses will obtain:

$$\boldsymbol{\sigma}_i = \frac{1}{2\Delta V_i} \mathbf{L}_{0_i} \mathbf{N}_i^T \mathbf{F}_{S_i} = \mathbf{M}_i \mathbf{F}_{S_i}. \quad (3.18)$$

From the classical linear-elastic theory [26], the stress-strain relation is introduced as

$$\boldsymbol{\sigma}_i = \mathbf{D} \boldsymbol{\varepsilon}_i \quad (3.19)$$

where $\mathbf{D}_{6 \times 6}$ is the classical elastic stiffness matrix. By substituting Eq. (3.11) into Eq. (3.18), and equating Eqs. (3.18) and (3.19); the following constitutive relation of SPPM for particle i derived:

$$\mathbf{D} = \mathbf{M}_i \mathbf{K}_i \mathbf{N}_i \quad (3.20)$$

For every particle in the reference configuration, Eq. (3.20) represents a linear system of equations, including 9 equations and 3 unknowns (which are SPPM micro-elastic module's); therefore, by solving Eq. (3.20) for every particle, the SPPM micro-elastic constants (a_i , b_i , and d_i) can be obtained. Note that Eq. (3.20) would be only valid for the linear-elastic solids (small deformation problems). Considering that assumption, Eq. (3.20) can be once solved for the reference configuration and the obtaining constant values for a_i , b_i , and c_i , can be stored for each particle and employed for the rest of the analysis. Note that, assuming a linear-elastic isotropic material, the obtained micro-elastic constants will be a function of Young's modulus (E), and the Poisson's ratio (ν). Note that depending on the arrangements of particles on each horizon, Eq. (3.20) may have multiple or even none solutions for special cases. This issue should be addressed as one of the difficulties with SPPM (similar to the issue of ill-conditioned or singular matrixes in the calculation of

meshless shape functions [6]). There are solutions for these such problems in the literature which are out of aims of this study. In this study, as a starting point, the simplest case (a lattice particle distribution) is considered which avoids such mentioned problems and verifies the practicality of the whole formulation (see chapter 4).

3.3. SPPM Two Spring Damage Model

In this section, a novel damage approach is proposed for SPPM framework; which provides the capability of modelling partial/complete damage. In this method so called “two-spring damage approach”, the classical theory of springs is correlated and combined with the SPPM framework. In following, first off, the classical theory of serial spring system is discussed; afterwards, the integrated form of the mentioned theory with SPPM is proposed.

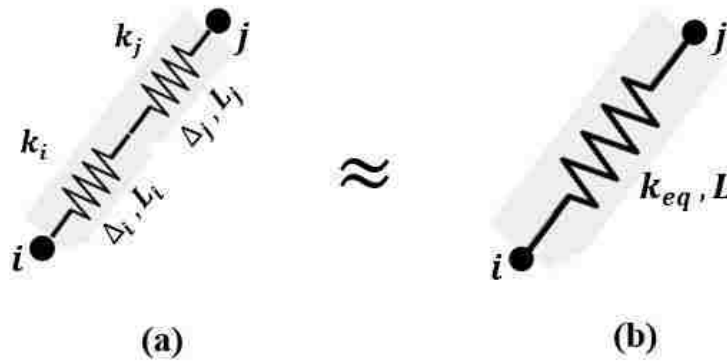


Figure 3.2. (a) An assuming system of serial springs, and (b) the equivalent spring system.

Consider Fig. (3.2-a) showing a system of serial springs between two nodes i and j ; where the stiffness of the springs are defined as k_i and k_j . Presuming having a linear-elastic spring system, the following constitutive relations are dominant based on the serial springs classical theory:

$$F_{eq} = F_i = F_j , \quad (3.21)$$

and

$$\Delta_{eq} = \Delta_i + \Delta_j . \quad (3.22)$$

Here, F_i is the internal force in spring i , F_j is the internal force in spring j ; Δ_i and Δ_j are, respectively, the displacements of springs i and j , and F_{eq} and Δ_{eq} are, respectively, the equivalent internal force and equivalent displacement in the equivalent spring system (Fig. (3.2-b)). The following relations for the forces can also be written as

$$F_i = k_i \Delta_i = k_i L_i S_i , \quad (3.23)$$

$$F_j = k_j \Delta_j = k_j L_j S_j , \quad (3.24)$$

and

$$F_{eq} = k_{eq}\Delta_{eq} = k_{eq}LS_{eq}, \quad (3.25)$$

Here, L_i and L_j are, respectively, the reference lengths of the springs i and j ; and S_i and S_j are, respectively, the stretches in springs i and j ; $L = L_i + L_j$, and S_{eq} is the equivalent stretch of the springs system. By substituting Eqs. (3.23) and (3.24) into Eq. (3.21). The following relation between the relative displacements of two serial springs can be obtained; as

$$\frac{\Delta_j}{\Delta_i} = \frac{k_i}{k_j}. \quad (3.26)$$

Moreover, by substituting Eqs. (3.23), (3.24), and (3.25) into Eq. (3.22), the equivalent stiffness of the serial springs system, k_{eq} , can be derived as

$$\frac{1}{k_{eq}} = \frac{1}{k_i} + \frac{1}{k_j} \rightarrow k_{eq} = \frac{k_i k_j}{k_i + k_j} \quad (3.27)$$

In SPPM method, in general, every bond is presumed to be equivalent to a system of spring-damper, as shown in Fig. (3.3).

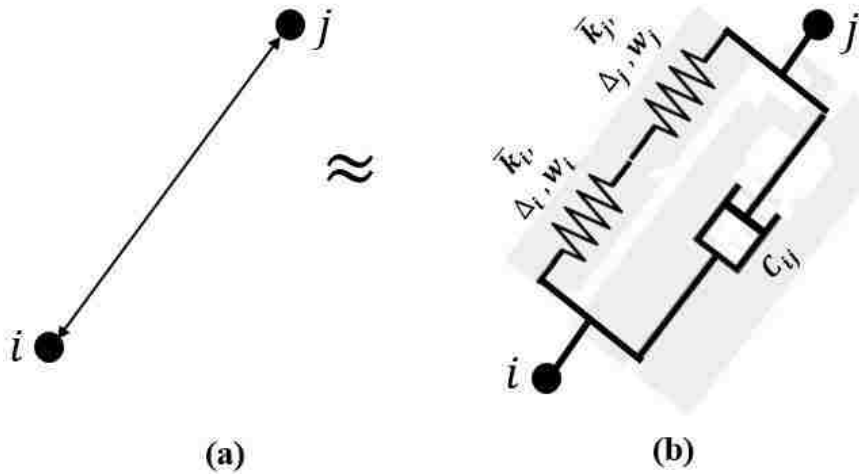


Figure 3.3. Equivalency of Peridynamic bonds to a system of spring-damper.

In other words, every bond is considered as a combination of two serial axial springs (with the same reference length) plus an internal damper acting between two interacting particles. Ignoring the existence of a damper here, in order to integrate the State-based Peridynamics Particle Model and the theory of springs; the equivalent stiffness of the serial spring system (Eq. (3.27)) is corresponded to peridynamic micro-elastic modulus's in SPPM (derived in Eq. (3.20)). Starting with a simpler case, considering only the bond-based term of the force

state defined in Eq. (3.8) for linear elastic materials, $(F_s)_{ij} = a_i(S_e)_{ij}$, the bond ij is assumed to be an equivalent spring with the equivalent stiffness of $k_{eq} = a_i/L_0$; therefore, the following relation can be written considering Eq. (3.27), as

$$k_{eq} = \frac{a_i}{L_0} = \frac{k_i k_j}{k_i + k_j} \quad (3.28)$$

By assuming $(\Delta_e)_{ij} = L_0(S_e)_{ij}$ substituting Eq. (3.26) into Eq. (3.28), and solving Eq. (3.28) for k_i and k_j we have

$$k_i = \left(\frac{a_i}{L_0}\right) \left(\frac{\Delta_i + \Delta_j}{\Delta_i}\right) = \left(\frac{a_i}{L_0}\right) \left(\frac{(\Delta_e)_{ij}}{\Delta_i}\right), \quad (3.29)$$

$$k_j = \left(\frac{a_i}{L_0}\right) \left(\frac{\Delta_i + \Delta_j}{\Delta_j}\right) = \left(\frac{a_i}{L_0}\right) \left(\frac{(\Delta_e)_{ij}}{\Delta_j}\right). \quad (3.30)$$

Here, we re-name k_i and k_j as an undamaged stiffness of each of the serially connected springs. Furthermore, for the linear-elastic material, the assumption of $\Delta_i = \Delta_j = (\Delta_e)_{ij}/2$ will be acceptable (by considering the assumption of having two spring with equal length and equal stiffness); therefore, Eqs. (3.29) and (3.30) can be simplified as

$$k_i = k_j = \frac{2a_i}{L_0}. \quad (3.31)$$

Now, by having the undamaged stiffness of the serial springs system, the damaged stiffness associated with each spring, \bar{k}_i and \bar{k}_j , are proposed as

$$\bar{k}_i = (1 - w_i)k_i = (1 - w_i) \left(\frac{2a_i}{L_0}\right), \quad (3.32)$$

$$\bar{k}_j = (1 - w_j)k_j = (1 - w_j) \left(\frac{2a_i}{L_0}\right). \quad (3.33)$$

where w_i and w_j are, respectively, the damage parameters associated with particle i and j . Note that one could consider the proposed approach as an isotropic damage model; since a single damage parameter, ω , is associated with each particle. In other words, all the bonds associated with particle i in the material horizon would have the same amount of damage in each time step.

Therefore, the equivalent stiffness of the damaged serially connected springs can be calculated as

$$\bar{k}_{eq} = \frac{\bar{k}_i \bar{k}_j}{\bar{k}_i + \bar{k}_j} = \left(\frac{2a_i}{L_0}\right) \left(\frac{(1 - w_i)(1 - w_j)}{2 - w_i - w_j}\right), \quad (3.34)$$

and the bond-based force state can be finally derived as

$$(F_s)_{ij} = \bar{k}_{eq}(\Delta_e)_{ij} = \left(\frac{2a_i}{L_0}\right) \left(\frac{(1-w_i)(1-w_j)}{2-w_i-w_j}\right) (\Delta_e)_{ij} = \left(\frac{2(1-w_i)(1-w_j)}{2-w_i-w_j}\right) a_i (S_e)_{ij}. \quad (3.35)$$

The obtained formulation in Eq. (3.35) can be easily expanded to state-based theory; by considering all the bonds connected to particle i each as a mentioned two-spring system and following the same proposed procedures.

Hence, the general formulation of the force state, $(F_s)_{ij}$, by integrating the state-based peridynamic particle approach and serial springs theory; and in a more sophisticated way by distinguishing the tensile and compressive bonds, is proposed as follows

$$(F_s)_{ij} = d_{ij} a_i (S_e)_{ij} + b_i \sum_{m=1}^{Np_i} d_{im} (S_e)_{im} + c_i \sum_{\bar{m}=1}^{Nr_{II}} d_{i\bar{m}} (S_e)_{i\bar{m}}, \quad (3.36)$$

where d_{ij} is call the “damage factor” associated with particles i and j , defined as

$$d_{ij} = \begin{cases} (d_t)_{ij} = \frac{2(1-w_{t_i})(1-w_{t_j})}{2-w_{t_i}-w_{t_j}} & (S_e)_{ij} \geq 0 \\ (d_c)_{ij} = \frac{2(1-w_{c_i})(1-w_{c_j})}{2-w_{c_i}-w_{c_j}} & (S_e)_{ij} < 0 \end{cases}, \quad (3.37)$$

where d_t and d_c are, respectively, a tensile and compressive damage factor, w_{t_i} and w_{c_i} are, respectively, the tensile and compressive damage parameters associated with particle i , and so on. The damage parameters, w_t and w_c , can vary between 0 (undamaged case) and 1 (fully damaged case), or, similar to what is done in many continuum peridynamics models [21], can be defined as a binary value (0 or 1). The damage parameters should be defined regarding the failure criterion and the material type. Note that although the proposed formulation allows having two different damage parameters, defining only one damage parameter is also possible for special materials. In this study, damage parameters are calibrated and proposed for concrete problems (expressed in chapter 4).

The Two-Spring Damage Method, proposed in this chapter, should be considered as a robust damage approach for the SPPM framework; in the sense that:

- (1) It will ensure the correctness of the obtaining results (satisfying the Newton’s laws).
- (2) It will provide the capability of modelling partial damage as well as ultimate failure.
- (3) It will ensure the symmetry of the results in symmetric problems (symmetric particle configuration with symmetric boundary conditions).
- (4) As it can be verified from Eq. (3.36), once one of the adjacent particles, i or j , get fully damaged, the force states associated with both particles, $(F_s)_{ij}$ and $(F_s)_{ji}$, (in other word the bond force) will become zero. This capability will lead to more localized and

more realistic damage patterns. In addition, in the case of having no damage (linear-elastic case), Eq. (3.36) will, basically, simplify to Eq. (3.8).

3.4. SPPM Plasticity Model

In this section, the proposed plasticity approach for State-based Peridynamic Particle Model is expressed. The plastic yield criterion and plastic flow rule for SPPM is proposed and discussed. The yield condition, presented in this study, is based on the concepts of the simplest multiaxial yield criterion (J2 plasticity and Von Mises yield surface); and can be only valid under small deformations.

Consider Eq. (3.9) which is introduced in the previous chapter. Eq. (3.9) can be expressed in a more general form as

$$(S_e)_{ij} = (S_T)_{ij} - (S_P)_{ij} = (S_T)_{ij} - \frac{1}{2}((S_{PS})_{ij} + (S_{PS})_{ji}), \quad (3.38)$$

where $(S_P)_{ij}$ is the plastic stretch of bond ij , and $(S_{PS})_{ij}$ is defined the plastic stretch state associated to particle i (similar to definition of force states). Therefore, the calculation of $(S_{PS})_{ij}$ and $(S_{PS})_{ji}$, and subsequently $(S_P)_{ij}$, is of our interest in this section.

Starting with the yielding condition, from the classical continuum theory, the stress tensor, at a particle in a solid domain, $\underline{\sigma}_i$, can be written in terms of deviatoric, σ_d , and hydrostatic, σ_h , parts as

$$\underline{\sigma}_i = (\sigma_d)_i + (\sigma_h)_i. \quad (3.39)$$

The hydrostatic stress is known as the average of the diagonal components (normal components) of the stress tensor $((\sigma_h)_i = \frac{1}{3}(tr(\underline{\sigma}_i))I)$; therefore, the deviatoric stress tensor can be computed as

$$(\sigma_d)_i = \underline{\sigma}_i - (\sigma_h)_i = \underline{\sigma}_i - \frac{1}{3}(tr(\underline{\sigma}_i))I. \quad (3.40)$$

From the SPPM linear-elastic theory, the relation between the classical stress vector, σ_i , and SPPM force state vector is defined as (same as Eq. (3.18))

$$\sigma_i = [\sigma_x, \sigma_y, \tau_{xy}, \sigma_z, \tau_{yz}, \tau_{xz}]_i^T = \mathbf{M}_i \mathbf{F}_{S_i}, \quad (3.41)$$

where \mathbf{M}_i and \mathbf{F}_{S_i} are defined in Eq. (3.18). By performing the matrix multiplication of Eq. (3.41), the stress components of σ_i (acting on particle i) can be obtained in terms of force states. Therefore, by substituting the obtained values of stress components from Eq. (3.41) into Eq. (3.40), the equivalent deviatoric stress tensor in terms of force states can be

computed. Finally, the equivalent J_2 Plasticity for SPPM approach can be defined (for particle i) as a half of the L_2 norm of the equivalent deviatoric stress tensor as follows

$$(J_{2eq})_i = \frac{1}{2} \|(\boldsymbol{\sigma}_d)_i\|_2. \quad (3.42)$$

The yielding condition is then implemented by defining a parameter, $YieldRatio_i$, as shown as follows

$$YieldRatio_i = \frac{(J_{2eq})_i}{(F_{eff})^{2/3}}. \quad (3.43)$$

Here, F_{eff} is defined as the effective yielding function (which can be defined as a function of yielding strength and ultimate strength of the material). Therefore, the yield condition is reached when $YieldRatio_i \geq 1$. Note that our effective yielding function for concrete is proposed in the next chapter.

For calculating the plastic flow, two general approaches are proposed here, which are:

- (1) The evolution of the plastic stretch (*calculation of* $(\Delta S_{ps})_{ij}$).
- (2) The evolution of the plastic strain (*calculation of* $(\Delta \boldsymbol{\varepsilon}_{pl})_i$).

The first approach was introduced by Gerstle [14], and the modified, generalized version is presented in this study. It is assumed that in analogy to Eq. (3.40), the plastic flow can be caused by the deviatoric components of the force state vector, as follows.

$$(\mathbf{F}_{Sdev})_i = \mathbf{F}_{S_i} - (F_{Savg})_i. \quad (3.44)$$

Here, $(\mathbf{F}_{Sdev})_i$ is the deviatoric part of the force state vector, \mathbf{F}_{S_i} (from Eq. (3.11)), calculating at particle i ; and $(F_{Savg})_i$ is the average of the components of \mathbf{F}_{S_i} , defined as

$$(F_{Savg})_i = avg(\mathbf{F}_{S_i}) = \frac{1}{Np_i} \sum_{j=1}^{Np_i} (F_s)_{ij}. \quad (3.45)$$

The change in the plastic stretch vector, $\Delta \mathbf{S}_{ps_i} = [(\Delta S_{ps})_{ij}]_{Np_i \times 1}$, is then defined as

$$\Delta \mathbf{S}_{ps_i} = \Delta \lambda \frac{(\mathbf{F}_{Sdev})_i}{\|(\mathbf{F}_{Sdev})_i\|_2}, \quad (3.46)$$

where

$$\Delta \lambda = \frac{\sigma_{yield}}{E} (YieldRatio_i - 1). \quad (3.47)$$

Here, $\Delta \lambda$ is defined the SPPM flow constant, analogues to Levy-Mises flow constant from the classical theory, E is the young's module and σ_{yield} is the yielding strength of material.

Therefore, the plastic stretch state of particle i (at time step $n + 1$), $(S_{Ps})_{ij}^{n+1}$, can be computed as

$$(S_{Ps})_{ij}^{n+1} = (S_{Ps})_{ij}^n + (\Delta S_{Ps})_{ij}^n, \quad (3.48)$$

and by substituting Eq. (3.48) into Eq. (3.38), the elastic stretch of bond ij can be calculated.

The second approach for calculating the plastic flow for the SPPM framework is calculation of plastic strain increments, $(\Delta \boldsymbol{\varepsilon}_{pl})_i$. From Eq. (3.18), the relation between force states and classical stress components are known (From Eq. (3.18)); hence, the plastic strain increments can be calculated from classical stress components and by using any method that is presented in classical plasticity literature (for instance, Levy-Mises approach). Afterwards, the plastic strain and stretch can be calculated from below

$$\begin{aligned} (\boldsymbol{\varepsilon}_{pl})_i^{n+1} &= (\boldsymbol{\varepsilon}_{pl})_i^n + (\Delta \boldsymbol{\varepsilon}_{pl})_i^n \rightarrow \\ \mathbf{S}_{Ps_i} &= \mathbf{N}_i (\boldsymbol{\varepsilon}_{pl})_i. \end{aligned} \quad (3.49)$$

Here, $\mathbf{S}_{Ps_i} = [(S_{Ps})_{ij}]_{Np_i \times 1}$, and \mathbf{N}_i is defined in Eq. (3.15). The obtained values from Eq. (3.49) are then can be substituted into Eq. (3.38) for calculating the elastic stretch of bond ij .

Note that damage due to the excess plastic strain (or stretch) can be included into SPPM approach via the proposed damage model, and by defining a proper damage parameter (See Eq. (3.37)). In that, the damage parameters should be defined in correspondence with the employed ultimate strength of material and the effective yielding function (Eq. (3.43)). In this study, calibrated damage parameters are proposed for concrete (see the next chapter).

3.5. Numerical Implementation of SPPM

In this section, the SPPM damage-plasticity approach is summarized, the numerical discretization of SPPM is discussed, and the constitutive numerical algorithm is proposed by flowcharts.

Consider Fig. (3.1). As it is discussed in section 3. 2, in SPPM method, the problem domain will initially discretize with the finite number of (randomly distributed) particles. The horizon radius of particle i is assumed to be calculated as follows.

$$\begin{aligned} r_i &= (L_0)_{i6}, & \text{(for 2D)} \\ r_i &= (L_0)_{i18}, & \text{(for 3D)} \end{aligned} \quad (3.50)$$

where, $(L_0)_{i6}$ is the distance of the sixth nearest particle from i (an so on). By having this definition for the horizon radius, we would end up in the same number of particles in each horizon in spite of having arbitrary particle distribution (it significantly increases the computational efficiency of the model). Afterwards, the neighbor-list of particle i can be obtained by considering only the particles near i that their distance from particle i are less than or equal the horizon radius (r_i) (the neighbor-list should be sorted in a descending order and stored for every particle). Note that in order to decrease the computational cost, the neighbor-list will only generate based on the reference configuration and keep constant in the rest of the analysis (it means that no matter how particle j will deform, it will remain in the neighbor-list of particle i till the end). The total number of particles in the horizon of particle i is named Np_i . The general form the equation of motion for SPPM can be written for particle i and in time step n , as

$$m_i \ddot{u}_i^n = \sum_{j=1}^{Np_i} \left\{ (F_b)_{ij}^n + (F_{dampin})_{ij}^n \right\} - (F_{dampex})_{ij}^n + B_i^n, \quad (3.51)$$

Where \ddot{u}_i^n is the acceleration of particle i at time step n ($\ddot{u}_i^n = \ddot{u}(\mathbf{x}_i, t^n)$), $(F_b)_{ij}^n$ is the bond force acting on particle i in the direction of j (Eq. (3.7)), B_i^n is the body force; m_i is the mass of particle i which is defined proportional to the mass density of particle i (ρ_i), and the associated finite volume (ΔV_i) of particle i , as

$$m_i = \rho_i \Delta V_i, \quad (3.52)$$

where the volume of particle i is assumed the volume of the Voronoi diagram of particle i . In order to increase the computational efficiency, ΔV_i can be also approximated as a half of the horizon volume; therefore:

$$\Delta V_i = \iiint \text{Voronoi}(\text{particle } i) dV \approx \frac{1}{2} \left(\frac{4}{3} \pi r_i^3 \right). \quad (3.53)$$

Note that m_i and ΔV_i should be calculated once, and only for the reference particle configuration. In Eq. (3.51), F_{dampex} and F_{dampin} are, respectively, the external and internal damping forces acting on particle i (defined in following). Eq. (3.51) in global coordinates (cartesian), in a matrix form, can be written as

$$m_i \ddot{\mathbf{u}}_i^n = \mathbf{N}_i^n \left((\mathbf{F}_b)_i^n + (\mathbf{F}_{dampin})_i^n \right) - (\mathbf{F}_{dampex})_i^n + \mathbf{B}_i^n, \quad (3.54)$$

Where \mathbf{N}_i^n is introduced in Eq. (3.15), and $(\mathbf{F}_{dampin})_i^n$ and $(\mathbf{F}_{dampex})_i^n$ are defined as

$$(\mathbf{F}_{dampin})_i^n = C_{in} \mathbf{V}_{Axial} = (2m_i \xi_{in} \omega_i) [\dot{\mathbf{u}}_{ij}^n]_{Np_i \times 1}, \quad (3.55)$$

and

$$(\mathbf{F}_{dampex})_i^n = C_{ex} \dot{\mathbf{u}}_i^n = (2m_i \xi_{ex} \omega_{struct}) \dot{\mathbf{u}}_i^n. \quad (3.56)$$

Here, $C_{in} = 2m_i\xi_{in}\omega_i$ (shown in Fig. (3.3)) and $C_{ex} = 2m_i\xi_{ex}\omega_{struct}$ are, respectively, the internal and external damping coefficients; ξ_{in} and ξ_{ex} are, respectively, the internal and external damping ratios that should be defined by user, ω_i is the highest natural frequency associated with the horizon of particle i , and ω_{struct} is considered as the fundamental natural frequency of the entire structure. ω_i can be approximated, assuming the shortest wavelength that can be physically represented by the material horizon is $\lambda_i \approx r_i/2$, as follows

$$\omega_i \approx \frac{2\pi\tilde{c}_i}{\lambda_i} \approx \frac{4\pi}{r_i} \sqrt{\frac{\kappa}{\rho}}, \quad (3.57)$$

where, \tilde{c}_i is the speed of sound associated to particle i , κ is the Bulk's modulus ($\kappa = E/3(1 - 2\nu)$), and r_i is the horizon radius. Note that ω_{struct} can be also approximated or estimated thorough the numerical simulations.

Note that $\ddot{\mathbf{u}}_i^n$ in Eq. (3.54), can be discretized in time using the central deference method as shown below:

$$\ddot{\mathbf{u}}_i^n = \frac{\mathbf{u}_i^{n+1} - 2\mathbf{u}_i^n + \mathbf{u}_i^{n-1}}{\Delta t^2}, \quad (3.58)$$

where Δt is the time stepping increment.

Calculation of principal stresses and strains are also of interest, particularly for considering the failure criteria. The components of the classical stress tensor (in cartesian coordinate system), $\underline{\boldsymbol{\sigma}}_i$, can be obtained by solving Eq. (3.18) ($\boldsymbol{\sigma}_i = \mathbf{M}_i \mathbf{F}_{S_i}$); therefore, the principal stresses can be simply obtained by calculating the eigenvalues of stress tensor as

$$\boldsymbol{\sigma}_{p_i} = [\sigma_1 \quad \sigma_2 \quad \sigma_3]^T = \mathbf{eig}(\underline{\boldsymbol{\sigma}}_i). \quad (3.59)$$

The strain tensor, $\underline{\boldsymbol{\varepsilon}}_i$, can be also obtained by calculating the strain components, substituting Eq. (3.18) into Eq. (3.19), as follows.

$$\boldsymbol{\varepsilon}_i = \mathbf{D}^{-1} \boldsymbol{\sigma}_i = \mathbf{D}^{-1} \mathbf{M}_i \mathbf{F}_{S_i}, \quad (3.60)$$

where $\mathbf{M}_i = \frac{1}{2\Delta V_i} \mathbf{L}_{0_i} \mathbf{N}_i^T$. Afterwards, the strain tensor can be generated and the principal strains can be also computed by calculating the eigenvalues of the strain tensor as

$$\boldsymbol{\varepsilon}_{p_i} = [\varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3]^T = \mathbf{eig}(\underline{\boldsymbol{\varepsilon}}_i). \quad (3.61)$$

The general algorithm of the SPPM approach is proposed (in form of a flowchart) in Fig. (3.4).

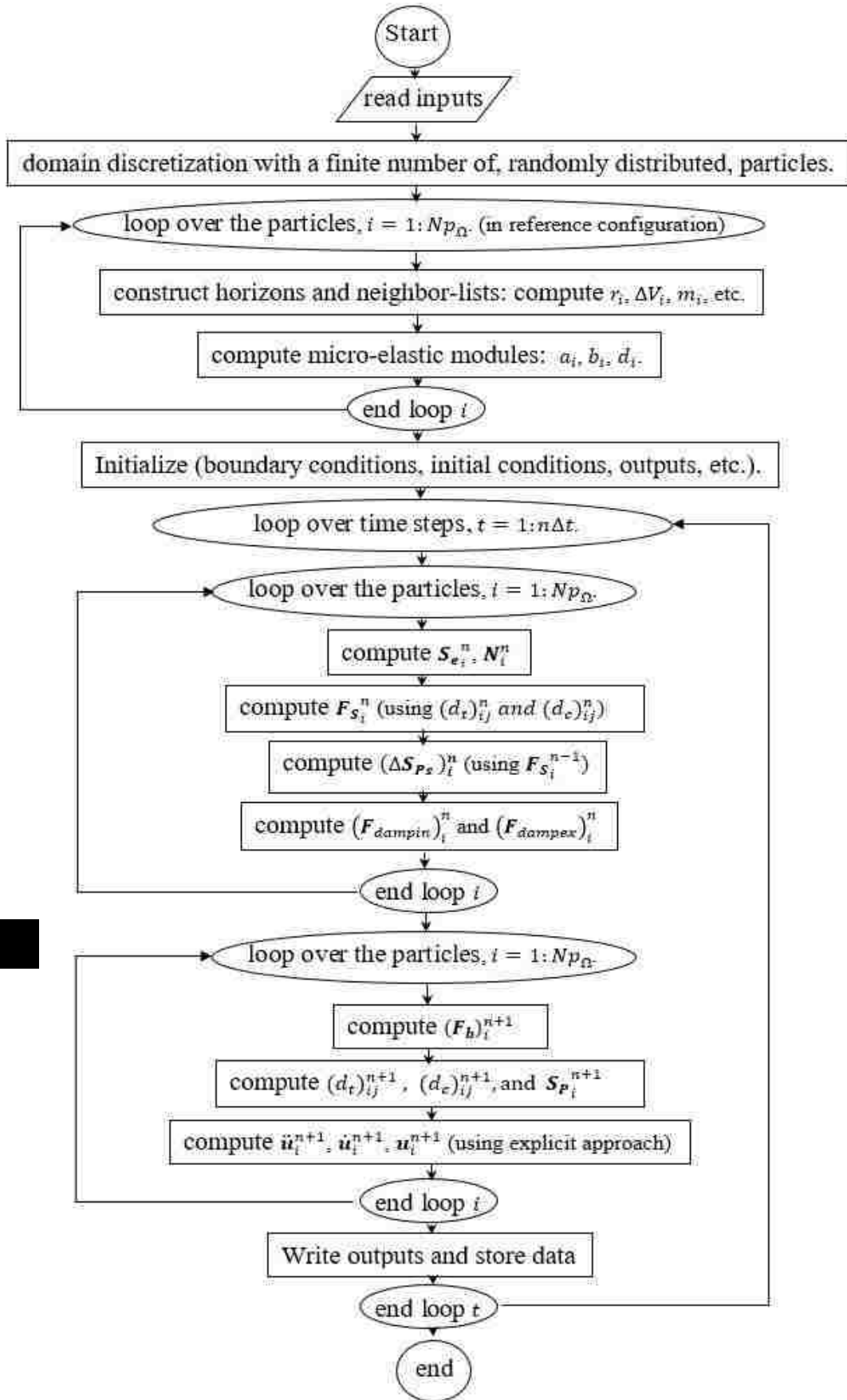


Figure 3.4. The general flowchart of SPPM method. (Serial code).

Chapter 4

Re-formulated State-Based Peridynamic Lattice Model

4.1. Introduction

The initial version of State-based Peridynamic Lattice Model (SPLM) was introduced by Gerstle [14] in 2015. Some other versions of SPLM, with minor modifications and the same algorithm with the original version, were also proposed by some other graduate students in the University of New Mexico Department of Civil Engineering. It is concluded that neither the Gerstle's original version nor the prior efforts were successful. All the former versions of SPLM had the following major issues:

- (1) Incorrect results. None of the previous versions of SPLM could provide the results that could be either verified with the classical solutions (even for the simple benchmark problems), or validated with the experimental tests.
- (2) Non-objectivity. It was evident that the obtained results from the former versions were mesh-sensitive; in that, for a typical problem, by rotating the lattice configuration different results were obtained.
- (3) Convergence issues. Indeed, convergence does not have any meaning in the previous versions of SPLM. Increasing the number of particles was only providing different results and not making any improvement.
- (4) Asymmetry. It was seen that for a given completely symmetric problem (geometry and boundary conditions), and with the defined isotropic-homogeneous material properties; the asymmetric crack patterns were obtained, despite expecting symmetric cracking pattern.

The main response to the mentioned issues, frequently stated by the former SPLM developers, was: SPLM is something totally different than classical approaches. In view of the author of this study, SPLM is a re-formulation of continuum peridynamics theory, as well as classical continuum approaches. Therefore, obtaining the same results as the continuum methods, at least for the elastic (pre-peak) region and for the benchmark cracking problems, is a requirement.

In this chapter, the re-formulated version of SPLM method (based on the introduced SPPM algorithm) is proposed and calibrated for simulating cracking and damage in concrete. The new SPLM method provides objective, symmetric, convergent, and correct results (in accordance with the mentioned issues).

4.2. Linear-Elastic SPLM

The proposed SPLM formulation is the simplified form of the proposed SPPM method (in chapter 3). In this approach, instead of considering an arbitrary reference particle distribution, a lattice configuration is employed to discretize the problem domain. The fundamentals of generating the lattice topology is presented in [14]. In accordance with the chosen lattice configuration, the neighbor-list of particle i possesses six particles in 2D and eighteen particles in 3D. The 2D lattice topology used in this study is the close-packed hexagonal configuration, shown in Fig. (4.1). Using this configuration leads to symmetric and equally spaced particles in every horizon. Therefore, except for boundary particles, the number of particles in every particle horizon in the bulk will be $Np_i = 6$. The reference bond length matrix (defined in Eq. (3.17)) is simplified as

$$L_{0i} = L_0 = \begin{bmatrix} L_0 & 0 & \dots & 0 \\ 0 & & & \\ \vdots & \ddots & & \vdots \\ 0 & \dots & 0 & L_0 \end{bmatrix}_{6 \times 6}, \quad (4.1)$$

Note that the radius of the horizon of particle i is fixed as the lattice spacing ($r_i = r = L_0$). The associated finite volume with each particle (defined in Eq. (3.53)) in the material bulk can also be simplified, by calculating the volume of the Voronoi diagram of particle i (shown in Fig. (4.1)), to $\Delta V_i = \Delta V = (\sqrt{3}/2)tL_0^2$, where t is the material thickness.

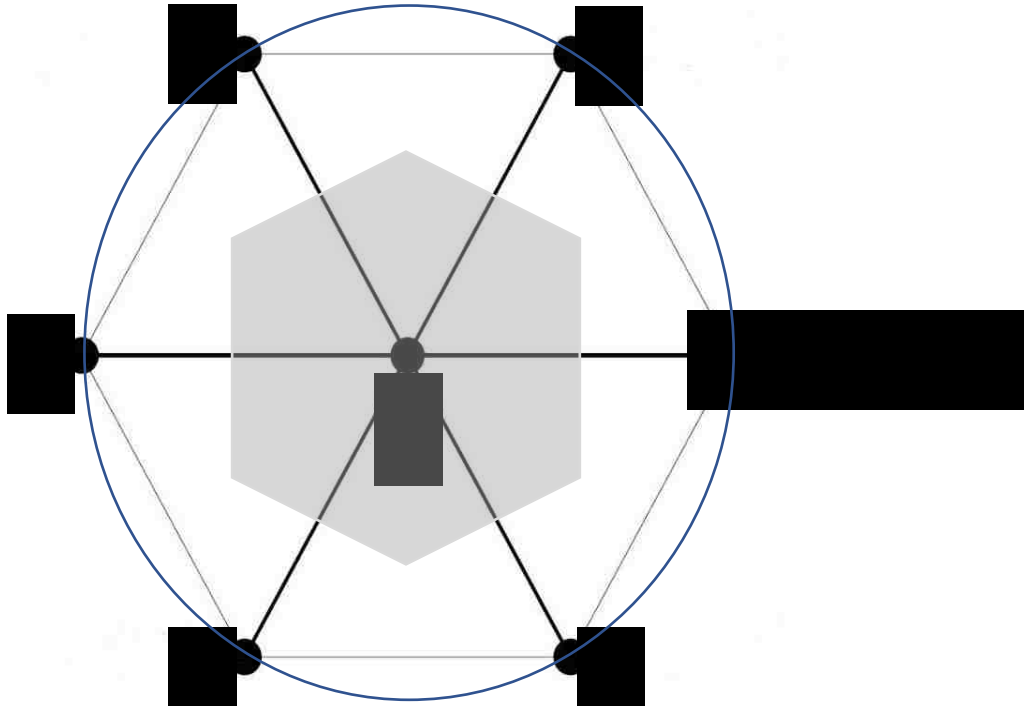


Figure 4.1. Lattice topology and bond numbering order of particles. The Voronoi diagram of particle i ($= \Delta V$) is shown with gray hatch.

For the SPLM approach, the equation of motion for particle i , shown in Fig. (4.1), is written similar to Eq. (3.51). Note that regarding the considered lattice topology, j varies from 1 to 6 (as shown in Fig. (4.1)); in other words, in SPLM method, the numbering order of all the surrounding particles are fixed for every horizon, which extremely simplifies the computations.

The definition of force states is also simplify in SPLM. For 2D formulation, since we only have six neighbor particles with identical reference bond length, the third term of Eq. (3.8), related to second nearest neighbors, can be canceled. Hence, the force state can be formulated, for 2D problems, as

$$(F_s)_{ij} = a(S_e)_{ij} + b \sum_{m=1}^6 (S_e)_{im}, \quad (4.2)$$

where a is the bond-based micro-elastic modulus, and b is the only state-based micro-elastic modulus considered for 2D SPLM. Subsequently, the \mathbf{K} matrix, defined in Eq. (3.12), is simplified as

$$\mathbf{K}_i = \mathbf{K} = \begin{bmatrix} a+b & b & \dots & b & b \\ b & a+b & \dots & & b \\ \vdots & & \ddots & & \vdots \\ b & & \dots & a+b & b \\ b & b & \dots & b & a+b \end{bmatrix}_{6 \times 6}. \quad (4.3)$$

By (1) neglecting the effects of having fewer bonds in boundary particles (assuming \mathbf{K} is valid for all the horizons in the problem domain), (2) associating the same material volume (ΔV) to all the particles in reference configuration, and (3) assuming constant \mathbf{N} matrix (Eq. (3.15)); Eq. (3.20) can be solved only once and constant micro-elastic modulus can be obtained for all the horizons; therefore, Eq. (3.20) can be written as

$$\mathbf{D}_{3 \times 3} = \mathbf{M}_{3 \times 6} \mathbf{K}_{6 \times 6} \mathbf{N}_{6 \times 3}. \quad (\text{in 2D}) \quad (4.4)$$

and the micro-elastic constants, a and b , can be obtained for 2D plane stress problems as follows

$$a = \frac{2EL_0t}{\sqrt{3}(1+\nu)},$$

and

$$b = \frac{EL_0t(1-3\nu)}{6\sqrt{3}(\nu^2-1)}; \quad (4.5)$$

and for the 2D plane strain case as

$$a = \frac{2EL_0t}{\sqrt{3}(1+\nu)},$$

and

$$b = \frac{EL_0t(1-4\nu)}{6\sqrt{3}(2\nu-1)(\nu+1)}; \quad (4.6)$$

where E is the Young's modulus and ν is the poisson's ratio.

For the 3D problems, the same procedure is followed and Eqs. (3.8) and (3.51) is used for the Face Centered Cubic (FCC) lattice configuration [14] with $\Delta V_i = \Delta V = (\sqrt{2}/2)L_0^3$, and Eq. (4.4) is re-written for 3D FCC lattice as

$$\mathbf{D}_{6 \times 6} = \mathbf{M}_{6 \times 18} \mathbf{K}_{18 \times 18} \mathbf{N}_{18 \times 6}. \quad (\text{in 3D}) \quad (4.7)$$

Hence, the micro-elastic constants (a , b and c) can be computed for the reference lattice configuration, accordingly; obtained as

$$\begin{aligned} a &= \frac{EL_0^2}{\sqrt{2}(1+\nu)}, \\ b &= \frac{(\sqrt{2}-1)EL_0^2(1-4\nu)}{24(2\nu-1)(\nu+1)}, \\ c &= \frac{(1-2\sqrt{2})EL_0^2}{4(1+\nu)} = \left(\frac{1}{2\sqrt{2}} - 1\right) a. \end{aligned} \quad (\text{for 3D case}) \quad (4.8)$$

4.3. SPLM Damage-Plasticity Model for Concrete

The general formulation of SPLM method (including plasticity and damage) can be generated based on the proposed SPPM framework (chapter 3). The general form of a particle horizon in the re-formulated SPLM approach is shown in Fig. (4.2).

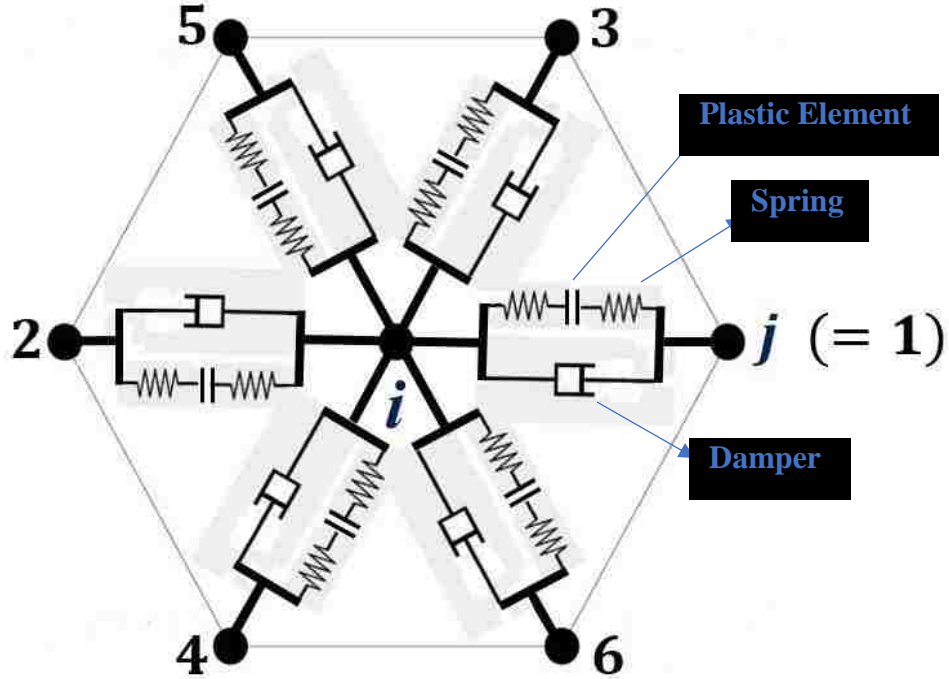


Figure 4.2. General SPLM horizon scheme.

Here, some of the introduced equations in chapter 3 are simplified for SPLM and re-written. More details about the definition of the variables are provided in chapter 3. Considering Fig. (4.2) and Eq. (3.51), the equation of motion can be re-written as follows

$$m_i \ddot{u}_i^n = \sum_{j=1}^6 \left\{ (F_b)_{ij}^n + (F_{dampin})_{ij}^n \right\} - (F_{dampex})_{ij}^n + B_i^n, \quad (4.8)$$

where the bond force is defined as

$$(F_b)_{ij}^n = \frac{1}{2} \left((F_s)_{ij}^{n-1} + (F_s)_{ji}^{n-1} \right), \quad (4.9)$$

and by applying the Two-Spring damage method, the force state (Eq. (4.2)) can be re-formulated (for 2D), based on Eq. (3.36), as

$$(F_s)_{ij} = d_{ij} a(S_e)_{ij} + b \sum_{m=1}^6 d_{im} (S_e)_{im}. \quad (4.10)$$

Note that for calculating the damping forces, Eqs. (3.55-3.57) can be used assuming $\lambda_i \approx 2L_0$, where L_0 is the lattice spacing.

The damage factor (d_{ij}), defined in Eq. (3.37), is

$$d_{ij} = \begin{cases} (d_t)_{ij} = \frac{2(1 - w_{t_i})(1 - w_{t_j})}{2 - w_{t_i} - w_{t_j}} & (S_e)_{ij} \geq 0 \\ (d_c)_{ij} = \frac{2(1 - w_{c_i})(1 - w_{c_j})}{2 - w_{c_i} - w_{c_j}} & (S_e)_{ij} < 0 \end{cases} \quad (4.11)$$

In this chapter, suitable and calibrated damage parameters (w_t and w_c) are proposed in correspondence with the physical mechanical behavior of concrete. As is briefly discussed in chapter 3, in the SPLM/SPPM approach, the damage phenomenon is associated with particles instead of bonds, in that the damage parameters are defined and calculated based upon the “stress” and “strain” states of each particle in global coordinates. In this view, one could name this method as an “isotropic damage” model or “particle based” damage model.

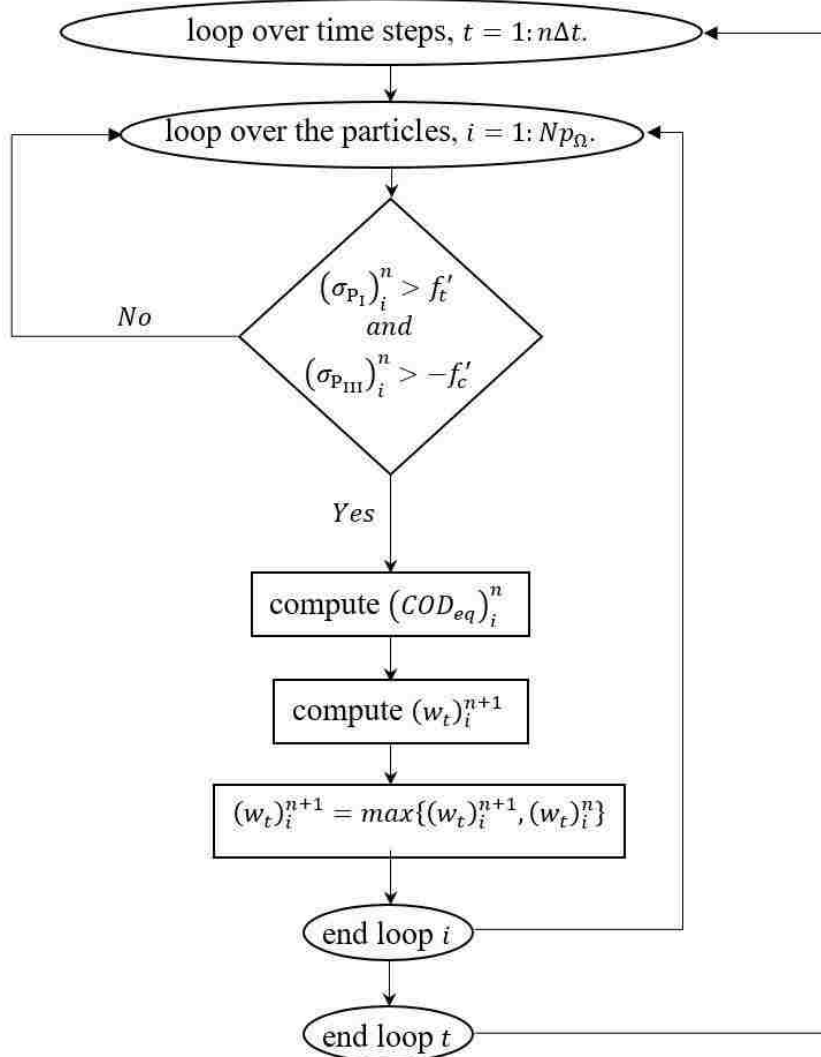


Figure 4.3. Flowchart of calculating the tensile damage parameter (w_t).

In the SPLM approach, tensile damage behavior of concrete is modeled based on Hillerborg fictitious crack model [15]. The process of simulating the tensile damage in SPLM (damage initiation and evolution) is shown in Fig. (4.3) as a flowchart. The tensile damage initiation condition for SPLM is based on the stress condition, as shown in Fig. (4.3). Once the damage initiation condition satisfied, tensile damage initiates in particle i . In Fig. (4.3), $(\sigma_{pI})_i^n$ is the maximum positive principal stress component, defined as

$$(\sigma_{pI})_i^n = \max \left\{ \max \left\{ (\sigma_p)_i^n \right\}, 0 \right\}, \quad (4.12)$$

and $(\sigma_{pIII})_i^n$ is defined as the minimum of the other two principal stress components (σ_{pI} is defined in Eq. (3.59)). The user defined parameters in tensile damage initiation condition are the tensile strength of concrete, f'_t , and the compressive strength of concrete, f'_c .

The evolution of the tensile damage (tension softening) of concrete is simulated via a bilinear tensile strength- COD curve shown in Fig. (4.4). By taking the multiaxial state of stress-strain into account, the equivalent crack opening displacement (COD_{eq}) associated with particle i and its representative material volume is formulated as a function of maximum principal strain at particle i , defined as ε_{pI} , as

$$(COD_{eq})_i^n = (2L_0)\varepsilon_I, \quad (4.13)$$

where L_0 is the lattice spacing, $\varepsilon_I = \max \left\{ (\varepsilon_p)_i^n \right\}$, and $(\varepsilon_p)_i^n$ is defined in Eq. (3.61).

Following the algorithm shown in Fig. (4.3), the tensile damage parameter $(w_t)_i$ is defined based on bilinear tensile strength- COD curve (Fig. (4.4)), as follows.

$$(w_t)_i = \begin{cases} 0 & COD_{eq} < 0 \\ 1 - \left(\frac{f'_t}{E COD_1 \varepsilon_I} \right) (COD_1 + (\gamma - 1) COD_{eq}) & 0 \leq COD_{eq} < COD_1 \\ 1 - \left(\frac{\gamma f'_t}{E \varepsilon_I} \right) \left(\frac{COD_c - COD_{eq}}{COD_c - COD_1} \right) & COD_1 \leq COD_{eq} < COD_c \\ 1 & COD_{eq} \geq COD_c \end{cases}, \quad (4.14)$$

where E is the Young's modulus, f'_t is the tensile strength, γ is the tensile damage parameter at "knee", COD_c is the critical crack opening displacement, and $COD_1 = \alpha COD_c$; see Figs. (4.4-4.5). In this study, $\alpha = 0.1$ and $\gamma = 0.25$ are assumed. It follows that the tensile damage parameter, w_t , varies in a nonlinear fashion and goes to 1 as the

COD_{eq} increases. In addition, when COD_{eq} reaches COD_0 , the damage parameter immediately jumps to a finite value, rather than starting from zero (see Fig. (4.5)).

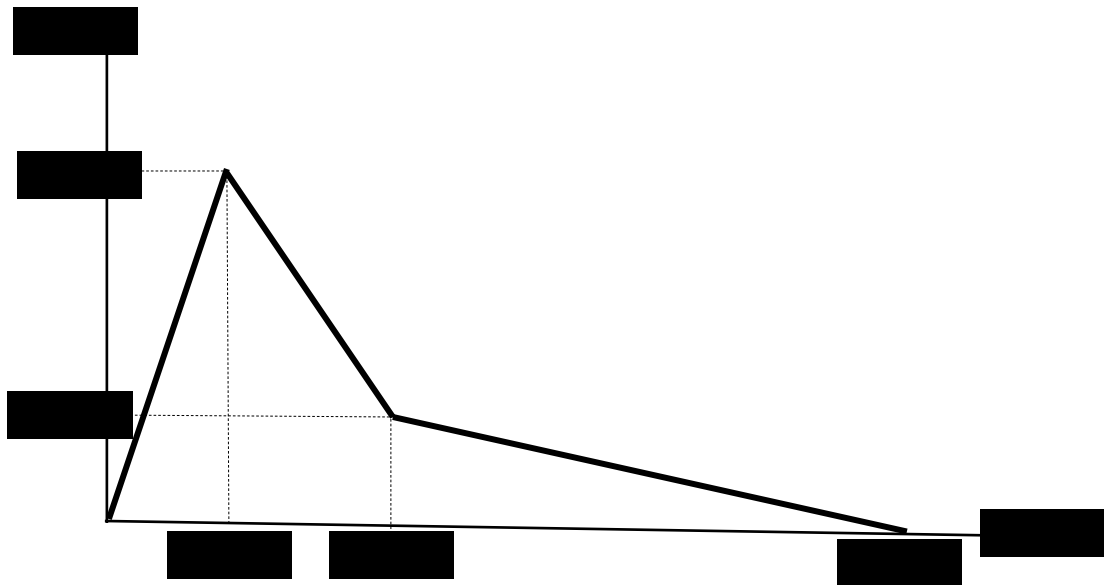


Figure 4.4. Bilinear tensile strength-COD (tension softening) curve.

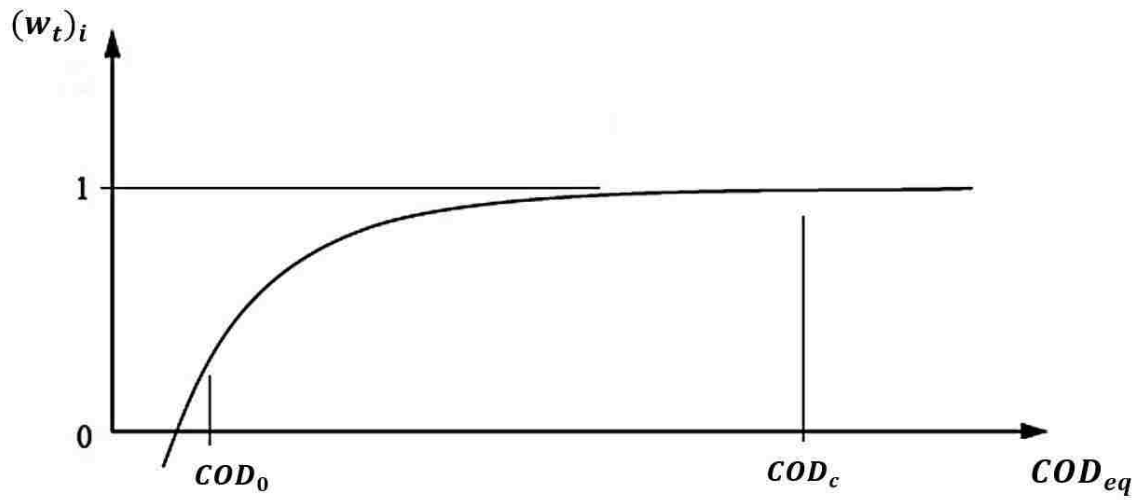


Figure 4.5. Tensile damage parameter, w_t - COD_{eq} curve.

Note that, as can be seen from the proposed formulation, plasticity and damage are integrated in SPLM approach. The plasticity method proposed in chapter 3 is applied to SPLM and the linear hardening model, shown in Fig. (4.6), is employed. The effective yielding function, F_{eff} , (introduced in Eq. (3.43)) is defined and calibrated for concrete so that hardening can be modelled, as

$$F_{eff} = f_y + \left(\frac{f_{ult} - f_y}{\varepsilon_{ult}} \right) \varepsilon_{eff}, \quad (4.15)$$

where, f_y and f_{ult} are, respectively, the yielding strength and ultimate strength of concrete, and ε_{ult} is the ultimate strain that concrete can carry, ε_{eff} is defined in Eq. (4.17).

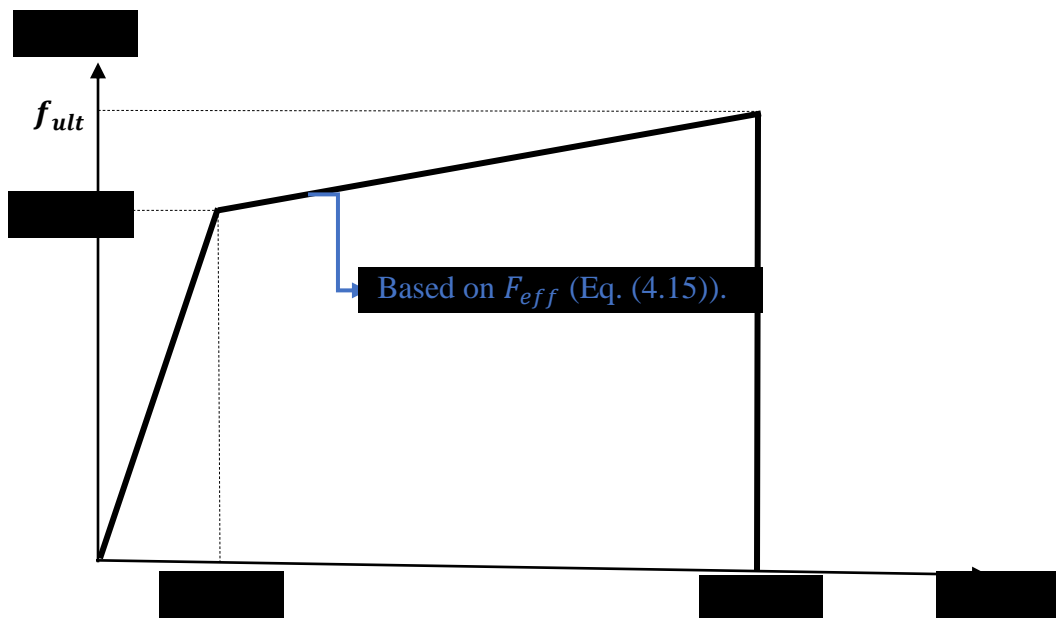


Figure 4.6. Compressive stress-strain relation with linear hardening for SPLM.

In the SPLM approach, damage due to excessive compression is also modeled via defining the compressive damage parameter. It is understood that once some of the particles become damaged, the stress state of the neighboring particles can possibly change from tension to compression mode (and vice versa). In other words, some of the bonds that have a tensile force in time step n , may become compressive in timestep $n+1$ (and vice versa). Consider we have a bond with 30% of tensile damage in time step n , if the direction of load changes to compression in time step $n+1$, it would not be realistic to assume that bond will have the same 30% damage in compression as well; it is so plausible that the mentioned bond will have some damage but the damage percentage under compression would be different than what it was under tension. In another view, if we consider a concrete specimen that is failed under tension, it does not mean that it cannot carry any compression anymore (it will still sustain some compression). Therefore, applying the same value of tensile damage parameter to the force state of the bonds under compression is conceptually, theoretically,

and numerically wrong (one of the major issues with the former SPLM versions). In another physical point of view, consider a concrete specimen that is partially damaged under uniaxial tension, if we change the direction of the loading (making it uniaxial compression), it would be so predictable that the specimen will fail under lower compressive peak load than the undamaged specimen (and vice versa).

All the mentioned subtle issues lead to define a separate compressive damage parameter for SPLM/SPPM method. Failure due to excessive plastic strain is also considered in the definition of compressive damage parameter. The compressive damage parameter for particle i , w_{c_i} , is modeled as a function of w_{t_i} and ultimate strain (ϵ_{ult}), shown in Figs. (4.6-4.7), and defined as.

$$w_{c_i} = \begin{cases} w_{c_{max}} \left(\frac{\sqrt{w_{t_i}} - \sqrt{(w_{t_{cr}})_i}}{1 - \sqrt{(w_{t_{cr}})_i}} \right) & w_{t_i} > (w_{t_{cr}})_i \\ 0 & w_{t_i} \leq (w_{t_{cr}})_i \end{cases} \quad (4.16)$$

In Eq. (4.16), w_{t_i} is the tensile damage parameter, $w_{t_{cr}}$ is the critical tensile damage parameter which is assumed as a threshold for initiating partial compressive damage, and $w_{c_{max}}$ is the maximum value of the compressive damage parameter defined in model. The current SPLM model is calibrated for concrete by the assuming values of $w_{t_{cr}} = 1/3$, and $w_{c_{max}} = 1$. The procedure of calculating the compressive damage parameter in SPLM is shown in Fig. (4.8).

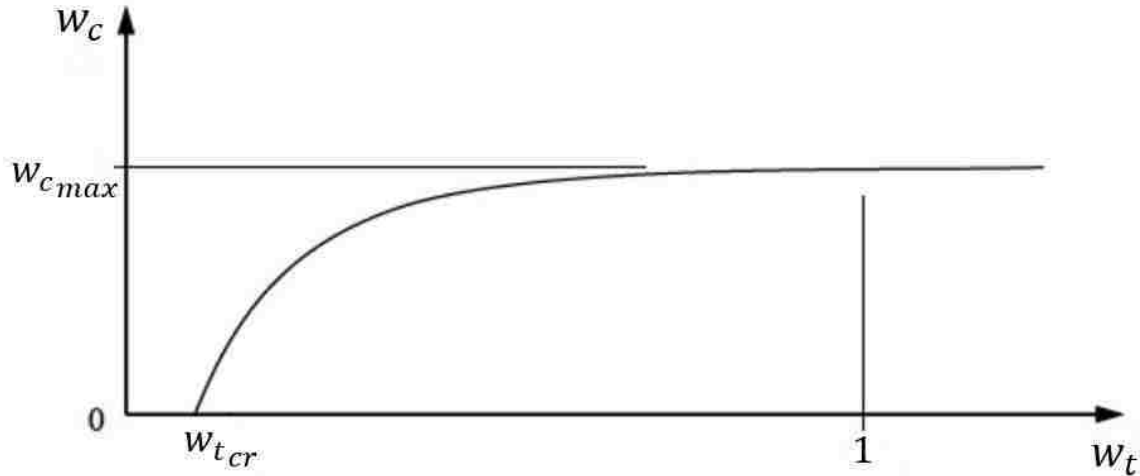


Figure 4.7. compressive damage parameter, w_c , curve in SPLM.

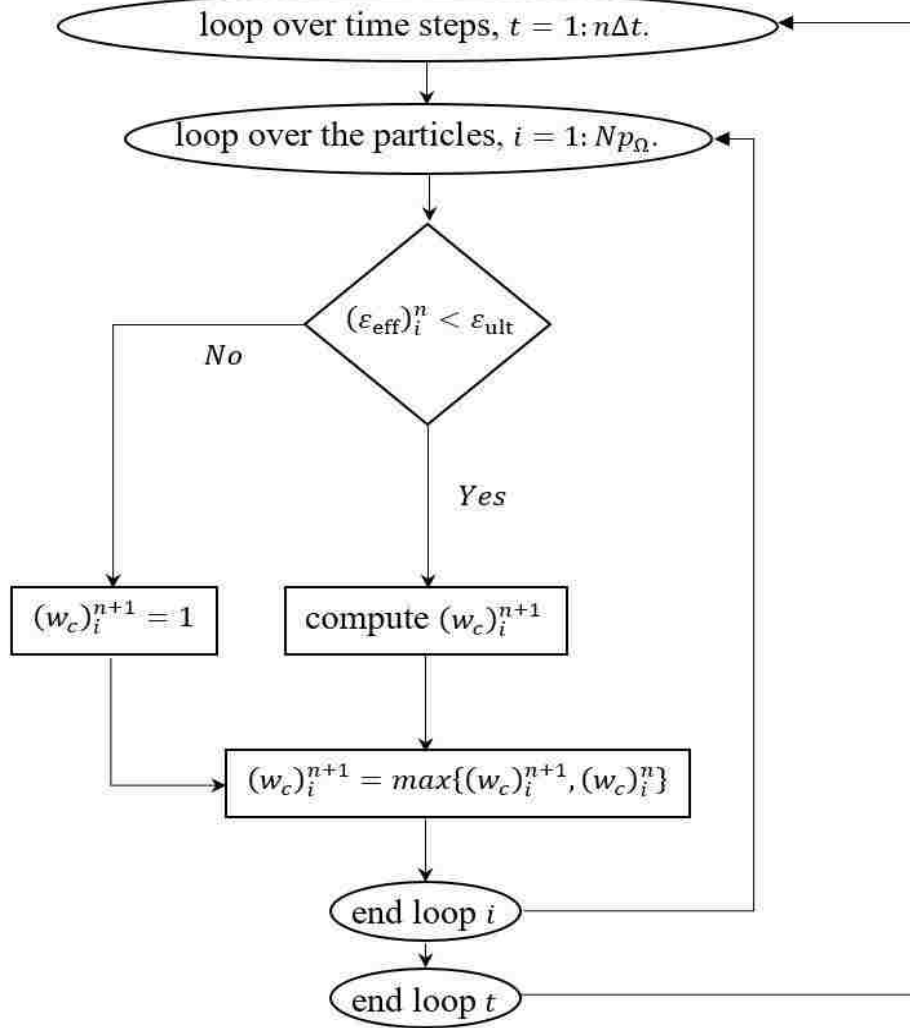


Figure 4.8. Flowchart of calculating the compressive damage parameter (w_c).

The plastic failure condition (as shown in Fig. (4.8)) is defined based on the multiaxial yielding function, ε_{eff} , which is assumed as a function of principal strains, as

$$\varepsilon_{\text{eff}} = \sqrt{\frac{1}{2}((\varepsilon_1 - \varepsilon_2)^2 + (\varepsilon_2 - \varepsilon_3)^2 + (\varepsilon_3 - \varepsilon_1)^2)} , \quad (4.17)$$

where ε_1 , ε_2 , and ε_3 are the principal strain components (defined in Eq. (3.61)). Once ε_{eff} at particle i becomes greater than the ultimate strain value, ε_{ult} , (user defined), the compressive damage parameter will jump to 1 (full damage). In other words, partial plastic damage is directly not modeled in this approach. However, since ε_1 is involved in the calculation of tensile damage parameter, and ε_1 is a function of plastic strains; partial damage due to plasticity is indirectly included in this model.

Chapter 5

Numerical Results

5.1. Introduction

In this chapter, the numerical results of simulating the plain concrete are presented and discussed. In this study, generally, the obtained results from the re-formulated SPLM is compared with older SPLM versions, the mentioned Abaqus models, and the lab tests.

In section 5.2, the re-formulated SPLM is compared with the older versions. The comparison between SPLM, Abaqus, and experimental tests are conducted in section 5.3, and some convergence studies are performed in section 5.4.

5.2. Re-formulated SPLM versus the older versions

In this section, three benchmark problems of uniaxial tension, uniaxial compression, and Brazilian split cylinder are considered; simulated with the re-formulated SPLM approach (named SPLM-2017 in this chapter) and compared with the results of a version namely SPLM-2016 (which is a modified and debugged version of the older versions of SPLM). The mentioned problems are also solved for three different lattice rotations to verify the objectivity of the proposing approach. The results of the static analysis of SAP2000 (fictitious crack model) are also included to examine the capability of the SPLM codes to provide similar solutions with the classical methods.

To have a fair comparison between new and older SPLM codes, SPLM-2016 is implemented by the author and considered as the best possible representative of the versions following the wrong algorithm (all the minor issues are fixed).

5.2.1. Material properties and parameters

Note that arbitrary material properties are considered for this part of the study (material properties do not perfectly match with a specific lab test data here; however, they can be considered as typical common properties for normal concrete as frequently reported and employed in various concrete references).

The concrete properties and parameters, defined in the models, are shown in Table. 1. The tensile strength of concrete is calculated using ACI 318-92 correlation [27] as follows

$$f'_t = 6\sqrt{f'_c}, \quad (\text{in US units}) \quad (5.1)$$

and, the elasticity modulus, E , is calculated based on ACI 318-14 design code [28] as

$$E = 33\rho^{\frac{3}{2}}\sqrt{f'_c}, \quad (\text{in US units}) \quad (5.2)$$

where ρ and f'_c are, respectively, mass density and compressive strength of concrete. In this study, the fracture energy, G_f , is considered as a material property and the critical Crack Opening Displacement, COD_c , is calculated based on the fracture energy. The following correlation for estimating the fracture energy based on the maximum aggregate size is proposed by CEB-FIP-90 code [29] as

$$G_f = G_{f0} \left(\frac{f_{cm}}{f_{cm0}} \right)^{0.7}, \quad (5.3)$$

where G_{f0} is the base fracture energy, f_{cm} is the mean compressive strength which is defined as a function of the characteristic compressive strength as $f_{cm} = f'_c + 8 \text{ Mpa}$, and $f_{cm0} = 10 \text{ Mpa}$. The value of G_{f0} depends upon the maximum aggregate size, d_{max} [29]; Once the fracture energy is estimated from Eq. (5.3), the critical Crack Opening Displacement, COD_c , can be calculated from the area underneath the considered tension softening curve (inelastic region) of concrete, Fig. (4.4), as

$$COD_c = \frac{2G_f}{f'_t(\alpha + \gamma)}, \quad (5.4)$$

and COD_1 (see Fig. (4.4)) is calculated as $COD_1 = \alpha COD_c$, where α is assumed to be 0.1. All the other parameters are introduced in Table 1.

Table 1. Material properties and constant parameters for section 5.2.

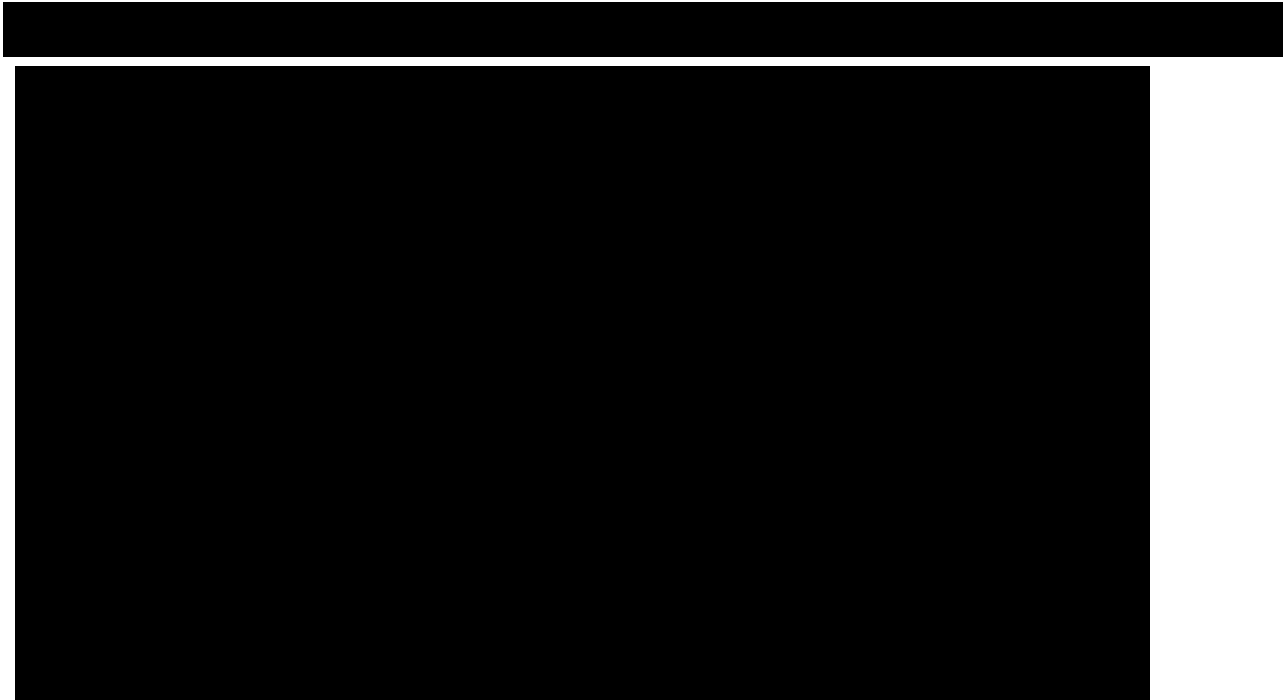
Parameter	Uniaxial Tension (SI)	Uniaxial Compression (SI)	Split Brazilian Cylinder (SI)
Compressive strength, f'_c	27.580 Mpa	27.580 Mpa	27.580 Mpa
Yielding strength, f_y	22.983 Mpa	22.983 Mpa	22.983 Mpa
Tensile strength, f'_t	2.6168 Mpa	2.6168 Mpa	2.6168 Mpa
Mass density, ρ	2323.0 kg/m ³	2323.0 kg/m ³	2323.0 kg/m ³
Maximum aggregate size, d_{max}	0.019 m	0.019 m	0.019 m
Base fracture energy, G_{f0}	33.49 J/m ²	33.49 J/m ²	33.49 J/m ²
Poisson's ratio, ν	0.2	0.2	0.2
Damping ratio external, ξ_{ex}	0.2	0.2	0.2
Damping ratio internal, ξ_{in}	0.2	0.2	0.2
Tensile damage parameter, γ	0.25	0.25	0.25
Lattice spacing, L_0	0.01 m	0.01 m	0.005 m

5.2.2. Uniaxial Tension Problem

In this problem, a cubic concrete specimen of size $0.3m \times 0.15m$ with the thickness of $0.15m$ is considered. The boundary conditions are defined such that the bottom boundary is fixed only in y direction and the time-varying displacement (Eq. (5.5)) is applied to the top boundary. In SPLM, the mentioned boundary conditions are imposed to two rows of particles at top and bottom of the specimen (shown in the results by green particles).

$$\Delta_y(t) = \left(\frac{\Delta_{max}}{2} \right) \left(1 - \cos \left(\frac{\pi t}{t_{rampend}} \right) \right). \quad (5.5)$$

In Eq. (5.5), Δ_{max} is the maximum applied displacement assumed to be calculated as $\Delta_{max} = 1.5COD_c$, $t_{rampend} = 0.8t_{end}$, and t_{end} is calculated based on the fundamental period of vibration of the specimen and the time stepping increment (more details can be found here [14]). Note that the fundamental periods of the specimens are estimated from the linear elastic modal FEM analysis of each problem. For this problem, fundamental period of $0.00042 s$ is estimated and used. The mentioned problem is solved with both SPLM-2016 and SPLM-2017; each for three different lattice rotations of 0 , 15 , and 30 degrees. The obtained Force-displacement curves are shown in Fig. (5.1), the cracking pattern obtained for each lattice rotation is shown in Fig. (5.2), and data comparisons and some additional information regarding the outputs (i.e. obtained peak loads for each analysis, number of particles, etc.) are provided in Table 2. The results show significant improvements (specifically in post-peak region) in SPLM-2017; also much better match with the classical solutions.



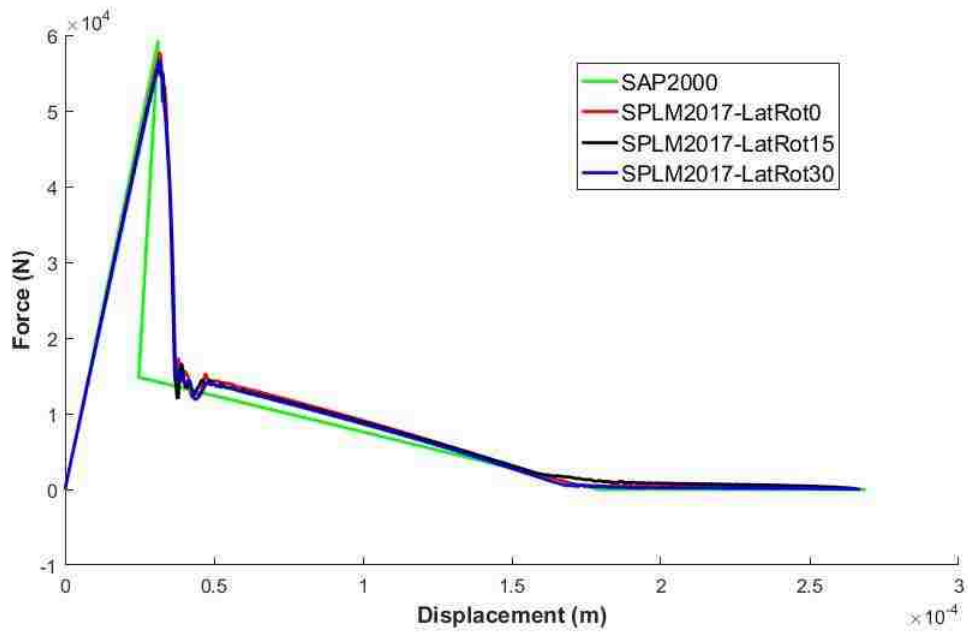
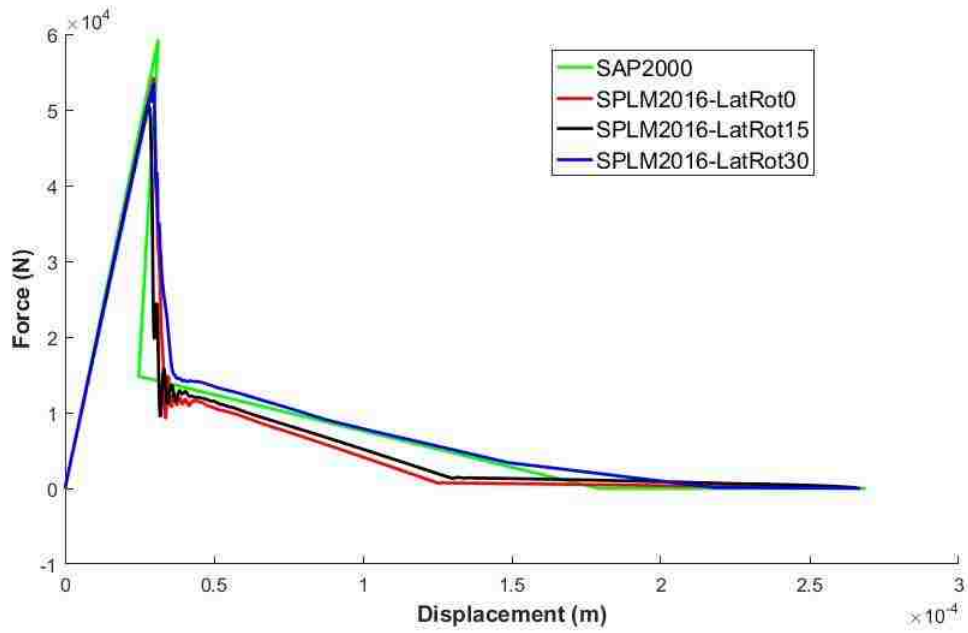


Figure 5.1. Comparison between SPM-2016 and SPM-2017 for Uniaxial Tension problem

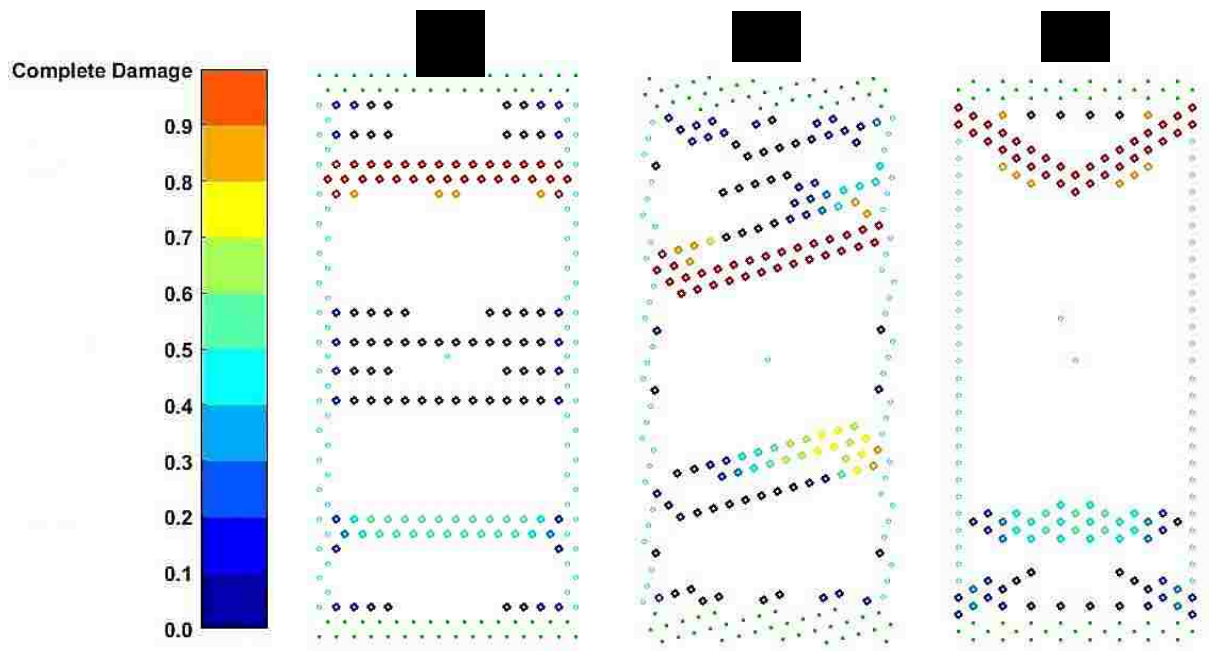


Figure 5.2. Obtained cracking patterns for Uniaxial tension problem at the final timestep, for SPLM-2017, with lattice rotations of: (a) 0 degree, (b) 15 degree, and (c) 30 degree.

5.2.3. Uniaxial Compression Problem

In this problem, a cubic concrete specimen of size $0.3m \times 0.15m$ with the thickness of $0.15m$ is considered (the same as tension problem). The same boundary conditions as uniaxial tension problem are considered for this case, except the direction of the applied displacement is flipped in order to apply compression to the specimen.

This problem is also solved with both SPLM-2016 and SPLM-2017; each for three different lattice rotations of 0, 15, and 30 degrees. The obtained Force-displacement curves are shown in Fig. (5.3), the cracking pattern obtained for each lattice rotation is shown in Fig. (5.4). Note that the yielded particles due to plasticity are shown with the black color. The data comparisons are provided in Table 3. Significant improvements can be recognized for the new SPLM code based on the presented results.



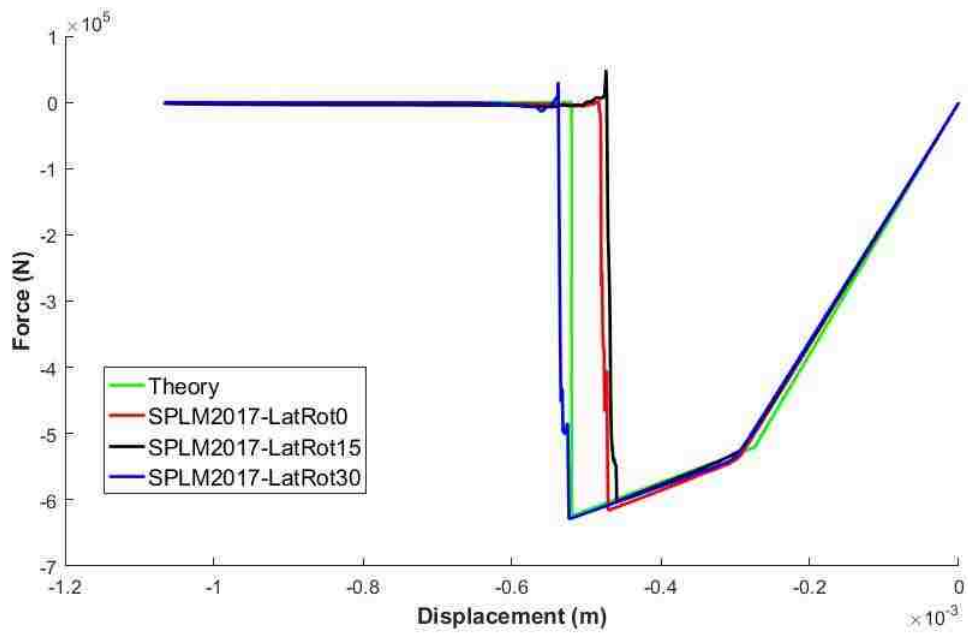
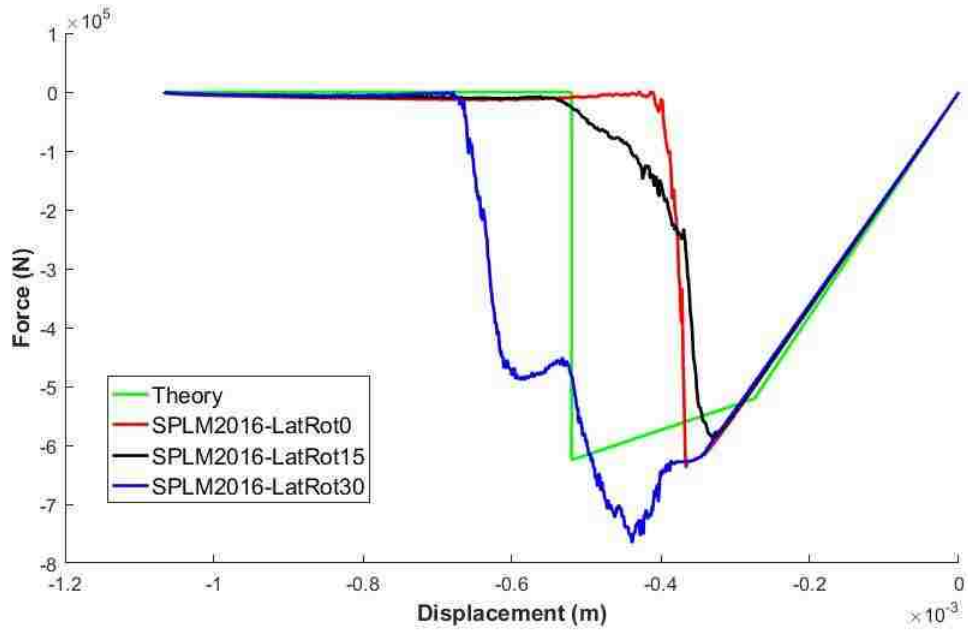


Figure 5.3. Comparison between SPLM-2016 and SPLM-2017 for Uniaxial Compression problem

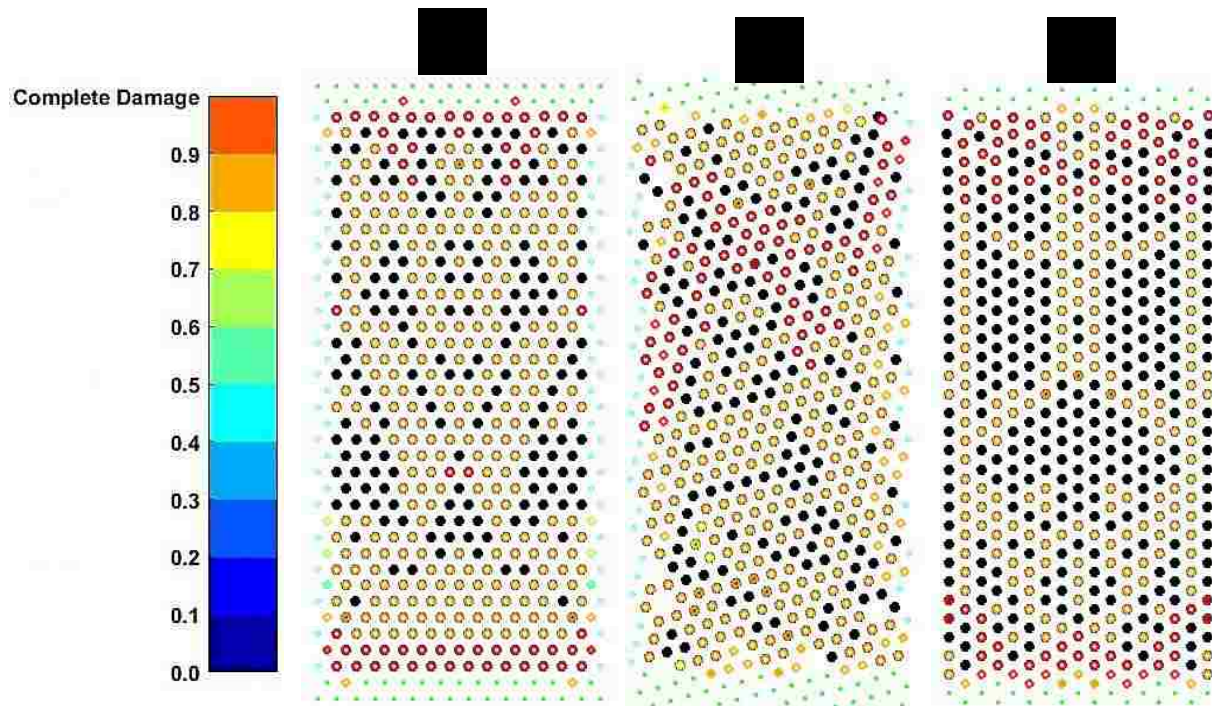


Figure 5.4. Obtained cracking patterns for Uniaxial Compression problem at the final timestep, for SPLM-2017, with lattice rotations of: (a) 0 degree, (b) 15 degree, and (c) 30 degree.

5.2.4. Split Brazilian Cylinder

In this problem, a circular cylinder with a diameter of 0.15m and height of 0.30m, subjected to compression along its diameter, is simulated. The same time varying displacement (Eq. (5.5)) is applied at top and bottom of the Split Cylinder models. In SPLM the displacement is applied to certain particles, defined at top and bottom of the specimen (the green particles shown in the results), to emulate the loading plates.

This problem is also solved with both SPLM-2016 and SPLM-2017; each for three different lattice rotations of 0, 15, and 30 degrees. The obtained Force-displacement curves are shown in Fig. (5.5), the cracking pattern obtained for each lattice rotation is shown in Fig. (5.6). Note that the yielded particles due to plasticity are shown with the black color. The data comparisons are provided in Table 4. According to the obtained results, the superiority and robustness of the re-formulated SPLM method versus the older versions can be verified.



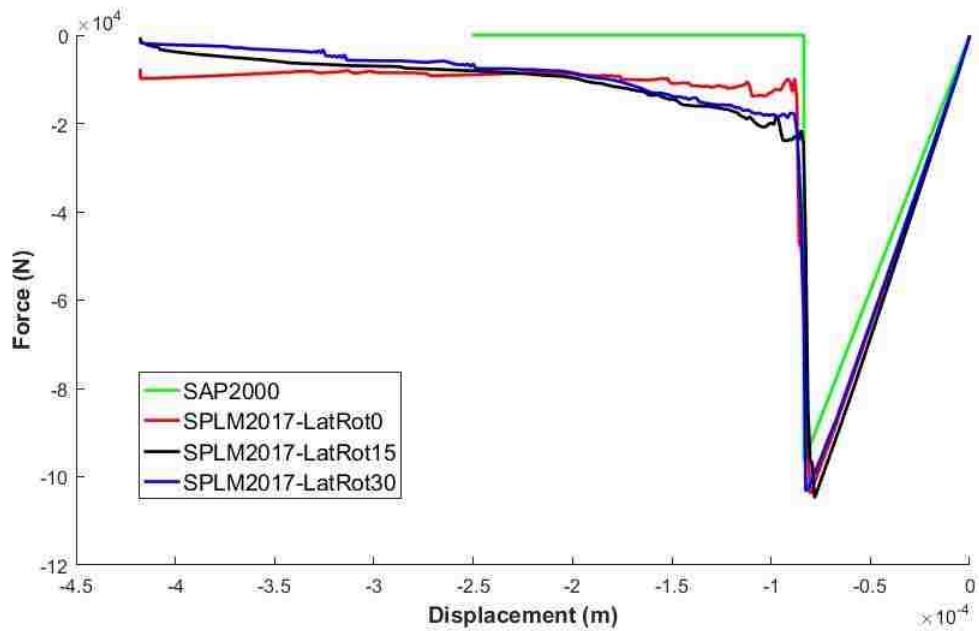
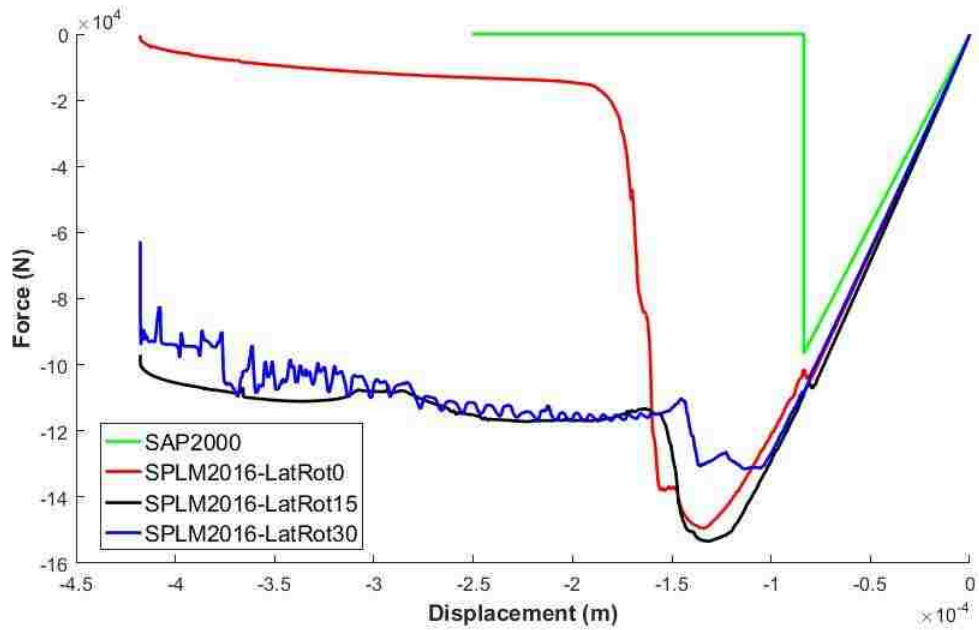
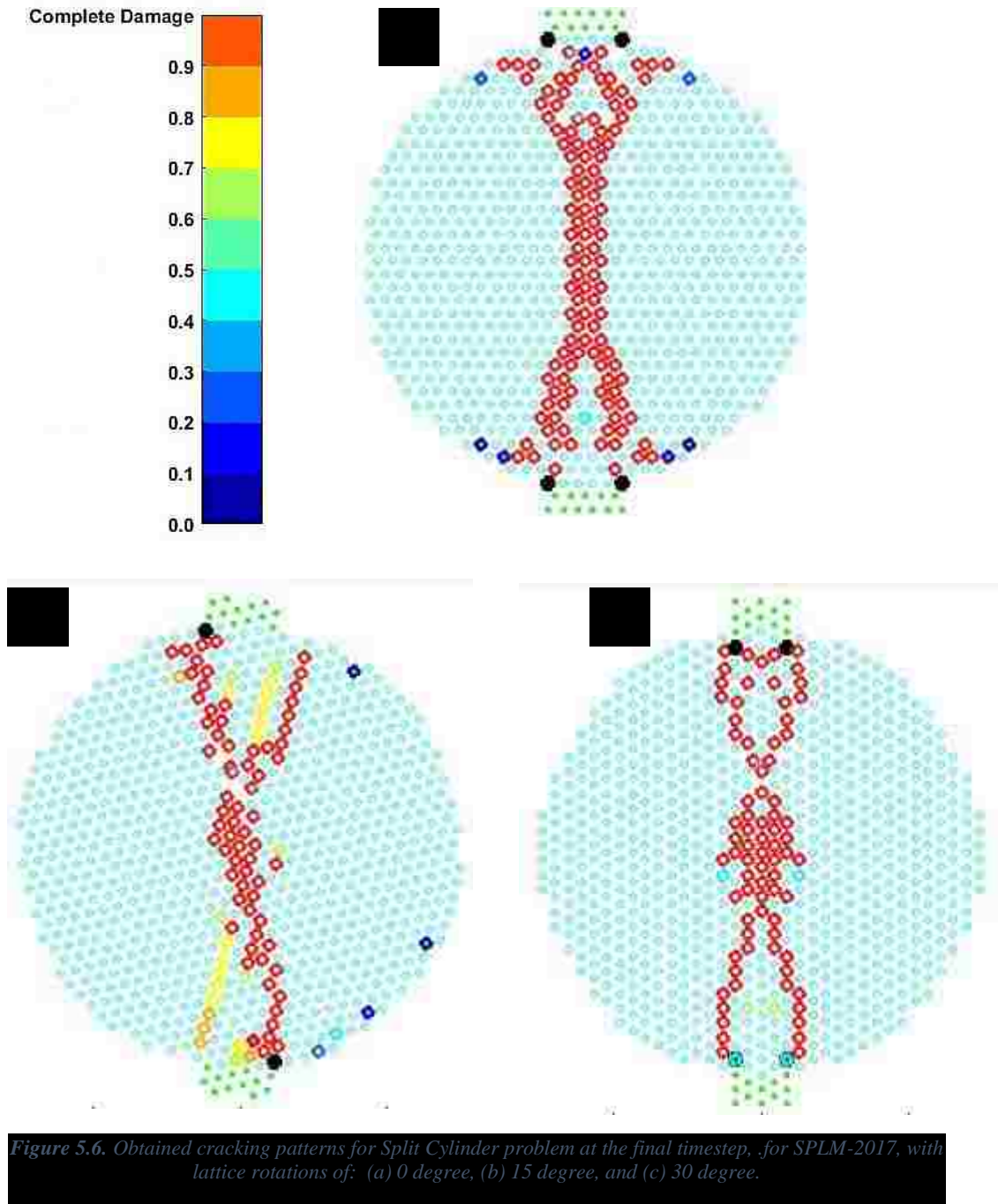


Figure 5.5. Comparison between SPLM-2016 and SPLM-2017 for Split Cylinder problem



5.3. Re-formulated SPLM versus Abaqus and Experimental data

In this section, the problems of Dog-bone shaped concrete specimens, with two different sizes [1], and Split Brazilian Cylinder based on [30] are simulated with all the mentioned numerical approaches. The obtained results are then compared together and validated with the lab tests.

The material properties and parameters used for the simulations are presented in sec. (5.3.1). the Dog-bone problems are discussed in sec. (5.3.2), and Split cylinder is considered in sec. (5.3.3).

5.3.1. Material properties and parameters

For the Dog-bone problems, concrete properties are mostly defined based on [1] and for the Split Cylinder problem based on [30]; however, some of the properties that were not clearly presented in the lab data are decided to be calculated from the design codes.

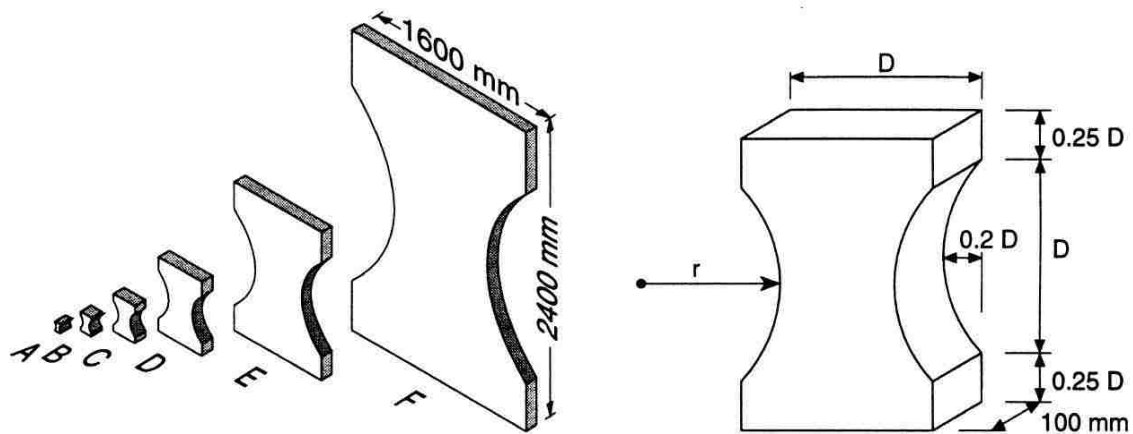
The concrete properties and parameters, defined in the models, are shown in Table. 5. For all problems solved in this section, since the reported value of tensile strength in reference papers are not reliable (presumably not represent the real tensile strength because of the difficulties in performing the experiments); the tensile strength of concrete is calculated using the ACI formula (Eq. (5.1)). The Young's modulus is also estimated from Eq. (5.2), and Eq. (5.4) is considered for calculating critical crack opening displacement. Note that the applied displacement-rate is considered slow enough so that avoid probable pre-peak nonlinearities and dynamic effects. Moreover, the external damping is not defined in either Abaqus or SPLM models.

Table 5. Material properties and constant parameters in section 5.3.

Parameter	Split Brazilian Cylinder (SI)	Dog-Bone specimens (SI)	
Compressive strength, f'_c	41.23 Mpa	50.0 Mpa	
Tensile strength, f'_t	3.20 Mpa	2.415 Mpa	
Yielding strength, f_y	34.36 Mpa	41.67 Mpa	
Mass density, ρ	2323.0 kg/m ³	2183.0 kg/m ³	
Maximum aggregate size, d_{max}	0.019 m	0.008 m	
Base fracture energy, G_{f0}	33.49 J/m ²	25.00 J/m ²	
Poisson's ratio, ν	0.2	0.2	
Damping ratio internal, ξ_{in}	0.2	0.2	
Damping ratio external, ξ_{ex}	0.002	0.002	
Tensile damage parameter, γ	0.25	0.25	
Element size, \bar{L}	0.005 m	0.008 m	
Lattice spacing, L_0	0.005 m	0.008 m (for D)	0.004m (for B)

5.3.2. Dog-Bone Specimens Under Uniaxial Tension

As it mentioned, the specimens of sizes B and D [1] are selected and simulated for this study. The specimen dimensions are shown in Fig. (5.7). The boundary conditions are defined such that the bottom boundary is fixed only in y direction and the time-varying displacement, $\Delta_y(t)$, (see Eq. (5.5)) with the maximum displacement, Δ_{max} , of 0.5mm is applied to the top boundary. The fundamental periods of 0.0017s for Specimen D, and 0.00042s for specimen B are considered. In SPLM, the mentioned boundary conditions are imposed to 2~3 rows of particles at top and bottom of the specimen (the green particles shown in Figs. (5.9) and (5.10)); while in Abaqus models, the boundary conditions are directly applied to the boundary edges. A structured, symmetric, 4-node plane stress quadrilateral elements (CPS4R), with the total number of 220 elements for specimen B and 3680 elements for specimen D, are exploited in Abaqus models. The Abaqus dynamic-explicit solver is considered for analyzing the problems. The force-displacement curves for specimens, B and D, comparing SPLM and Abaqus results, are plotted in Fig. (5.8). The obtained cracking patterns at final time step are shown in Figs. (5.9) and (5.10). Note that the contour levels shown in Figs. (5.9) and (5.10) are defined the same in both SPLM and Abaqus models. the obtained peak loads are shown in Table 6; and compared with those of theoretical engineering peak loads. Some other information (number of particles/elements) are also gathered in Table. 6.



Type	A	B	C	D	E	F
D (mm)	50	100	200	400	800	1600
r (mm)	36.25	72.5	145	290	580	1160

Figure 5.7. Dog-Bone specimen dimensions (from [1]).

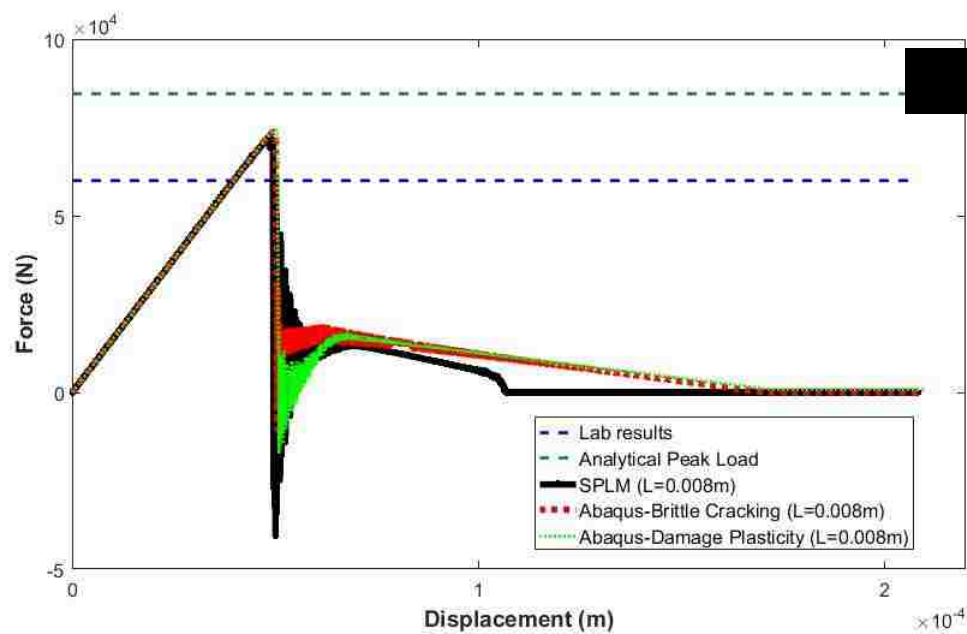
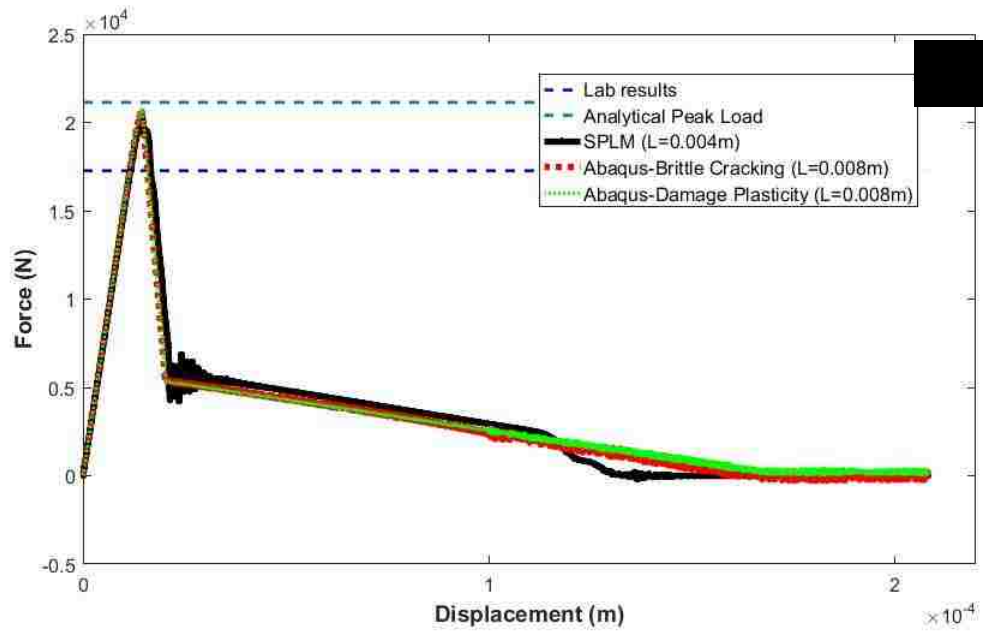


Figure 5.8. Force-displacement curves of Dog-bone problems. (a) Specimen B, (b) Specimen D.

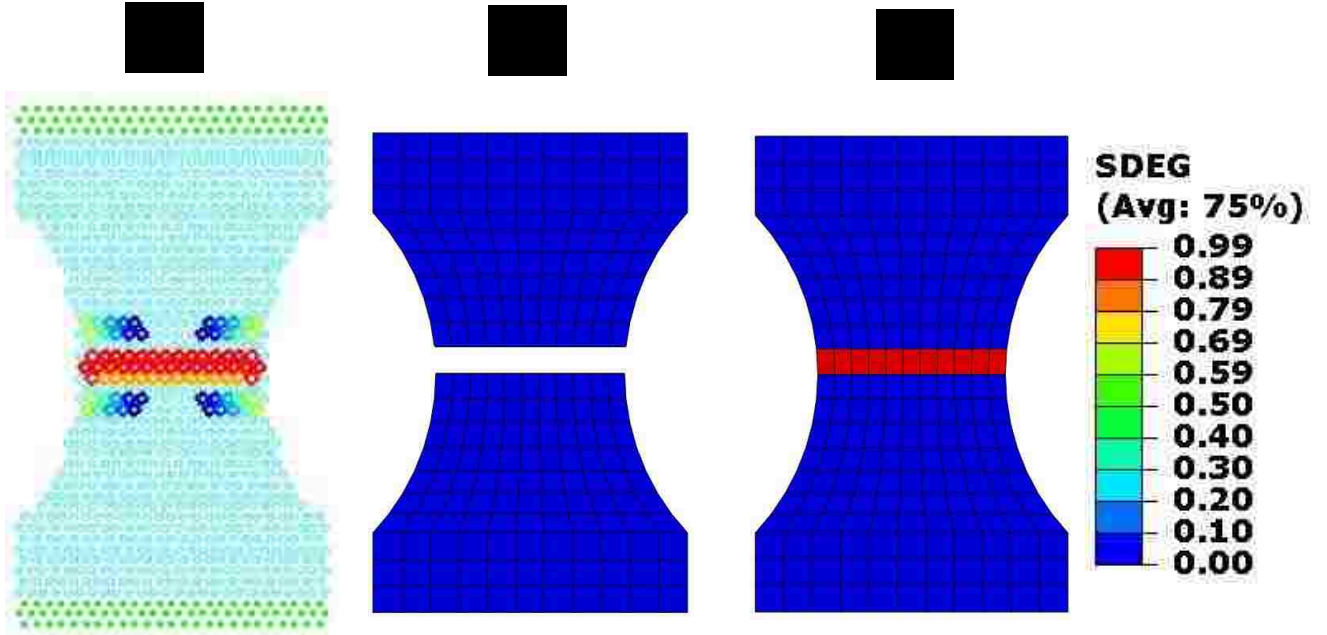


Figure 5.9. Obtained cracking patterns for specimen B, at the final timestep. (a) SPLM, (b) Abaqus brittle Cracking (STATUS), (c) Abaqus Damage Plasticity (SDEG).

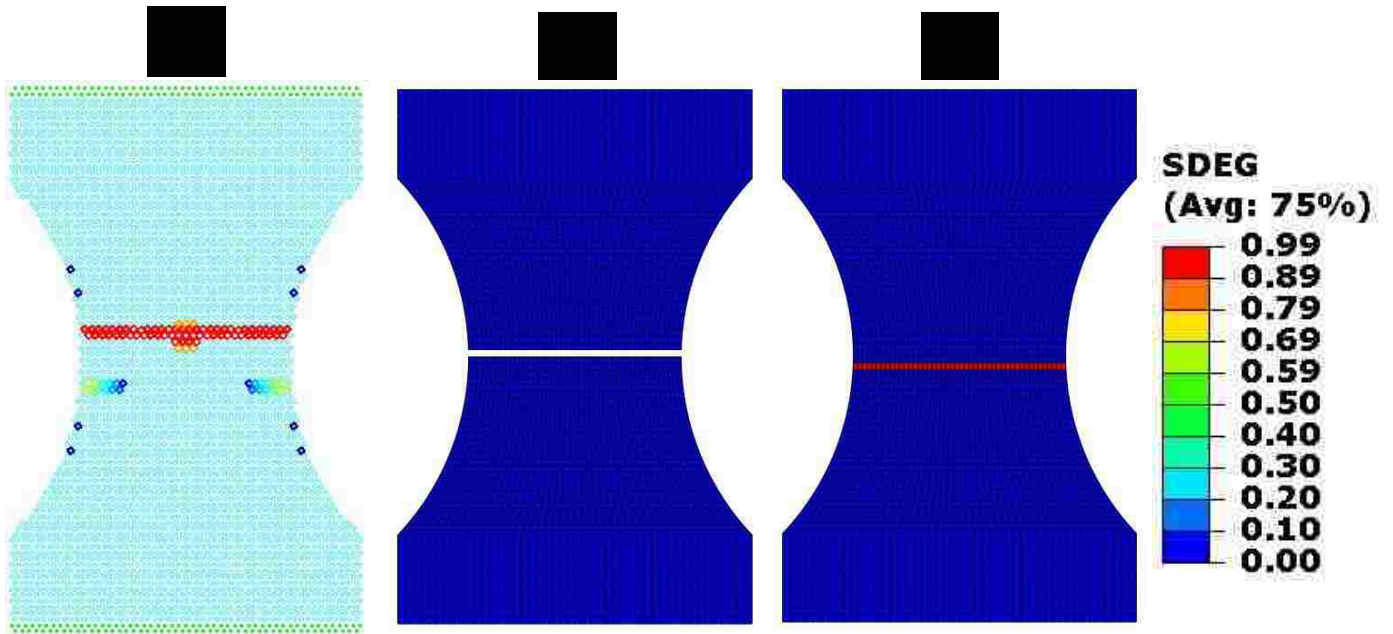


Figure 5.10. Obtained cracking patterns for specimen D, at the final timestep. (a) SPLM, (b) Abaqus brittle Cracking (STATUS), (c) Abaqus Damage Plasticity (SDEG).

Before starting the interpretation of the outcomes, the following difficulties of all the Abaqus concrete cracking models, mentioned in chapter 2, should be addressed. The SPLM also has some issues listed as follows.

- 1) The proposed lattice topology lead to a condition of having less number of generated particles in certain rows along the width of the specimens, and subsequently less bonds. Therefore, the failure in SPLM models will occur along the smaller defined width of material (which means lower estimation of the peak load).
- 2) One of the differences between SPLM and continuum based methods is the difference in the stiffness of the boundary particles with the particles in the bulk. In Abaqus approaches, before getting damage, the stiffness of all the elements (on the boundaries or inside the domain) are the same. However, in SPLM approach, even in undamaged case, all the boundary particles are connecting to less bonds and subsequently having lower stiffness. This issue is mostly related to the estimation of the bond-based micro-elastic modulus a (defined in Eq. (4.2)) which is calculating based on having 6 bonds in the horizon, while for boundary particles we have less than 6 bonds (in other words less stored elastic energy per particle in boundaries) [14]. This issue implies having intrinsically less stiffness in boundaries and affect the obtained peak loads and cracking patterns.

In light of the mentioned difficulties, the obtained results For the Dog-bone problems can be interpreted as follows.

As it can be seen from the figures and Table 6, by comparing the numerical results, in both case of small and big concrete specimens, the Brittle cracking, Damage plasticity, and

SPLM models roughly reached the same peak load (within 2% of variation). On the other hand, for specimen B, the obtained peak loads are almost matching with the analytical solution; however, the obtained peak loads are lower than the analytical solution in case of Specimen D. The probable reason of getting lower peak loads for Specimen D can be because of ignoring the size effects in both calculation of the analytical solution and also calculation of the tensile strength of concrete (lower strength should be considered for bigger specimens).

Comparing numerical and lab results, the predicted peak loads from all the numerical simulations are higher. It should be noted that in this study, the size dependency of the tensile strength (as it can be verified from [1]) is ignored; and the values based on ACI formula (Eq. (5.1)) for the tensile strengths are employed for numerical simulations (shown in Table 5). Therefore, since ACI correlation is overestimating the tensile strength, using the exact reported values for the tensile strength from [1] will lead to lower peak loads. Nevertheless, it should be considered that the correctness of the reported nominal tensile strengths are also questionable; hence, using ACI correlation seems to be more reliable.

Comparing the force-displacement curves, the pre-peak behavior of all the numerical approaches are almost identical. However, the post peak behaviors are slightly different which is rational and happens due to having different damage techniques and failure methods in the models. For the smaller specimen, specimen B, less post peak vibrations are captured from the numerical simulation which is reasonable due to the small size of the specimen B (small mass). On the other hand, for the bigger specimen, specimen D, more vibrations can be seen in the post-peak regions of the curves, as it is expected.

As it can be seen from the cracking patterns of Dog-bone models (Figs. (5.9) and (5.10)), in general, Abaqus models are showing more localized crack (a single crack in the middle) than the SPLM approach. Theoretically, the element sizes should not be defined smaller than the aggregate size (minimum size of the fracture process zone in concrete). In the case of specimen B, since the size of the specimen is so small comparing the specified element size, the convergence problem is so probable because of having insufficient number of elements. This issue, Inaccuracy and lack of objectivity in modelling the small specimens due to meshing limitations, should be addressed as one of the major difficulties of FEM based fracture models. As it can be seen from the Abaqus results for specimen D, a single crack is propagated exactly at the center of the specimen. However, in SPLM, a complete crack (from above the center line of the shape) and some partial damages are propagated. Therefore, too much localized cracking issue is also evident in the results of Abaqus approaches for the bigger specimen which is not so realistic.

Furthermore, based on the obtained results, SPLM shows the superior capability of predicting size effect in concrete specimens. According to the size effect theory, which is verified in the reference paper for this study [1] and also other literature [31], by increasing the size of the specimen we should expect lower tensile strength. As it can be seen from Fig. (5.10), in the bigger specimen the ultimate damage occurred in the thicker region of the specimen which has a bigger effective area. Therefore, if we recalculate the nominal

strength using that bigger area, we would get lower strength. Hence, this issue could justify the capability of SPLM method in predicting the size effect.

In conclusion, SPLM is showing reasonable conformity (in both pre-peak and post-peak) with the well-developed Abaqus codes. The obtained solutions are also corresponding with those of theoretical solutions and experimental results.

5.3.3. Brazilian Split Cylinder

In this problem, a circular cylinder with a diameter of 0.15m and height of 0.30m, subjected to compression along its diameter, is simulated. The same time varying displacement (Eq. (5.5)) is applied at top and bottom of the Split Cylinder models. In Abaqus models, in order to apply a uniform displacement, a length of one-sixth of diameter (0.025m) from top and bottom boundaries of cylinder is flattened; while, in SPLM the displacement is applied to certain particles, defined at top and bottom of the specimen (the green particles shown in Fig.(5.12)), to emulate the loading plates. In Abaqus models, the same element types as the Dog-Bone problems used. The total number of 864 elements in Abaqus models and 839 particles in SPLM are exploited. The obtained force-time curves of the simulations are plotted in Fig. (5.11). The cracking patterns are shown in Fig. (5,12). The theoretical solution for the peak load of the split cylinder, P , considering the effect of the loading block, is given [32] as

$$P = \frac{\pi}{2} (f'_t DH) / (1 - \beta^2)^{1.5}, \quad (5.6)$$

where D is the diameter, H is the height, $\beta = (\text{block width})/D$, and f'_t is the tensile strength of the cylinder. Although the extent of validity of Eq. (5.6) is not clear, the mentioned correlation is employed to estimate the analytical peak load. The obtained peak loads are presented in Table 6. Note that the theoretical peak load, shown in Fig. and Table 6, is calculated based on the value of f'_t obtained from ACI correlation (Eq. (5.1)).

As it can be seen from Fig. (5.1), the pre-peak and post-peak behavior obtained from SPLM and damage plasticity approaches are the same; however, the obtained peak load and post peak behavior of Brittle cracking model is different. Brittle Cracking method shows much higher peak load than other methods which is essentially because of inability of this method in modelling the compressive failure. Comparing the cracking patterns (Fig. (5.12)), in spite of having much simple damage method in SPLM; Damage-Plasticity and SPLM are showing reasonable and similar crack propagation (tensile fracture at the center and compressive crushing at the loading boundaries). However, Brittle Cracking method is showing unrealistic cracking patterns (i.e. horizontally propagated cracks).

Considering the obtain peak loads (shown in Table 6), SPLM and Damage-Plasticity approaches are showing conformity with the theoretical solution and lab test results. As it mentioned, the presented experimental peak loads in Table 6 are estimated based on the reported tensile strengths of cylinder, values of 2.79 Mpa and 3.24 Mpa [30]. Assuming the validity of Eq. (5.6) for calculating the peak load based on cylinder tensile strength, the cylinder peak load values of 204 KN, and 236 KN can be estimated (β is assumed zero in this part to comply with [30]).

Therefore, SPLM is showing identical results to Damage Plasticity method and also consistent with the experimental and theoretical solutions.

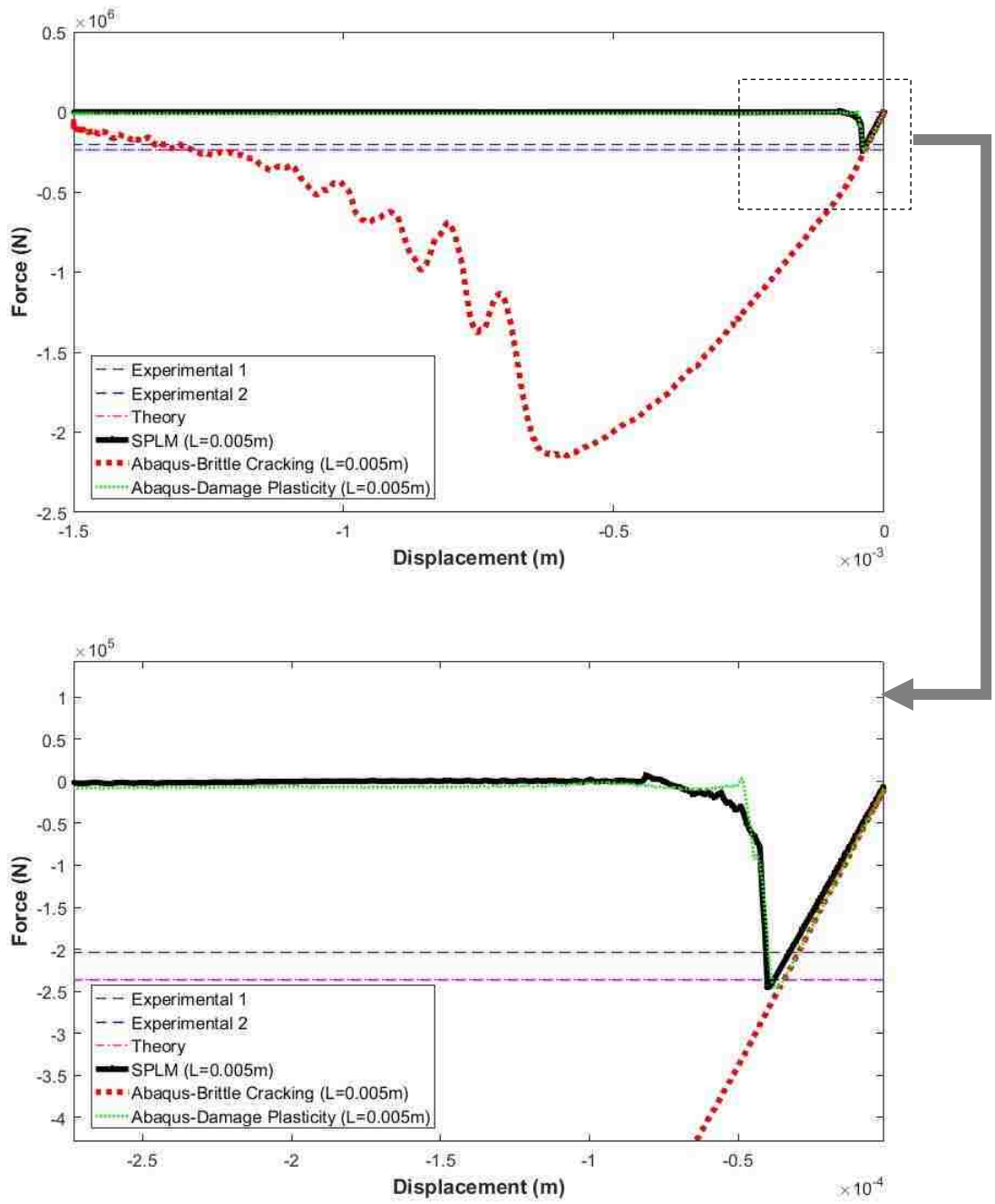


Figure 5.11. Force-displacement curves obtained for Split Brazilian Cylinder.

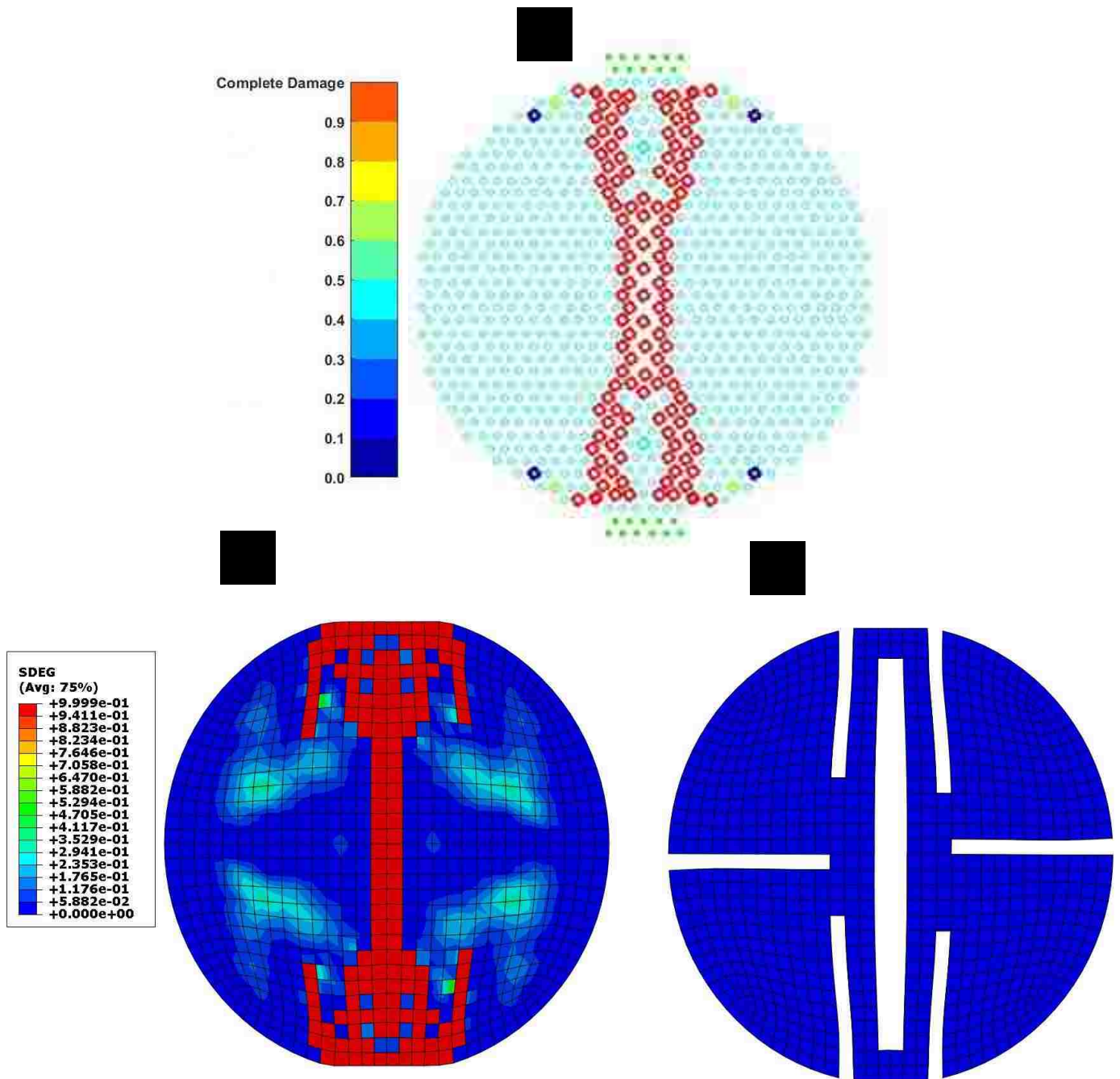


Figure 5.12. Obtained cracking patterns for Split Cylinder problem at the final time step. (a) SPLM, (b) Abaqus Damage-Plasticity (SDEG), (c) Abaqus Brittle Cracking (STATUS).

5.4. Convergence Study

In this section, a small convergence study is performed to show the capability of the new SPLM approach to converge to the classical solutions. Here, the small Dog-bone specimen (introduced in section 5.3.2) is solved with different lattice spacings. The results are then compared with Abaqus models. Note that the mesh sizes used in Abaqus analysis are not changed here. The lattice spacings of 0.004m and 0.008m are chosen for this study. The obtained cracking patterns are shown in Fig. (5.13), and force-displacement plots are presented in Fig. (5.14). As it can be seen from Fig. (5.14), by making the lattice spacing twice smaller, the pre-peak and post-peak results are both improve, and almost match with the Abaqus outputs.

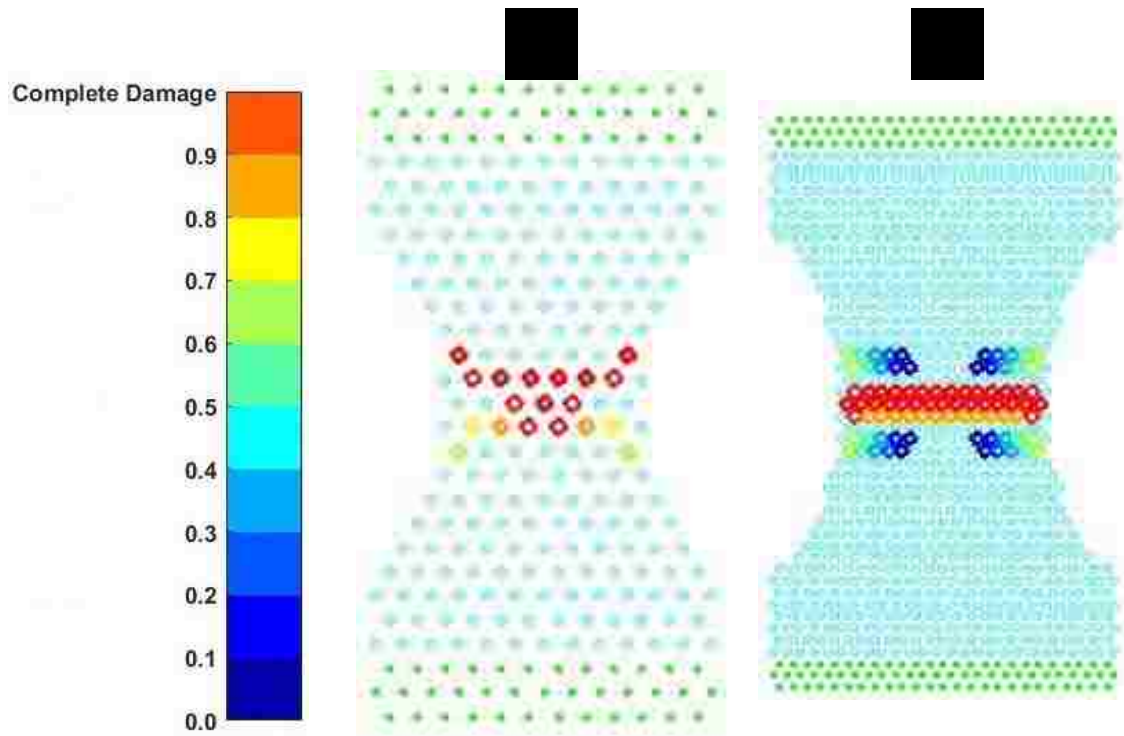


Figure 5.13. Obtained cracking patterns for Dog-bone Specimen B.
(a) Course mesh (287 particles), (b) Fine mesh (1035 particles).

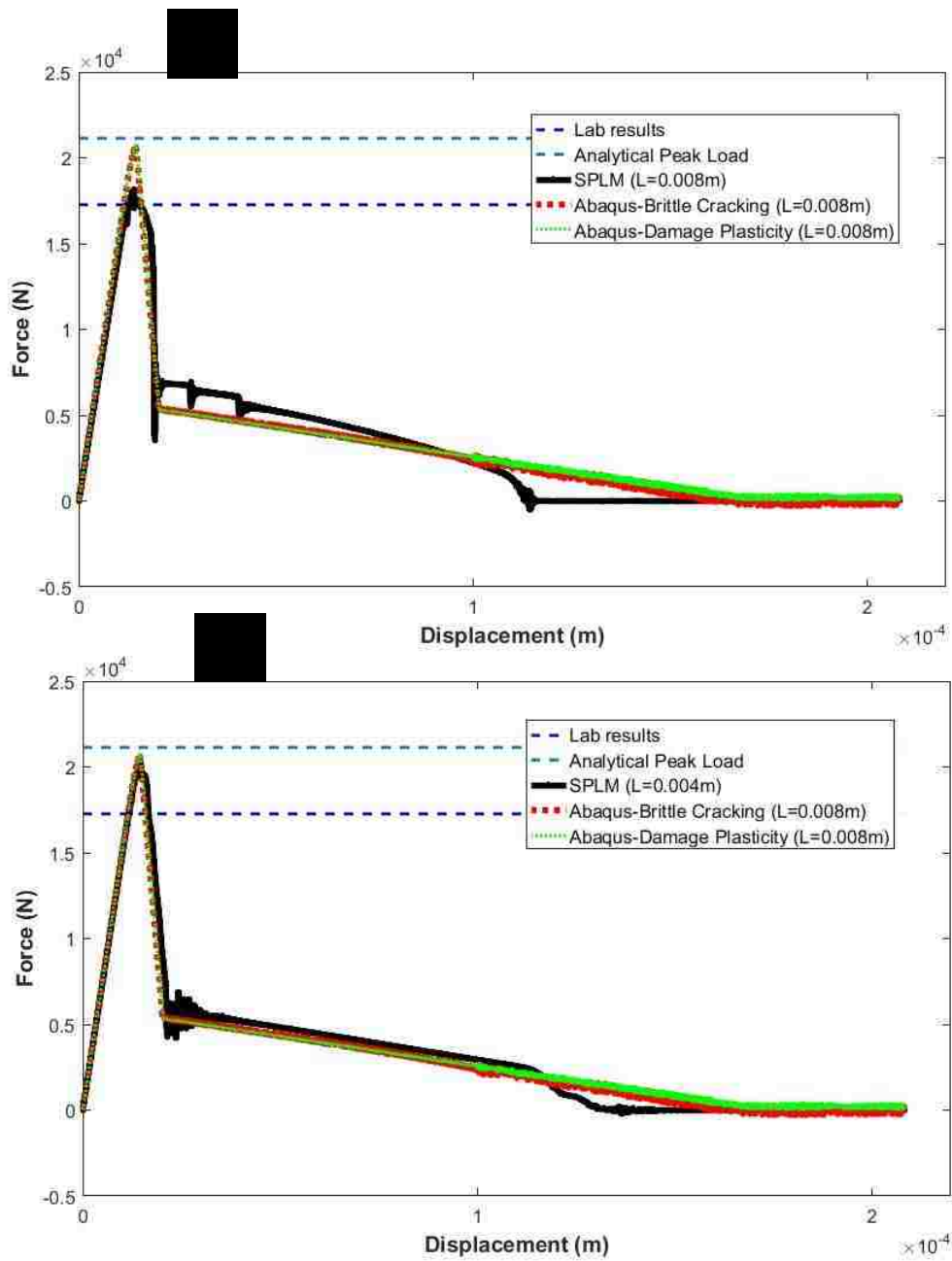


Figure 5.14. Obtained Force-displacement curves for Dog-bone specimen B (convergence study).
 (a) Course mesh (287 particles), (b) Fine mesh (1035 particles).

Chapter 6

Discussions and Conclusions

6.1. Summary

In this study, the State-Based Peridynamics Particle Approach (SPPM) is proposed for solid mechanics. SPPM can be considered as a re-formulation of continuum state-based peridynamics, based on discrete, randomly-positioned, particles. In SPPM, instead of solving integral equations, discrete equations in the form of finite summations are solved. A spring theory is combined with the state-based peridynamics concept, and a novel damage model called the “Two-Spring Damage Model” is introduced. A suitable and simple plasticity model is also proposed for SPPM. The advantages of SPPM method are listed as follows.

- (1) Computational implementation of SPPM is more efficient than continuum peridynamics. Since SPPM starts with discrete equations, the equation discretization cost is lower than with continuum peridynamics (numerical integration may need more nodes than what is used for domain discretization, while in SPPM there is no need to define extra integration points). In addition, the proposed meshless algorithm, in general, decreases the computational cost of the domain discretization.
- (2) SPPM formulation is more accurate than continuum peridynamics. Considering the mentioned issues in number (1), SPPM directly deals with discrete equations and no numerical integration is needed in this approach. SPPM has the capability to model complex geometries, and boundaries can be modeled with more accuracy than the continuum peridynamics.
- (3) Implementation of damage theories are more convenient within the SPPM framework. The SPPM formulation is easier to implement than continuum peridynamics.

Next, the SPLM method was re-formulated and calibrated for concrete. SPLM can be considered as a lattice-based form of SPPM since the lattice mesh is employed to distribute the particles. The reason of proposing a lattice-based approach while having a meshfree formulation is listed as follows.

- (1) Computational efficiency. In general, all the Peridynamics-based approaches need computation power since lots of floating point operations are involved in Peridynamics computations. Having a lattice configuration simplifies the calculations and increases the computational efficiency.
- (2) Having a more deterministic model. Having a random particle distribution may lead to varying solutions (specifically varying cracking patterns). Although this issue can be considered as one of the advantages of the SPLM/SPPM model, performing this study was not in the scope of this research.
- (3) Parallel processing. Generally, parallel programming (particularly the Message Passing Interface (MPI) method) is more complicated in cases of random particle distributions (for instance, defining the overlapping domains). The efficiency of

parallel domain discretization (allocating different cores to different domain region in MPI (load balancing issue)) would not be very high; since having a dense particle distribution in on region of the problem domain might result in performing most of the computations with only a few cores. The mentioned issues will affect the performance of the parallel approach.

By employing SPLM the accuracy in modelling geometries and boundary conditions will decrease. Therefore, although using lattice increases the computational efficiency of the approach, it will have the tradeoff of lower expected accuracy.

The re-formulated SPLM is then used to solve some 2D planer concrete problems. The obtained outputs are then compared with those of Abaqus concrete cracking models, theoretical solutions, and some available experimental data. The obtained results demonstrate that SPLM is capable of simulating concrete reasonably, and produces similar results to FEM-based methods and other theoretical solutions.

6.2. Discussion and Remarks

According to the results of our simulations, the following observations are made:

- (1) The SPLM formulation facilitates computations and decreases the computational cost. However, in addition to the difficulties of modeling smooth boundaries with the lattice, the following assumptions are also made in SPLM formulation to simplify the model, that cannot be neglected while analyzing the results:
 - a. Constant transformation matrix is used (Eq. (3.15)) based on the reference lattice configuration. (N_i will not update in each time step.)
 - b. Constant micro-elastic modules (a , and b) are employed for all the particles. (Boundary particles should have different a and b because of having less bonds.)
 - c. The same material volume is associated with all the particles, including boundary particles.
- (2) Comparing the re-formulated SPLM version to the older versions, the new SPLM is more accurate, objective, symmetric, convergent, and in general more reliable.
- (3) The changing the state of particles from compression to tension (and vice versa), after getting damage in the neighboring particles, is recognized and fixed in this study; which is a complex and realistic phenomenon.
- (4) Despite getting almost the same results for different lattice rotations, more lattice rotation sensitivity is seen in compression problems (although the differences in the results were not significant). The possible reasons for this issue probably relates to the mentioned assumptions in number (1); the performance of the proposed plasticity model in matching with the classical theory is questionable. Hence, further studies are recommended on this issue.

- (5) Although SPLM/SPPM is a novel numerical approach and in the initial stages of development, the obtained results showed the capability of this method to compete with well-known, well-developed commercial FEM codes such as Abaqus.
- (6) The SPLM is much simpler than comparable finite element models. Computational implementations of the method, such as damage evolution approaches, are much easier in SPLM than FEM continuum based methods. Furthermore, defining and understanding the input parameters is also easier and the model needs less effort to calibrate.
- (7) The small differences between SPLM and the theoretical and FEM results (elastic region slope, and also post-peak behavior), can be justified by considering the mentioned issues expressed in number (1). Nevertheless, the obtained results demonstrated the ability of SPLM to seriously challenge continuum FEM approaches.
- (8) Despite having similar pre-peak and post-peak behaviors for different lattice rotations in SPLM, different cracking patterns were obtained (not unlike to real physical behaviors). It can be interpreted as the ability of SPLM/SPPM in producing more realistic solutions. This issue should be considered as one of the advantages of SPLM/SPPM compared to FEM.
- (9) The main weakness of SPLM compared with FEM models, in general, can be expressed as more computational effort. However, since lots of complex optimizations have been made in Abaqus codes, comparing the SPLM method and Abaqus FEM models in terms of analysis timing is not rational, and is not performed in this study.
- (10) Comparing only different Abaqus models with each other, the Abaqus damage plasticity approach illustrated better and closer solutions to theoretical results. On the other hand, defining the input parameters (specifically, defining the damage parameters) for the damage plasticity model is much harder for the user than other methods. In addition, other Abaqus approaches have more limitations and less capabilities in modeling the real behavior of concrete. The brittle cracking model can only simulate the tensile damage and cannot model the plastic behavior of concrete; however, it has fewer input parameters and the model can be defined and calibrated with less effort by the user.
- (11) Mesh sensitivity issues, related to use of characteristic length of elements, can be considered as another major difficulty of Abaqus concrete cracking approaches. In addition, difficulties in performing convergence (mesh refinement) studies is another deficiency of the FEM concrete cracking approaches.

6.3. Final Thoughts

The proposed re-formulated SPLM can reasonably model concrete structures and provide even more realistic results than the commercial FEM software packages.

In conclusion, SPLM/SPPM, and peridynamics-based methods generally, are promising numerical methods for solid mechanics simulations. Further development of the SPLM/SPPM may make it competitive with FEM and other conventional continuum approaches for simulating cementitious materials

6.4. Future Studies

The following suggestions for future research (for SPLM framework) are presented (sorted based on the level of complexity):

- (1) Define better SPLM damage functions.
- (2) Develop more realistic plasticity models (for both plastic flow and plastic yielding envelope). For instance, the Drucker Prager plasticity model [33] is recommended to be implemented in the SPLM framework for simulating concrete, because of the reported correspondence of this approach to concrete behavior.
- (3) Evaluate SPLM for simulating other materials, and propose calibrated SPLM models for other materials. Models for steel structures are desirable.
- (4) Investigate the current SPLM applied to cyclic loading and fatigue damage.
- (5) Study the behavior of concrete structures under high loading rates. Study the abilities of SPLM in simulating highly dynamic problems.
- (6) Modify the current SPLM model to include creep and shrinkage. Develop a creep-cracking model.
- (7) Propose adaptive refinement approaches suitable for the SPLM method. Refinement methods for SPLM can be categorized into two general forms of (1) increasing the horizon radius (considering more bond while keeping the lattice spacing constant), and (2) using more particles in certain regions (changing the lattice spacing). For the first mentioned scenario, considering bigger horizon sizes for the boundary particles are suggested. In order to implement adaptive refinement procedures, defining an error indicator/estimator for SPLM/SPPM framework is necessary. Note that more complex and more objective adaptive refinement techniques can be implemented using SPPM formulation.

More general (and probably more complicated) study suggestions are proposed based on SPPM/SPLM approach:

- (1) Implement a hybrid SPLM/SPPM approach. The obtained results from this study indicate that having a lattice configuration would be sufficient to match with the classical solutions with less computational effort than a pure SPPM approach. However, boundary problems and other mentioned issues with lattice formulations remain. Use the SPPM formulation for boundary particles, and use SPLM for the particles in the material bulk.

- (2) Propose a non-deterministic approach (random method) based on the SPPM formulation.
- (3) Investigate updating the particle neighbor-lists, also re-calculation of micro-elastic modulus, in every time step (particularly, after damage initiation). The outcomes of this study may lead to developing more realistic damage models.
- (4) Combine SPPM/SPLM with other numerical approaches (for instance FEM, or other meshless methods).
- (5) Going further from the classical elastic-plastic behaviors and formulate different concepts. Consider impact forces or contact forces for the damaged particles, even defining new material behavior.

References

- [1] Van Vliet MR, Van Mier JG. Experimental investigation of size effect in concrete and sandstone under uniaxial tension. *Engineering Fracture Mechanics*. 2000;65:165-88.
- [2] Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. *International journal for numerical methods in engineering*. 1994;37:229-56.
- [3] Gingold RA, Monaghan JJ. Smoothed particle hydrodynamics-theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*. 1977;181:375-89.
- [4] Arzani H, Afshar M. Solving Poisson's equations by the discrete least square meshless method. *WIT Transactions on Modelling and Simulation*. 2006;42:23-31.
- [5] Nikravesht Kazeroni S, Afshar MH. An adaptive node regeneration technique for the efficient solution of elasticity problems using MDLSM method. *Engineering Analysis with Boundary Elements*. 2015;50:198-211.
- [6] Liu G-R, Gu Y-T. *An introduction to meshfree methods and their programming*: Springer; 2005.
- [7] Bažant Z. Microplane model for strain controlled inelastic behaviour, Chapter 3. CS Desai and RH Gallagher eds. 1984.
- [8] Jirásek M. Damage and smeared crack models. *Numerical modeling of concrete cracking*. 2011:1-49.
- [9] ABAQUS Analysis user's manual 2016, Dassault Systems Simulia Corp., Providence, RI, USA. 2016.
- [10] ABAQUS Theory user's manual 2016, Dassault Systems Simulia Corp., Providence, RI, USA. 2016.
- [11] Silling SA. Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids*. 2000;48:175-209.
- [12] Gerstle W, Sau N, Silling S. Peridynamic modeling of concrete structures. *Nuclear engineering and design*. 2007;237:1250-8.
- [13] Silling SA, Epton M, Weckner O, Xu J, Askari E. Peridynamic states and constitutive modeling. *Journal of Elasticity*. 2007;88:151-84.
- [14] Gerstle WH. *INTRODUCTION TO PRACTICAL PERIDYNAMICS: Computational Solid Mechanics Without Stress and Strain*: World Scientific Publishing Co Inc; 2015.
- [15] Hillerborg A, Modéer M, Petersson P-E. Analysis of crack formation and crack growth in concrete by means of fracture mechanics and finite elements. *Cement and concrete research*. 1976;6:773-81.
- [16] Lubliner J, Oliver J, Oller S, Onate E. A plastic-damage model for concrete. *International Journal of solids and structures*. 1989;25:299-326.
- [17] Lee J, Fenves GL. Plastic-damage model for cyclic loading of concrete structures. *Journal of engineering mechanics*. 1998;124:892-900.
- [18] Genikomsou AS, Polak MA. Finite element analysis of punching shear of concrete slabs using damaged plasticity model in ABAQUS. *Engineering Structures*. 2015;98:38-48.
- [19] Silling S, Lehoucq R. Peridynamic theory of solid mechanics. *Advances in applied mechanics*. 2010;44:73-168.

- [20] Macek RW, Silling SA. Peridynamics via finite element analysis. *Finite Elements in Analysis and Design*. 2007;43:1169-78.
- [21] Silling SA, Askari E. A meshfree method based on the peridynamic model of solid mechanics. *Computers & structures*. 2005;83:1526-35.
- [22] Seleson P, Littlewood DJ. Convergence studies in meshfree peridynamic simulations. *Computers & Mathematics with Applications*. 2016;71:2432-48.
- [23] Gerstle W, Sau N, Aguilera E. Micropolar peridynamic constitutive model for concrete. 2007.
- [24] Gerstle W, Sau N, Silling S. Peridynamic modeling of plain and reinforced concrete structures. *SMiRT18, Int Conf Structural Mech Reactor Technol2005*.
- [25] Huang D, Zhang Q, Qiao P. Damage and progressive failure of concrete structures using non-local peridynamic modeling. *Science China Technological Sciences*. 2011;54:591-6.
- [26] Sadd MH. *Elasticity: theory, applications, and numerics*: Access Online via Elsevier; 2009.
- [27] ACI Committee 318 (1992) *Building Code Requirements for Reinforced Concrete and Commentary (ACI 318-92 / ACI 318R-92)*. American Concrete Institute, Detroit, 353pp.
- [28] Committee A. *Building code requirements for structural concrete (ACI 318-14) and commentary*. ACI, Farmington Hills, United States. 2014.
- [29] MC90 C. *Design of Concrete Structures. CEB-FIP Model Code 1990*. Thomas Telford; 1993.
- [30] Wright P. Comments on an indirect tensile test on concrete cylinders. *Magazine of Concrete Research*. 1955;7:87-96.
- [31] Bazant ZP, Planas J. *Fracture and size effect in concrete and other quasibrittle materials*: CRC press; 1997.
- [32] Rocco C, Guinea G, Planas J, Elices M. Review of the splitting-test standards from a fracture mechanics point of view. *Cement and concrete research*. 2001;31:73-82.
- [33] Drucker DC, Prager W. Soil mechanics and plastic analysis or limit design. *Quarterly of applied mathematics*. 1952;10:157-65.